Tat-Jen Cham
Jianfei Cai
Chitra Dorai
Deepu Rajan
Tat-Seng Chua
Liang-Tien Chia (Eds.)

LNCS 4351

Advances in Multimedia Modeling

13th International Multimedia Modeling Conference, MMM 2007
Singapore, January 2007
Proceedings, Part I



Springer

Lecture Notes in Computer Science 4351

Commenced Publication in 1973
Founding and Former Series Editors:
Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Editorial Board

David Hutchison
Lancaster University, UK

Takeo Kanade
Carnegie Mellon University, Pittsburgh, PA, USA

Josef Kittler
University of Surrey, Guildford, UK

Jon M. Kleinberg
Cornell University, Ithaca, NY, USA

Friedemann Mattern
ETH Zurich, Switzerland

John C. Mitchell
Stanford University, CA, USA

Moni Naor
Weizmann Institute of Science, Rehovot, Israel

Oscar Nierstrasz
University of Bern, Switzerland

C. Pandu Rangan
Indian Institute of Technology, Madras, India

Bernhard Steffen
University of Dortmund, Germany

Madhu Sudan
Massachusetts Institute of Technology, MA, USA

Demetri Terzopoulos
University of California, Los Angeles, CA, USA

Doug Tygar
University of California, Berkeley, CA, USA

Moshe Y. Vardi
Rice University, Houston, TX, USA

Gerhard Weikum
Max-Planck Institute of Computer Science, Saarbruecken, Germany

Tat-Jen Cham Jianfei Cai
Chitra Dorai Deepu Rajan
Tat-Seng Chua Liang-Tien Chia (Eds.)

Advances in Multimedia Modeling

13th International Multimedia Modeling Conference, MMM 2007
Singapore, January 9-12, 2007
Proceedings, Part I

Springer

Volume Editors

Tat-Jen Cham
Jianfei Cai
Deepu Rajan
Liang-Tien Chia
Nanyang Technological University, School of Computer Engineering
Block N4, Nanyang Avenue, Singapore 639798
E-mail: {astjcham,asjfcai,asdrajan,asltchia}@ntu.edu.sg

Chitra Dorai
IBM T.J. Watson Research Center
P.O. Box 704, Yorktown Heights, NY 10598, USA
E-mail: dorai@watson.ibm.com

Tat-Seng Chua
National University of Singapore
School of Computing, Department of Computer Science
Singapore
E-mail: chuats@comp.nus.edu.sg

Library of Congress Control Number: 2006939130

CR Subject Classification (1998): H.5.1, H.5, I.4, H.2.4, I.3, H.3-4, E.4

LNCS Sublibrary: SL 3 – Information Systems and Application, incl. Internet/Web and HCI

ISSN 0302-9743
ISBN-10 3-540-69421-8 Springer Berlin Heidelberg New York
ISBN-13 978-3-540-69421-2 Springer Berlin Heidelberg New York

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, re-use of illustrations, recitation, broadcasting, reproduction on microfilms or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer. Violations are liable to prosecution under the German Copyright Law.

Springer is a part of Springer Science+Business Media

springer.com

© Springer-Verlag Berlin Heidelberg 2007
Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 11968061 06/3142 5 4 3 2 1 0

Preface

The 13th International Multimedia Modeling Conference (MMM) was held in Singapore on January 9–12, 2007, organized by the School of Computer Engineering, Nanyang Technological University (NTU). The conference venue was the Nanyang Executive Centre, located within NTU's 200 hectare Jurong campus in the west of Singapore, and it also served as the main conference accommodation.

The main technical sessions were held on January 10–12, 2007, comprising 2 keynote talks, 18 oral presentation sessions in 2 parallel tracks, and 2 poster sessions. A wide range of topics was covered in the conference, including multimedia content analysis (retrieval, annotation, learning semantic concepts), computer vision/graphics (tracking, registration, shape modeling), multimedia networking (coding, peer-to-peer systems, adaptation), multimedia access (databases, security) and human–computer interaction (user interfaces, augmented reality).

This year a bumper crop of 392 paper submissions were received for publication in the main conference. In order to achieve our goal of instantiating a high–quality review process for the conference, a large and motivated Technical Program Committee had to be formed. Thankfully, we were able to rely on the help of many committed senior researchers and eventually a review structure was created comprising 18 Area Chairs, 152 Program Committee members and 36 additional reviewers. The review process was rigorous and double blind, with each paper assigned to three to four reviewers, and further managed by an Area Chair who provided additional input and recommendations to the Program Chairs. In addition, there was collaboration with other conferences with overlapping review periods to avoid accepting papers which were submitted simultaneously to different conferences. Through the conscientious efforts of the reviewers, all submissions received at least two reviews, while over 97% of submissions received at least three to four reviews. Subsequently, all papers were considered carefully, with significant deliberation over borderline papers. Eventually, only 72 papers were accepted for oral presentation and 51 papers accepted for poster presentation, resulting in a competitive acceptance rate of 31.4%. The only distinguishing difference between the oral and poster papers was the mode of presentation – all accepted papers were considered full papers and allocated the same number of pages. Additionally, there were two paper awards given out at this conference: the Best Paper Award, and the Best Student Paper Award.

Outside of the main technical sessions, there were also four special sessions on Networked Graphics Applications (NGA), Services and the Assurance in Multimedia Mobile Information Systems (SAMM), Networked Multimedia Systems and Applications Focusing on Reliable and Flexible Delivery for Integrated Multimedia (NMS) and Ubiquitous Multimedia Service (UMS). The paper review for these special sessions was handled separately by different organizers and Program Committees, and accepted papers were presented on January 9, 2007.

There was also a conference banquet on January 11, 2007, which featured a dinner boat cruise along Singapore's harbor front on the Imperial Cheng Ho.

We are heavily indebted to many individuals for their significant contribution. In particular, Linda Ang was very helpful in maintaining the Web-based review management system and solving technical crises almost instantly. Su-Ming Koh was crucial in creating, maintaining and handling all registration-related matters effectively and efficiently. Poo-Hua Chua promptly handled all matters related to the main conference Web site. Hwee-May Oh consistently kept the Organizing Committee in tow by checking and monitoring the action plans before and after every meeting. We thank the MMM Steering Committee for their invaluable input and guidance in crucial decisions. We would like to express our deepest gratitude to the rest of the Organizing Committee: Industrial Track Chair Chang-Sheng Xu, Local Arrangements Chairs Wooi Boon Goh and Kin-Choong Yow, Publicity and Sponsorship Chairs Sabu Emmanuel and Kap-Luk Chan, and Workshop Chairs Chiew Tong Lau and Fang Li. We are also most sincerely appreciative of the hard work put in by the Area Chairs and members of the Technical Program Committee, whose detailed reviews under time pressure were instrumental in making this a high-quality conference.

We would like to thank the Lee Foundation and PREMIA for their generous sponsorship, as well as help from the School of Computing, National University of Singapore, ACM SIGMM, and the Singapore Tourism Board. Finally, this conference would not have been possible without strong and unwavering support from NTU's Centre for Multimedia & Network Technology (CeMNet).

January 2007

Tat-Jen Cham
Jianfei Cai
Chitra Dorai
Deepu Rajan
Tat-Seng Chua
Liang-Tien Chia

Organization

Organizing Committee

General Co-chairs:	Tat-Seng Chua (National University of Singapore, Singapore) Liang-Tien Chia (Nanyang Technological University, Singapore)
Program Co-chairs:	Tat-Jen Cham (Nanyang Technological University, Singapore) Jianfei Cai (Nanyang Technological University, Singapore) Chitra Dorai (IBM T.J. Watson Research Center, New York)
Industrial Track Chair:	Changsheng Xu (Institute for Infocomm Research, Singapore)
Workshop/Tutorials Co-chairs:	Chiew Tong Lau (Nanyang Technological University, Singapore) Fang Li (Nanyang Technological University, Singapore)
Publications Chair:	Deepu Rajan (Nanyang Technological University, Singapore)
Local Arrangements Co-Chairs:	Wooi Boon Goh (Nanyang Technological University, Singapore) Kin-Choong Yow (Nanyang Technological University, Singapore)
Publicity and Sponsorship Co-chairs:	Sabu Emmanuel (Nanyang Technological University, Singapore) Kap-Luk Chan (Nanyang Technological University, Singapore)
Registration:	Su-Ming Koh
Webmaster:	Linda Ang Poo-Hua Chua
Secretary:	Hwee-May Oh

Steering Committee

Yi-Ping Phoebe Chen (Deakin University , Australia)
Tat-Seng Chua(National University of Singapore, Singapore)
Tosiyasu L. Kunii (Kanazawa Institute of Technology, Japan)
Wei-Ying Ma (Microsoft Research Asia, China)
Nadia Magnenat-Thalmann (University of Geneva , Switzerland)
Patrick Senac (Ensica, France)

Program Committee

Area Chairs

Edward Chang
Lap-Pui Chau
Shu-Ching Chen
Ajay Divakaran
Alan Hanjalic
Mohan Kankanhalli
Zhengguo Li
Chiawen Lin
Wolfgang Mller-Wittig
Wei Tsang Ooi
Silvia Pfeiffer
Mei-Ling Shyu
Qibin Sun
Daniel Thalmann
Marcel Worring
Jiankang Wu
Changsheng Xu
Roger Zimmerman

Members

Lalitha Agnihotri
Terumasa Aoki
Pradeep Kumar Atrey
Noboru Babaguchi
Selim Balcisoy
Qiu Bo
Shen Bo
Ronan Boulic
Djeraba Chabane
Lekha Chaisorn
Ee-Chien Chang
Kai Chen
Lei Chen
Mei-Juan Chen
Shoupu Chen
Xilin Chen
Yi-Shin Chen
Shao-Yi Chien
Eng Siong Chng
Hao-hua Chu
Jen-Yao Chung
Pablo de Heras Ciechomski
Serhan Dagtas
Ravindra Dastikop
Michel Diaz
Zoran Dimitrijevic
LingYu Duan
Kun Fu
Sheng Gao
John M. Gauch
Yu Ge
Enrico Gobbetti
Romulus Grigoras
William I. Grosky
Junzhong Gu
Xiaohui Gu
Zhenghui Gu
Mario Gutierrez
Jiro Gyoba
Daniel Haffner
Xian-Sheng Hua
Haibin Huang
Qingming Huang
Weimin Huang
Zhiyong Huang
Benoit Huet
Andres Iglesias
Horace Ip
Xing Jin
Xuan Jing
James Joshi
Marcelo Kallmann
Li-Wei Kang
Ahmed Karmouch
Pavel Korshunov
Jose Lay
Clement Leung
Chung-Sheng Li
He Li
Huiqi Li
Liyuan Li
Mingjing Li
Qing Li
Te Li
Xuelong Li
Ying Li
Rainer Lienhart
Joo-Hwee Lim
Jian-Liang Lin
Weisi Lin
Karen Liu
Tiecheng Liu
Yang Liu
Ying Liu
Alexader Loui
Kok-Lim Low
Guojun Lu
Hanqing Lu
Zhongkang Lu
Hongli Luo
Jian-Guang Luo
Jiebo Luo
Jianhua Ma
Namunu Maddage
Nadia Magnenat-Thalmann

Enrico Magli
Stephane Marchand-Maillet
Bernard Merialdo
Kazunori Miyata
Soraia Raupp Musse
P.J. Narayanan
Luciana Nedel
Chong Wah Ngo
Noel O'Connor
Ee Ping Ong
Vincent Oria
Pietro Pala
Feng Pan
Nilesh Patel
Wen-Hsiao Peng
Julien Pettre
B. Prabhakaran
Regunathan Radhakrishnan
Kalpathi Ramakrishnan
Lloyd Rutledge
Shin'ichi Satoh
Dieter Schmalstieg
Guus Schreiber
Nicu Sebe
Ho Kim Seon
Ishwar Sethi
Timothy Shih
P. Shivakumara
Haiyan Shu
Alan Smeaton
Cees Snoek
Luiz Fernando Gomes Soares
Yuqing Song
Alexei Sourin
Yeping Su
Lifeng Sun
Hari Sundaram
Jo Yew Tham
Yu-Kuang Tu
Jean-Marc Valin
Svetha Venkatesh
Frederic Vexo
Kongwah Wan
Jinjun Wang
Xingwei Wang
Yu Wang
Jongwook Woo
Yi Wu
Yi-Leh Wu
Lexing Xie
Ruiqin Xiong
Ziyou Xiong
Xiangyang Xue
Xiaokang Yang
Susu Yao
Kim-Hui Yap
Chai Kiat Yeo
Rongshan Yu
Xinguo Yu
Chengcui Zhang
Haihong Zhang
Lei Zhang
Zhongfei Zhang
Jinghong Zheng
Xiaofang Zhou
Guangyu Zhu
Yongwei Zhu

Additional Reviewers

Marco Agus
Alia Amin
Michael Blighe
Rui Cai
Kuan-Ta Chen
Songqing Chen
Rodrigo Mendes Costa
Carlos Augusto Dietrich
JL Dugelay
Arjan Egges
Eric Galmar
Stephane Garchery
Zhen Guo
Michiel Hildebrand
Keith Jacoby
Minseok Jang
Xiaoxi Jiang
Saubhagya Ram Joshi
Mustafa Kasap
Andrew Kinane
Duy-Dinh Le
Bart Lehane
Dongyu Liu
Mentar Mahmudi
Joanna Marguier
Jean Martinet
Simon Moncrieff
Manuel Menezes de Oliveira Neto
Ciaran O'Conaire
Marcelo Soares Pimenta
Dimitris Protopsaltou
Tele Tan
Ba Tu Truong
Changhu Wang
Jian Yao
Ruofei Zhang

Keynote Speakers

Keynote Speaker I

Multimedia and Web 2.0: Challenge and Synergy

Professor Edward Chang received his MS in Computer Science and PhD in Electrical Engineering at Stanford University in 1994 and 1999, respectively. He joined the department of Electrical and Computer Engineering at University of California, Santa Barbara, in September 1999. He received his tenure in March 2003, and was promoted to full professor of Electrical Engineering in 2006. His recent research activities are in the areas of machine learning, data mining, high-dimensional data indexing, and their applications to image databases, video surveillance, and Web mining. Recent research contributions of his group include methods for learning image/video query concepts via active learning with kernel methods, formulating distance functions via dynamic associations and kernel alignment, managing and fusing distributed video-sensor data, categorizing and indexing high-dimensional image/video information, and speeding up support vector machines via parallel matrix factorization and indexing. Professor Chang has served on several ACM, IEEE, and SIAM conference program committees. He co-founded the annual ACM Video Sensor Network Workshop and has co-chaired it since 2003. In 2006, he co-chaired three international conferences: Multimedia Modeling (Beijing), SPIE/IS&T Multimedia Information Retrieval (San Jose), and ACM Multimedia (Santa Barbara). He serves as an Associate Editor for *IEEE Transactions on Knowledge and Data Engineering and ACM Multimedia Systems Journal*. Professor Chang is a recipient of the IBM Faculty Partnership Award and the NSF Career Award. He is currently on leave from UC, heading R&D effort at Google/China.

Keynote Speaker II

Dr. Dick Bulterman is a senior researcher at CWI in Amsterdam, where he heads Distributed Multimedia Languages and Interfaces since 2004. From 1988 to 1994 (and briefly in 2002), he led CWI's Department of Computer Systems and Telematics and from 1994 to 1998, he was head of Multimedia and Human Computer. In 1999, he and two other brave souls started Oratrix Development BV, a CWI spin-off company that transfered the group's SMIL-based GRiNS software to many parts of the developed world. In 2002, after handing the responsibilities of CEO over to Mario Wildvanck, he returned to CWI and started up a new research activity at CWI on distributed multimedia systems. Prior to joining CWI in 1988, he was on the faculty of the Division of Engineering at Brown, where he was part of the Laboratory for Engineering Man/Machine Systems. Other academic appointments include visiting professorships in computer

science at Brown (1993-94) and in the information theory group at TU Delft (1985) and a part-time appointment in computer science at the Univeristy of Utrecht (1989-1991). Dr. Bulterman received a PhD in Computer Science from Brown University (USA) in 1982. He also holds an M.Sc. in Computer Science from Brown (1977) and a BA in Economics from Hope College (1973). He started his academic journey at Tottenville High School on Staten Island, NY, where he learned (among other things) to play trombone and string bass. He was born in Amstelveen (The Netherlands) in 1951; after 35 years in the USA, he now resides with his family in Amsterdam. His hobbies (in as much as one can speak of hobbies with two children under the age of 12 ...) include flying airplanes (he holds an FAA private ASEL license with instrument rating and a Dutch commercial pilot's license with IR), singing in the Cantorij of the Oude Kerk in Amsterdam and trying to learn piano and cello (which is a much lighter instrument than a string bass). He is on the editorial board of *ACM Trans. on Multimedia Communications, Computing and Applications (TOMCCAP), ACM/Springer Multimedia Systems Journal and Multimedia Tools and Applications*. He is a member of Sigma Xi, the ACM and the IEEE.

Table of Contents

Learning Semantic Concepts

Temporal Video Segmentation on H.264/AVC Compressed Bitstreams 1
Sarah De Bruyne, Wesley De Neve, Koen De Wolf, Davy De Schrijver, Piet Verhoeve, and Rik Van de Walle

Ontology-Based Annotation of Paintings Using Transductive Inference Framework 13
Yelizaveta Marchenko, Tat-Seng Chua, and Ramesh Jain

Interactive Visual Object Extraction Based on Belief Propagation 24
Shiming Xiang, Feiping Nie, Changshui Zhang, and Chunxia Zhang

Modeling Modifications of Multimedia Learning Resources Using Ontology-Based Representations 34
Marek Meyer, Sonja Bergstraesser, Birgit Zimmermann, Christoph Rensing, and Ralf Steinmetz

Graphics

Region-Based Reconstruction for Face Hallucination 44
Jeong-Seon Park, Junseak Lee, and Seong-Whan Lee

A Shape Distribution for Comparing 3D Models 54
Levi C. Monteverde, Conrado R. Ruiz Jr., and Zhiyong Huang

3D Facial Modeling for Animation: A Nonlinear Approach 64
Yushun Wang and Yueting Zhuang

Normalization and Alignment of 3D Objects Based on Bilateral Symmetry Planes 74
Jefry Tedjokusumo and Wee Kheng Leow

Image Registration, Matching and Texture

Extraction of Anatomic Structures from Medical Volumetric Images 86
Wan-Hyun Cho, Sun-Worl Kim, Myung-Eun Lee, and Soon-Young Park

Dual-Space Pyramid Matching for Medical Image Classification 96
Yang Hu, Mingjing Li, Zhiwei Li, and Wei-ying Ma

An Image Registration Method Based on the Local and Global Structures 106
Nan Peng, Zhiyong Huang, and Zujun Hou

Automated Segmentation of Drosophila RNAi Fluorescence Cellular Images Using Graph Cuts 116
Cheng Chen, Houqiang Li, and Xiaobo Zhou

Human-Computer Interaction

Recommendation of Visual Information by Gaze-Based Implicit Preference Acquisition 126
Atsuo Yoshitaka, Kouki Wakiyama, and Tsukasa Hirashima

The 3D Sensor Table for Bare Hand Tracking and Posture Recognition 138
Jaeseon Lee, Kyoung Shin Park, and Minsoo Hahn

Legible Collaboration System Design 147
Toshiya Fujii, Wonsuk Nam, and Ikuro Choh

Presentation of Dynamic Maps by Estimating User Intentions from Operation History 156
Taro Tezuka and Katsumi Tanaka

Tracking and Motion Analysis

An Object Tracking Scheme Based on Local Density 166
Zhuan Qing Huang and Zhuhan Jiang

Modeling Omni-Directional Video 176
Shumian He and Katsumi Tanaka

Temporally Integrated Pedestrian Detection from Non-stationary Video 188
Chi-Jiunn Wu and Shang-Hong Lai

Visual Perception Theory Guided Depth Motion Estimation 198
Bing Li, De Xu, Songhe Feng, and Fangshi Wang

Advanced Media Coding and Adaptation

Adaptive Data Retrieval for Load Sharing in Clustered Video Servers ... 207
Minseok Song

A User-Friendly News Contents Adaptation for Mobile Terminals 217
Youn-Sik Hong, Ji-Hong Kim, Yong-Hyun Kim, and Mee-Young Sung

An Efficient Predictive Coding of Integers with Real-Domain Predictions Using Distributed Source Coding Techniques 227
Mortuza Ali and Manzur Murshed

A Distributed Video Coding Scheme Based on Denoising Techniques . . . 237
Guiguang Ding and Feng Yang

Media Annotation

Fusion of Region and Image-Based Techniques for Automatic Image Annotation 247
Yang Xiao, Tat-Seng Chua, and Chin-Hui Lee

Automatic Refinement of Keyword Annotations for Web Image Search 259
Bin Wang, Zhiwei Li, and Mingjing Li

Mining Multiple Visual Appearances of Semantics for Image Annotation 269
Hung-Khoon Tan and Chong-Wah Ngo

Automatic Video Annotation and Retrieval Based on Bayesian Inference 279
Fangshi Wang, De Xu, Wei Lu, and Weixin Wu

Image and Video Coding

Density-Based Image Vector Quantization Using a Genetic Algorithm 289
Chin-Chen Chang and Chih-Yang Lin

Multilayered Contourlet Based Image Compression 299
Fang Liu and Yanli Liu

Iterative Image Coding Using Hybrid Wavelet-Based Triangulation 309
Phichet Trisiripisal, Sang-Mook Lee, and A. Lynn Abbott

A Novel Video Coding Framework by Perceptual Representation and Macroblock-Based Matching Pursuit Algorithm 322
Jianning Zhang, Lifeng Sun, and Yuzhuo Zhong

Context-Aware Media Modeling

MetaXa—Context- and Content-Driven Metadata Enhancement for Personal Photo Books 332
Susanne Boll, Philipp Sandhaus, Ansgar Scherp, and Sabine Thieme

Context-Sensitive Ranking for Effective Image Retrieval 344
Guang-Ho Cha

Visual Verification of Historical Chinese Calligraphy Works 354
Xiafen Zhang and Yueting Zhuang

Discovering User Information Goals with Semantic Website Media Modeling 364
Bibek Bhattarai, Mike Wong, and Rahul Singh

Multimedia Databases

Online Surveillance Video Archive System 376
Nurcan Durak, Adnan Yazici, and Roy George

Hierarchical Indexing Structure for 3D Human Motions 386
Gaurav N. Pradhan, Chuanjun Li, and Balakrishnan Prabhakaran

Similarity Searching Techniques in Content-Based Audio Retrieval Via Hashing 397
Yi Yu, Masami Takata, and Kazuki Joe

Fast Answering k-Nearest-Neighbor Queries over Large Image Databases Using Dual Distance Transformation 408
Yi Zhuang and Fei Wu

Media Retrieval

Subtrajectory-Based Video Indexing and Retrieval 418
Thi-Lan Le, Alain Boucher, and Monique Thonnat

DR Image and Fractal Correlogram: A New Image Feature Representation Based on Fractal Codes and Its Application to Image Retrieval 428
Takanori Yokoyama and Toshinori Watanabe

Cross-Modal Interaction and Integration with Relevance Feedback for Medical Image Retrieval 440
Md. Mahmudur Rahman, Varun Sood, Bipin C. Desai, and Prabir Bhattacharya

A New Multi-view Learning Algorithm Based on ICA Feature for Image Retrieval 450
Fan Wang and Qionghai Dai

P2P Networks

A P2P Architecture for Multimedia Content Retrieval 462
E. Ardizzone, L. Gatani, M. La Cascia, G. Lo Re, and M. Ortolani

Optimizing the Throughput of Data-Driven Based Streaming in Heterogeneous Overlay Network 475
Meng Zhang, Chunxiao Chen, Yongqiang Xiong, Qian Zhang, and Shiqiang Yang

LSONet: A Case of Layer-Encoded Video Transmission in Overlay Networks 485
Hui Guo, Kwok-Tung Lo, and Jiang Li

A Strategyproof Protocol in Mesh-Based Overlay Streaming System 495
Rui Sun, Ke Xu, Zhao Li, and Li Zhang

Semantic Video Concepts

Utility-Based Summarization of Home Videos 505
Ba Tu Truong and Svetha Venkatesh

Performance Analysis of Multiple Classifier Fusion for Semantic Video Content Indexing and Retrieval 517
Rachid Benmokhtar and Benoit Huet

Video Semantic Concept Detection Using Multi-modality Subspace Correlation Propagation 527
Yanan Liu and Fei Wu

Enhancing Comprehension of Events in Video Through Explanation-on-Demand Hypervideo 535
Nimit Pattanasri, Adam Jatowt, and Katsumi Tanaka

Audio and Video Coding

Low-Complexity Binaural Decoding Using Time/Frequency Domain HRTF Equalization 545
Rongshan Yu, Charles Q. Robinson, and Corey Cheng

Complexity Reduction of Multi-frame Motion Estimation in H.264 557
Linjian Mo, Jiajun Bu, Chun Chen, Zhi Yang, and Yi Liu

A Novel Intra/Inter Mode Decision Algorithm for H.264/AVC Based on Spatio-temporal Correlation 568
Qiong Liu, Shengfeng Ye, Ruimin Hu, and Zhen Han

Switchable Bit-Plane Coding for High-Definition Advanced Audio Coding 576
Te Li, Susanto Rahardja, and Soo Ngee Koh

Content I

Neighborhood Graphs for Semi-automatic Annotation of Large Image Databases . . . 586
Hakim Hacid

Bridging the Gap Between Visual and Auditory Feature Spaces for Cross-Media Retrieval . . . 596
Hong Zhang and Fei Wu

Film Narrative Exploration Through the Analysis of Aesthetic Elements . . . 606
Chia-Wei Wang, Wen-Huang Cheng, Jun-Cheng Chen, Shu-Sian Yang, and Ja-Ling Wu

Semantic Image Segmentation with a Multidimensional Hidden Markov Model . . . 616
Joakim Jiten and Bernard Merialdo

Semi-supervised Cast Indexing for Feature-Length Films . . . 625
Wei Fan, Tao Wang, JeanYves Bouguet, Wei Hu, Yimin Zhang, and Dit-Yan Yeung

Linking Identities and Viewpoints in Home Movies Based on Robust Feature Matching . . . 636
Ba Tu Truong and Svetha Venkatesh

An Efficient Automatic Video Shot Size Annotation Scheme . . . 649
Meng Wang, Xian-Sheng Hua, Yan Song, Wei Lai, Li-Rong Dai, and Ren-Hua Wang

Content Based Web Image Retrieval System Using Both MPEG-7 Visual Descriptors and Textual Information . . . 659
Joohyoun Park and Jongho Nang

Applications I

A New Method to Improve Multi Font Farsi/Arabic Character Segmentation Results: Using Extra Classes of Some Character Combinations . . . 670
Mona Omidyeganeh, Reza Azmi, Kambiz Nayebi, and Abbas Javadtalab

Modeling Television Schedules for Television Stream Structuring . . . 680
Jean-Philippe Poli and Jean Carrive

Automatic Generation of Multimedia Tour Guide from Local Blogs . . . 690
Hiroshi Kori, Shun Hattori, Taro Tezuka, and Katsumi Tanaka

Computer Vision

A Robust 3D Face Pose Estimation and Facial Expression Control for Vision-Based Animation 700
Junchul Chun, Ohryun Kwon, and Peom Park

Hierarchical Shape Description Using Skeletons 709
Jong-Seung Park

Motion Structure Parsing and Motion Editing in 3D Video 719
Jianfeng Xu, Toshihiko Yamasaki, and Kiyoharu Aizawa

Tamper Proofing 3D Motion Data Streams 731
Parag Agarwal and Balakrishnan Prabhakaran

Image Processing I

A Uniform Way to Handle Any Slide-Based Presentation: The Universal Presentation Controller 741
Georg Turban and Max Mühlhäuser

A Tensor Voting for Corrupted Region Inference and Text Image Segmentation 751
Jonghyun Park, Jaemyeong Yoo, and Gueesang Lee

A Novel Coarse-to-Fine Adaptation Segmentation Approach for Cellular Image Analysis 762
Kai Zhang, Hongkai Xiong, Lei Yang, and Xiaobo Zhou

Vehicle Classification from Traffic Surveillance Videos at a Finer Granularity 772
Xin Chen and Chengcui Zhang

A Fuzzy Segmentation of Salient Region of Interest in Low Depth of Field Image 782
KeDai Zhang, HanQing Lu, ZhenYu Wang, Qi Zhao, and MiYi Duan

Author Index 793

Temporal Video Segmentation on H.264/AVC Compressed Bitstreams

Sarah De Bruyne[1], Wesley De Neve[1], Koen De Wolf[1], Davy De Schrijver[1], Piet Verhoeve[2], and Rik Van de Walle[1]

[1] Department of Electronics and Information Systems - Multimedia Lab
Ghent University - IBBT
Gaston Crommenlaan 8 bus 201, B-9050 Ledeberg-Ghent, Belgium
sarah.debruyne@ugent.be
http://multimedialab.elis.ugent.be
[2] Televic, Belgium

Abstract. In this paper, a novel method for temporal video segmentation on H.264/AVC-compliant video bitstreams is presented. As the H.264/AVC standard contains several new and extended features, the characteristics of the coded frames are different from former video specifications. Therefore, previous shot detection algorithms are not directly applicable to H.264/AVC compressed video bitstreams. We present a new concept, in particular, 'Temporal Prediction Types', by combining two features: the different macroblock types and the corresponding display numbers of the reference frames. Based on this concept and the amount of intra-coded macroblocks, our novel shot boundary detection algorithm is proposed. Experimental results show that this method achieves high performance for cuts as well as for gradual changes.

1 Introduction

Recent advances in multimedia coding technology, combined with the growth of the internet, as well as the advent of digital television, have resulted in the widespread use and availability of digital video. As a consequence, many terabytes of multimedia data are stored in databases, often insufficiently cataloged and only accessible by sequential scanning. This has led to an increasing demand for fast access to relevant data, making technologies and tools for the efficient browsing and retrieval of digital video of paramount importance. The prerequisite step to achieve video content analysis is the automatic parsing of the content into visually-coherent segments, called shots, separated by shot boundaries [1].

The definition of a shot change is important to stress, since the object or camera motions may drastically change the content of a video sequence. A shot is defined as *"a sequence of frames continuously captured from the same camera"* [2]. According to whether the transition between consecutive shots is abrupt or not, boundaries are classified as cuts or gradual transitions, respectively.

Algorithms for shot boundary detection can be roughly classified in two major groups, depending on whether the operations are done on uncompressed data

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 1–12, 2007.
© Springer-Verlag Berlin Heidelberg 2007

or whether they work directly with compressed domain features. The two major video segmentation approaches operating in the uncompressed domain are based on color histogram differences [3] and changes in edge characteristics [4]. On the other hand, full decompression of the encoded video and the computational overhead can be avoided by using compressed domain features only. Since most video data are compressed to preserve storage space and reduce band width, we focus on methods operating in the compressed domain. Existing techniques in this domain mostly concentrate on the MPEG-1 Video, MPEG-2 Video, and MPEG-4 Visual standards. These algorithms are for the most part based on the correlation of DC coefficients [5], macroblock prediction type information [6,7], or the bit consumption (or bit rate) of a frame [8].

Due to the compression performance of the newest video compression standard H.264/AVC [9], more video content will probably be encoded in this format. This video specification possesses features like intra prediction in the spatial domain and multiple reference frames, which were not included in previous standards. In this paper, we investigate whether the earlier mentioned compressed domain methods are still applicable for H.264/AVC compressed data. Since these methods turn out to be inadequate, we propose a new shot detection algorithm for H.264/AVC compressed video.

The outline of this paper is as follows. In Sect. 2, the main characteristics of H.264/AVC are elaborated from a high-level point of view. Section 3 discusses the influences of these characteristics on existing compressed domain algorithms. A new shot boundary detection algorithm based on temporal prediction types is proposed in Sect. 4. Section 5 discusses a number of performance results obtained by our method. Finally, Sect. 6 concludes this paper.

2 Design Aspects of H.264/AVC

The H.264/AVC specification contains a lot of new technical features compared with prior standards for digital video coding [9]. With respect to shot boundary detection, H.264/AVC has three important design aspects, which are either new or extended compared to previous standards: intra prediction, slice types, and multi-picture motion-compensated prediction.

In contrast to prior video coding standards, intra prediction in H.264/AVC is conducted in the spatial domain, by referring to neighboring samples of previously-decoded blocks [9]. Two primary types of intra coding are supported: Intra_4×4 and Intra_16×16 prediction. In Intra_4×4 mode, each 4×4 luma block is predicted separately. This mode is well suited for coding parts of a picture with significant detail. The Intra_16×16 mode uses a 16×16 luma block and is more suited for coding very smooth areas of a picture. Another intra coding mode, I_PCM, enables the transmission of the values of the encoded samples without prediction or transformation. Furthermore, in the H.264/AVC *Fidelity Range Extensions* (FREext) amendment, Intra_8×8 is introduced. The latter two types are hardly used and are therefore not supported in the following algorithms, but these algorithms can easily be extended to cope with this prediction type too.

In addition, each picture is partitioned into MBs, which are organized in slices [9]. H.264/AVC supports five different slice types. In I slices, all MBs are coded using intra prediction. Prior-coded images can be used as a prediction signal for MBs of the predictive-coded P and B slices. Whereas P MB partitions can utilize only one frame to refer to, B MB partitions can use two reference frames. The remaining two slice types, SP and SI, which are specified for efficient switching between bitstreams coded at various bit rates, are rarely used.

The third design aspect, multi-picture motion-compensated prediction [9], enables efficient coding by allowing an encoder to select the best reference picture(s) among a larger number of pictures that have been decoded and stored in a buffer. Figure 1 illustrates this concept. A multi-picture buffer can contain both short term and long term reference pictures and allows reference pictures containing B slices. When using inter prediction for a MB in a P (or B) slice, the reference index (or indices) are transmitted for every motion-compensated 16×16, 16×8, 8×16, or 8×8 luma block.

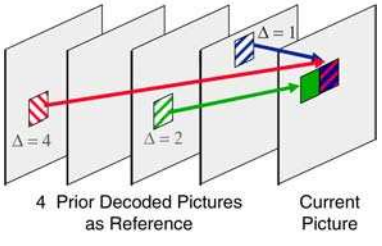


Fig. 1. Multi-picture motion-compensated prediction. In addition to the motion vector, picture reference parameters (Δ) are transmitted [9].

3 Temporal Segmentation Algorithms for H.264/AVC

In this section, we verify whether the existing compressed domain algorithms are still applicable to H.264/AVC compressed video bitstreams, keeping the new and improved characteristics of this specification in mind.

3.1 DC Coefficients

Shot boundary detection methods in compressed domain often use DC coefficients to generate DC images [5]. For intra-coded MBs, these DC coefficients represent the average energy of a block (i.e., 8×8 pixels), that can be extracted directly from the MPEG compressed data. For P and B frames, the DC coefficients of the referred regions in the reference frame are used to obtain the corresponding DC image. Based on these DC images, shot detection algorithms, such as color histograms, can be directly transformed to the compressed domain.

Unlike previous MPEG standards, DC coefficients of intra-coded MBs in H.264/AVC only present an energy difference between the current block and the adjacent pixels instead of the energy average. In case we want to apply the

proposed algorithm, we need to calculate the predicted energy from adjacent pixels to obtain the average energy. Therefore, almost full decoding is inevitable, which diminishes the advantages of this compressed domain method.

3.2 Bit Rate

In previous coding standards, frames located at a shot boundary consist for the greater part of intra coded MBs using prediction conducted in the transform domain only. This leads to high peaks in the bit rate, which makes shot boundary detection possible [8]. When looking at H.264/AVC-compliant bitstreams, frames coincided with shot boundaries normally have much lower bit rates than those of MPEG-2 Video for example, due to the intra prediction in the spatial domain. The height of these peaks decreases, which makes is more difficult to make a distinction between shot boundaries and movement. From these observations, one can conclude that this algorithm is hard to apply to H.264/AVC.

3.3 Macroblock Prediction Type Information

The distribution of the different MB prediction types [6,7] was used to detect shot boundaries in previous coding standards. This method exploits the decisions made by the encoder in the motion estimation phase, which results in specific characteristics of the MB type information whenever shot boundaries occur. As shown in Fig. 2, when a B frame does not belong to the same shot as one of its reference frames, the B frame will hardly refer to this reference frame. It is clear that the "amount" of the different prediction types of the MBs in a B frame can be applied to define a metric for locating possible shot boundaries.

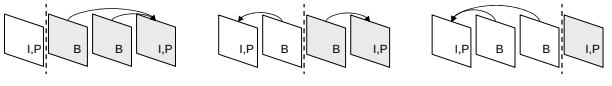


Fig. 2. Possible positions of a shot boundary

Due to the multi-picture motion-compensated prediction and the possibility of B frames to be used as reference pictures, the existing methods based on macroblock prediction types cannot directly be applied to H.264/AVC. However, features such as MB types, MB partitions, and the display numbers of reference pictures contain important information regarding the semantic structure of a video sequence. Moreover, these features can be extracted directly from the compressed data. In Sect. 4, a shot boundary detection algorithm is presented based on the above mentioned features.

4 Shot Boundary Detection in H.264/AVC

Within a video sequence, a continuous strong inter-frame correlation is present, as long as no significant changes occur. As a consequence, the different prediction

types and the direction of the reference frames in a frame can be applied to define a metric for locating possible shot boundaries.

To determine the direction of the reference frames, the 'display number' of the frames needs to be checked. This number represents the location of a frame in the decoded bitstream and can be derived from the Picture Order Count (POC) of the frame and the display number of the last frame prior to the previous Instantaneous Decoding Refresh (IDR) picture [9]. By comparing the display number of the current frame and the reference frames, we can derive whether the reference frames are displayed before or after the current frame.

In the context of shot boundary detection, we present the concept 'Temporal Prediction Types' combining the different macroblock types and the direction of the reference frames. Each MB type in a P or B slice corresponds to a specific partitioning of the MB in fixed-size blocks. For each macroblock partition, the prediction mode and the reference index or indices can be chosen separately. As each MB partition corresponds to exactly one prediction mode and the smallest partition size is 8×8 pixels, the following discussion is based on this 8×8 blocks.

Depending on the prediction mode of a MB partition, this partition consists of zero to two reference indices. In case no reference pictures are used, we speak of **intra temporal prediction**. Partitions that only use one reference picture to refer to, belong to one of these two temporal prediction types:

Forward temporal prediction in case the display number of the referred frame precedes the display number of the current frame.

Backward temporal prediction in case the current frame is prior to the referred frame.

This subdivision is used for MB partitions in a P slice or for partitions in a B slice that only use one reference frame. In case a MB partition in a B slice refers to two frames, the following classification is applied:

Forward temporal prediction in case the display numbers of both referred frames are prior to the current frame.

Backward temporal prediction in case the current frame is displayed earlier than both the referred frames.

Bi-directional temporal prediction in case the current frame is located in between the reference frames (which is very similar to the well-know concept used for B frames in MPEG-2 Video).

Summarized, we have four possible temporal prediction types, i.e. intra, forward, backward, and bi-directional temporal prediction.

According to the specification, it is allowed to construct coded pictures that consist of a mixture of different types of slices. However, in current applications where shot boundary detection is applicable, frames will normally be composed of slices with similar slice types. Therefore, in the remainder of this paper, we will refer to I, P, and B slice coded pictures as I, P, and B frames.

As mentioned before, there are two major types of shot changes: abrupt and gradual transitions. Since their characteristics are divergent, the detection of these transitions needs to be separated.

4.1 Detection of Abrupt Changes

One could expect that abrupt changes always occur at I frames, but it should be mentioned that this notion is not enough to detect shot boundaries. This is due to the fact that a certain type of I frames, in particular IDR pictures, are often used as random access points in a video. Therefore, I frames do not always correspond to shot boundaries. Further, depending on the encoder characteristics or the application area, the GOP structure of the video can either be fixed or adapted to the content, which can result in shot boundaries occurring at P or B frames. Considering this observation, a distinction is drawn between I, P, and B frames in order to detect the transitions.

Shot Boundaries Located at an I Frame. All MBs in an I frame are coded without referring to other pictures within the video sequence. As a consequence, they do not represent the temporal correlation between the current frame (F_i) and the previous depicted frame (F_{i-1}). However, in case this previous frame is a P or B frame, it contains interesting information, such as the temporal prediction types of the blocks. When the percentage of blocks with backward and bi-directional temporal prediction in F_{i-1} is large, there is a high correlation between F_{i-1} and F_i. As a consequence, the amount of blocks with intra and forward temporal prediction is low and the chance that a shot boundary is located between these two frames is very small. On the other hand, when the previous frame does not refer to the current frame, which results in a high percentage of intra and forward temporal predicted blocks, we cannot conclude that these two frames belong to different shots. During the encoding of the previous frame, for example, the current and following frames are not always at hand in the multi-picture buffer. In this case, backward and bi-directional temporal prediction in the previous frame are impossible.

To solve this problem, a second condition, based on the distribution of the intra prediction modes within two successive I frames, is added [10]. (These two I frames do not need to be located next to each other, as there can also be P and B frames in between them). Whereas MBs with 16×16 prediction modes are more suited for coding very smooth areas of a picture, those with 4×4 prediction are used for parts of a picture with significant detail, as can be seen in Fig. 3.

When two successive I frames belong to different shots, the distribution of the intra prediction modes of the two frames will highly differ. Consequently, comparing the intra prediction modes between the consecutive I frames at corresponding positions will reflect the similarity of the frames. However, when there are fast moving objects or camera motion, this approach would lead to false alarms. Instead, the MBs are grouped in sets of 5×5 MBs, named sub-blocks, which are then compared to each other. Now, let S^k be the set of MBs

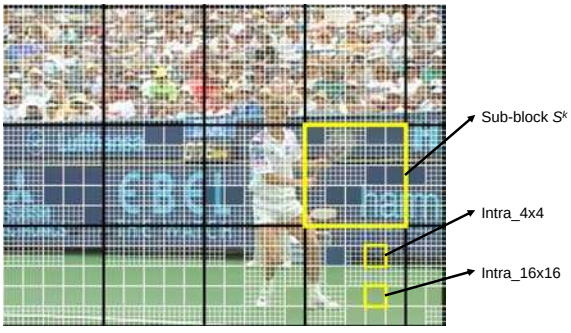


Fig. 3. The distribution of intra prediction modes

included within the k^{th} sub-block of an I frame and i the current and j the previous I frame. The decision function between two consecutive I frames can be defined as follows:

$$\Omega(i) = \frac{1}{\#\text{MB}} \sum_{\forall k} \left| \sum_{l \in S^k} Mode_4 \times 4_i^l - \sum_{l \in S^k} Mode_4 \times 4_j^l \right| \quad (1)$$

If the percentage of blocks with intra and forward temporal prediction in the previous frame is higher than a predefined threshold T_1 and the dissimilarity $\Omega(i)$ between the current frame and its preceding I frame is higher than a second predefined threshold T_2, we declare a shot boundary located, at the current I frame. The values of both thresholds were selected in order to maximize the performance of the algorithm and were set to 80% and 15% respectively.

Shot Boundaries Located at a P or B Frame. P and B frames, in contrast to I frames, use temporal prediction to exploit the similarity between consecutive frames in a shot. In case the current frame is the first frame of a new shot, this frame will have hardly any resemblance to the previously depicted frames. Therefore, the current frame will mainly contain blocks with intra and backward temporal prediction. Blocks in the previous frame, on the other hand, will mostly use intra and forward temporal prediction. Bi-directional temporal prediction will hardly be present in this situation, since this type is only advantageous when the content of the neighboring pictures is resemblant. If the percentage of blocks with intra and forward temporal prediction in the previous frame and the percentage of blocks with intra and backward temporal prediction in the current frame are both higher than the predefined threshold T_1, we declare a shot boundary located at the current P or B frame. This threshold is the same as for I frames as the principle behind the metric is similar.

It is insufficient to take only the percentage of intra coded blocks into account. In case a future depicted frame is already coded and stored in the buffer at the moment the current frame is coded, this future frame can be used as a reference. This reference picture will represent the content of the new shot, which makes the use of backward temporal predicted blocks in the current frame preferable to intra coded MBs.

Generally speaking, the intra mode is only used to code a MB when motion estimation gives no satisfactory results. In H.264/AVC, even if a block can be predicted well, the encoder might prefer intra coding when the block can be better predicted by adjacent pixels instead of temporal prediction. As a result, statistical information for shot boundary detection, based on the percentage of intra coded MBs only, is insufficient to draw a conclusion. By making use of the distribution of the different temporal prediction types, a more accurate detection of shot boundaries can be accomplished.

Summary. Let $\iota(i)$, $\varphi(i)$, $\beta(i)$, and $\delta(i)$ be the number of blocks with intra, forward, backward, and bi-directional temporal prediction, respectively, i and $i-1$ the current and previous frame, and $\#B$ the number of blocks in a frame. Using (1), the detection of abrupt transitions can be summarized as follows:

```
if (f_i is an I frame)
    if ( 1/#B (ι(i−1) + φ(i−1)) > T_1  and  Ω(i) > T_2 )
        {declare a shot boundary}

if (f_i is a P or B frame)
    if ( 1/#B (ι(i−1) + φ(i−1)) > T_1  and  1/#B (ι(i) + β(i)) > T_1 )
        {declare a shot boundary}
```

4.2 Detection of Gradual Changes

Another challenge is the detection of gradual changes as they take place over a variable number of frames and consist of a great variety of special effects. A characteristic, present during most gradual changes, is the increasing amount of intra-coded MBs. The distribution of the percentage of intra-coded MBs in a frame is connected with the duration of the transition. If the transition consists of a few frames, the mutual frame difference is relatively big and most frames will consist of intra-coded MBs. In case the transition is spread out over a longer interval, the resemblance is higher and therefore, a lot of B frames may use bi-directional temporal prediction as well.

To smooth the metric $\Delta(i)$ defined by the percentage of the intra-coded MBs and to diminish the peaks, a filter with Gaussian impulse response is applied. The result can be seen in Fig. 4(a).

In contrast to the detection of abrupt changes, a fixed threshold cannot be applied in the context of gradual changes, since the height of the peaks in this metric is linked to the duration of the transition. Instead, we make use of two variable thresholds T_a and T_b based on characteristics of preceding frames. Within a shot, the frame-to-frame differences are normally lower than during a gradual change. Therefore, the mean and variation of the metric for a number of preceding frames is taken into account to determine the adaptive threshold T_a. Once a frame is found which exceeds this threshold T_a (Fig. 4(b)), the following frames are examined to determine whether or not they also belong to the transition. This is done by comparing each frame to a threshold T_b based on the mean and

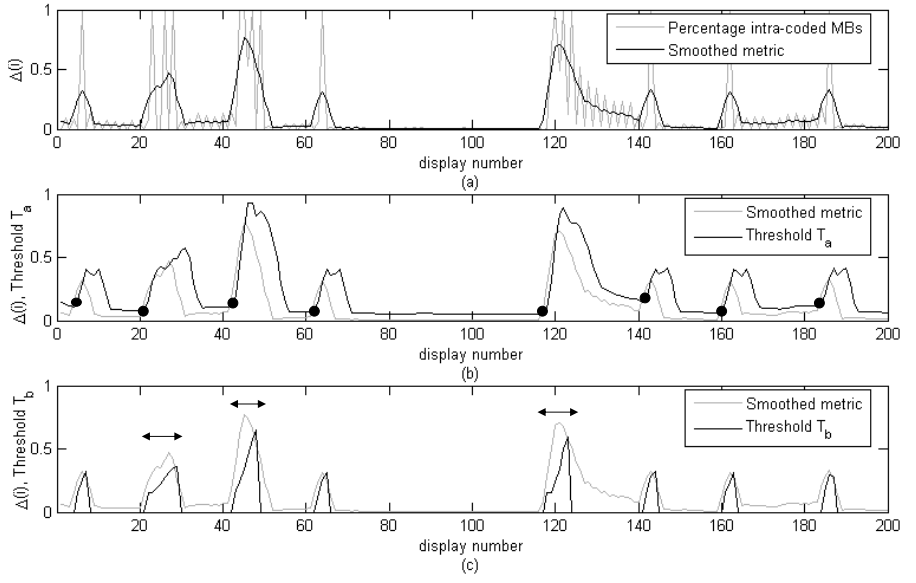


Fig. 4. Gradual changes. (a) Smoothening of the gradual metric, (b) detection of the beginning of a gradual change, (c) detection of the end of a gradual change.

variation of the previous frames belonging to the gradual change (Fig. 4(c)). When the value $\Delta(i)$ for this frame is below the threshold T_b, the end of the gradual change is found. For both thresholds, a lower boundary and a minimal variation are taken into account to avoid small elevations in a smooth area being wrongly considered as a shot boundary. Without this adjustment, a gradual transition would be falsely detected around frame 86. Furthermore, the angles of inclination corresponding to the flanks of the gradual changes are taking into consideration to determine the actual length of the transition. Afterwards, the duration of the obtained transition is examined to remove false alarms, such as abrupt transitions or fixed I frames. The detection of gradual transitions is executed before the detection of abrupt changes to avoid that gradual changes would be falsely considered to be multiple abrupt changes. In Fig. 4, for example, the narrow peaks at frames 63 and 142 correspond to abrupt changes, while the wide peaks around frames 45 and 122 represent gradual changes.

For video sequences coded with former MPEG standards, shot detection algorithms in the compressed domain were not able to distinguish the different types of gradual changes. Since H.264/AVC supports several intra coding types, a difference can be made. In smooth frames, most of the time, MBs using Intra_16×16 or Skipped mode are utilized. By examining the distribution of the MB coding types, a distinction is made between fade ins, fade outs, and other gradual changes.

Nowadays, long term reference pictures belonging to previous shots are seldom used. As the computational power increases tremendously and more intelligent encoding algorithms are developed, these long term reference pictures could be

used in the future to store the backgrounds of recurring scenes. As a result, our algorithm needs to be extended since forward temporal prediction to this long term reference frame can then be used in the first coded frame belonging to a new shot. The display numbers therefore need to be compared to the previous detected shot boundary.

5 Experiments

To evaluate the performance of the proposed algorithm, experiments have been carried out on several kinds of video sequences. Five trailers with a resolution around 848×448 pixels were selected as they are brimming of abrupt and gradual transitions and contain a lot of special effects. "Friends with money" mainly contains shots with lots of moving objects and camera motions alternated with dialogs. "She's the man", "Little miss sunshine", and "Accepted" are all trailers brimming with all kinds of shot changes, variations in light intensity, and motion. Especially "Basic instinct 2" is a challenge, as it is full of motion, gradual changes, et cetera. These sequences were coded with variable as well as with fixed GOP structures in order to evaluate the influence hereof on the algorithm.

5.1 Performance

The evaluation of the proposed algorithm is performed by comparing the results with the ground truth. For this purpose, the "recall" and "precision" ratios based on the number of correct detections ($Detects$), missed detections (MDs), and false alarms (FAs) are applied:

$$Recall = \frac{Detects}{Detects + MDs} \quad Precision = \frac{Detects}{Detects + FAs}$$

In Table 1, the performance of the proposed algorithm is presented for the above mentioned video sequences coded with a variable GOP structure based on the content of the video. This table also depicts the performance for these video sequences coded with a fixed GOP structure described by the regular expression IB(PB)* and an intra period of 20 and 200 frames.

This table shows that the proposed algorithm performs well for video sequences coded with a variable as well as with a fixed GOP structure. The causes of the missed detections and the false alarms are similar in both cases.

For these test results, the major part of the missed detections are caused by long gradual changes, since there is almost no difference between two consecutive frames. This is a problem which most of the shot boundary detection algorithms have to cope with. Furthermore, brief shots containing quite a lot of motion will sometimes be considered as a gradual changes between the previous and the following shot as their characteristics bear resemblance to gradual changes. Consequently, this shot will not be detected.

The false alarms have various reasons. Sudden changes in light intensity, such as lightning, explosions, or camera flashlights often lead to false alarms. This

Table 1. Performance based on Recall (%) and Precision (%) of the algorithm on sequences coded with a variable as well as a fixed GOP structure. A distinction is made between the abrupt (CUT) and the gradual changes (GC).

Test sequences	# original shots		CUT		GC	
	CUT	GC	Precision	Recall	Precision	Recall
Variable GOP structure						
Friends with money	48	1	96.00	100.00	50.00	100.00
She's the man	120	41	95.83	95.83	89.13	100.00
Little miss sunshine	81	24	86.36	93.83	92.00	95.83
Accepted	117	6	94.12	95.73	38.46	83.33
Basic instinct 2	91	47	86.46	91.21	91.49	91.49
Fixed GOP structure: Intra period 20						
Friends with money	48	1	100.00	97.92	50.00	100.00
She's the man	120	41	96.46	90.83	95.24	97.56
Little miss sunshine	81	24	100.00	97.53	81.48	91.67
Accepted	117	6	95.87	99.15	41.67	83.33
Basic instinct 2	91	47	95.18	86.81	97.06	70.21
Fixed GOP structure: Intra period 200						
Friends with money	48	1	100.00	95.83	100.00	100.00
She's the man	120	41	96.61	95.00	88.89	97.56
Little miss sunshine	81	24	95.12	96.30	91.30	87.50
Accepted	117	6	93.60	100.00	60.00	100.00
Basic instinct 2	91	47	92.05	89.01	89.13	87.23

is due to the fact that the current image cannot be predicted from previous reference frames since the luminance highly differs. Afterwards, future frames could use reference frames located before the light intensity change for prediction. However, nowadays, video sequences usually do not consist of a large buffer and therefore do not contain these reference frames. When a shot contains lots of movement, originating from objects or the camera, false alarms will sometimes occur. Due to this motion, successive frames will have less similarity and it will be more difficult for the encoder to find a good prediction. This leads to a lot of intra-coded MBs, and therefore, the structure of the MB type information in successive frames bears resemblance to gradual changes. Experiments have shown that looking at the distribution of the motion vectors does not offer a solution to this problem since the vectors do not always give a good representation of real movement. Here, a trade-off must be made between recall and precision.

6 Conclusion

This paper introduces an algorithm for automatic shot boundary detection on H.264/AVC-compliant video bitstreams. Therefore, a new concept 'Temporal Prediction Types' was presented combining two features available in a compressed bitstream, i.e., the different macroblock types and the corresponding

display numbers of the reference frames. These features can easily be extracted from compressed data, making the decompression of the bitstream unnecessary and thereby avoiding computational overhead. Moreover, the experimental results show that the performance is promising for sequences coded with fixed as well as with variable GOP structures.

Acknowledgements

The research activities as described in this paper were funded by Ghent University, the Interdisciplinary Institute for Broadband Technology (IBBT), the Institute for the Promotion of Innovation by Science and Technology in Flanders (IWT), the Fund for Scientific Research-Flanders (FWO-Flanders), the Belgian Federal Science Policy Office (BFSPO), and the European Union.

References

1. Gargi, U., Kasturi, R., Strayer, S.: Performance Characterization of Video-Shot-Change Detection Methods. IEEE Transactions on Circuits and Systems for Video Technology **10**(1) (2000) 1–13
2. Lelescu, D., Schonfeld, D.: Statistical Sequential Analysis for Real-Time Video Scene Change Detection on Compressed Multimedia Bitstream. IEEE Transactions on Multimedia **5**(1) (2003) 106–117
3. Zhang, H.J., Kankanhalli, A., Smoliar, S.: Automatic Partitioning of Full-Motion Video. Multimedia Systems **1**(1) (1993) 10–28
4. Zabih, R., Miller, J., Mai, K.: A Feature-Based Algorithm for Detecting and Classifying Scene Breaks. In: Proceedings of ACM '95. (1995) 189–200
5. Yeo, B.L., Liu, B.: Rapid Scene Analysis on Compressed Video. IEEE Transactions on Circuits and Systems for Video Technology **5**(6) (1995) 533–544
6. Pei, S.C., Chou, Y.Z.: Efficient MPEG Compressed Video Analysis Using Macroblock Type Information. IEEE Transactions on Multimedia **1**(4) (1999) 321–333
7. De Bruyne, S., De Wolf, K., De Neve, W., Verhoeve, P., Van de Walle, R.: Shot Boundary Detection Using Macroblock Prediction Type Information. In: Proceedings of WIAMIS '06. (2006) 205–208
8. Li, H., Liu, G., Zhang, Z., Li, Y.: Adaptive Scene-Detection Algorithm for VBR Video Stream. IEEE Transactions on Multimedia **6**(4) (2004) 624–633
9. Wiegand, T., Sullivan, G., Bjontegaard, G., Luthra, A.: Overview of the H.264/AVC Video Coding Standard. IEEE Transactions on Circuits and Systems for Video Technology **13**(7) (2003) 560–576
10. Kim, S.M., Byun, J., Won, C.: A Scene Change Detection in H.264/AVC Compression Domain. In: Proceedings of PCM '05. (2005) 1072–1082

Ontology-Based Annotation of Paintings Using Transductive Inference Framework

Marchenko Yelizaveta[1], Chua Tat-Seng[1], and Jain Ramesh[2]

[1] National University, Singapore
{marchenk,chuats}@comp.nus.edu.sg
[2] UC Irvine, USA
jain@ics.uci.edu

Abstract. Domain-specific knowledge of paintings defines a wide range of concepts for annotation and flexible retrieval of paintings. In this work, we employ the ontology of artistic concepts that includes visual (or atomic) concepts at the intermediate level and high-level concepts at the application level. Visual-level concepts include artistic color and brushwork concepts that serve as cues for annotating high-level concepts such as the art periods for paintings. To assign artistic color concepts, we utilize inductive inference method based on probabilistic SVM classification. For brushwork annotation, we employ previously developed transductive inference framework that utilizes multi-expert approach, where individual experts implement transductive inference by exploiting both labeled and unlabelled data. In this paper, we combine the color and brushwork concepts with low-level features and utilize a modification of the transductive inference framework to annotate art period concepts to the paintings collection. Our experiments on annotating art period concepts demonstrate that: a) the use of visual-level concepts significantly improves the accuracy as compared to using low-level features only; and b) the proposed framework out-performs the conventional baseline method.

Keywords: Transductive inference, Multi-expert, Concepts Ontology, Paintings.

1 Introduction

Visual characteristics of paintings such as color, brushwork, and composition constitute a large body of artistic concepts that facilitate expert analysis in the paintings domain. They closely relate to high-level semantic information of painting such as the artist names, painting styles and art periods. These concepts have been used for painting analysis to support applications such as brush-stroke detection and image annotation [3, 6, 9, 12, 13]. Several studies [6, 9] performed automatic brushwork analysis for the annotation of paintings with artist names. These methods directly modeled the artist profile based on low-level features. Such approach yields limited accuracy because of two drawbacks. First, it does not incorporate domain-specific knowledge for the disambiguation of results. Second, since visual-level concepts are not represented explicitly, the introduction of other high-level concepts

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 13 – 23, 2007.
© Springer-Verlag Berlin Heidelberg 2007

in arts domain will require additional training. To alleviate these problems, in our previous work [12], we proposed a framework for ontology-based annotation of paintings where meta-level artistic concepts such as the color and brushwork are introduced as the basis for annotating higher-level concepts such as the periods of art, artist names and painting styles. In this work, we adopt the proposed framework by utilizing the artistic color and brushwork concepts extracted to support annotation of paintings with the concepts of art periods.

For this task, we first perform annotation of paintings with artistic color concepts based on our earlier proposed method [13] that utilizes color theory of Itten [7] similarly to other studies [3]. We next perform the annotation of brushwork concepts by employing the previously developed framework for brushwork annotation using serial combinations of multiple experts [14, 15]. The paper describes our approach on utilizing the color and brushwork concepts in our ontological and transductive inference framework for the annotation of the high-level concepts of art period.

2 Ontology of Artistic Concepts

In our study we employ the ontology of artistic concepts that includes visual, abstract and application concepts as shown in Figure 1. This ontology is based on external Getty's AAT and ULAN ontologies [16]. It has several advantages. First, the explicit assignment of visual and abstract concepts offers more flexibility for paintings annotation and retrieval. Second, the use of domain-specific ontologies within the proposed framework facilitates concept disambiguation and propagation. Lastly, ontology includes retrieval concepts for both expert and novice user groups.

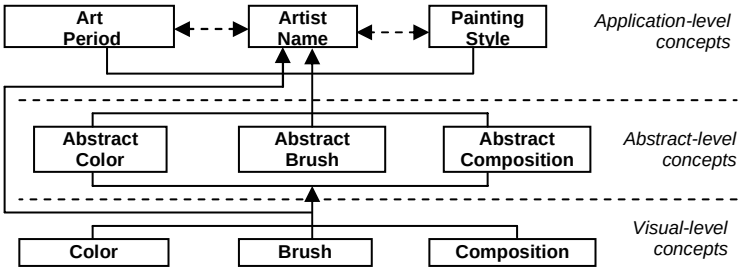


Fig. 1. Three-level ontology of artistic concepts. Double-edged arrows between concepts denote that these concepts are inter-connected.

Concepts of the visual level (atomic concepts) include color, brushwork and composition concepts. In our system, we utilize the visual-level concepts in two ways. First, they represent large vocabulary for retrieval of painting by the expert users. For example, such queries as paintings in *warm* colors, paintings with *temperature* contrast and *impasto* brushwork class are possible. Second, these concepts serve as cues for the annotation of higher-level concepts in abstract and application levels [1, 7]. Abstract-level concepts include concepts defined by artistic theories for the art

experts. Application-level concepts denote the widely used concepts for retrieval by novice users in online galleries such as the artist names, painting styles and periods of art etc.

In this paper, we focus on the annotation of paintings with the concepts of art period. Our collection includes paintings by various artists from *Medieval* and *Modern* periods of art. To perform the annotation, we exploit heuristics available in the domain knowledge. For example, paintings of *Medieval* period often exhibit *primary* palette of colors such as red, blue, *light-dark* color contrasts, *mezzapasta*, *glazing* and *shading* brushwork classes. Paintings of *Modern* art often exhibit *complimentary* colors, *temperature* contrasts and variety of brushwork classes such as *scumbling*, *impasto*, *pointillism*, *divisionism* and *grattage* [1]. To account for such heuristics, we utilize the visual-level concepts as mid-level features to assist in the annotation of paintings with high-level concepts. In this section, we also briefly discuss the visual-level concepts.

2.1 Visual-Level Color Concepts

Itten's theory [7] proposes the mapping between colors and artistic color concepts, and is primarily used by artists. Itten defines twelve fundamental hues and arranges them in color circle. Fundamental hues vary through five levels of intensity and three levels of saturation, thus creating their respective subsets of colors. Fundamental colors are arranged along the equatorial circle of sphere, luminance varies along medians and saturation increases as the radius grows. Itten locates the shades of gray colors in the center of the sphere and white and black colors at the poles of the sphere.

Based on the color sphere, Itten defined color temperatures concepts (*warm*, *cold* and *neutral*), color palette concepts (primary, *complimentary* and *tertiary*) and color contrasts (*complimentary*, *light-dark* and *temperature*). We discussed these concepts in detail in our previous work [13].

2.2 Visual-Level Brushwork Concepts

In our study, we employ eight brushwork classes widely used in Medieval and Modern periods of art. Table 1 summarizes information of brushwork. It demonstrates that our brushwork collection includes mostly stochastic textures. They exhibit a variety of properties such as directionality, contrast, regularity etc. In terms of the spatial homogeneity, we can roughly group brushwork patterns as homogeneous (*mezzapasta* and *pointillism*), weakly homogeneous (*divisionism*) and inhomogeneous (*scumbling*, *shading* and *glazing*).

3 Transductive Inference of Concepts Using Serial Multi-expert Approach

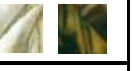
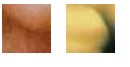
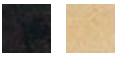
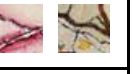
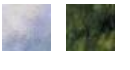
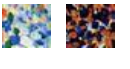
To annotate paintings with artistic concepts, we employ previously developed transductive inference framework. We briefly discuss its major components in this section.

3.1 Serial Multi-expert Approach

The decision process within the serial multi-expert framework starts with all classes and the original dataset including both labeled and unlabelled patterns. It progressively reduces the subset of candidate classes to which a pattern might belong to based on the manually pre-defined decision hierarchy, which guides the experts in splitting the input dataset into individual classes.

We denote the subset of candidate classes as *the target set*. We formalize the reduction of the target size as follows. The expert at the *i-th* level has the input vector (X,S_{i-1}) received from the ancestor node and generates the output vector S_i, where X represents a pattern. S_i represents the set of classes to which the expert of *i-th* level believes the pattern X might belong and the set S_i is a subset of its respective set S_{i-1} $(S_n \subset S_{n-1} \subset S_i \ldots \subset S_0)$. During the annotation process, if the terminal node is reached, then the unlabelled patterns under this node are labeled with a single element of S_i.

Table 1. Visual-level brushwork concepts

Class	Characteristics	Background	Examples
Shading	Depiction of foldings in Medieval Period	Edges and gradients, often directional, intensity contrast, weakly or non-homogeneous	
Glazing	Depiction of nudity/face in Medieval Period	Subset of hues (yellow, red, orange), intensity contrast, gradients, non-homogeneous, may contain edges	
Mezzapasta	Widely used technique in paintings. The color palette used varies with respect to the art period.	Homogeneous, low intensity contrast and small gradients	
Grattage	Depiction of objects and patterns in Fauvism and Expressionism painting styles of Modern Art period	Edges, high gradients, intensity contrast, inhomogeneous	
Scumbling	Depiction of sky, clouds, greenery and atmosphere in various painting styles of Modern art	Soft gradients, low intensity and hue contrast, low directionality, weakly homogeneous	
Impasto	Widely used in Impressionism, Post-impressionism, Pointillism styles of Modern art	Edges, high gradients, often directional, low hue contrast, high intensity contrast	
Pointillism	Often used for depiction of atmosphere/air in Pointillism painting style of Modern art	Medium intensity contrast, medium roughness, no directionality, homogeneous	
Divisionism	Widely used in Pointillism, demonstrates the Color Mixing Principle	High gradients, high roughness, high intensity and hue contrast, no directionality, weakly homogeneous	

We employ Class Set Reduction and Class Set Reevaluation strategies for annotation using the serial multi-expert framework. The Class Set Reduction requires that the experts generate a subset of candidate class labels from the original set of candidate class labels received from the ancestor node. The Class Set Reevaluation extends the intermediate nodes to facilitate additional analysis: if the unlabelled

patterns are assigned labels with high confidence, then these assignments become final and the decision process does not evaluates these patterns further.

3.2 Class Weighted Feature Score

To provide the expert with the feature relevance information, we calculate feature scores with respect to each analyzed class. For this we first calculate tight partitions in the feature space using iterative K-means method. Since the K-means clustering minimizes the intra-cluster distance, the data points within a partition are somewhat close to each other in the feature space and exhibit relatively small variances along some of the feature dimensions. Thus, feature dimension is more likely to be relevant to the partition if the projection of the partition on this dimension has a smaller variance. Second, we employ Chi-square statistics to compare the feature value distributions between this partition and the whole dataset. Intuitively, if the distributions are similar, then the analyzed feature is not representative of the cluster and its Chi-square statistics is comparatively low. We represent the feature distributions using the normalized histograms of each feature in the cluster and the whole dataset. To measure the similarity of distributions, we employ Pearson's Chi-Square test: $X^2=\Sigma(O_i - E_i)^2/E_i$, where we treat the *i-th* histogram bin of the feature distribution in a cluster and the overall dataset as the observed counts O_i, and expected counts E_i respectively. Using the Chi-square statistics we obtain the relevance score of the analyzed feature with respect to a partition. Third, we combine the feature scores of the partitions to calculate the feature scores of the classes. The experts utilize the class weighted feature scores during the model selection step to be discussed in Section 3. 4.

3.3 Individual Experts

For each individual expert, the decision hierarchy predefines its input target set TS_i and output target sets TS_{O1} and TS_{O2}. To implement individual experts, we train probabilistic mixture model GMM using EM algorithm. This model approximates the patterns of TS_i as k clusters in the feature space using parametric Gaussian distributions $G(\mu_1,\Sigma_1)….G(\mu_K,\Sigma_K)$. Next, the expert maximizes the calculated posterior probabilities $p(x_j,G(\mu_i,\Sigma_i))$ to estimate the cluster membership of each pattern x_j. Using this information, the expert performs annotation of the unlabelled patterns using *the cluster purity* measure. We define pure cluster of class X as the cluster in which more than 75% of the labeled patterns are of that class (or a subset of classes). The cluster purity represents the degree to which the calculated cluster contains labels of class X and is defined as $p(c)=N_X/N_{all}$, where N_X and N_{all} denote the number of labeled patterns of class X and the overall number of patterns in cluster c respectively. The expert measures the purity of clusters based on the class labels in its output target sets. The unlabelled patterns that fall in the pure clusters receive the candidate class label of that cluster. The unlabelled patterns in impure clusters are assigned the label of the biggest labeled class in the input target set.

To perform the model selection step, the system first trains several models using varying input parameters. Next, it select the least erroneous model using Vapnik's combined bound [4] as shown in Figure 2. For each trained model we have its

respective hypothesis h, the full sample risk $R(X_{l+u})$, the transduction risk (or test error) $R(X_u)$ and the training error $R(X_l)$. The Vapnik's criterion estimates of the testing error based on training error $R(X_l)$ and on the bounded deviation between the two random variables $R(X_u)$ and $R(X_l)$ around their mean $R(X_{l+u})$.

4 Annotation of Artistic Concepts

To annotate paintings with the concepts of art periods, we perform a three-step procedure. First, we sub-divide paintings in the fixed size blocks and perform iterative K-means clustering of painting blocks using low-level color and texture features. Second, we perform the analysis of visual color concepts using the method to be discussed in Section 4. 1. Since we perform the analysis of color concepts at the level of fixed-size blocks, we employ the majority vote to assign color concepts to clusters. For the annotation of a cluster with brushwork concepts, we utilize low-level color and texture features of a cluster and employ the transductive inference framework (see Section 4. 2). Using a combination of low-level color and texture features and mid-level color and brushwork concepts, we again employ the transductive inference framework as described in Section 3 to perform the annotation of application-level concepts.

Input:
A full sample set X_{l+u} and training sample set X_l,
Feature weighted scores $F_S(L_j)$ for the candidate class labels L_j,
A maximum number of mixture components K,
A set of cut-off thresholds for the feature ranks T_f
Output:
Candidate class labels of the test set X_u
Algorithm:
1. For each cut-off threshold $tf \in T_f$ and number k of mixture components, $2 \le k \le K$, train GMM on X_{l+u} to generate $(K-1)\times|T_f|$ number of models $\{M_{k,tf}\}$;
2. Based on the training set, employ the cluster purity measure to generate a set $\{h_{k,tf}\}$ of $(K-1)\times|T_f|$ hypotheses corresponding to the models;
3. For each hypothesis $\{h_{k,tf}\}$ calculate its training error $R(X_l)$ and its Vapnik combined bound
4. Output a candidate class labels for X_u using $\{h_{k,tf}\}$ with the smallest Vapnik's bound.

Fig. 2. The model selection algorithm

4.1 Visual-Level Color Concepts

For the analysis of color concepts we utilize CIE L*u*v color space. We employ a two-step procedure to assign *warm*, *cold* or *neutral* color temperature concept to a region. First, we model the distribution of various color temperatures within a block. For this, we back-project image colors to the corresponding reference colors in the Itten color space using the following formulae:

$$ref = \arg_{M_c} \min_{1 \le i \le N} dist\,(Rc, Mc\,(i)) \qquad (1)$$

where *dist* denotes the normalized Euclidean distance, R_c denotes the image colors, $Mc(i)$ denotes the reference color i on the Itten's chromatic sphere, and N denotes the

number of Itten colors *(N = 187,* including 5 shades of grey, black and white colors). The feature vector of a block includes the number of pixels of each color temperature concept, color values of dominant colors extracted from 316-color histogram in HSI color space, spatial coherence of block pixels of each color temperature calculated based on a modification of the color coherence vector. Second, the system utilizes probabilistic SVM [18] and winner takes all strategy to assign color temperature concept to each block. Using the same two-step procedure, we classify blocks with respect to *complimentary*, *primary* and *tertiary* color palette concepts.

To calculate color contrast concepts, we represent each block as a set of color pairs based on its dominant colors [2]. Using formula 1, we calculate the corresponding reference colors. Based on the relative location of reference colors on the chromatic color sphere, we calculate the *complimentary*, *temperature* and *light-dark* contrast values. Lastly, we average the contrast values of all color pairs within the blocks to derive the contrast values for each block.

4.2 Visual-Level Brushwork Concepts

In order to employ the transductive inference framework as described in Section 3 to annotate brushwork concepts, we extract color and texture features and derive the decision hierarchy for the annotation.

We employ variety of feature extraction techniques for adequate representation of brushwork concepts [13] such as major colors [10], directional histograms of image edges and gradients, multi-resolution Gabor Texture features [11], wavelet-based features, Hurst coefficient [8] and Zernike moments [17]. We extract these features based on the fixed-size blocks and average their values to calculate one feature vector per cluster. Figure 3 demonstrates the decision hierarchy for brushwork.

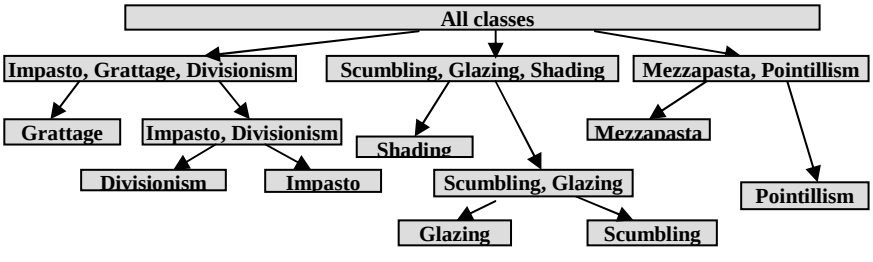


Fig. 3. The decision hierarchy for brushwork annotation

4.3 Application-Level Art Period Concepts

For each image cluster, we now have low-level color and texture features as well as intermediate-level artistic concepts for color and brushwork. We utilize this information to annotate high-level concepts of art periods. Overall, we employ a two-step procedure to perform annotation. First, we annotate the image clusters with high-level concepts. To perform this task, we employ transductive inference framework. However, since our collection includes paintings of only two periods of art, the decision tree has only three nodes: a root node and two leaf nodes. In accord to the

decision tree, the framework employs a single expert that annotates the image clusters with one of the two mutually exclusive concepts. To facilitate feature selection, we calculate class weighted feature scores for periods using the method discussed in Section 3. 2. The framework utilizes feature scores during the model selection step as described in Section 3. 3. Second, we back-project clusters onto their respective paintings and employ the majority vote technique to annotate the art period concept to the whole painting.

5 Experiments

For our experiments, we employ 200 and 700 paintings of various artists and painting styles for training and testing respectively. The testing set includes 120 paintings in Medieval and 580 paintings of Modern period of art. To preserve color and brushwork information, we employ the fixed-size blocks of size 32x32 for the concept analysis.

5.1 Annotation of Visual-Level Color Concepts

To measure the accuracy of labeling with color temperature and color palette concepts we employ 5,000 randomly sampled blocks from the training set. We utilize this dataset to perform training and testing of probabilistic SVM classifiers for annotation of color temperature and color palette concepts respectively. We use 75% of the dataset for training and 25% for testing. We found that we could achieve 91.2% of accuracy in color temperature annotation task and 93.7% in color palette annotation task. We did not evaluate the annotation of blocks with color contrast concepts due to the lack of ground truth, but we have demonstrated its performance for region-based retrieval task [13].

5.2 Annotation of Visual-Level Brushwork Concepts

For this set of experiments, we extract 4880 blocks from 30 paintings of Renaissance, Fauvism, Impressionism, Post-Impressionism, Expressionism and Pointillism painting styles. We randomly select 75% of the dataset for training and use the remaining patterns for testing. Figure 4 demonstrates the distribution of brushwork.

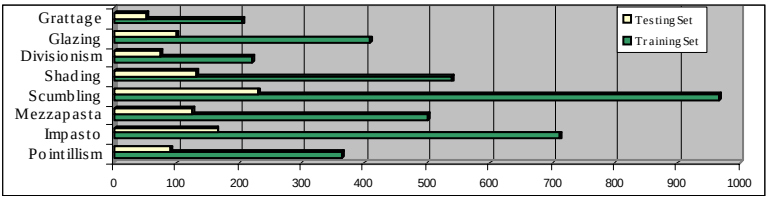


Fig. 4. Distribution of brushwork classes in the training and testing datasets

Table 2 summarizes the performance of the systems in terms of overall annotation accuracy. We employ a single GMM model as the baseline system for our experiments with brushwork.

Table 2. Performance of the systems for brushwork concepts annotation

System	Class Reduction	Class Reevaluation
Baseline	**80.07%**	
Baseline with feature selection	**83.6%**	
Multi-expert with model selection	**93.7%**	**87.45%**

Baseline performs the annotation of the unlabelled instances into the brushwork classes on the basis of pure clusters. It can be viewed as a single expert operating on the full feature set. During our experiments, we found that baseline generates the best results using *K=30* mixture components. To evaluate feature selection, we perform another baseline with feature selection as discussed in Section 3. 2. The proposed multi-expert transductive inference framework achieves higher accuracy due to the several reasons. First, it sequentially disambiguates patterns, which yields high annotation accuracy at the leaf nodes. Second, it employs the model selection step that finds most appropriate number of mixture components as well as the cut-off threshold for the features scores with respect to each individual expert.

5.3 Annotation of Paintings with Art Period Concepts

For our experiments of application-level concept annotation, we perform clustering of blocks from each painting in 60 clusters. The first baseline system (Baseline 1) for our experiments is a binary SVM classification method based on low-level color and texture features. To test the contribution of the visual-level concepts to the overall result, we employ the variation of the baseline system (Baseline 2) that combines visual-level concepts and low-level features with the class weighted feature scores above 0.7. Lastly, we evaluate the proposed transductive inference framework using both low-level features and intermediate-level concepts. Table 3 demonstrates the performance of the systems.

Table 3. Performance of the systems for application-level concepts annotation

System	Accuracy of cluster annotation, %	Accuracy of image annotation, %
Baseline 1	**68.72%**	**81.48%**
Baseline 2	**79.02%**	**93.56%**
Transductive inference with model selection	**86.84%**	**98.71%**

From these results, we draw the following observations. Baseline 2 results in higher accuracy as compared to Baseline 1 system due to the several reasons. First, the use of visual-level concepts facilitates more accurate mapping from feature vectors to the art period concepts. Second, the use of the weighted feature scores results in the reduction of the noise in the feature space. Next, our proposed method achieves even higher accuracy of 98% at the image-level as compared to Baseline 2 because of several improvements. First, the transductive inference yields higher accuracy due to the use of unlabeled data samples. Second, during the model selection

step, the framework finds the parameter values that lead to the least erroneous results in accord to Vapnik's combined bound. Figure 5 illustrates misclassified paintings. All of them belong to Modern art period. However, they all exhibit dark and red colors with large areas of mezzapasta brushwork class similarly to the paintings of Medieval art period.



Fig. 5. Examples of misclassifications by the proposed system

6 Conclusions

In this paper we proposed a framework for ontology-based annotation of paintings with application-level concepts of art period. Within this framework, we utilize domain-specific knowledge to facilitate annotation. Our experimental results demonstrate that the use of meta-level artistic concepts results in higher annotation accuracy and that the proposed framework outperforms conventional classification approach for annotation of high-level concepts. In our future work, we will focus on several tasks. First, we will perform the annotation of paintings with artist names and painting style concepts. Second, we will develop a methodology to share and integrate the concept ontology used in our study with external ontologies. Third, we will extend the proposed framework to utilize external textual descriptions such as concept definitions in external ontologies and WWW textual information.

References

[1] Canaday J. Mainstreams of Modern Art, Saunders College Publishing, 1981.
[2] Chua T.-S., Lim S.-K., Pung H.-K.. "Content-based retrieval of segmented images". *ACM MM*, 211 – 218, 1994.
[3] Corridoni J. M., Del Bimbo A., and Pala P. Retrieval of Paintings Using Effects Induced by Color Features, CAIVD, pp. 2-11, 1998.
[4] El-Yaniv, R., and Gerzon, L. Effective Transductive Learning via PAC-Bayesian Model Selection. *Technical Report CS-2004-05, IIT*, 2004.
[5] Friedman J. H., An overview of predictive learning and function approximation, From *Statistics to Neural Networks*, Springer Verlag, NATO/ASI, 1-61,1994.
[6] Herik, H.J. van den, Postma, E.O. Discovering the Visual Signature of Painters. In *Future Directions for Intelligent Systems and Information Sciences*, 129-147, 2000.
[7] Itten J. The Art of Color, Reinhold Pub. Corp., NY, 1961
[8] Kaplan L. M. and Kuo C.-C. J., Texture roughness analysis and synthesis via extended self-similar (ESS) model, *IEEE Trans. Pattern Anal. Machine Intell*, 1043–1056, 1995.
[9] Li J., Wang J. Z. Studying Digital Imagery of Ancient Paintings by Mixtures of Stochastic Models, *IEEE Trans. on Image Proc*, vol. 13 (3), 2004.

[10] Low W.-C., Chua T.-S., "Color-Based Relevance Feedback for Image Retrieval". IW-MMDBMS 1998, pp.116-123
[11] Manjunath B. S., Ma W. Y., Texture features for browsing and retrieval of image data. *IEEE Trans. Pattern Anal. Machine Intell* (18), 837–842, 1996.
[12] Marchenko Y., Chua T.-S., Aristarkhova I., Jain R. Representation and Retrieval of Paintings based on Art History Concepts. *IEEE Int'l Conf. on Multimedia and Expo (ICME)*, 2004.
[13] Marchenko Y., Chua T.-S., Aristarkhova I., Analysis of paintings using Color Concepts. *IEEE Int'l Conf. on Mm and Expo (ICME)*, 2005.
[14] Marchenko Y., Chua T.-S., Jain R., Semi-supervised Annotation of Brushwork in Painting Domain using Serial Combinations of Multiple Experts, *ACM Multimedia*, 2006.
[15] Marchenko Y., Chua T.-S., Jain R., Transductive Inference Using Multiple Experts for Brushwork Annotation in Paintings Domain, *ACM Multimedia*, 2006.
[16] Paul Getty Trust. Art and Architecture Thesauri and United List of Artist names. 2000. Available at http://www.getty.edu/research/conducting_research/vocabularies/
[17] Teague, M.R. Image Analysis via the General Theory of Moments, *Journal of the Optical Society of America*, 70 (8), 920-930.
[18] Vapnik, V. Estimation of Dependences Based on Empirical Data. Springer Verlag, New York, 1982.

Interactive Visual Object Extraction Based on Belief Propagation

Shiming Xiang[1], Feiping Nie[1], Changshui Zhang[1], and Chunxia Zhang[2]

[1] State Key Laboratory of Intelligent Technology and Systems,
Department of Automation, Tsinghua University, Beijing 100084, China
{xsm, nfp03, zcs}@mail.tsinghau.edu.cn
[2] School of Computer Science, Software School,
Beijing Institute of Technology, Beijing 100081, China
cxzhang@bit.edu.cn

Abstract. Interactive foreground/background segmentation in a static image is a hot topic in image processing. Classical frameworks focus on providing one class label for the user to specify the foreground. This may be not enough in image editing. In this paper, we develop an interactive framework which can allow the user to label multiply foreground objects of interest. Our framework is constructed on belief propagation. The messages about the foreground objects and background are propagated between pixel grids. Finally, each pixel is assigned a class label after finishing the message propagation. Experimental results illustrate the validity of our method. In addition, some applications in color transfer, image completion and motion detection are given in this paper.

1 Introduction

Extracting the foreground objects in static images is one of the most fundamental tasks in image content analysis, object detection, object recognition and image editing. The task can be formulated as an image segmentation problem. In spite of many thoughtful attempts, it is still very difficult to find a general method which can yield good results in a large variety of natural images. The difficulties lie in the complexity of modelling the numerous visual patterns and the intrinsic ambiguity of grouping them to be visual objects.

To reduce the complexity and intrinsic ambiguity, one method is to design interactive frameworks, which can allow the user to specify the objects and the background according to her/his own understanding about the image. In view of image perception, the user specifications about the image give us the visual hints to model and group the visual patterns.

Most existing interactive segmentation frameworks aim at extracting the foreground from the background, and the classical graph mincut is used to solve the optimization problem [1,2]. Such existing frameworks are initially developed to provide the user one label to specify the foreground, although there are more than one objects of interest in the foreground. Naturally, assigning different objects with different labels is desired in many applications, especially in image editing. This is known as the multi-label problem.

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 24–33, 2007.
© Springer-Verlag Berlin Heidelberg 2007

Fig. 1(a) shows an example of image editing. The task is to transfer the color of one rose to the other [3]. If we are given only one label for the foreground, we would have to label the two roses as one object and could only get the segmentation result as illustrated in the upper right panel of Fig. 1(a). Thus we can not achieve our goal (unless we segment the foreground once again). However, if we are given two labels for the foreground, we could separate them from each other (bottom right panel of Fig.1(a)) and directly perform color transfer between them.

In this paper, we develop an interactive framework which can allow the user to label multiple objects. The framework is constructed on belief propagation [4,5], which can naturally deal with the problem with multi-class labels. First, the foreground objects and background specified by the user are modelled by K-means method. Then the information about the objects and background is propagated between pixel grids via belief propagation. After iterations, each pixel receives a belief vector, which records the probabilities of class labels. Finally we assign a label to each pixel according to the maximum a posterior (MAP) criterion. In this way, we solve the task of multi-label problem.

2 Related Works

Early interactive methods include magic wand and intelligent scissors, which are now used as plus tools in Photoshop products. Magic wand starts with user specified points to generate a region with similar color statistics to those of the specified points. Intelligent scissors need the user to label the points near the object boundary. These points are further used as seeds to generate an accurate object boundary.

Current methods [1,2,6,7,8] change the interaction styles. The user can label the background and foreground by dragging the mouse. The main advantage is that it does not require the user to stare at and stroke along the object boundary. In algorithm, the colors of the background and foreground are learned by expectation maximization (EM) [6] or K-means [7] methods. Then the graph mincut algorithm is used to solve the energy minimum problem. Good performances can be achieved in a large variety of natural images [1,2,7].

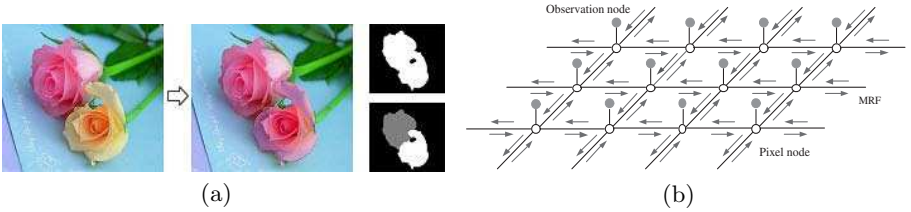


Fig. 1. (a): An example of color transfer between two roses; (b): The graphical model for belief propagation. The shadowed nodes are the observation nodes.

3 Overview of the Framework

The problem we consider can be described as follows. Given an image $\mathcal{I}$ with m pixels, $\mathcal{P} = \{p_1, \cdots, p_m\}$, and $n+1$ point sets, $\mathcal{O}_1, \cdots, \mathcal{O}_n, \mathcal{B}$, each of which consists of the user specified pixels. Each $\mathcal{O}_i$ $(i = 1, \cdots, n)$ corresponds to a foreground object of interest, and $\mathcal{B}$ corresponds to the background. Let $\mathcal{L}$ be an indicator set of $\mathcal{O}_1, \cdots, \mathcal{O}_n$ and $\mathcal{B}$, that is, $\mathcal{L} = \{1, \cdots, n, n+1\}$. Then the task is to assign a class label $l_p \in \mathcal{L}$ to each unlabelled pixel $p \in \mathcal{P}$.

The image is first loaded into the software system we developed. Through the interactive tools, the user labels the foreground objects and the background by dragging the mouse over the image. In this way, we are given $n+1$ point sets $\mathcal{O}_1, \cdots, \mathcal{O}_n, \mathcal{B}$. Then all the left work is finished automatically.

First, for each $\mathcal{O}_i$ $(i = 1, \cdots, n)$ and $\mathcal{B}$, we use K-means to calculate the mean colors of the clusters, and denote them by $\{K_i^{\mathcal{O}}(j)\}$, and $\{K^{\mathcal{B}}(j)\}$. In our experiments, the K-means method is initialized to have 64 clusters. Then, the belief propagation is used to solve the task. Based on the labels of pixels, the objects are finally extracted from the background.

4 Belief Propagation for Visual Object Extraction

4.1 MRF Construction and the Graphical Model

Our task is to assign a unique label to each pixel. To solve the pixel-level task, we model the image as a Markov random field (MRF), where each pixel is treated as a node and each node is considered to connect with its spatial neighbors. We assume that the labels should be piecewise smooth and the labelling should also fit to the learned mean color models. This MRF formulation for our task yields the following energy minimization problem:

$$E(l) = \sum_{(p,q)\in\mathcal{E}} V(l_p, l_q) + \sum_{p\in\mathcal{P}} D_p(l_p) \tag{1}$$

where $\mathcal{E}$ includes all pairs of neighbors in the MRF, $V(l_p, l_q)$ is the cost of assigning l_p and l_q to two neighbors p and q, and $D_p(l_p)$ is the data cost associated to the mean color models, which can be viewed as a likelihood cost.

Eq. (1) is a discrete optimization problem which may be NP hard. Here we relax it as a probabilistic inference problem and use belief propagation [8,9] approach to solve it. To be simplified, for each pixel we consider its spatial neighbors with four-connectivity. Fig. 1(b) shows the graph, in which an observation node is attached to each pixel to transfer the observation information.

In terms of MRF, a clique of this graphical model contains a node and its four neighbors. The description and analysis on this model with pairwise cliques of belief propagation become more specific and simpler [4]. In this way, finding a labelling with minimum energy in the MRF is to obtain a MAP estimation.

4.2 Message Update Rules

The belief propagation algorithm can be run in an iterative way. At each iteration, each node not only receives a message from each of its four neighbors, but also sends a message to each of them. We denote by m_{pq}^t the message that node p sends to node q at time t, by D_p the message from the observation node, by b_p the belief at p after T iterations. Then the max-product update rules in our system can be described as follows:

$$m_{pq}^t(l_q) \leftarrow \alpha \max_{l_p}(V_{pq}(l_p, l_q) \cdot D_p(l_p) \cdot \prod_{s \in \mathcal{N}} m_{sp}^{t-1}(l_p)) \tag{2}$$

$$b_p(l_p) \leftarrow \alpha \cdot D_p(l_p) \cdot \prod_{s \in \mathcal{N}(p)} m_{sp}^T(l_p) \tag{3}$$

where α denotes a normalizing constant, and $\mathcal{N} = \mathcal{N}(p) - \{q\}$.

The equivalent computation can be implemented with negative log probabilities. Thus, eq. (2) and (3) can be rewritten as follows:

$$m_{pq}^t(l_q) = \min_{l_p}(V_{pq}(l_p, l_q) + D_p(l_p) + \sum_{s \in \mathcal{N}} m_{sp}^{t-1}(l_p)) \tag{4}$$

$$b_p(l_p) = D_p(l_p) + \sum_{s \in \mathcal{N}(p)} m_{sp}^T(l_p) \tag{5}$$

After iterations, the label l_p^* that minimizes $b_p(l_p)$ is finally selected as the optimal assignment of pixel p.

4.3 Computing Messages

Each node p is only assigned a unique label l_p, which indicates that its associated pixel belongs to the l_p-th visual entity (object/background). Thus the data cost $D_p(l_p)$ can be calculated as the minimum color distance [7]. We have

$$d(p, l_p) = \begin{cases} \min\limits_j \|C(p) - K_{l_p}^{\mathcal{O}}(j)\| & l_p = 1, \cdots, n \\ \min\limits_j \|C(p) - K^{\mathcal{B}}(j)\| & l_p = n+1 \end{cases}$$

where $C(p)$ is the color of p. Further, we normalize the distances to keep them into a same scale:

$$\begin{cases} D_p(i) = 0; & D_p(l_p) = \infty, & l_p = 1, \cdots, n+1,\ l_p \neq i; & \forall\, p \in \mathcal{O}_i \\ D_p(n+1) = 0; & D_p(l_p) = \infty, & l_p = 1, \cdots, n; & \forall\, p \in \mathcal{B} \\ D_p(l_P) = \frac{d(p,l_p)}{d(p,1)+\cdots+d(p,n+1)}, & & l_p = 1, \cdots, n+1; & \forall\, p \in \mathcal{U} \end{cases} \tag{6}$$

here $\mathcal{U} = \mathcal{P} - (\mathcal{O}_1 \cup \cdots \cup \mathcal{O}_n \cup \mathcal{B})$ includes the unlabelled pixels. We can see that user specified pixels are all hardly constrained. The infinite distance can set to be the maximum distance in RGB color space. In this paper, we set ∞ to be 2.

$V(l_p, l_q)$ is used to represent the prior energy (cost) due to the discontinuity at the object boundaries. According to the assumption of piecewise constants, we define it as the following function:

$$V(l_p, l_q) = \begin{cases} 0 & \text{if } l_p = l_q \\ d & \text{otherwise} \end{cases} \tag{7}$$

where d is a constant to punish a label jumping. It is calculated as the standard deviation of all the distances $\{d(p, l_p)\}_{l_p=1:n+1}^{p\in\mathcal{P}}$.

4.4 Updating Messages

The grid graph shown in Fig. 1(b) is constructed with four-connectivity. Thus we can treat it as a bipartite graph. In this way, the belief propagation can be alternatively performed on two subsets of nodes. Let $\mathcal{P} = \mathcal{A} \cup \mathcal{B}$ and $\mathcal{A} \cap \mathcal{B} = \emptyset$. The message can be updated as follows:

$$m_{pq}^t(l_p) = \begin{cases} m_{pq}^t(l_q) & \text{if } p \in A \text{ (if } p \in B) \\ m_{pq}^{t-1}(l_q) & \text{otherwise} \end{cases} \tag{8}$$

Note that the message should be normalized by α when updating the message in max-product algorithm. We omit the term $-log(\alpha)$ in the updating rules in min-sum algorithm. But correspondingly, the message should also be normalized. In a negative log probability framework, the normalization is to perform a zero-mean centralization for the $n + 1$ components of each message vector.

4.5 Coarse-to-Fine Performance

The belief propagation based on rules (2) and (3) or (4) and (5) needs to iteratively update the messages in an iterative way. The process includes receiving, computing and delivering messages. In the whole graph, every node must wait its neighbors when treating the messages. Actually, only when every node has computed the messages, the messages can then be delivered among neighbors. Thus the messages are treated in a synchronous way. As a result, it may take many iteration times to deliver them to a far distance. Here we use a coarse-to-fine strategy to speed up the perfromance.

We first construct a pyramid with a granularity of 2×2 pixels [5]. The nodes in each level are connected into a graph, also according to four-connectivity.

Note that it is unnecessary to construct an image pyramid via traditional down-sampling techniques to calculate the data costs in different levels. Only in the zero-th level the computation of data costs needs the mean color models of $n+1$ visual entities (objects and background). For a node p in the j-th level, its associated data cost is calculated as follows:

$$D_p^{(j)}(l_p) = \tfrac{1}{4}(D_{p_1}^{(j-1)}(l_p) + D_{p_2}^{(j-1)}(l_p) + D_{p_3}^{(j-1)}(l_p) + D_{p_4}^{(j-1)}(l_p)) \tag{9}$$

where p_1, p_2, p_3 and p_4 are its four father nodes in the $(j-1)$-th level.

The coarse-to-fine computation is started in the coarsest level, and all the messages are initialized to zero. After several iterations, the messages at each node in the j-th level will be equally delivered to its four father nodes in the $(j-1)$-th level. The messages will be delivered to the zero-th level. After they are propagated in the zero-th level, the belief vectors are finally calculated.

4.6 The Algorithm

The steps of object extraction can be summarized as follows:

Algorithm: Object extraction via belief propagation
Input: Color image $\mathcal{I}$, $n+1$ pixel sets $\mathcal{B}$ and $\mathcal{O}_1, \cdots, \mathcal{O}_n$, number of coarse-to-fine levels J, and iteration times T
Output: Labels of each pixel $p \in \mathcal{P}(=\mathcal{I})$

(1) Learn $\{K^{\mathcal{B}}(j)\}$, $\{K_i^{\mathcal{O}}(j)\}$, $i = 1, \cdots, n$; $j = 1, \cdots, 64$
(2) Calculate $D_p(l_p)$, $l_p = 1, \cdots, n+1$; $p \in \mathcal{P}$
(3) Calculate $D_p^{(j)}(l_p)$, $l_p = 1, \cdots, n+1$; $j = 1, \cdots, J-1$; $p \in \mathcal{P}$
(4) $m_{pq}^{(J-1),0}(l_q) \longleftarrow 0$, $(p,q) \in \mathcal{E}$; $l_p = 1, \cdots, n+1$
(5) for $j = J-1$ to 0
(6) for $t = 0$ to T
(7) for each node p
(8) Calculate message $m_{pq}^{(j),t}(l_q)$, according to eq. (4)
(9) end
(10) end
(11) Copy messages to the $(j-1)$-th level
(12) end
(13) Calculate $b_p(l_p)$, $l_p = 1, \cdots, n+1$; $p \in \mathcal{P}$, according to eq. (5)
(14) $l_p \longleftarrow \arg\min_i \{b_p(i)\}$, $p \in \mathcal{P}$

In the above algorithm, $m_{pq}^{(j),t}$ denotes the message that node p sends to node q in the j-th level at time t. The label assignment in step (14) is equivalent to a MAP in max-product algorithm.

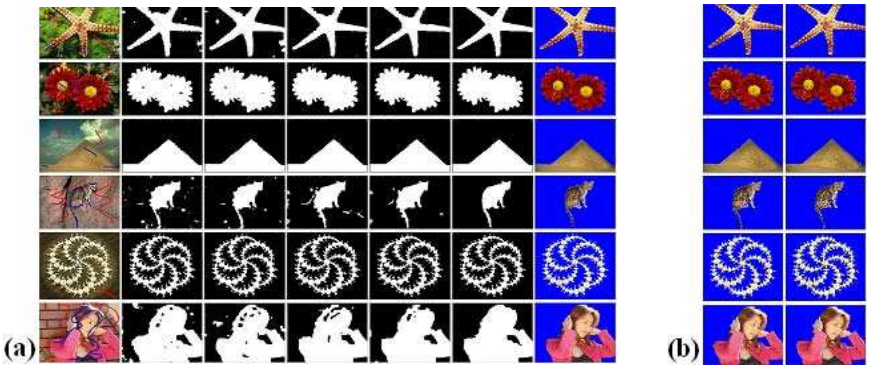


Fig. 2. (a): Results with no coarse-to-fine performance; (b): Results of coarse-to-fine performance with two and three coarse-to-fine levels

5 Results, Comparisons and Applications

We evaluated the algorithm on a variety of different natural images. Here some experimental results are first reported. Then we compare our method with graph

cut and label propagation [10]. Finally, we illustrate some applications in color transfer, image completion, and motion detection.

5.1 Performances of Belief Propagation

Our method has two integer parameters, T and J. Fig.2(a) shows the results of different iterations in six examples, with no coarse-to-fine performance, i.e., $J = 0$. The first column demonstrates the original images with user specified foreground and background strokes. From the second to the sixth columns are the results with $T = 10, 20, 30, 40, 50$, respectively. The extracted foreground objects are shown in the last column. Generally, after about forty times of iterations, we can get satisfactory results. Fig.2(b) shows the results of coarse-to-fine performance with $J = 1$ (the first) and $J = 2$ (the second). In these experiments, we fix T to be 20. As can be seen, we can obtain good results with two-level performance. Fig. 3 shows three examples with multi-class labels. In the first two images, the user specified three visual objects, while in the third image the user specified six visual objets (textures). The segmented results with $J = 2$ and $T = 20$ are illustrated in the second column. All the visual entities are accurately extracted from the images.

For an image with 481×321 pixels, belief propagation needs about 1.8s in case of $J = 0$ and $T = 50$ in a PC with with 1.7GHz CPU and 512 RAM. In case of $J = 2$ and $T = 20$, it needs about 0.8s.

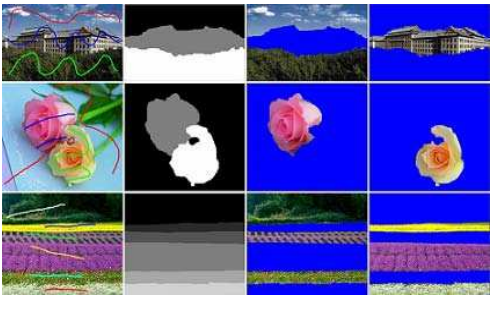


Fig. 3. Results of three examples with multi-labels

5.2 Comparisons

We first compare our approach with the graph mincut algorithm. The background and foreground labelled by the user are respectively learned by EM and K-means method. The Gaussian mixture model in EM algorithm has 5 components. In these methods, eq. (7) is used to calculate $V(l_p, l_q)$. Fig. 4 gives five examples for comparison. In each group, the first column shows the original image with user strokes. The results obtained by graph cut with EM, by graph cut with K-means, and by our method are shown in the second to the forth columns, respectively. In experiments, we also take $J = 2$ and $T = 20$. Comparative work

shows that our performance of belief propagation generates more details at the object boundaries.

For an image with 481×321 pixels, the graph mincut needs about 0.1s in a PC with 1.7GHz CPU and 512 RAM.

In view of machine learning, the class labels of the pixels labelled by the user are all known. The task is to infer the class labels of unlabelled pixels in the image. This is a typical transductive learning problem. Many semi-supervised learning methods are proposed to deal with this task. Here we use Zhou's regularization method based on data manifold [10], which can be viewed as a special label propagation method [11]. The neighborhood of each pixel is defined as a 5×5 patch with its center at the pixel. Fig. 5 shows two examples. Generally, semi-supervised learning method needs more labelled data points to get good results. For an image with 241×161 pixels, Zhou's method needs about 15.0s on a PC with 1.7GHz CPU and 512 RAM, among which 95% time is spent on the construction of the weight matrix.

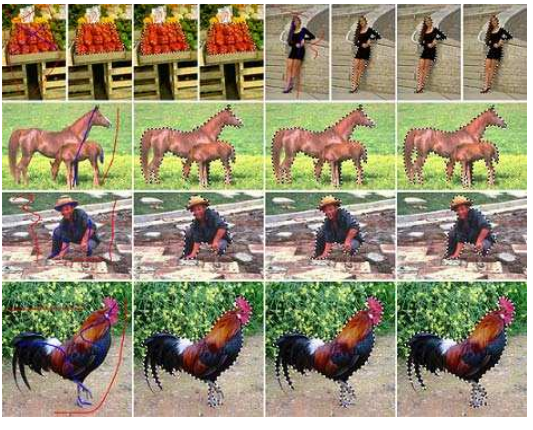


Fig. 4. In each group, the second to the fourth columns show the results obtained by the graph cut with EM, the graph cut with K-means and our method, respectively

5.3 Applications

This subsection introduces the applications of our method in color transfer, image completion with texture synthesis and motion detection.

The original goal of color transfer is to transfer the color of the source image to the destination image such that the latter looks like the former in color appearance. Here we develop the performance to transfer between two visual objects in an image. Fig.6(a) shows two examples. In each group, the middle column shows the segmented result obtained by our method. Based on these results, the color transfer algorithm [3] is used between the two kinds of roses. The transferred results are demonstrated in the third column in each group. We can see that the color is faithfully transferred.



Fig. 5. In each group, the second column illustrates the result obtained by Zhou's method, and the third shows the result with our method

It is worth pointing out that the interactive framework with two-class labels can not directly solve this task. Actually, only one class label is not enough for us to distinguish between those two kinds of roses.

Fig.6(b) shows two examples of image completion with texture synthesis. First, the image is segmented via our interactive object extraction system. Then a texture ensemble with 300 "L" shapes [12] is constructed from the background. Each "L" is a half of a patch with 5×5 pixels. Finally, Wei's method [12] is used to fill the area of the foreground.

Many existing motion detection algorithms can only be used in the occasions where the background is static. Here we extend our method to detect motions in videos where the background is not static. Fig.6(c) shows an example. We only label the first frame. Then the background and foreground are learned by K-means method. Based on the learned mean color models, the belief propagation is used to detect the motion in the second frame. After the foreground and background are extracted, they are learned again by K-means method to update the mean color models. Then we use the updated models to detect the third frame, and so on. The limit of this approach is that colors of the background and the objects should not change dramatically.



Fig. 6. (a): Two examples of color transfer between objects in an image; (b): Two examples of texture transfer; (c): Results of motion detection

6 Summary and Conclusion

We have proposed and demonstrated an iterative framework to extract the foreground objects in an image. Our framework is constructed on belief propagation, which can be naturally used to treat the tasks with multi-class labels. As a result, we provide a mechanism for the user to specify more than one objects of interest in the foreground. This may be very useful in image editing as shown in the examples in color transfer.

Our method is based on the mean color models of user specified foreground and background. When the background and the foreground or foreground objects have similar color distributions, the framework would give us results with noises. This is still an open problem. How to learn more visual hints from the user strokes on the image is a challenging problem. Only using the color information is not enough to model the background and foreground which have similar colors. In the future, we will add the texture information to analyze this problem. For another future work, we will compare our method with the multiway min-cut algorithm [13].

Acknowledgements

This work is supported by the Projection (60475001) of the National Nature Science Foundation of China.

References

1. Boykov, Y. Y., Jolly M. P.: Interactive graph cuts for optimal boundary & region segmentation of objects in N-D images. In: International conference on Computer Vision (ICCV). Vancouver, Canada (2001) 105–112
2. Blake, A., Rother, C., Brown, M., Perez, P., Torr, P.: Interactive image segmentation using an adaptive GMMRF model. In: ECCV. Prague, Czech (2004) 428–441
3. Reinhard, E., Ashikhmin, M., Gooch, B., Shirley, p.: Color transfer between images. IEEE Computer Graphics and Applications. **21** (2001) 34–41
4. Weiss, Y., Freeman, W. T.: On the optimality of solutions of the max-product belief propagation algorithm in arebitrary graphs. IEEE Trans. on Information Theory. **47** (2001) 723–735
5. Felzenszwalb, P. F., Huttenlocher, D. P.: Efficient belief propagation for early vision. In: CVPR. Washington DC, USA (2004) 261–268
6. Rother, C., Kolmogorov, V., Blake, A.: "GrabCut" — interactive foreground extraction using iterated graph cuts. In: SIGGRAPH. Los Angeles, (2004) 309–314
7. Li, Y., Sun, J., Tang, C. K., Shum, H. Y.: Lazy snapping. In: SIGGRAPH. Los Angeles , USA (2004) 303–307
8. Sun, J., Yuan, L., Jia, J. Y., Shum, H. Y.: Image completion with structure propagation. In: SIGGRAPH. Los Angeles, USA (2005) 861–868
9. Wang, J., Cohen, M.F.: An iterative optimization approach for unified image segmentation and matting. In: ICCV. Beijing, China (2005) 936–943
10. Zhou, D. Y., Weston, J., Gretton, A., et al: Learning with local and global consistency. Advances in NIPS 16, MIT Press, Cambridge, USA (2004)
11. Zhu, X. J., Ghahramani, Z.: Semi-supervised learning using Gaussian fields and harmonic functions. In: ICML. Washington DC, USA (2005) 912–919
12. Wei, L. Y., Levoy, M.: Fast texture synthesis using tree-structured vector quantization. In: SIGGRAPH. New Orleans, USA (2000) 479–488
13. Birchfield, S., Tomasi, C.: Multiway cut for stereo and motion with slanted surfaces. In: ICCV. Corfu, Greece (1999) 489–495

Modeling Modifications of Multimedia Learning Resources Using Ontology-Based Representations

Marek Meyer[1], Sonja Bergstraesser[2], Birgit Zimmermann[1], Christoph Rensing[2], and Ralf Steinmetz[2]

[1] SAP AG, SAP Research CEC Darmstadt,
Bleichstr. 8, 64283 Darmstadt, Germany
{marek.meyer, birgit.zimmermann}@sap.com
[2] KOM Multimedia Communications Lab, Darmstadt University of Technology,
Merckstr. 25, 64283 Darmstadt, Germany
{bergstraesser, rensing, steinmetz}@kom.tu-darmstadt.de

Abstract. Repurposing of multimedia-based Learning Resources is an important issue in E-Learning, as economic success of content production depends on how intensively content is used. Repurposing does not only mean reuse "as is", but also comprises modifications of the contents to suit a different learning or teaching context, as well as reuse of fragments of a large Learning Resource. This paper introduces a method for modeling multimedia content modifications based on an ontology-based content representation. A theoretical background for modeling modifications of multimedia contents independent of the particular format is provided. Also a practical implementation is presented and discussed.

1 Introduction

Reusability is an important concept in the E-Learning community. Reusing existing E-Learning contents saves costs and also enables to benefit from the knowledge of other domain experts. For Web-Based Trainings (WBT) the Shareable Content Object Reference Model (SCORM) is the most common exchange format, which enables reuse of WBTs in different systems [1]. But reuse does not only consist of reuse "as is", but also comprises repurposing. Repurposing means to adapt a Learning Resource to a new learning or teaching context. 15 different relevant adaptations for Learning Resources have been identified in a user survey [2]. These adaptations can be broken down into several different fine-granular modifications.

There are some approaches, such as adaptive Learning Resources, single source publishing or layout templates (e.g. Cascading Style Sheets), that facilitate the adaptation to a few well-known scenarios; most contents though are and probably will be available only in non-adaptive form.

However, SCORM is only one format, but there are others, as well. Also, even if SCORM is used, different formats, such as HTML, XML or Flash, may

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 34–43, 2007.
© Springer-Verlag Berlin Heidelberg 2007

be used for the actual contents. Developing tools for repurposing is therefore a difficult and complex task. Hence, it is not advisable to develop a new tool for each adaptation and each format combination completely from scratch. Instead, frequently used functionality should be moved into a framework, which enables the developer of a repurposing tool to focus on the adaptation itself, instead on file and format handling details. Such a repurposing framework is described in [3]. This framework comprises a content ontology that contains concepts of content elements that are part of Learning Resources [4]. These concepts are independent from the particular format. In addition a Learning Resource Content Representation (LRCR) is described which is a graph representation of a Learning Resource's content where concepts defined in the ontology are instantiated to describe each node in the graph. But an abstract content representation is not sufficient to support the development of repurposing tools. Modifications have to be modeled also in a format-independent way for completely outsourcing format-specific methods. In this paper, modeling efforts for content modifications are discussed. Modifications of Learning Resource content are considered regarding two aspects: a theoretical approach and an actual implementation.

The need for modeling content modifications is illustrated by a practical use case: Consider a user Alice who is responsible for advanced training in her company. She is asked by her management to provide a course on the fundamentals of accounting to some employees. Because her budget is low, she decides not to produce a new course, but to buy an existing one from Bob. Unfortunately, this course does not comply to the corporate design of Alice's company. Furthermore, the terminology of Bob's course is partly unknown to Alice's target group and needs to be replaced. Alice examines the obtained course and observes that it is a SCORM package that contains about 100 separate HTML documents. Each of these pages has a defined background and text color and Bob's company logo in the upper right corner. Now, Alice has to open each of these 100 HTML documents to change the background color, replace Bob's logo by her company's logo, move the logo to the upper left corner and replace unsuitable terminology by her own terms. This scenario motivates the use of an adaptation tool, which enables Alice to adapt the layout and terminology of all documents of a Learning Resource at once and more easily. But two weeks later, Alice notices that some employees have not yet participated in the online training, because they work in the field and do not have access to an online Learning Management System most of the time. They would rather prefer learning the contents while traveling by train to their customers. Fortunately, Bob also offers a Microsoft Word document which describes the fundamentals of accounting. But again, layout and terminology do not suit Alice's company. Hence, she opens Microsoft Word for changing layout and terminology manually. Would not it be great if Alice could use only one adaptation tool for the adaptation of different document types? Changing layout, replacing unsuited terminology, translating a document - from an end user's point of view all these adaptations are always the same, no matter which underlying format is used. Thus, Alice desires one tool for performing adaptations of all her Learning Resources.

And that is what this paper deals with - modifications of multimedia Learning Resources are modeled in a format-abstracted manner for supporting the development of format-independent adaptation tools.

This paper is structured as follows. Related work is presented in section 2 before the Learning Resource Content Representation is explained in section 3. Section 4 focuses on the modeling of content modifications. In section 5, the modification model is illuminated in the context of a practical implementation of an adaptation tool.

2 Related Work

The model driven architecture (MDA) approach separates application logic from underlying platform technology [5]. Platform independent models document the behavior of an application separated from the technology-specific implementation. Model transformation in the sense of the MDA approach can be seen as an application of a graph transformation [6]. This approach has found a large community and is used in different applications. Beside the different scenario, the idea to abstract from the implementation and specify generic models is also the motivation behind the approach presented here. It abstracts from format-specific resources and format-specific modifications of these resources by generating a resource model and by modeling the modifications which can be performed on the resources.

The ALOCoM framework [7] is an ontology based framework to enable reuse of Learning Objects. The framework is focused on slide presentations. A slide presentation is disaggregated into its components and mapped to a Java Object Model. Out of the Java Object Model a RDF representation is generated and stored in a Learning Object Repository. Components are reused by copying existing components into a new slide presentation. This scenario allows the reuse of complete slide presentations, and of parts of these slide presentations, e.g. one image. The generated slide presentation uses default presentation styles. The format can be chosen out of a list of supported formats. The major difference to this approach is that ALOCoM converts Learning Resources into an intermediate format and transforms it back into another format for reuse. This may cause a loss of information. Also, the ALOCoM approach requires modifications of the contents to happen in the source or target format.

Kashyap and Shklar [8] propose an RDF model based approach to adapt content resources for different devices. In their work they use a representation of the features of the different devices and components which represent the content resources. A XML resource can be adapted to the different devices using device-specific style sheets. Depending on the device which is requesting content resources an appropriate style sheet can be generated based on the information in the RDF model. No library or collection is needed, containing specific style sheets for all the possible requirements a device might have. This approach follows the idea to uncouple information from presentation and to adapt certain properties of a content resource; it focuses on web applications.

3 Content Representation

To abstract the user from the details of a Learning Resource, a representation of the processed Learning Resource is needed. This representation has to deal with all the elements a Learning Resource may consist of. It needs to be able to deliver the information about the Learning Resource, which is needed to manage the Repurposing process. Hence, a mapping from the Learning Resource into a model which can provide all the required information is needed. Beside these two main requirements there exist several others [4]. The Resource Description Framework (RDF) is used for the content representation of the Learning Resource, because it fulfils all the identified requirements.

As a base for this model a conceptualization of a Learning Resource and the parts it consists of is required. The approach presented here uses an ontology for the conceptualization. In a repurposing process resources in multiple formats are involved, so it must be possible to find an instantiation of a concept defined in the ontology for all of them. A Content Ontology (CO) for describing Learning Resources has been developed. Details about the development of the Content Ontology and the corresponding Learning Resource Content Representation (LRCR) can be found in [4]. The LRCR is based on the representation of the structure of the Learning Resource. A separation of concerns has been realized by distinguishing between structural information and semantic information. The structural concepts can be used to describe the type of elements and also their relations and order. Additional semantic information can be added to the content representation for making the meaning of elements explicit. The meaning of elements can be marked with concepts at different levels of detail. For example a concrete description of an element is an example. A more abstract description of the same element is a SubjectEntity, an Entity which belongs to a certain subject. The level of detail in which a certain element is described depends on the information which is needed about this element and the analysis which is used to identify the element. Attributes and additional information about an element, such as information given in the metadata, are also included in the content representation (see Fig. 1).

By using the concepts defined in the Content Ontology a content representation of a Learning Resource can be build. This representation includes all information which is required in a Repurposing process and enables a view for the user which is uncoupled from all the details of the Learning Resource, e.g. in construction and formats.

4 Modeling Modifications

In the previous section, a format-independent, ontology-supported Learning Resource Content Representation has been introduced. That content representation is static; changes of the contents cannot be specified within the scope of the LRCR. Therefore, another model for content modifications has to be provided. This section deals with format-independent modeling of modifications of multimedia Learning Resources.

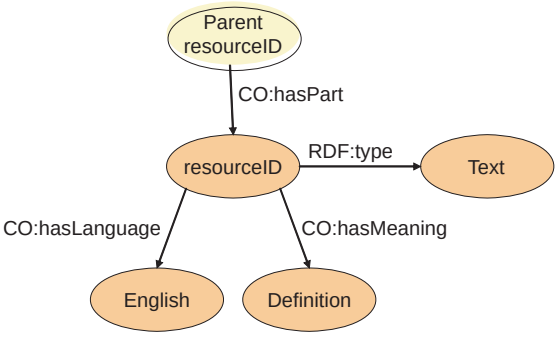


Fig. 1. Example for a LRCR clipping

4.1 Granularity of Modifications

An important design decision is the granularity of modifications. Is, for example, the replacement of a corporate design a single modification or a combination of several modifications? Zimmermann et al. have identified a structure of adaptation processes, which is helpful for the consideration of granularity [9]. On the most general layer, whole adaptation processes are resident, e.g. the adaptation to a different corporate design. An adaptation process divides into several process fragments. Process fragments are composed of adaptation functions, which may either read or modify the contents of a Learning Resource. Which of these granularity levels is best suited for modeling of content modifications?

The goal of modification modeling is to provide an abstraction layer for separating the concerns of repurposing tools and the format-specific content modification methods. Also, reuse of modifications, which are implemented once and reused for several repurposing applications, is a central motivation. In this respect, adaptation processes are too large to be reused easily and often. Process fragments are also application dependent, may rely on information from other process fragments, and sometimes comprise interaction with a user. They are also reused rarely. Adaptation functions, finally, are reusable for multiple process fragments, require no user interaction and need only a manageable amount of parameters to work. Therefore, content modifications are best modeled at the granularity of adaptation functions - restricted to those adaptation functions which cause changes of the content. These modifications are mainly insertion, deletion, replacement and rearrangement of elements, as well as changes of attributes and relations.

4.2 Theoretical Approach

The Learning Resource Content Representation is a graph and is considered to be a mapping of the whole contents, containing the information which is relevant for performing adaptations. Modifications at the granularity of adaptation functions produce only delimited local changes of the Learning Resource Content

Representation. These changes can be expressed as graph operations. Consider there is a Learning Resource r in which one logo should be replaced by another one. If the Learning Resource consists of HTML documents, a logo is usually embedded by using a reference to the image file, which contains the logo. Replacing an image in HTML documents requires only changing the image reference. For other formats (e.g. Microsoft Word), images are physically embedded in documents; hence a replacement works different. Let H be the set of all valid HTML documents and W the set of all valid Microsoft Word documents.

$$exchangeLogo_H : r_1 \mapsto r_1' \quad |r_1, r_1' \in H$$
$$exchangeLogo_W : r_2 \mapsto r_2' \quad |r_2, r_2' \in W$$

And for the general case:

$$exchangeLogo_F : r \mapsto r' \quad |r, r' \in F, mod \in M$$

where r is a document from a given format space F and mod is a modification out of the set of all modifications M.

Consider the projection of Learning Resource r into the Learning Resource Content Representation $r \mapsto \varphi(r)$, where φ is the projection function from the document format space F into the abstract LRCR space A. The modification from the previous example can now be observed in the LRCR space. The function, which modifies $\varphi(r)$ into $\varphi(r')$, is called mod_φ and represents an abstract modification of the Learning Resource content.

This algebra helps developing content adaptation tools. Adaptations have no longer to be implemented directly as format-specific methods. Instead, an adaptation tool analyzes $\varphi(r)$ (the LRCR) and specifies adaptations as a concatenation of modifications mod_φ . Each modification mod is transformed by an underlying layer into a format-specific modification mod_F. This transformation from LRCR space into the actual document format space is also called interpretation of an abstract modification. Fig. 2 illustrates these transformations.

5 Implementation

A repurposing tool for SCORM-based Learning Resources has been developed as part of the Content Sharing project. This repurposing tool is built upon the generic framework for format-independent content modifications, which implements a LRCR and provides an interface for executing abstract modifications as they have been sketched in the previous section. The whole repurposing tool has been implemented in Java (J2SE 1.4.2). The next section describes how abstraction modifications have been realized in practice.

5.1 Implementation of Modifications

First of all, it has to be distinguished between modification types, which are classes of modifications (e.g. "deletion of an element") and modification instances, which are actual modification requests at run-time (e.g. "delete element 1234").

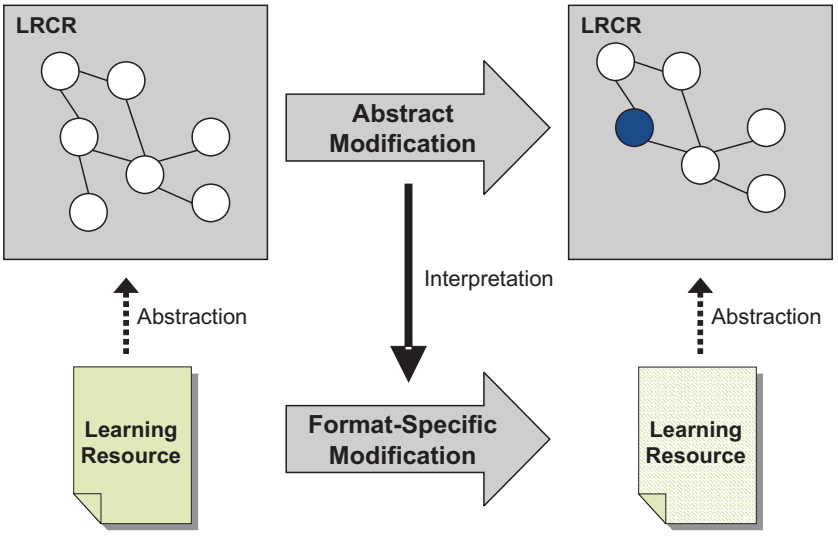


Fig. 2. Interpretation of abstract modifications

Modification types are modeled as Java classes, which are all derived from a common interface called *IModification*. All classes provide a method for retrieving the primary target element of the modification, i.e. the element that is changed. Furthermore, each modification class may define further specific variables and methods, which are regarded as parameters for the particular modification type. A modification instance is an instance of one of the modification classes. There are currently three sub interfaces of *IModification* for structural modifications, layout modifications and content modifications.

At design-time, new modification types can be specified by implementing new Java classes. Notice that modification classes do not provide methods for actually performing a modification. Similarly, a modification instance does not change content by itself, but *represents* what has to be performed.

At run-time, an adaptation application instantiates one of the modification classes to express what needs to be changed. This modification instance is then passed to a framework, which performs the modification.

By now, 17 different modification classes have been implemented (including their format-specific interpretation), and more are yet to come.

5.2 Repurposing Framework

The multimedia content repurposing framework provides content analysis and modification services to repurposing applications. As interfaces to the application it provides access to an abstract content representation of a Learning

Resource - the LRCR - and it accepts and executes modification requests. The overall framework is explained in detail in [3].

A repurposing application breaks the intended changes of a document down into a series of modifications. These modifications are instantiated as Java objects and passed to a modification transaction engine (MTE), which is part of the framework. The repurposing framework contains a number of format-specific plug-ins (FP) for the supported document formats. Based on the element of the content representation that is primarily targeted by the modification, the corresponding format-plug-in is identified and invoked. This format-plug-in then interprets the format-independent modification instance in a format-specific way by executing the appropriate Java method. Arguments for a modification method may be derived from variables of the particular modification class.

After all modifications have been performed, the LRCR is updated to represent the new state of the Learning Resource. Fig. 3 illustrates the components of the repurposing framework.

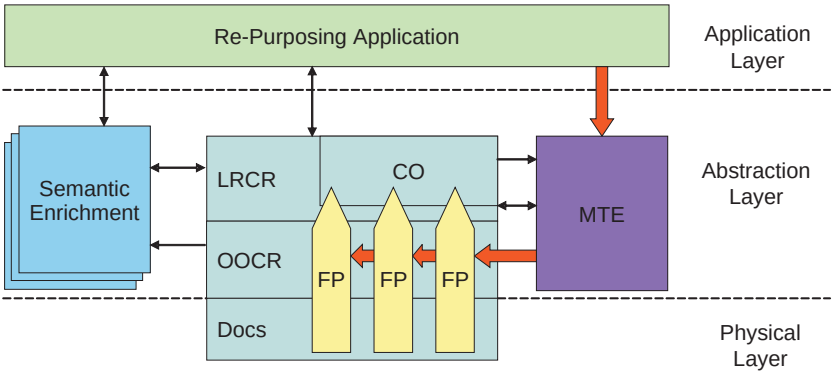


Fig. 3. Repurposing framework

5.3 An Exemplary Repurposing Application

An adaptation tool has been implemented on top of the repurposing framework. This tool already supports layout adaptations (e.g. replacing logos, background images, background and text colors) and adaptation for better printability (removing fixed widths of page elements).

Adaptations are realized as guided processes. As one part of the layout adaptation process, the tool searches for all background image definitions and background colors in all documents of a Learning Resource. Fig. 4 shows a screenshot of a dialog, where a user may specify new background images and colors for elements, which he has selected in an earlier step. The changes are instantiated by the adaptation tool as a set of modification objects, which are passed to the repurposing framework.

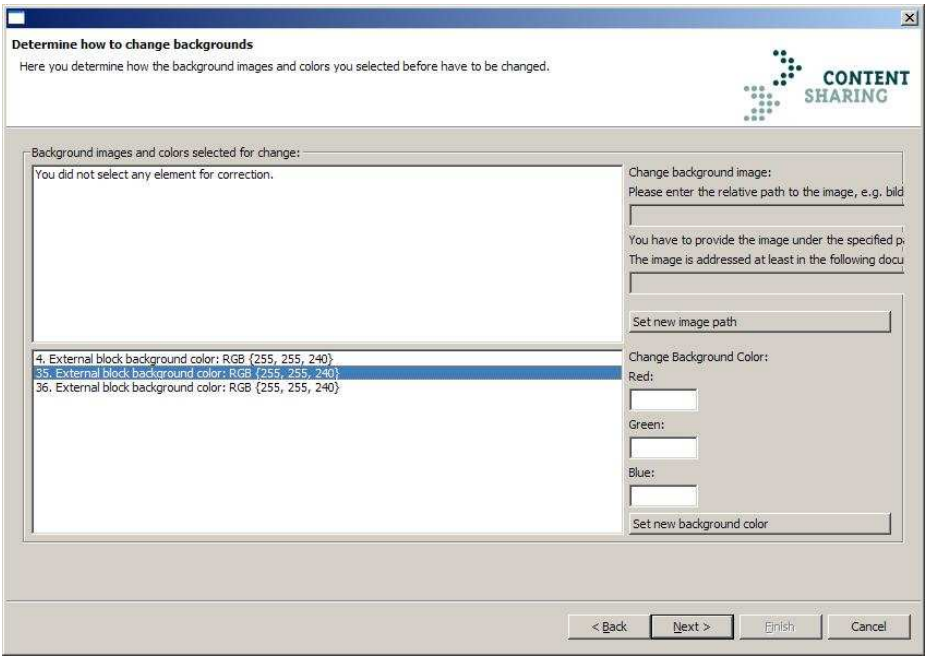


Fig. 4. Application example: dialog for changing layout information

6 Conclusions

In this paper a theoretical background is presented for modeling modifications of multimedia-based contents of Learning Resources independent of a particular format. Also, the issue of granularity of modifications has been considered. The theoretical approach has been realized in practice by representing content modifications as Java classes at design time and instantiated objects at run time. A number of modifications have been implemented, which support design and layout changes. More modification types are planned, which then enable other kinds of adaptations. The recent experiences are very promising. The concept of the repurposing framework - a relatively complex approach at first sight - and the investment in its development proved first positive results: Adaptation applications can now be implemented with reduced effort [9]. One example for such an adaptation tool has been presented in this paper. And even more important: As all modifications have to be carefully modeled, the applications tend to work more reliable and fewer bugs occur.

For the future, we plan to implement more modification types and new adaptation applications on top of the framework. There are also plans for modularizing existing Learning Resources based on this framework. Furthermore, one or two additional documents formats will be supported soon.

Acknowledgments

This work is supported by the German Federal Ministry of Economics and Technology in the context of the project Content Sharing.

References

1. Advanced Distributed Learning: Sharable content object reference model (SCORM) 2004, (http://www.adlnet.org)
2. Zimmermann, B., Bergsträßer, S., Rensing, C., Steinmetz, R.: A requirements analysis of adaptations of re-usable (e-learning) content. (2006) accepted for ED-MEDIA 2006.
3. Meyer, M., Hildebrandt, T., Rensing, C., Steinmetz, R.: Requirements and an architecture for a multimedia content re-purposing framework. In: Proceedings of the First European Conference on Technology Enhanced Learning. (2006)
4. Bergstraesser, S., Faatz, A., Rensing, C., Steinmetz, R.: A semantic content representation supporting re-purposing of learning resources. In: accepted for I-KNOW 2006. (2006)
5. Object Management Group: Model driven architecture. (http://www.omg.org/mda/)
6. Rensink, A., Nederpel, R.: Graph transformation semantics for a qvt language. In: Proceedings of the Fifth International Workshop on Graph Transformation and Visual Modeling Techniques. (2006)
7. Verbert, K., Gasevic, D., Jovanovic, J., Duval, E.: Ontology-based learning content repurposing. In: WWW '05: Special interest tracks and posters of the 14th international conference on World Wide Web, New York, NY, USA, ACM Press (2005) 1140–1141
8. Kashyap, V., Shklar, L.: Declarative rdf models for feature-based targeting of content to multiple devices. In: Proceedings of the Tenth International World Wide Web Conference. (2001)
9. Zimmermann, B., Rensing, C., Steinmetz, R.: Format-bergreifende anpassungen von elektronischen lerninhalten. In: accepted for Deutsche e-Learning Fachtagung Informatik 2006. (2006)

Region-Based Reconstruction for Face Hallucination

Jeong-Seon Park[1], Junseak Lee[1], and Seong-Whan Lee[2]

[1] Department of Multimedia Contents, Chonnam National University
Dundeok-dong, Yeosu, Chonnam 550-749, Korea
{jpark, iexpert}@chonnam.ac.kr
[2] Department of Computer Science and Engineering, Korea University
Anam-dong, Seongbuk-ku, Seoul 136-701, Korea
swlee@image.korea.ac.kr

Abstract. This paper proposes a new method for synthesizing high-resolution faces from single-frame low-resolution facial images, using a region-based reconstruction method of an extended morphable face model. In the method, we suppose that any noble facial image can be reconstructed by a linear combination of prototypes obtained from training facial images. Then, in order to maintain the local characteristics of local facial regions, we apply a region-based reconstruction method. The encouraging results show that the proposed methods can be used to improve the performance of face recognition systems, particularly to enhance the resolution of facial images captured from visual surveillance systems.

Keywords: Face Hallucination, Example-based reconstruction, Face recognition, Super-resolution, Extended morphable face model.

1 Introduction

Handling low-resolution(LR) images is one of the most difficult and common problems in various kinds of image processing applications, such as analysis of scientific, medical, astronomical, and weather images, archiving, retrieval and transmission of those images, as well as video surveillance or monitoring[1]. Numerous methods have been reported in the area of synthesizing or reconstructing high-resolution(HR) images from either a series of low-resolution images or a single-frame low-resolution image.

Most resolution enhancement approaches rely on a certain type of prior knowledge of image class to be reconstructed. The essence of these techniques is to use a training set of high resolution images and their low resolution counterparts, to build a co-occurrence model. When applying the example-based learning method, the goal is to predict high resolution data from the observed low resolution data[2]. Hardie et al.[3] used Markov Random Field(MRF) priors which are mainly applied to generic images.

However, for face hallucination, the domain knowledge of face images is used to generate high resolution face images. Baker and Kanade[4] adopted an image

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 44–53, 2007.
© Springer-Verlag Berlin Heidelberg 2007

pyramid to predict a prior under a Bayesian formulation. This method infers the high frequency components from a parent structure with the assistance of training samples. Gunturk et al. [2] applied Principal Component Analysis (PCA) to determine the prior model. Wang and Tang[5] developed a face hallucination algorithm using an eigen-transformation. However, the method only utilizes global information without paying attention to local details. Liu et al.[6] proposed a two-step approach to integrate a parametric global model with Gaussian assumption and a non-parametric local model based on MRF. Motivated by Liu et al., Li and Lin[7] also proposed a two-step approach to hallucinating faces by reconstructing the global image under a Maximum A Posterior(MAP) criterion, and re-estimating the residual image under the MAP criterion.

We are concerned with building a HR facial image from a LR facial image for visual surveillance systems. Our reconstruction method is example-based, object-class-specific or top-down approach. The example-based approach to interpreting images of deformable objects is now attracting considerable interest among many researchers[8][9] because of its potential of deriving high-level knowledge from a set of prototypical examples.

The proposed face reconstruction method is applied to an extended morphable face model, while most existing approaches are applied to the general face model. In the proposed model, a face is represented only by the pixel values in the normalized facial image. In the proposed extended morphable face model, an extended face is defined by a combined form of a low-resolution face, its interpolated high-resolution face from the low-resolution face, and its original high-resolution face, and then an extended face is separated by an extended shape vector and an extended texture vector.

2 Hallucinating Faces Using an Extended Morphable Face Model

In this section, we present an overview of our face hallucination methods derived from an example-based learning using an extended morphable face model.

Suppose that sufficiently large amount of facial images are available for off-line training, we could then represent any input face by a linear combination of facial prototypes[8]. Moreover, if we have a pair of LR facial image and its corresponding HR image for each person, we can obtain an approximation to the deformation required for the given LR facial image, by using the coefficients of examples. We can then obtain a HR facial image by applying the estimated coefficients to the corresponding HR example faces as shown in Fig. 1. Consequently, our goal is to find an optimal parameter set $\boldsymbol{\alpha}$ which can best represent the given LR facial image.

2.1 Definition of an Extended Morphable Face Model

In order to synthesize a HR facial image from a LR one, we defined an extended morphable face model in which an extended face is composed of a pair of LR

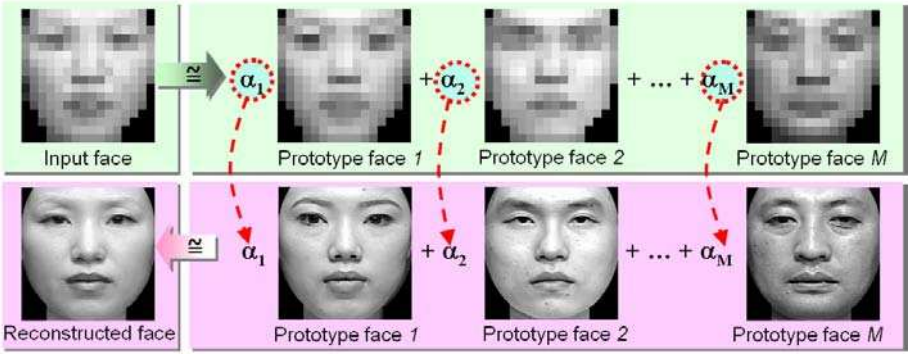


Fig. 1. Basic idea of the HR reconstruction using example-based learning

face and its corresponding HR one, and we separated an extended face by an extended shape and an extended texture according to the definition of morphable face model.

In addition to, we applied interpolation techniques to the extended shape and the extended texture[10] under the assumption that we can enlarge the amount of information from LR input image by applying interpolation techniques such as bilinear, bicubic, and so on. Fig. 2 shows an example of the facial image defined by the extended morphable face model, where bicubic interpolation is used for enlarging both LR shape and LR texture.

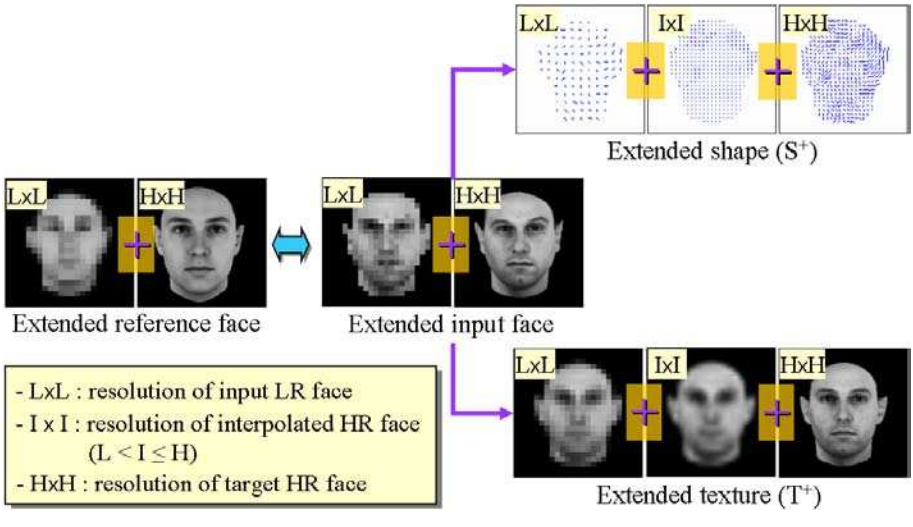


Fig. 2. An example facial image defined by the extended morphable face model

Then we can define S^+ to be an extended shape vector by simply concatenating a LR shape, the interpolated HR shape, and original HR shape:

$$S^+ = (d_1^x, d_1^y, \cdots, d_L^x, d_L^y, d_{L+1}^x, d_{L+1}^y \cdots, d_{L+I}^x, d_{L+I}^y, d_{L+I+1}^x, d_{L+I+1}^y \cdots, d_{L+I+H}^x, d_{L+I+H}^y)^T \quad (1)$$

where L, I and H are the number of pixels in input LR facial image,in the interpolated HR one, and in the original HR one, respectively. Similarly, let us define T^+ to be an extended texture vector:

$$T^+ = (i_1, \cdots, i_L, i_{L+1}, \cdots, i_{L+I}, i_{L+I+1}, \cdots, i_{L+I+H})^T. \quad (2)$$

Next, we transform the orthogonal coordinate system by PCA into a system defined by eigenvectors s_p^+ and t_p^+ of the covariance matrices C_S^+ and C_T^+ computed over the differences of the extended shape and texture, $\tilde{S^+} = S^+ - \bar{S^+}$ and $\tilde{T^+} = T^+ - \bar{T^+}$. Where $\bar{S^+}$ and $\bar{T^+}$ represent the mean of extended shape and that of extended texture, respectively. Then, an extended facial image can be represented by the following equation:

$$S^+ = \bar{S}^+ + \sum_{p=1}^{M} \alpha_p {s_p}^+, \quad T^+ = \bar{T}^+ + \sum_{p=1}^{M} \beta_p {t_p}^+ \quad (3)$$

where $\boldsymbol{\alpha},\ \boldsymbol{\beta} \in \Re^M$.

Based on the definition of the morphable face model[8], our face hallucination method is consists of following 4 steps, starting from a LR facial image to a HR facial image. Here the displacement of the pixels in an input LR face which correspond to those in the LR reference face is known.

Step 1. *Obtain the texture by warping an input LR face onto the reference face with its given LR shape.*
Step 2. *Reconstruct a HR shape from a given LR shape.*
Step 3. *Reconstruct a HR texture from the obtained LR texture at Step 1.*
Step 4. *Synthesize a HR face by warping the reconstructed HR texture with the reconstructed HR shape.*

Step 1(backward warping) and Step 4(forward warping) are explained from the previous study of morphable face model[8]. Step 2 and Step 3 are carried out by similar mathematical procedure except that the shape about a pixel is 2D vector and the texture is 1D(or 3D for RGB color image) vector.

2.2 Region-Based Face Hallucination

In this section, we explain the concept and method of region-based face hallucination method for improving the results of previous HR reconstruction.

The previously described face hallucination method is applied to the entire region of the facial image. As expected, the global reconstruction methods suffer from the weakness in that various significant features can affect the other regions'

results. Therefore, the characteristics of each local image are enfeebled in the reconstructed high-resolution facial image.

In order to separately preserve the characteristics of local regions, we applied the proposed example-based hallucination method for important local regions such as eyes, nose and mouth regions. Fig. 3 shows an example of global mask image and local mask image for region-based reconstruction.

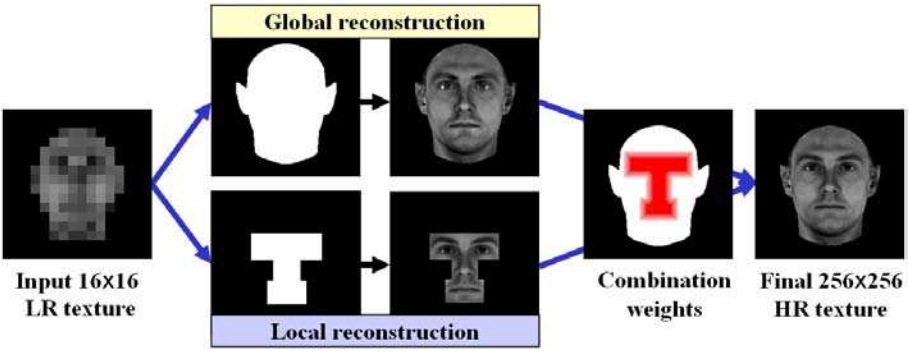


Fig. 3. Examples of global and local mask images for region-based reconstruction

It is important to obtain seamless and natural final images, in the case of region-based applications. Blending is applied to merge different regions, which were separately reconstructed for different regions of the face: eyes, nose and mouth regions.

In the proposed method, the transition area is defined as the distance(D) from local region bounding for every pixel in the local region. In other words, the weight for merging local reconstruction to global reconstruction is determined as the minimum distance to the boundary of the local region.

The blending weight for the local region is computed by the following equation:

$$\begin{aligned} &\omega_L(x_j) = 1/(D - d(x_j) + 1), \quad for \;\; 0 < d(x_j) < D \\ &\omega_L(x_j) = 1, \qquad\qquad\qquad\quad for \;\; d(x_j) >= D \end{aligned} \tag{4}$$

where $d(x_j)$ is the minimum distance from the boundary of local feature region, and D is the minimum distance that has the $\omega_L = 1$.

Then, the final shape and texture of each pixel in local region are computed as follows

$$\begin{aligned} S(x_j) &= \omega_L(x_j) \cdot S^L(x_j) + (1 - \omega_L(x_j)) \cdot S^G(x_j), \\ T(x_j) &= \omega_L(x_j) \cdot T^L(x_j) + (1 - \omega_L(x_j)) \cdot T^G(x_j), \end{aligned} \tag{5}$$

where $S^L(x_j)$ and $T^L(x_j)$ are the reconstructed shape and texture in the local region, and $S^G(x_j)$ and $T^G(x_j)$ are those in the global region.

3 Experimental Results and Analysis

3.1 Face Database

To test the performance of our reconstruction method, we used 200 facial images of Caucasian faces that were rendered from a database of 3D head models recorded by a laser scanner[8]. The original images were color images, set to the size of 256×256 pixels. They were converted to an 8-bit gray level and resized to 16×16 for LR facial images. PCA was applied to a random subset of 100 facial images for constructing bases of the defined face model. The other 100 images were used for testing our reconstruction methods.

In order to verify the potential of the proposed face hallucination method, we tested our face hallucination method using two other face databases, the Korean face database (KF DB)[11] and the XM2VTS database[12]. We used 200 facial images from 540 subjects' images of the KF DB and 200 facial images from 295 subjects' images of the XM2VTS DB. Reference face and examples of the KF DB and XM2VTS DB are shown in Fig. 4.

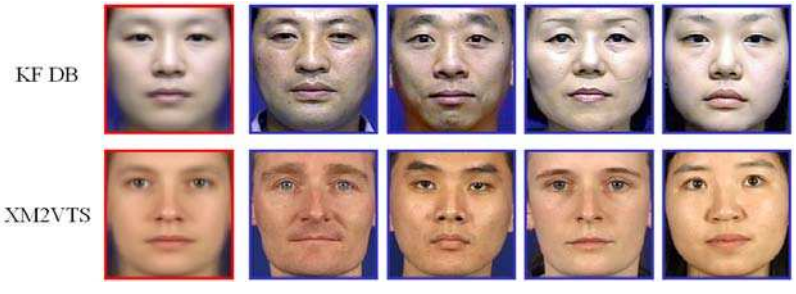


Fig. 4. Examples of KF DB and XM2VTS database

3.2 Reconstruction Results and Analysis

In order to verify the performance of the proposed method, we compared different resolution enhancement methods: the general BC interpolation method, classic example-based method using PCA transformation with the general face model (existing method), proposed method with the extended morphable face model, and proposed region-based reconstruction method.

Fig. 5 shows examples of the 256×256 high-resolution facial images reconstructed from 16×16 low-resolution images, and shows the mean intensity errors between each reconstructed image and its original high-resolution image. In the figure, (a) shows the input low-resolution images from different databases, (f) shows the original high-resolution facial images, and (b) to (e) show the reconstructed high-resolution images using the BC interpolation, existing example-based hallucination method with general face model, proposed extended morphable face model, and proposed region-based face hallucination method with the extended morphable face model, respectively.

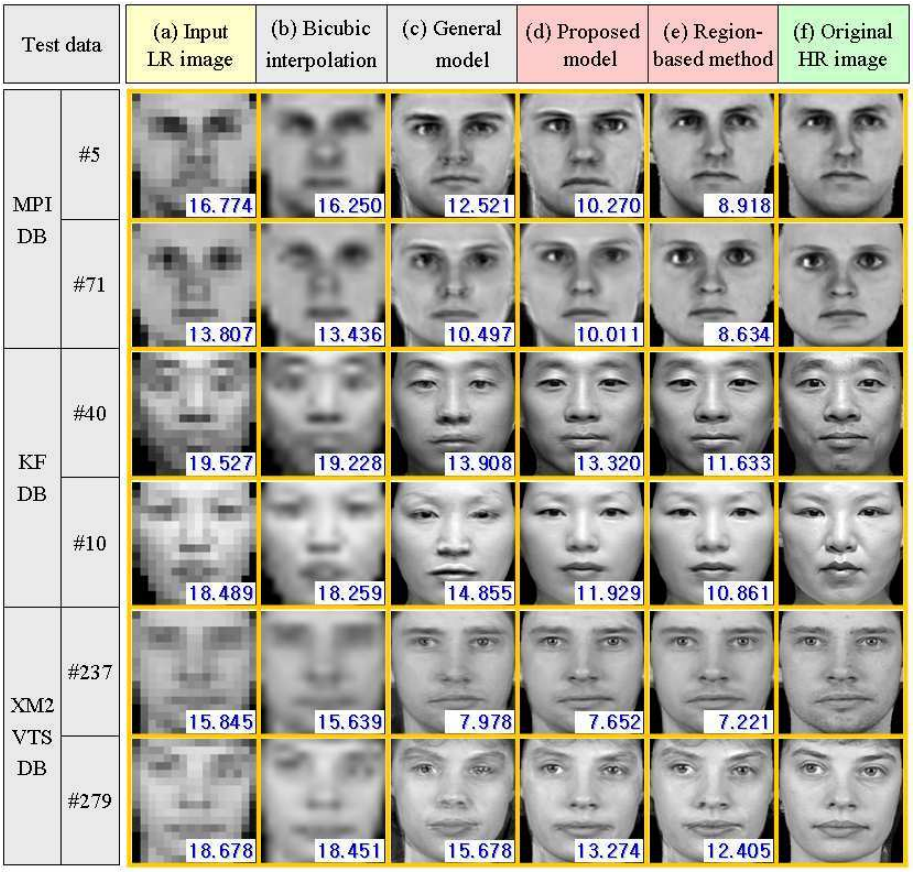


Fig. 5. Examples of 256×256 high-resolution facial images reconstructed from 16×16 low-resolution facial images in MPI, KF and XM2VTS DBs

As shown in Fig. 5, classifying each input low-resolution face from those images is almost impossible, even with the use of BC interpolation. On the other hand, the facial images reconstructed by the example-based learning methods, especially the reconstructed images by our proposed region-based face hallucination method with extended morphable face model, are more similar to the original faces than other methods. In addition, the mean intensity errors can be reduced by the proposed example-based face hallucination method.

For quantitative evaluation of the performance of our face hallucination methods, we measured the mean intensity errors per pixel between the original high-resolution facial images and their reconstructed versions, by using several different methods: low-resolution input image, BC interpolation method, example-based face hallucination using the general face model, proposed extended morphable face model, and proposed region-based hallucination method.

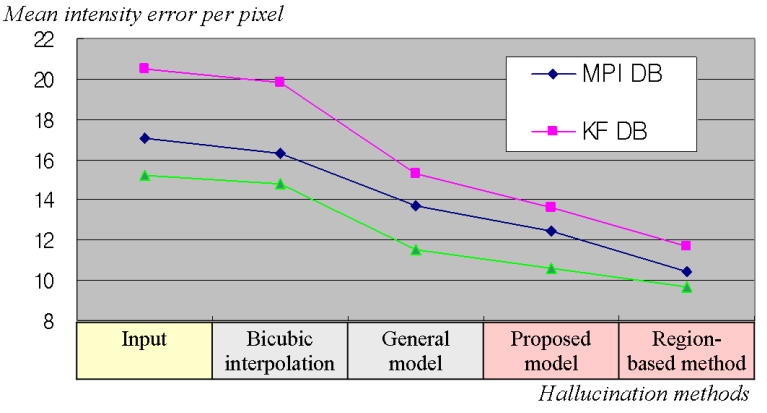


Fig. 6. Comparison of mean intensity errors by various hallucination methods

As shown in Fig. 6, we can reduce the mean reconstruction errors by using the proposed extended morphable face models, especially the region-based hallucination method, which combines the extended morphable face model and local hallucination method.

We also compared the enhancement rate(ER) of the face hallucination method by the following equation:

$$ER = (E^I - E^R)/E^I \times 100 \tag{6}$$

where E^I is the mean error of input low-resolution face to the original high-resolution face, and E^R is the mean error of reconstructed high-resolution face to the original high-resolution face. Fig. 7 shows the enhancement rate of different

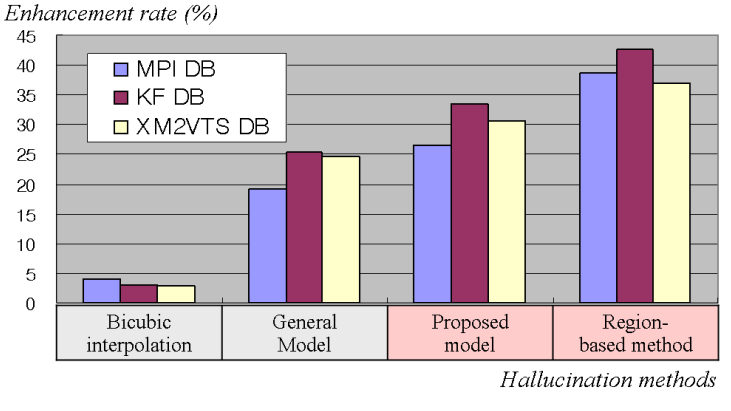


Fig. 7. Enhancement rate of different hallucination methods to the mean errors of input low-resolution

face hallucination methods. This figure shows the enhancement power of the proposed face hallucination method using extended morphable face model.

From the encouraging results of the proposed method, as shown in Figs. 5-7, the potential to improve the performance of face recognition systems exists, by reconstructing high-resolution facial images from low-resolution facial images captured in visual surveillance systems.

In order to verify the effect of face hallucination, we carried out simple face recognition experiment as described below. The original 256×256 facial images were registered to the recognition system, and the reconstructed high-resolution facial images from 16×16 facial images were used as test data. Fig. 8 shows the correct recognition rates of face recognition experiments with MPI face database (MPI DB), Korean face database (KF DB), and the XM2VTS database. As shown, the recognition performance was improved by employing the proposed face hallucination method.

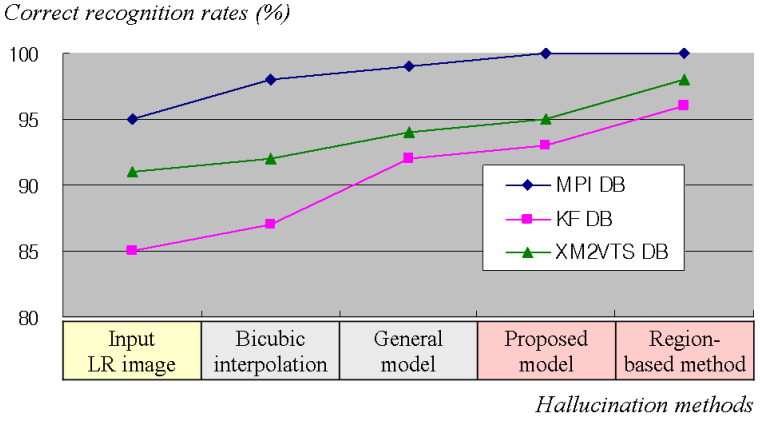


Fig. 8. Comparisons of recognition performance

4 Conclusions and Further Research

In this paper, we provided an efficient method for reconstructing high-resolution facial images, using region-based (global and local) reconstruction of the extended morphable face model. Our reconstruction method consisted of the following steps : estimating linear coefficients which minimize the error between the input low-resolution facial image and the represented linear combination of prototypes in the low-resolution image, and applying the estimated coefficients to the high-resolution prototypes. Moreover, we applied an region-based reconstruction method to improve the performance of high-resolution reconstruction by preserving the local characteristics of the facial images.

The experimental results appear very natural and plausible similar to original high-resolution facial images. This was achieved when displacement among the pixels in an input face which correspond to those in the reference face,

were known. Further studies on shape estimation with fractional accuracy from low-resolution facial images must be conducted for resolution enhancement of captured images in real-world, low-resolution situations.

Acknowledgments

This research was supported by Chonnam National University. We would like to thank the Max-Planck Institute for providing the MPI Face Database.

References

1. Tom, B., Katsaggelos, A.K.: Resolution Enhancement of Monochrome and Color Video Using Motion Comensation. IEEE Trans. on Image Processing, Vol. 10, No. 2 (Feb. 2001) 278–287
2. Gunturk, B. K., Batur, A. U., Altunbasak, Y., Hayes, M. H., and Mersereau, R. M. :Eigenface-Domain Super-Resolution for Face Recognition. IEEE Trans. on Image Processing, Vol. 12, No. 5, (May 2003) 597–606
3. Hardie, R. C., Barnar, K. J., and Armstrong, E. E. : Joint Map Registration and High-Resolution Image Estimation Using a Sequence of Undersampled Images. IEEE Trans. on Image Processing, Vol. 6, No. 12, (Dec. 1997) 1621–1633
4. Baker, S., Kanade, T.: Limit on Super-Resolution and How to Break Them. IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 24 No. 9 (Sep. 2002) 1167-1183
5. Wang, X. and TangX. : Hallucinating Face by Eigentransform. IEEE Trans. on Systems, Man and Cybernetics, Vol. 35, No. 3 (Aug. 2005) 425–434
6. Liu, C., Shum, H.-Y., and Zhang, C.-S. : A Two-Step Approach to Hallucinating Faces: Global Parametric Model and Local Nonparametric Model, Proc. of IEEE Conf. on Computer Vision and Pattern Recognition, Vol. 1, (Dec. 2001) 192–198
7. Li Y. and Lin X. : An Improved Two-Step Approach to Hallucinating Faces. Proc. of the 3rd Int'l Conf. on Image and Graphics (Dec. 2004) 298–301
8. Vetter, T., Troje, N. E.: Separation of Texture and Shape in Images of Faces for Image Coding and Synthesis. Journal of the Optical Society of America A. Vol. 14, No. 9 (1997) 2152–2161
9. Hwang, B.-W., Lee, S.-W.: Reconstruction of Partially Damaged Face Images Based on a Morphable Face Model. IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 25, No. 3 (2003) 365–372
10. Park, J.-S., Lee, S.-W.: Resolution Enhancement of Facial Image Using an Error Back-Projection of Example-based Learning. Proc. of the 6th Int'l Conf. on Automatic Face and Gesture Recognition, Seoul, Korea (May 2004) 831–836
11. Hwang, B.-W., Roh, M.-C., Lee, S.-W. : Performance Evaluation of Face Recognition Algorithms on Asian Face Recognition. Proc. of 6th IEEE Int'l Conf. on Automatic Face and Gesture Recognition, Seoul, Korea (May 2004) 278–283
12. Messer, K., Matas, J., Kittler, J., Luettin, J. and Maitre, G. : Xm2VTSDB: The extended M2VTS database. Proc. of 2nd Int. Conf. Audio- and Video-Based Personal Authentication, IAPR, Wachington D.C., (Mar. 1999) 72–77

A Shape Distribution for Comparing 3D Models

Levi C. Monteverde[1], Conrado R. Ruiz Jr.[2], and Zhiyong Huang[3,4]

[1] Citibank, International Technology Organization, 1 Temasek Avenue #26-00 Millenia Tower Singapore 039192, Singapore
Levijones.ait.monteverde@citigroup.com
[2] De La Salle University, College of Computer Studies, Taft. Avenue, Manila, Philippines
Ruizc@dlsu.edu.ph
[3] School of Computing, National University of Singapore
[4] Institute for Infocomm Research (I^2R), Singapore
Huangzy@comp.nus.edu.sg

Abstract. This study developed a new shape-based 3D model descriptor based on the D2 shape descriptor developed by Osada, et al of Princeton University. Shape descriptors can be used to measure dissimilarity between two 3D models. In this work, we advance it by proposing a novel descriptor D2a. In our method, N pairs of faces are randomly chosen from a 3D model, with probability proportional to the area of the face. The ratio of the smaller area over the larger area is computed and its frequency stored, generating a frequency distribution of N ratios which is stored as the second dimension of a 2D array, while the first dimension contains the frequency distribution of distances of randomly generated point pairs (the D2 distribution). The resulting descriptor, D2a, is a two-dimensional histogram that incorporates two shape features: the ratio of face areas and the distance between two random points.

1 Introduction

An important research area is the efficient storage and retrieval (shape-matching) of desired 3D models from an unorganized database of models (e.g. the World Wide Web). For example, if one needs a 3D model of an airplane, searching a database using "airplane" as keyword may not yield satisfactory results, since the filenames may not be descriptive (e.g. 001.wrl), may be in a foreign language, or might be misspelled. A better way is to combine keyword searching with an actual 3D model as query.

The shape dissimilarity between two 3D models is measured by applying a distance measure (such as Euclidian distance) to the shape descriptors of the two models being compared. The top k closest matches of the query model are those models with the smallest dissimilarity value compared to the query.

A shape descriptor that is used to compare 3D models based on shape similarity must be both accurate and efficient. In 2001, Osada [1] et al proposed the use of "shape distributions" as shape descriptors. These are frequency distributions

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 54 – 63, 2007.
© Springer-Verlag Berlin Heidelberg 2007

of certain shape features like angles, areas and distances randomly sampled from a 3D model. The most effective distribution was D2, which was the frequency distribution of distances of randomly selected point pairs on the surface of a 3D model. D2 was represented as a 1D array of integers, a histogram of frequencies.

D2 has several advantages. It is both rotation and translation invariant, is computationally cheap (both for generating the descriptor and comparing two descriptors), and describes the overall shape of an object, which means that it is not easily affected by minor shape distortions. For example, when a 3D model of a car is compared to a version of itself that has 5% less polygons, the overall shapes (and D2 values) of the two models remain very close [1].

In 2003, Ohbuchi [2], et al, extended Osada's D2 and added a second dimension (to the D2's one-dimensional histogram). The second dimension records the frequencies of the angles between the normal vectors of the two surfaces containing the two random points of a D2 sample. Instead of a 64-bin histogram, they used a 64x8 histogram for the AD (for "angle") descriptor, and 64x4 histogram for the AAD (for "absolute angle") descriptor. The AD and AAD enhanced descriptors outperformed D2 by 19% and 28% respectively. They used a database of 215 VRML models and implemented D2 in order to compare results with AD and AAD.

Prior to the Princeton Shape Benchmark (PSB) of Shilane, et al [3], researchers had to assemble their own test and training databases of 3D models from various sources. In order to compare the results of a shape matching algorithm, it was necessary to implement all shape descriptors that were to be compared. In 2003, the PSB was made available publicly for researchers in 3D model classification and retrieval. The PSB consists of a database of 1,814 3D models in the Object File Format (.OFF) and utility programs to measure the performance of any shape-matching algorithm that uses the PSB database. This allows researchers to directly compare a shape descriptor's 3D shape-matching performance with the results other studies that also used the PSB.

In 2004, Shilane [3] used the PSB to compare the performance of twelve shape descriptors in measuring shape dissimilarity, including Osada's D2. This study extends the D2 shape descriptor by extracting another shape feature – the ratio of areas between randomly chosen faces – and combining this with the original D2 histogram. A more detailed discussion of this method is presented in chapter 2. The PSB and the results of Shilane are used to gauge the relative effectiveness of the D2a shape descriptor in 3D shape matching.

2 Overview

In this section, we provide the theoretical framework behind the D2a shape descriptor, as well as a detailed description of D2a.

2.1 General Approach to the Shape Comparison of 3D Models

The most successful and popular approach so far in comparing the shape of two 3D models has been to apply two steps [4]:

Step 1. Apply some function on the shape feature(s) of a given 3D model to extract a "shape descriptor" for the 3D model. Shape features include areas, distances, angles, 2D projections, etc.

Step 2. Apply a distance formula to compare the shape descriptors of 2 models. Examples of distance formulas are the Manhattan, Euclidian, and Earth Mover's distance formulas. Figure 1 demonstrates this two-step process.

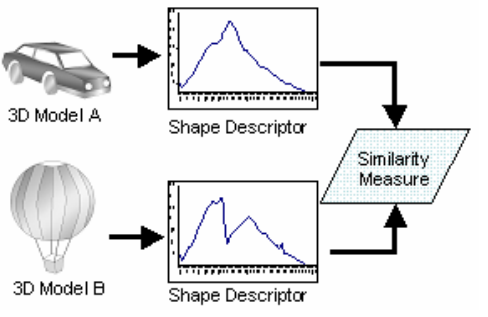


Fig. 1. The general approach to shape-based comparison of 3D models

2.2 Invariance to Transformation

The challenge in developing an ideal shape descriptor is twofold:

1. To develop the best shape descriptor that can represent 3D models, and
2. To remain unaffected by all transformations (scale, rotation and translation).

There are two ways to address these challenges:

1. Develop a *transformation-invariant descriptor* so that all rotations, scaling and translations of a model result in the same descriptor.
2. *Normalization*. 3D models can be normalized by finding a suitable transformation for each one. Unfortunately for this approach, there is no robust way to normalize rotation transformations [5], unlike with scale or translation. It is also possible to normalize the shape descriptor itself instead of the 3D model.

Figure 2 illustrates how 3D Shape Comparison is made when the descriptors are transformation invariant. In cases where the descriptor is not transformation invariant, normalization is applied to either the 3D model before the descriptor is extracted or on the descriptor itself.

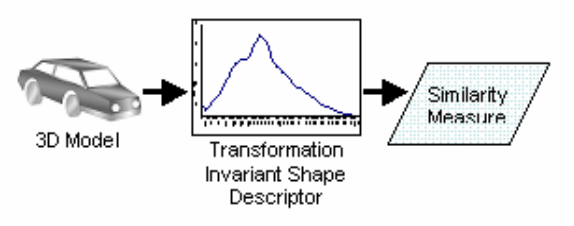


Fig. 2. Extracting a rotation, scale and translation invariant shape descriptor from a 3D model

2.3 The D2 Shape Descriptor

A 3D model is made up of a finite number of vertices and faces. Theoretically, on those faces lie an infinite number of points.

The distances between all pairs of points on the surface of the 3D model have a probability distribution. **This probability distribution is D2**. D2 is also called a shape distribution, because it is based on a feature of the model's shape, i.e. distances between all pairs of points. Osada noted that the D2 shape distribution is distinctive for each 3D model [1], and therefore represents the model's overall shape, i.e. it can be used as a shape descriptor.

However, since it is impossible to find the probability distribution of an infinite set (i.e. the set of all points on a 3D model), the actual implementation of D2 approximates the distribution by *randomly sampling* a sufficient number of points and recording the frequency of each range of distances. For example, the D2 distribution can be approximated by 1024 samples points, resulting in 1024*1024/2 + 1024 = 524,800 sample point-pair distances, since $|P_i\ P_j|$ is the same as $|P_j\ P_i|$, and is counted only once.

D2 is invariant to translation and rotation. Intuitively, no matter how the 3D model is rotated or translated, all of its vertices, faces and surface points move along with it, resulting in the same point-pair distances. However, it requires normalization for scale transformations. There several ways to normalize a D2 distribution. The two simplest and most effective are aligning by mean and aligning by maximum distance [2].

2.4 The D2a Shape Descriptor

The intuition behind D2a is that objects made up of faces with more varied sizes (i.e. has very big, very small and in-between sized surfaces) should look different from objects made up of faces with more uniform sizes (i.e. has mostly big, mostly small or mostly average-sized surfaces). For example, a 3D model of a car can have relatively large surfaces (making up the roof and windows), very small surfaces (making up the nuts and bolts), and many sizes in between (e.g. rear-view mirror) due to the discrete nature of mesh presentation for free form surfaces. A simple cube on the other hand, is made up of six equally-sized faces.

The **area ratio** *ar* of a face pair (F_i,F_j) of an object *O* is defined as the area of the smaller face over the area of the larger face:

$$ar(F_i, F_j) = \frac{\min(area(F_i), area(F_j))}{\max(area(F_i), area(F_j))}, \quad (1)$$

Allying the equation (1) to every face pair, we can derive the distribution of area ratio of the object.

A practical way to compute the area variability (or uniformity) of an object's faces is by sampling the **area ratios** of these faces in the following procedure:

```
ComputeAr(O)
    // Input: a 3D object O of M faces (F1..FM)
    // Output: histogram of area ratio of the faces
    choose N faces with probability of being chosen proportional to the area of
    each face
    for each face pair (Fi, Fj) of the selected N faces,
      if area(Fi)>area(Fj)
        ar = area(Fj)/area(Fi)
      else
        ar= area(Fi)/area(Fj)
      index = ⌊ar* numBins ⌋
      Ratio_Histogram[index] := Ratio_Histogram[index] + 1
```

Where `numBins` (=2 in our experiment) defines the granularity of the frequency distribution, and `Ratio_Histogram` contains the area ratio frequency distribution.

Since the car has varied polygon sizes, while the cube has uniform polygon sizes, one can expect the ratios of polygon areas on the car to be more variable than those of a simple cube. In fact, it can be easily observed that the only possible ratio between areas of faces in the cube is 1.0, since all faces have the same area. For the car, ratios of areas should tend to be lower than 1.0 and closer to 0.0 since one random face is likely to be much bigger or smaller than another random face. Thus, one can expect that in the frequency distribution of area ratios, a car's graph should have higher frequencies for lower ratios (i.e. the graph should skew higher to the left) due to the variances in area ratios.

It can also be conjectured that for 3D models which are made up mostly of same-sized polygons, the frequency distribution of area ratios should be lie near the value 1.0 (i.e. the graph should skew higher to the right), since one random face should not be much bigger or smaller than another random face. This conjecture can be verified by the graphs on Figure 3.

It can be seen that for objects with varied face areas (Figure 3a), the probability of two random faces having different areas is greater (therefore most ratios are below 1.0), while for objects with uniform face areas (Figure 3b), the probability of two random faces having the same area is greater (therefore most ratios are exactly 1.0).

The ratio of areas shape feature is stored in the second dimension of a 2D array whose first dimension contains the D2 distribution. The second dimension only has two bins for our experiment: the first to store the frequency of ratios that are < 1.0, and the second to store the frequency of ratios that are exactly 1.0.

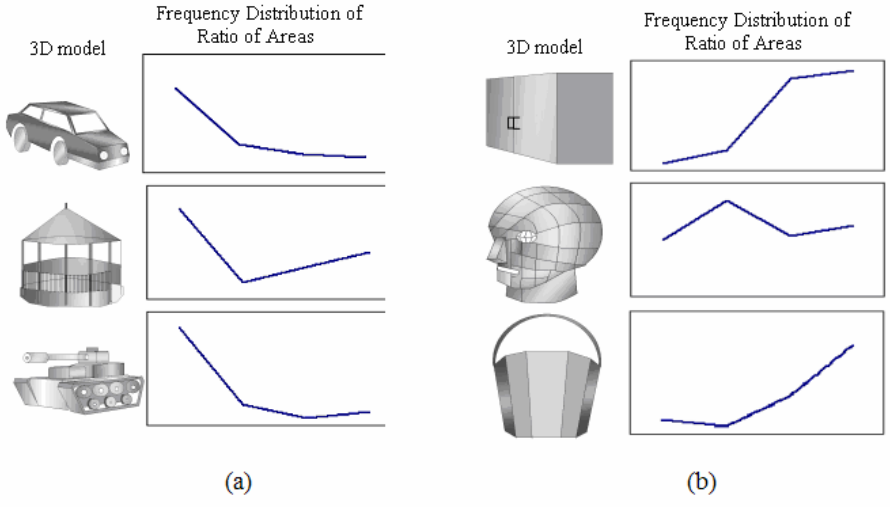


Fig. 3. Shown are 3D objects: (a) with **varied** face areas, (b) with relatively **uniform** face areas; and the graphs of their respective area ratio frequency distributions. The x-axis represents the ratio of areas (0..1), while the y-axis is the frequency. For the objects are made up of varied face areas, the ratios between these areas (small over big) tend to be small. Thus the graph shows higher frequency for small ratios. For the objects are made up of face areas of roughly similar size, the ratios between these areas (small over big) tend to be near 1.0. Thus the graph shows higher frequency for ratios near 1.0.

3 Experiments and Results

D2a was implemented in C++ and compiled using Microsoft Visual Studio.NET 2003. The Princeton Shape Benchmark was used to compare the results with Shilane's [3]. As with Shilane, only the test set of 907 models were used for all the algorithms (the training set was not used). For the D2 component (which is the first dimension of the D2a descriptor), a sample size of 1024 random points were generated and 64 bins were used. The dimensions of the D2a array were 64 x 2 unsigned integers, which took up 512 bytes (4 bytes for each unsigned integer).

The scale was normalized by aligning the mean [2, 3]. The L_1 Manhattan Distance was used to measure the distance between two D2a histograms. [2]

3.1 Performance Metrics

In order to compare the results directly with the results of Shilane, we used the same utility program provided by the PSB, `psbtable.exe`, to compute the same performance metrics: Nearest Neighbor, Tier 1, Tier 2, E-Measure, and Discounted Cumulative Gain (DCG). `psbtable.exe` generates these measurements given a distance matrix binary file (which is the output of our shape matching program) and a classification (.CLA) file. [3]

Nearest Neighbor, Tier 1 and Tier 2 measurements were also used as performance metrics by Osada [1], Kazhdan [5], and Ohbuchi [2] although these studies did not

use the Princeton Shape Benchmark 3D model database. For all performance metrics, higher numbers are better.

1. Nearest Neighbor (NN)

Using each model as query, NN is the percentage of cases where the returned top match is from the query model's class.

2. Tier 1 (T1)

Given a query model with class size N, retrieve the top N-1 matches. Tier 1 is the percentage of the N-1 matches retrieved that belong to the query model's class.

Each model in the class has only one chance to be in the first tier, since the number retrieved = number of possible correct matches. Therefore, only a perfect matching algorithm can return a T1 measure of 100%.

3. Tier 2 (T2)

Given a query model with class size N, retrieve the top 2(N-1) matches. Tier 2 is the percentage of the 2(N-1) matches retrieved that belong to the query model's class.

The number retrieved is 2 times the number of possible correct matches. Therefore, the highest possible score for T2 is 50%. However, a score of 50% doesn't mean that the matching algorithm is perfect, only that it is good enough to return all relevant matches if the retrieval size is big enough (twice the class size of the query -1).

4. E-Measure

The E-Measure is the precision and recall values combined into one value. The intuition is that a user of a search engine is more interested in the first page of query results than in later pages. So, this measure considers only the first 32 retrieved models for every query and calculates the precision and recall over those results.

The E-Measure is defined as:

$$E = 2 / (1/\text{Precision} + 1/\text{Recall})$$

5. Discounted Cumulative Gain (DCG)

DCG gives a sense of how well the overall retrieval would be viewed by a human. Correct shapes near the front of the list are more likely to be seen than correct shapes near the end of the list. With this rationale, discounted cumulative gain is calculated as: 1 + sum 1/lg(i) if the ith shape is in the correct class.

6. Normalized DCG

This metric normalizes the DCG values so that the average DCG becomes 0.0.

$$\text{Normalized } DCG_i = DCG_i / \text{Average DCG} - 1$$

Where:

DCG_i = DCG score of the i^{th} shape descriptor
Average DCG = average of all DCG scores

Normalized DCG shows how much better or worse than the average a shape descriptor's DCG score is. Since it is just another way of presenting DCG scores, it

always results in the same ranking as DCG. Therefore, although we present both DCG and Normalized DCG in the results, we count them as one metric (we use five metrics in all).

4 Results and Conclusions

Table 1 summarizes the results of Shilane [3] for 12 algorithms (first part) and our results for D2a (second part). Table 2 shows a summary of D2a's performance compared to D2. The blue horizontal bars on the first part of the table indicate where the D2a performance falls.

D2a significantly outperformed D2 in three out of five performance metrics. In T1, T2 and E-Measure, the improvements ranged from 15% – 92%. On the other hand, performance decline in the two other metrics was not as substantial: only 5.47% in NN and 3.39% in Normalized DCG.

Table 1. Summary of results for for the D2a algorithm, compared directly with 12 other algorithms. The blue horizontal bars on the first part of the table indicate where the D2a per-formance (using the same test set) would fall.

Shape Descriptor	Storage Size (bytes)	NN (%)	T1 (%)	T2 (%)	E – Meas. (%)	DCG (%)	Norm. DCG (%)
LFD	4,700	65.70	38.00	48.70	28.00	64.30	23.05
REXT	17,416	60.20	32.70	43.20	25.40	60.10	15.02
SHD	2,184	55.60	30.90	41.10	24.10	58.40	11.76
GEDT	32,776	60.30	31.30	40.70	23.70	58.40	11.76
EXT	552	54.90	28.60	37.90	21.90	56.20	7.55
SECSHEL	32,776	54.60	26.70	35.00	20.90	54.50	4.30
VOXEL	32,776	54.00	26.70	35.30	20.70	54.30	3.92
SECTORS	552	50.40	24.90	33.40	19.80	52.90	1.24
CEGI	2,056	42.00	21.10	28.70	17.00	47.90	-8.33
EGI	1,032	37.70	19.70	27.70	16.50	47.20	-9.67
D2	136	31.10	15.80	23.50	13.90	43.40	-16.94
SHELLS	136	22.70	11.10	17.30	10.20	38.60	-26.13
					Avg. DCG	52.45	
D2a (Test)	512	29.40	30.40	30.30	16.00	43.10	-17.52

Table 2. This shows a comparison of D2a's performance with that of D2. Positive numbers indicate an increase in performance, while negative numbers indicate a decline.

D2a Improvement over D2	NN (%)	T1 (%)	T2 (%)	E – Meas. (%)	DCG (%)	Normalized DCG (%)
	-5.47	92.41	28.94	15.11	-0.69	-3.39

4.1 Analysis

From the results obtained (shown on tables 1 and 2), we make the following observations. D2a underperformed in NN and DCG by 5.47% and 3.39%, respectively compared to D2. Both are "closest match" metrics, which indicates that D2 may be slightly better than D2a in placing the correct matches at the top of the retrieval list, even though D2a retrieves more correct matches than D2. D2a outperformed D2 in T1, T2 and E-Measure by 92.41%, 28.94% and 15.11%, respectively. T1, T2 and E-Measure are all related to precision and recall. T1 and T2 are precision values at specific retrieval sizes, while E-Measure is a combination of precision and recall for the top 32 matches.

The results suggest that D2a provides better precision-recall results than D2 when the retrieval size is at least 32 (and at most twice the query model's class size). This means that when a user of a search engine looks for a 3D model and requests at least 32 matches, D2a will likely return more correct matches than D2 (as much as 92% more, from our experiments). However, it is possible that D2 will show more correct matches (around 5% more) in the first page of the results than D2a.

This study shows that combining two shape features – distance between point pairs and ratio of areas of surfaces – into a single shape descriptor can result in better overall classification and retrieval performance. However, two problems may still persist in the D2a shape descriptor. First, the computation is high ($O(n^2)$) as all the ratios need to be calculated. Second, 3D models that are represented by surfaces having the same areas, such as a sphere and a cube, cannot be differentiated by the D2a descriptor, since both have the same ratio distribution. Further studies can be made to address these issues. On the other hand, the storage requirements of D2a are still well below other algorithms with comparable performance.

References

1. Osada, R., Funkhouser, T., Chazelle, B. and Dobkin, D.: Matching 3d models with shape distributions. Shape Matching International. (2001) 154-166
2. Ohbuchi, R., Minamitani, T., Takei, T.: Shape-Similarity Search of 3D Models by using Enhanced Shape Functions. Proceedings of the Theory and Practice of Computer Graphics. IEEE Computer Society (2003)

3. Shilane, P., Min, P., Kazhdan, M. and Funkhouser, T.: The Princeton Shape Benchmark. Shape Modeling International, Genova, Italy. (2004)
4. Funkhouser, T., Min, P., Kazhdan, M., Chen, J., Halderman, A. and Dobkin, D.: A search engine for 3d models. ACM Transactions on Graphics. Vol. 22(1), (2002)
5. Kazhdan, M., Funkhouser, T., and Rusinkiewicz, S.: Rotation invariant spherical harmonic representation of 3D shape descriptors. Eurographics Symposium on Geometry Processing. (2003)

3D Facial Modeling for Animation: A Nonlinear Approach*

Yushun Wang and Yueting Zhuang**

Digital media Computing & Design (DCD) Lab, Zhejiang University
MOE-Microsoft Key Laboratory of Visual Perception, Zhejiang University
{yswang, yzhuang}@cs.zju.edu.cn

Abstract. This paper presents an efficient nonlinear method for 3D facial modeling from a single image, with the support of 3D face examples. First a set of feature points is extracted from the image. The feature points are then used to automatically estimate the head pose parameters using the 3D mean face in our database as a reference model. After the pose recovery, a similarity measurement function is proposed to find the neighborhood for the given image. The scope of neighborhood can be determined adaptively using our cross-validation algorithm. Furthermore, the individual 3D shape is synthesized by neighborhood interpolation. Texture mapping is achieved based on feature points. The experimental results show that our algorithm can robustly produce 3D facial models from images captured in various scenarios.

Keywords: 3D facial modeling, nonlinear learning, head pose recovery, facial animation.

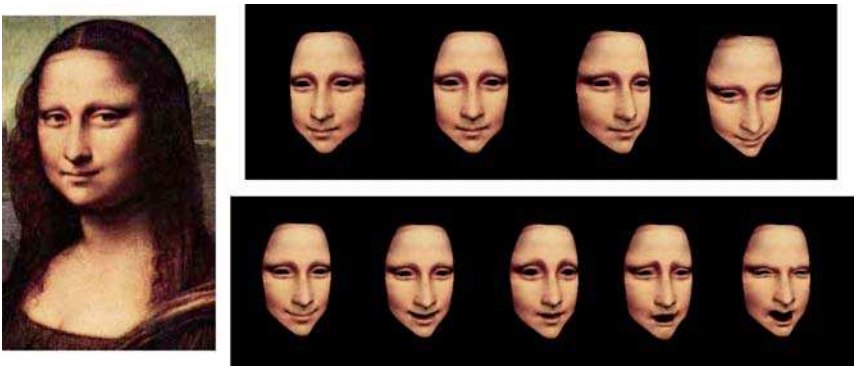


Fig. 1. The 3D modeling (top right) and animation (down right) of Mona Lisa (left)

* This work is supported by National Natural Science Foundation of China (No.60525108, No.60533090), Science and Technology Project of Zhejiang Province (2005C13032, 2005C11001-05), and China-US Million Book Digital Library Project (www.cadal.zju. edu.cn).
** Corresponding author.

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 64 – 73, 2007.
© Springer-Verlag Berlin Heidelberg 2007

1 Introduction

Generating of 3D human face models is important in computer graphics field as it is an essential part for interactive entertainment, video conference and virtual reality. The spread-out of facial modeling techniques mainly bring three practical requirements. First, the method should be easily applied to new individuals. Second, it should require no exorbitant equipments and computation cost. Third, the results should be robust and realistic.

1.1 Related Work

The pioneering work of facial modeling for animation was done by Parke in 1972 [1]. Currently, there are several main streams of available solutions.

Modeling by 3D scanners: Special equipments like 3D scanners can be used to capture the 3D shape of human heads. The data produced often needs a lot of post-processing in order to reduce noise and fill the holes. Besides, in order to animate 3D scanned models, the shape must also be combined with an animation structure, which can not be produced by the scanning process directly.
Physical based modeling: [2, 3, 4] One of the approaches to facial modeling is to approximate the anatomical structures of the face, i.e. skull, muscles and skin. The animation from physical models reflects the underlying tissue stresses. Due to the complex topology of human faces, it requires tedious tuning to model a new individual's face.
Feature points based modeling: [5, 6] Starting with several images or a 3D scan of a new individual, the generic model is deformed by the extracted facial feature points. Images are ubiquitous nowadays and a good source for facial modeling. In order to recover the 3D information, it needs orthogonal pair or more uncalibrated images.
Example based modeling: Blanz et al. [7] propose a method named morphable model, which builds new faces by a linear combination of examples. Their work can be applied to reanimating faces in images and videos [8, 9]. Supported by the examples, the input constraints can be released to only one image of the individual to generate plausible 3D results. The convergence process takes nearly an hour on SGI workstation, which limits its applications.

1.2 Our Approach

The example based approaches work well when there are a small number of examples. The iteration process converges and gets reasonable synthetic shapes and textures. However, as the number of examples increases, the structure of the 3D face space becomes more complicated and the global Euclidean distance measurement becomes invalid. The iterative optimization algorithms such as gradient descent need a lot of time to converge and easily get lost or trapped in local minimums. On the other side, in order to span a complete range of facial shapes, a large set of examples needs to be built. Due to the development of 3D scanners and the demand of realistic facial modeling and animation, the number of examples may increase dramatically.

For instance, the facial animations of Gollum in the feature film The Two Towers employed 675 example shapes [10].

In order to solve this problem, we introduce an algorithm from nonlinear dimension reductions called Locally Linear Embedding (LLE) [11], which maps its inputs into a single global coordinate system of lower dimensionality, and its optimizations - though capable of generating highly nonlinear embeddings—do not involve local minima. The idea of LLE is based on simple geometric intuitions that the data points can be linearly interpolated by its neighbors on a small piece of manifold patch. An n dimensional manifold is a topological space that is locally Euclidean (i.e. around every point, there is a neighborhood that is topologically the same as the open unit ball in R^n). The intuition is that if we can find the neighbors in the 3D face space for the given image, the synthesis process could be accelerated, as shown in Figure 2.

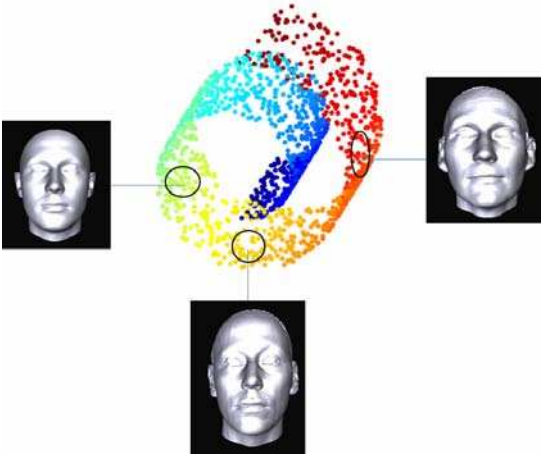


Fig. 2. 3D faces lie on a high dimensional manifold, where each 3D face can be reconstructed by its neighbors. The properly selected neighborhood will preserve the most salient features of the reconstructed 3D shape.

Based on the analysis above, we present a fast and efficient methodology to exploit a single photograph to get an animatable face model in a virtual world. The approach falls into the category of example based modeling, but also we extend this method by exploring the nonlinearity of 3D faces. Our algorithm efficiently finds the neighborhood for a given image in the 3D face space and synthesizes new faces using neighborhood interpolation.

The rest of the paper is organized as follows. In Section 2, we give an overview of our algorithm. Section 3 describes the locally embedding analysis of 3D face space. Section 4 presents the neighborhood interpolation algorithm for the synthesis of new faces. Experimental results are reported in Section 5. We conclude this paper and discuss some ideas for future work in Section 6.

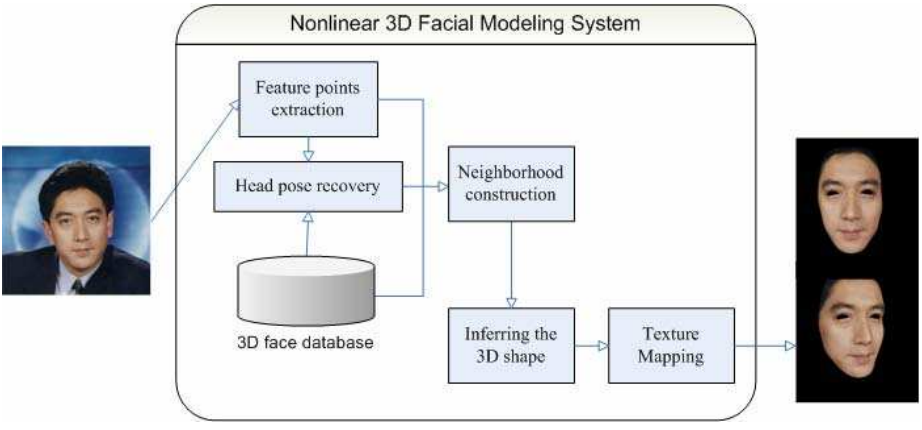


Fig. 3. Overview of our nonlinear approach to example based 3D facial modeling. Our system takes a single image as an input and gives a 3D textured model as output.

2 Algorithm Overview

As shown in Figure 3, our system takes a single image as input, and outputs textured 3D face. Our algorithm can be summarized into five steps. The first three steps directly relate to the nonlinear analysis i.e. locally embedding of the 3D face space, which provide a foundation for neighborhood interpolation. The latter two steps are about the synthesis of new faces.

Step 1: Given a frontal face image, a set of pre-defined feature points is extracted;
Step 2: Based on the feature points and a reference model, the head pose in the image is recovered automatically;
Step 3: The image finds its neighbors in the space of 3D faces by our similarity measurement;
Step 4: The 3D shape for the image is constructed by the neighborhood-based optimization;
Step 5: Texture coordinates are generated on the basis of the feature points to produce texture mapping of the model.

3 Locally Embedding of 3D Faces

In order to find the right position in the space of 3D faces for a given 2D image, a similarity measurement is needed. We employ the feature points on both images and face models as the input parameters for similarity measurement after an automatic head pose recovery.

3.1 Feature Points Extraction

MPEG-4 employs 84 feature points (Facial Definition Parameters, FDP) [12] to define a head model. For creating a standard conforming face, the set of facial

definition points used in our paper are derived from the MPEG-4 FDP. We exploit 58 feature points, as shown in the Figure 4, to define the frontal facial features. The feature points of a given image can be extracted manually or automatically.

The 3D models used in this paper are complete models with necks and ears besides the facial mesh. They are all preprocessed and in correspondence. The definition of the feature vertices on a reference model will also be made on other examples from their correspondence. The feature points defines a bounding box in which the part of mesh is our volume of interest from the facial animation point of view. During the operations of facial modeling and texture mapping, only the mesh in the bounding box is rendered.



Fig. 4. Standard-conforming feature points definition

3.2 Head Pose Recovery

The head pose of the image needs to be determined before calculating similarity. Various methods have been reported in the scenario of image sequences [13] or range data [14]. This paper proposes an efficient solution for pose recovery, which has three characteristics. First, with support of a 3D face example, we can recover the pose parameters from a single fronto-parallel image. The reference model employed here is the 3D mean face in our database for its generality. Second, similarity transformation parameters are used, i.e. the parameters to be estimated are rotation ***R***, translation ***t***, and scaling *s*. Third, using least squares estimation, the similarity transformation parameters can be calculated efficiently by matrix operations.

Our system chooses the feature points on eyes and mouth to estimate pose, for they are nearly on a plane. These 2D feature points on fronto-parallel images can be thought as on *xoy* plane in the 3D space. Then the problem of pose estimation is translated to the problem of similarity transformation parameters estimation between two point patterns, as shown in Figure 5. We use least squares estimation to minimize a cost function:

$$C(R,t,s)=\frac{1}{n}\sum_{i=1}^{n}\left\|p_i-(sRv_i+t)\right\|^2 \tag{1}$$

where $\{p_i=(x_i,y_i,0)\}$ is a set of feature points on image, $\{v_i=(x_i,y_i,z_i)\}$ is a set of feature vertices on the 3D model, R: rotation, *t*: translation:, and s : scaling are the

3D similarity transformation parameters, n is the number of feature points and vertices. Minimizing the cost function in Equation 1 will give the transformation parameters.

The estimated transformation parameters for Figure 5 are calculated and applied to the 3D model, as shown in Figure 6.

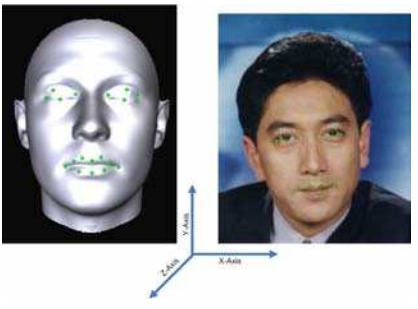


Fig. 5. Feature points used to estimate the head pose of the image (left) with a 3D face example (right)



Fig. 6. Estimated head pose parameters applied to the 3D face example

3.3 Neighborhood Construction

After the head pose estimation, the distance between the image and the 3D examples can be written as:

$$D(I, M_j) = \sqrt{\sum_{i=1}^{n} \left\| p_i - proj_{xoy}(sRv_i + t) \right\|^2} \tag{2}$$

where I is the input image, M_j is the jth example in 3D face space, and $proj_{xoy}$ is a mapping function to choose (x, y) from (x, y, z).

Once the distance function is determined, the only problem in the manifold analysis is how to choose the boundary of neighborhood. K nearest neighbors (k-NN) and ε-neighborhood: $N_\varepsilon(I) = \{M_j \,|\, D(I, M_j) \le \varepsilon\}$ are two strategies for selecting the size of local neighborhood. Our system combines k-NN and cross-validation to analyze and determine the value of k adaptively. We keep some 3D examples outside the database for cross-validation. As shown in Figure 7, the image on the left is input to our system for 3D reconstruction. The reconstruction result is compared with the real 3D data and gets a validation error. By testing the relationship between the error and the value of k, the optimum value of k is determined adaptively, as shown in Figure 8. The reconstruction error falls to its minimum where the neighbors represent most of the given example's salient features. As the number of k exceeds some value, the salient features tend to be smoothed out by averaging too many examples. Several such validation examples may be processed and the value of k is chosen by averaging these validation optimums. The properly selected neighborhood will preserve the



Fig. 7. Extra image and 3D shape of an individual who is not in the database for cross-validation

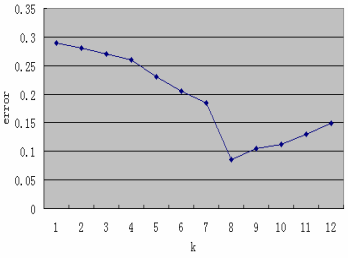


Fig. 8. The cross-validation result to choose *k* adaptively

most salient depth features of the individuals. This idea could also be applied to ε -neighborhood, where the value of ε can be chosen via validation by extra examples.

4 Synthesis of New Faces

Once the neighborhood for a given image is found, optimization techniques can be used to infer the 3D shape.

4.1 Inferring 3D Shapes by Neighborhood Interpolation

We construct a function that maps the 2D pixel positions $P = \{p_i\}$ to the 3D vertex coordinates $V = \{v_j\}$. Constructing such a function can be regarded as an interpolation or approximation problem, which solves a problem of approximating a continuous multivariate function $f(\vec{x})$ by an approximate function $F(\vec{x}, \vec{c})$ with an appropriate choice of parameter set $\vec{c}$ where $\vec{x}$ and $\vec{c}$ are real vectors ($\vec{x} = x_1, x_2, ..., x_n$ and $\vec{c} = c_1, c_2, ..., c_k$).

The family of radial basis functions (RBF) is well known for its power to approximate high dimensional smooth surfaces and it is often used in model fitting [6]. The network of RBF to infer the 3D shape of a given image is:

$$v_j = \sum_{i=1}^{k} c_{ji} \phi(D(I, M_i)) \tag{3}$$

where I is the input image represented by feature points, M_i is the ith 3D model in its neighborhood, $D(I, M_i)$ is the distance function described in Equation 2, k is the number of k-NN neighbors, c_{ji} denotes the parameters to be learned, j represents the jth element in the output vector, $\phi(r)$ is radially symmetric basis functions. Examples of basis functions are Gaussian functions $\phi(r) = e^{-(\frac{r}{c})^2}$, multi-quadrics

$\phi(r)=\sqrt{(r^2+c^2)}$ and thin plate splines $\phi(r^2)=r^2\log r$ with a linear term added. Plugging the Hardy basis function into Equation 3 results in:

$$v_j = F_j(I) = \sum_{i=1}^{k} c_{ji}\sqrt{D(I,M_i)^2 + s_i^2} \tag{4}$$

where $s_i = \min(D(I,M_i))$ is the stiffness coefficient for balancing the scope of neighborhood.

Substituting the k pairs of neighborhood training data $(\vec{p},\vec{v})$ into Equation 4 results in a linear system of k equations, where $\vec{p}$ is the vector concatenating all the elements of $proj_{xoy}(sRv_i+t)$ and $\vec{v}$ is the vector concatenating all the elements of vertex coordinates on the ith 3D model. Solving the linear system yields:

$$\vec{c} = H^{-1}\vec{v} \tag{5}$$

$$\vec{c} = (H+\lambda I)^{-1}\vec{v} \tag{6}$$

where λ=0.01 is a small disturbing factor determined empirically to decrease the impact of noise and I here is the identity matrix.

4.2 Texture Coordinates Extraction

Based on the feature points of the image, the texture coordinates can be interpolated to get texture mapping. Given a set of corresponding feature vertices on the 3D model and texture coordinates, the in-between vertices can get their texture coordinates via scattered data interpolation. We use the method similar to Section 4.1, except that the input of the RBF system is the 3D vertex and the output is the corresponding texture coordinates.

$$t = F(v) = \sum_{i=1}^{n} c_i\phi(\|v - v_i\|) \tag{7}$$

where *v* is the input 3D vertex, v_i is the *i*th feature vertex, *t* is the texture coordinates.

5 Experiments

The 3D face database was provided by the Max-Planck Institute for Biological Cybernetics in Tuebingen, Germany. The 3D scanned faces in the database provide a good start point for our supportive database. We have aligned all the 3D models with an animatable model and reduced its vertex density. The eyes and mouth areas were excided for animation purpose. Besides, we added extra examples to the database by face modeling software. After that, the database consists of 200 heads each with 5832 vertices.

In order to test our techniques, we have implemented a prototype system using Visual C++ and Matlab. We reconstructed the face models and animate them from the images either taken by us using a digital camera (Figure 9) or taken under arbitrary unknown conditions (Figure 10). We also applied our method to paintings such as

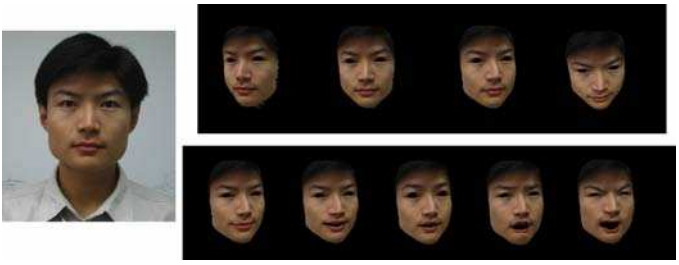


Fig. 9. An example of 3D modeling (top right) and animation (down right)



Fig. 10. Another facial modeling example

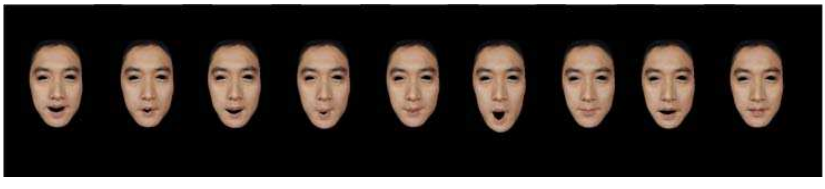


Fig. 11. Animation sequence generated by our system

Mona Lisa by Leonardo (Figure 1). We use the method described in our previous work [15] to animate them. An animation sequence is generated as shown in Figure 11.

We manually marked the feature points and the system takes approximately one second to reconstruct the 3D model with texture mapping. Although reconstructing the true 3D shape and texture from a single image is an under-determined problem, 3D face models built by our system look vivid from the frontal viewpoint and natural from other viewpoints.

6 Conclusions and Future Work

This paper proposes a novel efficient nonlinear approach to 3D facial modeling from a single image. In this algorithm, we measure the distance between the input image and the 3D models after estimating similarity transformation. Neighborhood interpolation is used to find the optimum of the 3D shape to preserve salient features. Furthermore, the image is mapped onto the synthesized model as texture. Vivid 3D animation can be produced from a single image through our system.

Our algorithm only needs matrix operations instead of iterative process to find optimums. Therefore it is efficient for many applications, such as teleconference, digital entertainment and video encoding.

There are several directions of improvement in the future. The inner properties of the face space need to be further explored in order to synthesize new faces efficiently and accurately. Currently the texture mapping just exploits the colors on the image that reflect the lighting conditions under which it was taken. Relighting techniques should be developed for integrating our facial model with the virtual environment. Furthermore, the wrinkles and detailed textures have not been properly tackled in the existing techniques. These problems ought to be considered in future work.

References

1. F.I. Parke: Computer generated animation of faces. Proceedings ACM annual conference., August 1972.
2. E. Sifakis, I. Neverov, R. Fedkiw: Automatic Determination of Facial Muscle Activations from Sparse Motion Capture Marker Data, SIGGRAPH 2005, ACM TOG 24, 417-425 2005.
3. K. Kahler, J. Haber, H. Yamauchi, H.- P. Seidel: Head shop: Generating animated head models with anatomical structure. In Proc. ACM SIGGRAPH Symposium on Computer Animation, pages 55--64, 2002.
4. D. Terzopoulos, K. Waters: Physically-based facial modelling, analysis, and animation. In The Journal of Visualization and Computer Animation.. 1(2):73-80. 1990.
5. W.Lee, N.Magnenat-Thalmann: "Fast Head Modeling for Animation", Journal Image and Vision Computing, Volume 18, Number 4, pp.355-364, Elsevier Sceince, 1 March, 2000.
6. F. Pighin, J. Hecker, D. Lischinski, R. Szeliski, D. H. Salesin: Synthesizing Realistic Facial Expressions from Photographs, Siggraph proceedings, pp. 75-84, 1998.
7. V. Blanz and T. Vetter: A Morphable Model for the Synthesis of 3D Faces, Proc. Siggraph 99, ACM Press, New York, pp. 187-194, 1999.
8. V. Blanz, C. Basso, T. Poggio, T. Vetter: Reanimating Faces in Images and Video, Computer Graphics Forum 22 (3), EUROGRAPHICS 2003, Granada, Spain, p. 641 - 650, 2003.
9. D. Vlasic, M. Brand, H. Pfister, J. Popovic: "Face Transfer with Multilinear Models", ACM Transactions on Graphics (TOG), ISSN: 0730=0301, Vol. 24, Issue 3, pp. 426-433, 2005.
10. J. Fordham: Middle earth strikes back. Cinefex, (92):71–142, 2003.
11. S. Roweis, L. Saul: Nonlinear Dimensionality Reduction by Locally Linear Embedding, Science, 290; 2323-2326, December 2000.
12. J. Ostermann: Animation of Synthetic Faces in MPEG-4. In Computer Animation, pages 49–51, Philadelphia, Pennsylvania, 8-10 June 1998.
13. Zhiwei Zhu, Qiang Ji: Real Time 3D Face Pose Tracking From an Uncalibrated Camera, Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'04) Volume 5, p. 73, 2004.
14. S. Malassiotis, M. G. Strintzis: Robust real-time 3D head pose estimation from range data. Pattern Recognition. 38(8): 1153-65. 26, 2005.
15. Y. Wang, Y. Zhuang, F. Wu, Data-driven facial animation based on manifold Bayesian regression, Journal of Zhejiang Univ SCIENCE A. 20067(4):556 556-563.

Normalization and Alignment of 3D Objects Based on Bilateral Symmetry Planes

Jefry Tedjokusumo and Wee Kheng Leow

Dept. of Computer Science, National University of Singapore,
3 Science Drive 2, Singapore 117543
{jefryted, leowwk}@comp.nus.edu.sg
www.comp.nus.edu.sg/∼leowwk

Abstract. Recent advancements in 3D scanning technologies have inspired the development of effective methods for matching and retrieving 3D objects. A common pre-processing stage of these retrieval methods is to normalize the position, size, and orientation of the objects based on PCA. It aligns an object's orientation based on PCA eigenvectors, and normalizes its size uniformly in all 3 spatial dimensions based on the variance of the object points. However, orientation alignment by PCA is not robust, and objects with similar shape can be misaligned. Uniform scaling of the objects is not ideal because it does not take into account the differences in the objects' 3D aspect ratios, resulting in misalignment that can exaggerate the shape difference between the objects. This paper presents a method for computing 3D objects' bilateral symmetry planes (BSPs) and BSP axes and extents, and a method for normalizing 3D objects based on BSP axes and extents. Compared to normalization methods based on PCA and minimum volume bounding box, our BSP-based method can normalize and align similar objects in the same category in a semantically more meaningful manner, such as aligning the objects' heads, bodies, legs, etc.

1 Introduction

Recent advancements in 3D scanning technologies have led to an increased accumulation of 3D models in databases and the Internet, and inspired the development of effective techniques for retrieving 3D objects that are similar in shape to a query model (e.g., [1,2,3,4,5,6]). 3D object matching and retrieval typically involve three basic stages: (1) object normalization, (2) feature extraction and object representation, and (3) object comparison. The first stage typically normalizes objects' positions, sizes, and orientations by translating the objects' centroids to the origin of the 3D coordinate frame, normalizing the variances of the points on the objects, and aligning their principal axes obtained using Principal Component Analysis (PCA) [1,7]. The second stage extracts various features from the objects and represents the objects in various forms such as histograms, 2D spherical maps, 3D grids, and abstract representations in terms of the extracted features [7]. The third stage typically uses very simple distance measures such as the Euclidean distance to perform efficient comparison.

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 74–85, 2007.
© Springer-Verlag Berlin Heidelberg 2007

The standard normalization method described above is not ideal. Orientation alignment based on PCA is not robust because PCA is sensitive to point distributions of the objects. Objects with similar shape may be misaligned [1] (Fig. 1). Moreover, this method does not take into account the difference in the objects' 3D aspect ratios. Normalization of objects with different 3D aspect ratios by the same scaling factors in all 3 spatial dimensions causes misalignments of their corresponding parts (Figs. 2, 7(a)). All these misalignments can result in an exaggeration of the difference between objects with similar shapes. Consequently, relevant objects (i.e., objects in the same category as the query) may be regarded by the matching algorithm as different from the query and are not retrieved. Therefore, it is important to normalize and align the objects properly.

A straightforward improvement over the standard normalization method is to scale the objects according to their 3D aspect ratios. This brings out a question: In which coordinate system should the objects' 3D aspect ratios be measured? A possibility is to measure 3D aspect ratios along the PCA axes. This method is not robust because, as discussed above, orientation alignment based on PCA is not robust. An alternative method is to compute the objects' minimum volume bounding boxes (MBB) [8], and normalize the objects based on MBB axes and widths. Our studies show that this method is even less robust than the PCA method, as will be discussed further in Sections 2 and 4.

It is observed that many natural and man-made objects exhibit bilateral (i.e., left-right) symmetry. It is a kind of reflectional symmetry that has an interesting semantic meaning: the bilateral symmetry plane (BSP) divides an object into a left and a right half, each is a mirror reflection of the other about the BSP. Moreover, the major axis that defines the object's top and bottom lies in the BSP. Therefore, by normalizing objects according to the principal axes and 3D aspect ratios defined on BSP, the objects' semantically corresponding parts such as head, body, legs can be aligned. Consequently, shape matching of objects aligned in this manner would be semantically more meaningful.

Note that PCA or MBB alone is insufficient for computing an object's BSP. The PCA and MBB planes (i.e., the planes normal to the PCA/MBB axes) may not be aligned to the BSP plane in terms of position and 3D orientation (Figs. 1 and 2). Furthermore, an object has three PCA planes and three MBB planes. Using only PCA and MBB algorithms, it is impossible to determine which of the three planes is nearest to the object's BSP. For objects that are not exactly bilaterally symmetric, the best fitting BSP may not pass through the objects' centroid. So, to determine an object's BSP, the algorithm needs to compute the correct 3D orientation and position of a plane that separates the object into two bilaterally symmetric parts.

This paper presents a method for (1) computing 3D Objects' BSPs and BSP axes and extents, and (2) normalizing and aligning 3D objects based on BSP axes and extents. Test results show that the algorithm can compute the exact BSPs of exactly bilaterally symmetric objects. For objects that are roughly bilaterally symmetric, the algorithm can compute the best fitting BSPs. Normalization of

objects according to BSPs yields better normalization and alignment between 3D objects in the same category compared to those using PCA and MBB.

2 Related Work

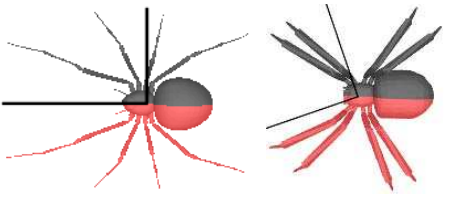
There is a substantial amount of work on 3D symmetry detection. Alt el al. [9] described algorithms for computing exact and approximate congruences and symmetries of geometric objects represented by point sets. Wolter et al. [10] presented exact algorithms for detecting all rotational and involutional symmetries in point sets, polygons and polyhedra. Jiang et al. [11,12] presented methods for determining rotational and involutional symmetries of polyhedra. Brass and Knauer [13,14] further developed methods for computing and testing symmetries of non-convex polyhedra and general 3D objects. Zabrodsky et al. [15] defined a Continuous Symmetry Measure to quantify the symmetries of objects. Minovic et al. [16] described an algorithm for identifying symmetry of 3D objects represented by octrees.

Sun and Sherrah [17] proposed algorithms for determining reflectional and rotational symmetries of 3D objects using orientation histograms. To reduce the search space, their algorithms search for the symmetries of an object along its principal axes and small orientation neighborhoods around them. The principal axes are obtained from a method similar to PCA. Our studies show that this approach is not robust because the reflectional symmetry plane of an object can be quite far from the PCA planes normal to the PCA axes (Section 4).

The above research work has focused on symmetry detection or quantification. On the other hand, Jiang and Bunke [18] applied symmetry detection in polyhedra for object recognition. Kazhdan et al. developed methods of matching 3D shape using reflectively symmetric feature descriptors [19] and rotationally symmetric descriptors [20].

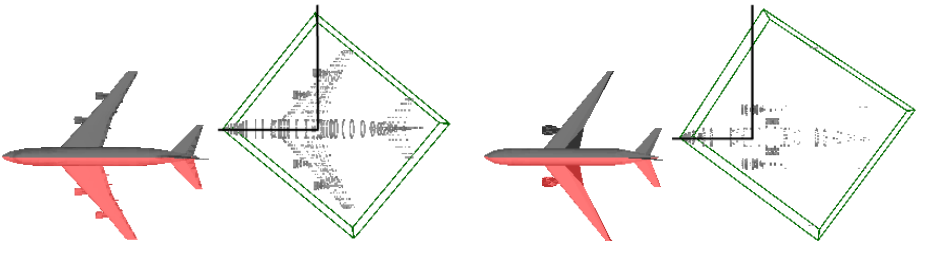
In this paper, we focus on determining bilateral symmetry planes (BSPs), BSP axes, and 3D aspect ratios for more robust and semantically meaningful normalization and alignment of 3D objects. The objects are represented as point-and-mesh models, and the object points need not be uniformly distributed over their surfaces. Indeed, many of our test objects are composed of highly non-uniformly distributed points.

Principal Component Analysis (PCA) is a well-known method for computing the principal axes of an object and the spread of points along the axes. It is the standard method for normalizing 3D objects' orientation. However, the principal axes obtained by PCA are sensitive to the distributions of points on the objects. Differences in point distributions between two objects of similar shape can cause their orientations to be misaligned [1]. This problem is most serious for objects that are not exactly bilaterally symmetric (Fig. 1). Moreover, the variances of the points along the PCA axes (i.e., the eigenvalues) are sensitive to non-uniformity of point distributions. Two objects with the same extents but different point distributions can have different variances (Fig. 2) As a result, the eigenvalues cannot be used as good estimates of the object's 3D aspect ratios.



(a) (b)

Fig. 1. PCA misalignment. Spider b is not exactly bilaterally symmetric, causing its PCA axes (black lines) to be misaligned with those of spider a. However, their computed BSPs are well aligned. Left and right sides of BSPs are denoted by different colors.



(a) (b) (c) (d)

Fig. 2. MBB misalignment. Airplanes a and c have different 3D aspect ratios. (b, d) Their first PCA axes (with the largest eigenvalues, horizontal black lines) are aligned with the BSPs but their MBBs (green boxes) are not.

The minimum volume bounding box (MBB) algorithm developed in [8] is less sensitive to the overall distribution of the points on the objects but is very sensitive to the positions of the points furthest from the objects' centroid. Typically, the objects' MBBs are not aligned with their BSPs (Fig. 2). However, it can compute the extents of the objects even if the point distributions are not uniform. So, MBB widths can be good estimates of the objects' 3D aspect ratios if MBB axes are aligned with the BSPs.

3 Bilateral Symmetry Plane

3.1 Computing BSP

For objects that are rotationally symmetric, such as a ball and an orange, each of the multiple rotational symmetry planes is a bilateral symmetry plane (BSP). However, for most natural and man-made objects with bilateral symmetry, they have only one BSP each (Figs. 1, 2, 5). To compute an object's BSP, we use the fact that each point on the object's surface has a mirror reflection with respect to the BSP.

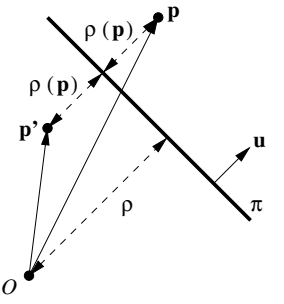


Fig. 3. Object point $\mathbf{p}$ and its mirror reflection $\mathbf{p}'$ with respect to the plane π

A plane π in 3D space can be parameterized by the equation

$$\mathbf{w} \cdot \mathbf{x} + w_0 = 0 \tag{1}$$

where w_0 and $\mathbf{w} = (w_1, w_2, w_3)$ are the parameters of the plane, and $\mathbf{x}$ is any 3D point on the plane. Consider any two points $\mathbf{x}_1$ and $\mathbf{x}_2$ lying on the plane. From Eq. 1, we obtain

$$\mathbf{w} \cdot (\mathbf{x}_2 - \mathbf{x}_1) = 0 \tag{2}$$

which means that $\mathbf{w}$ is normal to the plane. Thus, the plane's unit normal vector $\mathbf{u}$ is given by $\mathbf{w}/\|\mathbf{w}\|$. The perpendicular distance ρ of the plane from the origin is given by $\mathbf{u} \cdot \mathbf{x}$ for any point $\mathbf{x}$ on the plane. That is, $\rho = -w_0/\|\mathbf{w}\|$.

Now, consider a point $\mathbf{p}$ on the object's surface. From Fig. 3, it is obvious that the perpendicular distance of $\mathbf{p}$ from a plane π, denoted as $\rho(\mathbf{p})$, is

$$\rho(\mathbf{p}) = \mathbf{p} \cdot \mathbf{u} - \rho\,. \tag{3}$$

Then, the ideal mirror reflection $\mathbf{p}'$ of $\mathbf{p}$ with respect to the plane π is (Fig. 3):

$$\mathbf{p}' = \mathbf{p} - 2\,\rho(\mathbf{p})\,\mathbf{u}\,. \tag{4}$$

In practice, a 3D object is typically represented as a point-and-mesh model, which consists of a sparse set S of points on the 3D object's surface. So, for a point $\mathbf{p}_i \in S$, its ideal mirror reflection $\mathbf{p}'_i$ may not be in S. Let f denote the closest-point function and $f(\mathbf{p}'_i)$ denote a point in S closest to $\mathbf{p}'_i$. That is, $f(\mathbf{p}'_i)$ is the closest approximation to $\mathbf{p}'_i$. Then, the mean-squared error E between all $\mathbf{p}'_i \in S$ and its closest approximation $f(\mathbf{p}'_i)$ is:

$$E(\boldsymbol{\theta}) = \sum_{\mathbf{p}_i \in S} \|\,\mathbf{p}'_i - f(\mathbf{p}'_i)\|^2 = \sum_{\mathbf{p}_i \in S} \|\,\mathbf{p}_i - 2\,\rho(\mathbf{p}_i)\,\mathbf{u} - f(\mathbf{p}'_i)\|^2 \tag{5}$$

where the vector $\boldsymbol{\theta} = (w_0, w_1, w_2,\ w_3)$. Therefore, the problem of computing the bilateral symmetry plane is to find the plane π, parameterized by $\boldsymbol{\theta}$, that minimizes the error $E(\boldsymbol{\theta})$.

The algorithm for computing an object's BSP can be summarized as follows:

Compute BSP

1. Compute the three PCA axes (i.e., eigenvectors) of the object, and the three PCA planes normal to these axes.
2. Set the PCA plane with the smallest error E as the seed plane.
3. Rotate the seed plane in all three rotation angles by increments of δ to generate initial planes ω_j.
4. For each initial plane ω_j,
   perform gradient descent to obtain locally optimal BSP π_j and its error E_j.
5. Return the plane π_k with the smallest error E_k.

Given a sufficiently small δ, the above algorithm can find the globally optimal estimate of the object's BSP. In the tests, $\delta = 22.5°$ is used. This algorithm can also use MBB axes to obtain the seed plane. However, our tests show that initializing with PCA is more robust than initializing with MBB because the objects' BSPs tend to be closer to PCA planes than MBB planes (Section 4).

3.2 BSP-Based Object Normalization

Orientation alignment based on BSP offers an approach that can take into account the semantics of the object parts, such as head, body, legs, etc. We define the first BSP axis as the vector in BSP with the largest dispersion of points. This definition is analogous to that of PCA axis. The second BSP axis is the vector in BSP perpendicular to the first BSP axis. The third BSP axis is naturally the vector normal to BSP.

Based on the above definition, we can compute the BSP axes as follows:

Compute BSP Axes and Extents

1. Project 3D points on an object to its BSP.
2. Apply 2D PCA on the projected points and obtain principal axes in BSP.
3. Measure the extents (i.e., the distances between the furthest points) of the object along the two principal axes in BSP. The axis with a larger extent is defined as the first BSP axis, and the other one is the second BSP axis.
4. BSP's normal vector is defined as the third BSP axis. The extent along this axis is also computed.
5. The extents along the three BSP axes define the object's 3D aspect ratio.

In the third step, PCA eigenvalues should not be used as measures of the object's extents because they are sensitive to non-uniform distribution of points.

Similar to PCA axes, the BSP axes for different objects may be pointing in opposite directions even though their orientations are the same. A common technique of handling this problem is to reflect the principal axes before matching the objects [4]. With three principal axes, there are altogether eight reflected versions to be compared. The reflection with the smallest matching error would be the one with the semantically matching axis directions.

BSP-based normalization is performed by translating the objects centroids' to the origin of the 3D coordinate frame, aligning the objects' BSP axes, and normalizing their 3D aspect ratios to a standardized 3D aspect ratio according to their BSP extents. In case this method distorts the shapes of some objects too significantly, an alternative is to group objects into categories according to some criterion such as difference in aspect ratios, semantic class, etc., and scale the objects in each category to a different standardized 3D aspect ratio that minimizes shape distortion.

4 Experiments

The test set contains 1602 objects some of which are exactly bilaterally symmetric while the others are roughly bilaterally symmetric. This test set is compiled by combining 512 aircrafts in the Utrecht database [21] and 1090 objects in the Princeton database [22]. The Utrecht database contains 6 categories of aircrafts whereas the Princeton database contains about 50 categories of objects. Highly non-bilaterally symmetric objects in the Princeton database are excluded.

Two sets of tests were conducted to assess the performance of the algorithm for computing BSPs and BSP-based object normalization and alignment. The implementation of the MBB algorithm was downloaded from the web site valis.cs.uiuc.edu/~sariel/research/papers/98/bbox.html.

4.1 Test on BSP Computation

For this test, the following normalized error E' was computed for the estimated BSP $\boldsymbol{\theta}$ of each object S:

$$E'(\boldsymbol{\theta}) = \frac{1}{|S|\, v} \sum_{\mathbf{p}_i \in S} \| \mathbf{p}'_i - f(\mathbf{p}'_i) \| \tag{6}$$

where v is the variance of the points $\mathbf{p}_i$ from the object's centroid. This normalized error is independent of the number of points and the size of the objects, and so can be compared among the objects.

The algorithm for computing BSP was performed on all 1602 objects. It successfully computed the BSPs of 1589 (99.2%) objects. Among the successful cases, the computed BSPs of 487 (30.7%) bilaterally symmetric objects are practically exact, with $E' \leq 0.00001$ (Fig. 4). A total of 1348 objects (84.8%) with bilateral symmetry and approximate bilateral symmetry have errors $E' \leq 0.03$. For the other 241 (15.2%) successful cases, the computed BSPs have various amounts of error ranging from 0.03 to greater than 0.1. Sample results are shown in Fig. 5. For objects that are bilaterally symmetric (rows 1, 2), exact BSPs are found. For objects that are roughly bilaterally symmetric (row 3), the best fitting BSPs are computed. Therefore, the error E' is well correlated to the degree of bilateral symmetry of the test objects.

For the 13 (0.8%) failure cases (Fig. 5, row 4), all their errors are greater than 0.03, and 53.9% of them are greater than 0.1. The computed BSPs are all larger

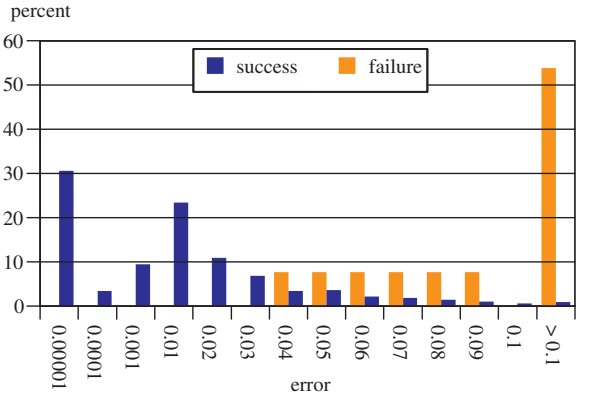


Fig. 4. Frequency distribution of the errors of computed BSPs of test objects

Table 1. Percentages of PCA and MBB planes nearest to test objects' BSPs

	1st	2nd	3rd
PCA	13.4%	61.7%	19.9%
MBB	31.1%	46.1%	22.8%

than 30° from the desired BSPs. The main reason of the failure is that these objects are not exactly bilaterally symmetric and there are very few points on them. In some cases, one or two outliers (i.e., points without mirror reflections and lying at large distances from the objects' centroids) are enough to severely tilt the orientation of the computed BSP. One method of solving this problem is to apply a robust method to exclude outliers while computing the BSP.

As discussed in Section 1, an object has three PCA planes and three MBB planes. Table 1 tabulates the percentage of PCA/MBB planes that are nearest, in terms of 3D orientation, to the computed BSPs of the test objects. It shows that most of the objects' BSPs are nearest to the second PCA plane (the plane normal to the second PCA axis). This is expected because most objects' second PCA axes run across their bodies in the left-right direction normal to their BSPs. Nevertheless, there are many other objects whose BSPs are nearest to other PCA/MBB planes. These results show that PCA and MBB, by themselves, are not able to determine the correct BSPs in general.

Figure 6 plots the frequency distribution of the angular difference between an object's BSP and its nearest PCA/MBB plane. 69.6% of the objects have BSPs exactly aligned with their nearest PCA planes (i.e., 0° difference). On the other hand, only 1.2% of the objects have BSPs exactly aligned with their nearest MBB planes. Most (30.7%) of the objects' BSPs are, in fact, more than 20° off the nearest MBB planes. This test result shows that it is better to use PCA planes to initialize the algorithm for computing BSP (Section 3.1).



Fig. 5. Sample BSP results. (Rows 1–3) Successful cases: (Rows 1, 2) Bilaterally symmetric objects, (Row 3) Approximately bilaterally symmetric objects. (Row 4) Failure cases. Left and right sides of BSPs are denoted by different colors.

4.2 Test on BSP-Based Normalization

Four types of normalization methods were compared:

1. PCA with uniform scaling (PCA):
   Normalize objects' centroids, PCA axes, and variance of points. This is the standard normalization method and serves as the base case.
2. PCA with 3D aspect ratio normalization (PCA3):
   Normalize objects' centroids, PCA axes, and 3D aspect ratio estimated by PCA eigenvalues.
3. MBB:
   Normalize MBB centroid, MBB axes, and MBB extents.
4. BSP:
   Normalize object's centroid, BSP axes, and BSP extents.

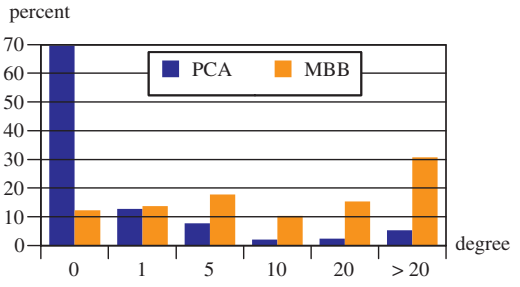


Fig. 6. Frequency distribution of the angular difference between objects' BSPs and their nearest PCA/MBB planes

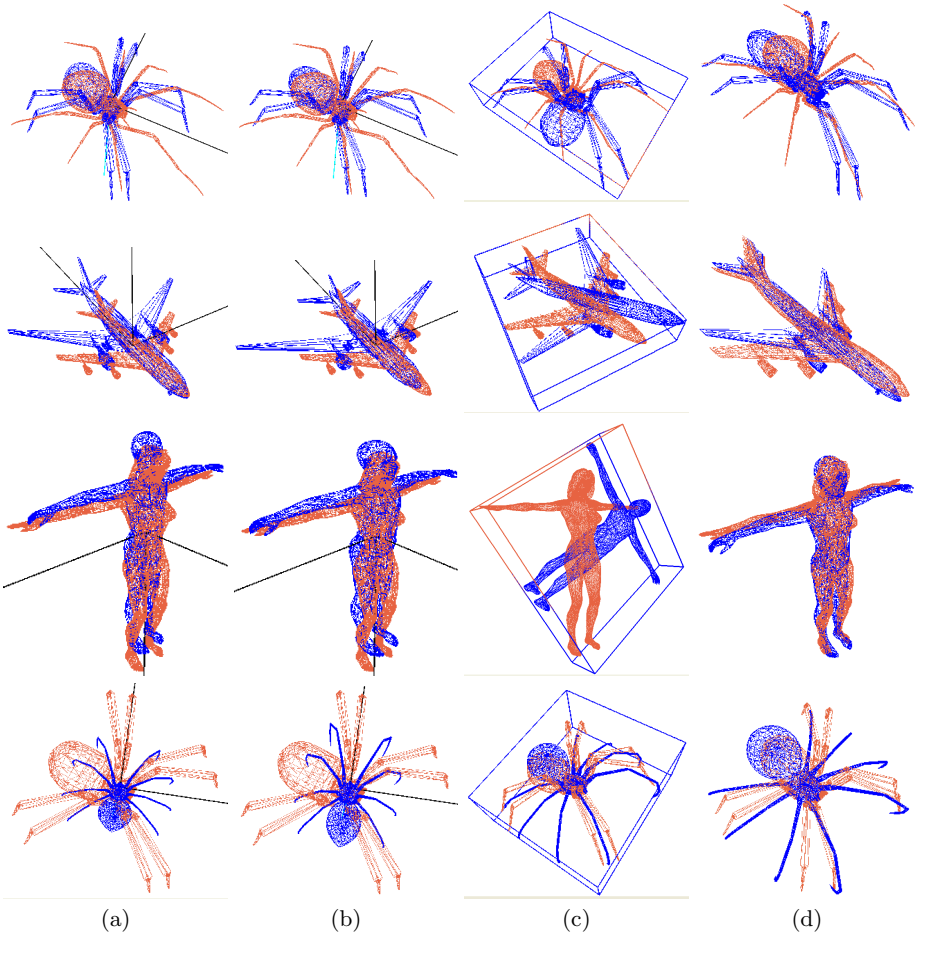


Fig. 7. Comparison of normalization methods. (a) PCA with uniform scaling, (b) PCA with normalization of 3D aspect ratio, (c) MBB, (d) BSP.

Figure 7 illustrates the difference between the various normalization methods. In many cases, both PCA and PCA3 can align the objects' principal axes well (Fig. 7(a, b), rows 1–3). But, sometimes, they give the wrong orientation alignment (Fig. 7(a, b), row 4). They are unable to normalize the 3D aspect ratios well enough to achieve semantically meaningful alignment of the object parts due to PCA's sensitivity to point distributions (Fig. 7(a, b), rows 2–4).

MBB can normalize the 3D aspect ratios relatively well. But, a slight difference in the lengths and widths of the objects can cause the orientation alignment to be off by as much as 90° (Fig. 7(c), rows 1–3). On the other hand, our BSP-based method consistently normalizes and aligns the objects well (Fig. 7(d)). In particular, semantically equivalent parts, such as heads, bodies, legs, wings, and tails, of different objects are correctly aligned.

5 Conclusions

This paper presented a method for computing 3D objects' bilateral symmetry planes (BSPs) and BSP axes and extents, and a method for normalizing and aligning 3D objects based on BSP axes and extents. The algorithm successfully computed the BSPs of 99.2% of the test objects. For exactly bilaterally symmetric objects, the exact BSPs are found. For roughly bilaterally symmetric objects, the best fitting BSPs are computed. Compared with normalization methods based on PCA and minimum volume bounding box, our method based on BSP can normalize and align similar objects in the same category in a semantically meaningful manner, such as aligning the objects' heads, bodies, legs, etc. Better normalization and alignment of objects are expected to improve the performance of shape matching and retrieval algorithms of 3D objects.

Acknowledgment

This research is supported by NUS ARF R-252-000-137-112.

References

1. Kazhdan, M., Funkhouser, T., Rusinkiewicz, S.: Rotation invariant spherical harmonic representation of 3D shape descriptors. In: Proc. Eurographics Symp. on Geometry Proc. (2003)
2. Osada, R., Funkhouser, T., Chazelle, B., Dobkin, D.: Matching 3D models with shape distribution. In: Proc. SMI. (2001)
3. Paquet, E., Rioux, M., Murching, A., Naveen, T., Tabatabai, A.: Description of shape information for 2-D and 3-D objects. Signal Processing: Image Communication **16** (2000) 103–122
4. Tangelder, J.W.H., Veltkamp, R.C.: Polyhedral model retrieval using weighted point sets. In: Proc. SMI. (2003)
5. Vranic, D.V., Saupe, D.: 3D shape descriptor based on 3D Fourier transform. In: Proc. Conf. Digital Signal Proc. Multimedia Comm. and Services. (2001) 271–274

6. Yu, M., Atmosukarto, I., Leow, W.K., Huang, Z., Xu, R.: 3D model retrieval with morphing-based geometric and topological feature maps. In: Proc. IEEE CVPR. (2003) II–656–II–661
7. Atmosukarto, I., Leow, W.K., Huang, Z.: Feature combination and relevance feedback for 3D model retrieval. In: Proc. MMM. (2005) 334–339
8. Barequet, G., Har-Peled, S.: Efficiently approximating the minimum-volume bounding box of a point set in three dimensions. J. Algorithms **38** (2001) 91–109
9. Alt, H., Mehlhorn, K., Wagener, H., Welzl, E.: Congruence, similarity and symmetries of geometric objects. Discrete Computational Geometry (1988) 237–256
10. Wolter, J.D., Woo, T.C., Volz, R.A.: Optimal algorithms for symmetry detection in two and three dimensions. Visual Computer **1** (1985) 37–48
11. Jiang, X.Y., Bunke, H.: A simple and efficient algorithm for determining the symmetries of polyhedra. CVGIP: Graphical Models & Image Proc. **54** (1992) 91–95
12. Jiang, X., Yu, K., Bunke, H.: Detection of rotational and involutional symmetries and congruity of polyhedra. Visual Computer **12** (1996) 193–201
13. Brass, P., Knauer, C.: Computing the symmetries of non-convex polyhedral objects in 3-space. In: Proc. European Workshop on Comp. Geometry. (2002)
14. Brass, P., Knauer, C.: Testing congruence and symmetry for general 3-dimensional objects. Comp. Geometry: Theory and Applications **27** (2004) 3–11
15. Zabrodsky, H., Peleg, S., Avnir, D.: Symmetry as a continuous feature. IEEE Trans. PAMI **17** (1995) 1154–1165
16. Minovic, P., Ishikawa, S., Kato, K.: Symmetry identification of a 3d object represented by octree. IEEE Trans. PAMI **15** (1993) 507–154
17. Sun, C., Sherrah, J.: 3D symmetry detection using the extended gaussian image. IEEE Trans. PAMI **19** (1997) 164–169
18. Jiang, X.Y., Bunke, H.: Determination of the symmetries of polyhedra and an application to object recognition. In: Proc. Comp. Geometry: Methods, Algorithms and Applications (LNCS 553). (1991) 113–121
19. Kazhdan, M., Chazelle, B., Dobkin, D., Finkelstein, A., Funkhouser, T.: A reflective symmetry descriptor. In: Proc. ECCV. (2002) 642–656
20. Kazhdan, M., Funkhouser, T., Rusinkiewicz, S.: Symmetry descriptors and 3D shape matching. In: Proc. Eurographics Symp. on Geometry Proc. (2004) 115–123
21. :(Utrecht Univeristy Object Database, www.cs.uu.nl/centers/give/imaging/3drecog/3dmatching.html)
22. : (Princeton University Object Database, shape.cs.princeton.edu/benchmark)

Extraction of Anatomic Structures from Medical Volumetric Images

Wan-Hyun Cho[1], Sun-Worl Kim[1], Myung-Eun Lee[2], and Soon-Young Park[2]

[1] Department of Statistics, Chonnam National University, Korea
[2] Department of Electronics Engineering, Mokpo National University, Korea
whcho@chonnam.ac.kr, {melee, sypark}@mokpo.ac.kr

Abstract. In this paper, we present the extraction method of anatomic structures from volumetric medical images using the level set segmentation method. The segmentation step using the level set method consists of two kinds of processes which are a pre-processing stage for initialization and the final segmentation stage. In the initial segmentation stage, to construct an initial deformable surface, we extract the two dimensional boundary of relevant objects from each slice image consisting of the medical volume dataset and then successively stack the resulting boundary. Here we adopt the statistical clustering technique consisting of the Gaussian mixture model (GMM) and the Deterministic Annealing Expectation Maximization (DAEM) algorithm to segment the boundary of objects from each slice image. Next, we use the surface evolution framework based on the geometric variation principle to achieve the final segmentation. This approach handles topological changes of the deformable surface using geometric integral measures and the level set theory. These integral measures contain the alignment term, a minimal variance term, and the mean curvature term. By using the level set method with a new defined speed function derived from geometric integral measures, the accurate deformable surface can be extracted from the medical volumetric dataset. And we also use the Fast Matching Method that can reduce largely the computing time required to deform the 3D object model. Finally, we use the marching cubes algorithm to visualize the extracted deformable models. The experimental results show that our proposed method can exactly extract and visualize the human organs from the CT volume images.

1 Introduction

Medical image processing has revolutionized the field of medicine by providing novel methods to extract and visualize 3D deformable models from medical volumetric data, acquired using various acquisition modalities. The extraction of three dimensional objects [1] is one of the most important steps in the analysis of the preprocessed medical image data, which can help diagnosis, treatment planning as well as treatment delivery.

The deformable process is to move a geometric surface toward the tissue type or anatomical structure to be detected. However, owing to the noise corruption and sampling artifacts of medical images, classical 3D segmentation techniques such as

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 86 – 95, 2007.
© Springer-Verlag Berlin Heidelberg 2007

the snake model may cause considerable problems. To address these difficulties, deformable models have been extensively studied and widely used in medical image segmentation with promising results. The deformable models are curves or surfaces defined within an image domain that can move under the influence of internal forces which are defined within the curve or surface itself, and external forces which are computed from the image data. The internal forces are designed to keep the model smooth during deformation. The external forces are defined to move the model toward an object boundary or other desired features within an image. By restricting extracted boundaries to be smooth and incorporating other prior information about the object shape, deformable models offer robustness to both image noise and boundary gaps and allow integrating boundary elements into a coherent and consistent mathematical description.

We are considering the segmentation of the volume dataset using geometric deformable models which are based on evolution theory and level set methods [1]. The evolution theory is to study the deformation process of curves or surfaces using only geometric measures such as the unit normal and curvature. The level set method is a mathematical tool for implementing the evolution theory. This method is used to account for automatic topology adaptation [1, 2], and it also provides the basis for a numerical scheme that is used by geometric deformable models. The curves or surfaces in the level set theory are represented implicitly as a level set of a scalar volume function which is usually defined on the range of the surface model.

Our segmentation framework consists of two stages, namely, a preprocessing technique for initialization and a level set segmentation process for the final segmentation. First, to obtain the proper initial segmentation, we extract the two dimensional boundary of relevant objects from each slice image consisting the medical volume dataset and then we successively stack the resulting boundary. Here we adopt the statistical clustering technique consisting of GMM and DAEM algorithm to segment the boundary of objects at each slice image. Second, we use the surface evolution framework based on geometric variation principle to achieve the final segmentation. This approach handles topological changes of the deformable surface using geometric integral measures and level set theory. This method contains three terms that are called the smoothing term, the alignment term and the minimal variance term.

The outline of this paper is given as follows. In this Section, we describe the general concept of 3D deformable model construction on the medical volumetric dataset. In Section 2, we propose a procedure how to obtain the initial segmentation using a statistical clustering method. And also we consider several measures that can be applied to the surface extraction from volume data, and we think about the problem to conduct the final segmentation combing these measures and the level set theory. In Section 3 we present the experimental results obtained by our algorithm and our conclusions are followed in Section 4.

2 Segmentation Framework

Recently, one method of extracting a deformable surface from a volumetric image is very often to use the level set approach. This approach is based on the theory of

surface evolution and geometric flow. The level set segmentation process has two major stages shown in Figure 1, initialization and level set surface deformation. Each stage is equally important for generating a correct segmentation.

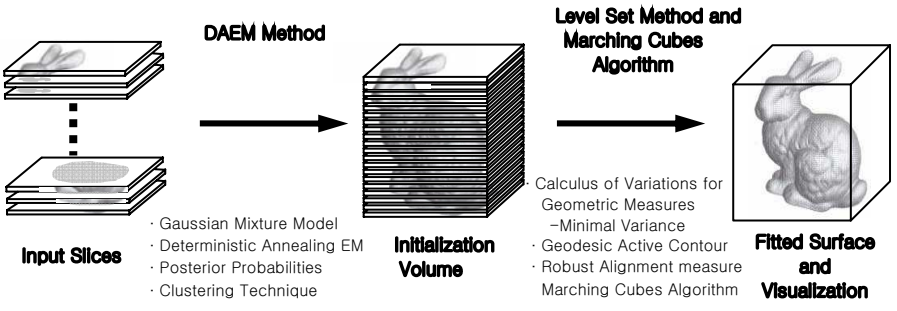


Fig. 1. Level set segmentation procedure

2.1 Initialization of Deformable Surface

Since the deformable model constructed by the level set process moves generally using gradient descent, it seeks local solutions, and therefore the results are strongly dependent on the initial values. Thus, we control the nature of the solution by specifying an initial model from which the deformation process proceeds. One of a various approach for extracting an initial surface model from the volumetric images data is to first extract two dimensional contours from each relevant image slice and then successively stack the resulting two dimensional contours together to form a surface model. Here we use a novel method for initial segmentation of each slice image called the Deterministic Annealing EM segmentation. This method incorporates GMM into DAEM algorithms.

First, we suppose that each slice image consists of a set of disjoint pixels labeled 1 to N , and that it is composed by the K distinct objects or classes. And we let y_i , $i = 1, \cdots, N$ denote the gray values observed from i th pixel. Furthermore, we let $\mathbf{z}_1, \cdots, \mathbf{z}_N$ denote the class indicator vectors for each pixel, where the k th element z_{ik} of $\mathbf{z}_i$ is taken to be one or zero according to the case in which the i th pixel does or does not belong to the k th cluster. Then, the joint probability model for the given image can be represent the following form as

$$p(y_1, \cdots, y_N; \mathbf{z}_1, \cdots, \mathbf{z}_N \mid \Theta, \boldsymbol{\pi}) = \prod_{k=1}^{K} \prod_{i=1}^{N} (\pi_k \phi(y_i; \mu_k, \sigma_k))^{z_{ik}} , \tag{1}$$

Here, in order to use this model for image segmentation, we need to determine the new technique that can be used to obtain the globally optimal estimators for parameter using in GMM. Now, we will use the deterministic annealing Expectation Maximization technique. This algorithm usually processes in the following manner.

Specifically, it starts with initial values β_0 for the temperature parameter β and $(\Theta^{(0)}, \boldsymbol{\pi}^{(0)})$ for the parameter vector $\boldsymbol{\Theta}$ and the prior probabilities $\boldsymbol{\pi}$ for the tissue classification and then we first generate iteratively successive estimates $(\Theta^{(t)}, \boldsymbol{\pi}^{(t)})$ at the given value β by applying the following Annealing E step and M step, for $t = 1,2,\cdots$ and next we repeat the Annealing EM step as we increase the value of temperature β.

DA-E-Step: Here, introducing an annealing parameter β, we consider the following objective function:

$$\vartheta(P_z^{(t)}, \Theta) = E_{P_z^{(t)}}(-\log p(y \mid \mathbf{z}, \Theta) p(z)) + \beta \cdot E_{P_z^{(t)}}(\log P_z^{(t)}). \quad (2)$$

The solution of the minimization problem associated with the generalized free energy in $\vartheta(P_{\mathbf{z}}^{(t)}, \boldsymbol{\Theta})$ with respect to probability distribution $p(\mathbf{z}; \boldsymbol{\pi})$ with a fixed parameter $\boldsymbol{\Theta}$ is the following Gibbs distribution:

$$P_\beta(\mathbf{z} \mid \mathrm{y}, \Theta) = \frac{(p(y \mid \mathbf{z}, \Theta) p(\mathbf{z}))^\beta}{\sum_{z \in \Omega_z} (p(y \mid \mathbf{z}', \Theta) p(\mathbf{z}'))^\beta}. \quad (3)$$

Hence we can obtain a new posterior distribution, $p_\beta(\mathrm{z} \mid \mathrm{y}, \Theta)$ parameterized by β. So, using a new posterior distribution $p_\beta(\mathbf{z} \mid \mathrm{y}, \Theta)$, we can obtain the conditional expectation of the hidden variable z_{ik} given the observed feature data as follows. This is the posterior probability where the i-th pixel belongs to the k-th cluster.

$$\tau_k^{(t)}(y_i) = E(Z_{ik}) = \frac{(\pi_k^{(t-1)} \phi(y_i;\, \mu_k^{(t-1)}, \sigma_k^{(t-1)}))^\beta}{\sum_{j=1}^{K} (\pi_j^{(t-1)} \phi(y_i;\, \mu_j^{(t-1)}, \sigma_j^{(t-1)}))^\beta}. \quad (4)$$

DA-M-Step: Next, we should find the minimum of $\vartheta(P_{\mathbf{z}}^{(t)}, \boldsymbol{\Theta})$ with respect to $\boldsymbol{\Theta}$ with fixed posterior distribution $p_\beta(\mathbf{z} \mid \mathrm{y}, \Theta)$. It means finding the estimates $\boldsymbol{\Theta}^{(t)}$ that minimize $\vartheta(P_{\mathbf{z}}^{(t)}, \boldsymbol{\Theta})$. Since the second term on the right hand side of the generalized free energy in Equation (1) is independent of $\boldsymbol{\Theta}$, we should find the value of $\boldsymbol{\Theta}$ minimizing the first term

$$Q_\beta(\Theta) = E_{p_z^{(t)}}(-\log p(y \mid \mathbf{z}, \Theta) p(\mathbf{z})). \quad (5)$$

From a minimizing trial, we can obtain the following estimators of mixing proportions, the component mean and the variance.

Finally, we can segment each slice image using the posterior probability obtained from the DAEM algorithm. Suppose that a given image consists of a set of the

K distinct objects or clusters $C_1,\cdots,C_K$. We usually segment an image to assign each pixel to the cluster with maximum posterior probability. To do this, we try to find what cluster has the maximum value among the estimated posterior probabilities obtained by using the DAEM algorithm. This is defined as

$$\hat{z}_i = \arg\max_{1\le k\le K} \tau_k^{(t)}(y_i)\ ,\ i=1,\cdots,N\ . \tag{6}$$

Then, we can segment a given slice image using the manner as assigning the i-th pixel to the $\hat{z}_i$-th cluster $C_{\hat{z}}$ having the maximum posterior probability.

2.2 Segmentation of Deformable Surface Using Level Set Method

2.2.1 Representing Deformable Surface with Volumetric Function

When considering a deformable model for segmenting 3D volume data, one option is an implicit level set model. This method specifies a surface S as a level set of a scalar volumetric function

$$\phi : U \to \Re\ ,$$

where $U \subset \Re^3$ is the range of the surface model. Thus, a surface S can be expressed as the following level set:

$$S = \{\mathbf{u} |\ \phi(\mathbf{u}) = k\ \}\ . \tag{7}$$

In other words, S is the set of points that composes the k isosurface of ϕ. Then, one approach to defining a deformable surface from a level set of a volumetric function is to consider that the volumetric function dynamically changes in time. It can mathematically express as

$$\phi(\mathbf{u},t) = k\ . \tag{8}$$

So, our model is based on geometric active surfaces that evolve according to geometric partial differential equations until they stop at the boundaries of the objects. Suppose the geometric surface evolution equation is given by

$$S_t = \frac{\partial S}{\partial t} = F\mathbf{n}\ , \tag{9}$$

where F is any speed quantity that does not depend on a specific choice of parameterization. Then, its implicit level set evolution equation that tracks the evolving surface is given by

$$\phi_t = \frac{\partial \phi}{\partial t} = F|\nabla\phi|\ . \tag{10}$$

In this case, we will take the speed function F as a weighted sum of three integral measures. Hence, to introduce geometric integral measures, we will consider two

types of functional measures that are related via the Green theorem. The first functional is defined along the surface by the general form of

$$E(S) = \int_0^{L_1} \int_0^{L_2} g(S(r,s))drds\,. \tag{11}$$

The second one integrates the value of the volume function $f(x,y,z)$ inside the surface, and is usually referred to as a volume based measure,

$$E(S) = \iiint_{\Omega_S} f(x,y,z)dxdydz\,, \tag{12}$$

where Ω_s is the volume inside the surface S. Formally, we search for the optimal planar surface that maximize the integral measure such as

$$S = \arg\max_S E(S)\,. \tag{13}$$

2.2.2 Geometric Alignment Measure

First, we would like to propagate an initial surface S that would stop as close as possible to an object's boundaries given medical volume images. For this end, we use the geometric functional that is expressed by an inner product between the volume image gradient and the surface normal. The reasonable motivation is that in many cases, the gradient direction is a good estimator for the orientation of the evolving surface. The inner product gets high values if the surface normal aligns with the image gradient direction.

First, we consider that a 3D gray level image is given as a function

$$I(x,y,z) : U \to \Re^+\,,$$

where $U \subset \Re^3$ is the image domain. One of geometric functional measures is the robust alignment term. This is the absolute value of the inner product between the image gradient and the surface normal.

It is given by

$$E_A(S) = \int_0^{L_1} \int_0^{L_2} < \nabla I(x(r,s), y(r,s), z(r,s)), \mathbf{n}(r,s) > drds \tag{14}$$

Here, our goal would be to find curves that maximize this geometric functional. Then, the Euler Lagrange equation gives us the following first variation,

$$\frac{\partial E_A(S)}{\partial S} = sign\,(<\nabla I, \nabla \phi>) \cdot \Delta I \cdot \mathbf{n}, \tag{15}$$

where $\Delta I = I_{xx} + I_{yy} + I_{zz}$ is the image Laplacian.

2.2.3 Minimal Variance Measure

The second geometric functional measure is a minimal variation criterion which was proposed by Vese and Chan [7]. It penalizes lack of homogeneity inside and outside the evolving surface. This functional measure is given by

$$E_{MV}(S) = \iiint_{\Omega_S} (I(x,y,z) - c_1)^2 \, dxdydz + \iiint_{\Omega \setminus \Omega_S} (I(x,y,z) - c_2)^2 \, dxdydz , \tag{16}$$

where S is the surface separating the two regions, Ω_S is the interior of the surface, and Ω / Ω_S is the exterior of the surface. Here, this measure will have induced the optimal surface that can best separate the interior and the exterior respectively of the evolving surface.

In the optimal process we look for the best separating surface as well as for the optimal expected values c_1 and c_2. Then, the first variation minimizing this functional is given as the mean intensity values of the image in the interior and exterior respectively of the surface S. So, the first variation equation is

$$\frac{\partial E_{MV}(S)}{\partial S} = (c_2 - c_1)(I - \frac{c_1 + c_2}{2}) \cdot \mathbf{n} \cdot \tag{17}$$

2.2.4 Geodesic Active Surface

One of the functionals related with these measures is known as the geodesic active surface model. This model was introduced in Caselles, Kimmel and Sapiro [4] as a geometric alternative for the snakes. The model is derived by a variation principle from a geometric measure and it is defined by

$$E_G(S) = \int_0^{L_1} \int_0^{L_2} g(S(r,s)) dr ds . \tag{18}$$

If the function $g(x,y,z)$ is given like as $g(x,y,z) = 1/(1+|\nabla I(x,y,z)|^2)$, then it is an integration of an inverse edge indicator function along the surface. The search would be for a surface along which the inverse edge indicator gets the smallest possible values. That is, we would like to find the surface S that minimizes this functional.

The geodesic active surface usually serves as a good regularization term in noisy image. The Euler Lagrange equation known as gradient descent process is given by the following evolution equation

$$\frac{\partial E_G(S)}{\partial S} = (g(S)\kappa - < \nabla g, \mathbf{n} >) \cdot \mathbf{n} \cdot \tag{19}$$

Here, κ is the mean curvature of the surface.

2.2.5 The Proposed Measure for Active Surface

In general, the speed term F represents the speed of the evolving surface in the direction of the normal to the surface. Here we will use the speed function in the evolving surface as a weighted sum of three geometric functional measures. It is given as

$$F = \alpha(g\kappa - < \nabla g, \mathbf{n} >) + sign(< \nabla I, \mathbf{n} >) \cdot \Delta I + \beta(c_2 - c_1)(I - (c_1 + c_2)/2). \tag{20}$$

So, the corresponding level set formulation of our surface evolution is

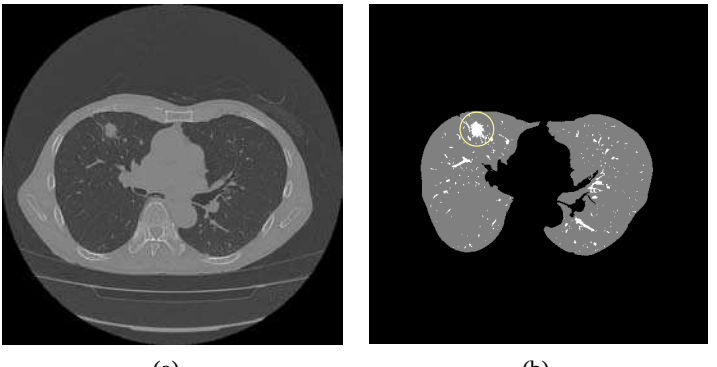
$$\phi_t = \phi_{t-1} + \Delta t\left(\alpha\{div(g(x,y,z)\frac{\nabla\phi}{|\nabla\phi|})\} + sign(< \nabla I, \nabla\phi >) \cdot \Delta I + \beta(c_2 - c_1)(I - \frac{c_1 + c_2}{2})\right)|\nabla\phi|. \tag{21}$$

Finally, for the solution of the partial differential equation to be both consistent and stable, it should be guaranteed that the small error in the approximation is not amplified as the solution is marched forward in time. That is, the stability of solution can be preserved by using the Courant-Friedreichs-Lewy (CFL) condition, which asserts that the numerical curves or surfaces should move at least in one grid cell at each time step. This gives us the CFL time step restriction of

$$\Delta t \le \frac{1}{\max\ |F| \cdot |\nabla\phi|}. \tag{22}$$

3 Experimental Results

To assess the performance of the level set procedure, we have conducted the experiment on the 3D volumetric dataset which consists of 2D slices by successively stacking one on top of the other. Fig. 2(a) shows one slice of 512x512x335 original

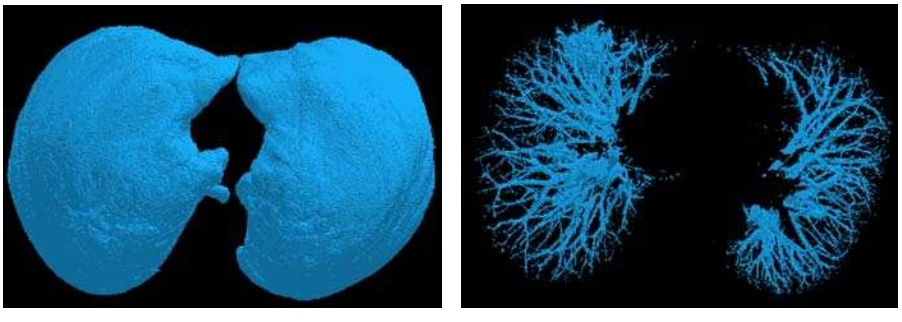


(a) (b)

Fig. 2. 2D lung CT image: (a) one slice of a 512x512x335 original lung CT image, (b) the segmented lung region

lung CT images and Fig. 2(b) shows the lung region obtained by applying the DAEM segmentation method to the original CT image in Fig. 2(a). Then the volumetric image is produced by stacking the original CT images, being the interior of the segmented lung region, and it is used as the initial model for the level set segmentation.

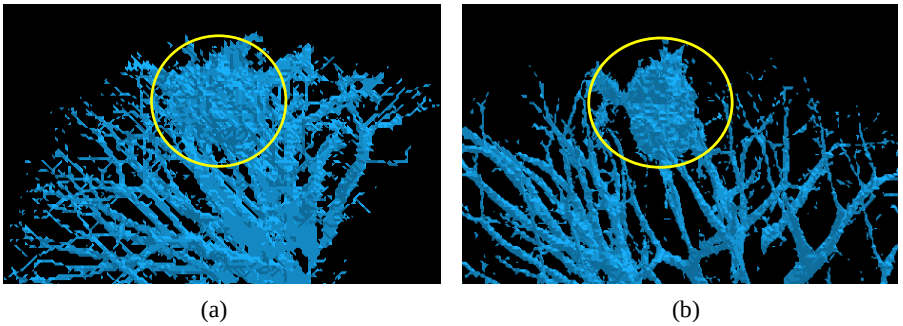
To display the three-dimensional volume datasets visibly, we have developed the deformable surface visualization technique. Here we used the Marching Cubes algorithm to create a surface by generating a set of three-dimensional triangles, each of which is an approximation to a piece of the iso-surface. Fig. 3(a) shows the 3D rendering result of the lung surface. Note that the lung surface is well visualized with smoothed surface. The 3D level set procedure is applied to the volumetric image with initial surfaces of two small balloons being located individually on either side of the lung region. We can observe that the anatomical object is well rendered after being extracted from the volumetric image as shown in Fig. 3(b).



(a) (b)

Fig. 3. 3D rendering result: (a) lung surface, (b) anatomic objects

In this image the blood vessels are well visualized and an object in a mass being suspected as a pulmonary tumor is also noted in the left upper end of the blood vessels. Fig. 4 shows the partly zoomed result of the location of the suspicious tumor and the corresponding tumor location is depicted as the circle on the 2D lung slice in Fig. 2(b).



(a) (b)

Fig. 4. Partly zoomed result of Fig. 3(b): (a) same view point, (b) other view point

4 Conclusion

In this paper, we have presented the extraction method of anatomic structures from volumetric medical images using the level set segmentation method. The segmentation procedure consists of a pre-processing stage for initialization and the final segmentation stage. In the initial segmentation stage, we have adopted the statistical clustering technique consisting of GMM and DAEM algorithms to segment the boundary of objects from each slice image. Next, we have used the surface evolution framework based on the geometric variation principle to achieve the final segmentation. This approach handles topological changes of the deformable surface using geometric integral measures and the level set theory. Finally, we have shown the 3D rendering results of the extracted anatomical objects by employing the marching cubes algorithm. The experimental results show that our proposed method can extract the anatomical objects from the CT volumetric images in an exact manner and visualize them for detail analysis.

References

1. R. Whitaker, D. Breen, K. Museth, and N. Soni.: Segmentation of Biological Volume Datasets Using a Level Set Framework, Volume Graphics (2001) 249-263.
2. S. G. Armato and W. F. Sensakovic.: Automated lung Segmentation for Thoracic CT, Academic Radiology, volume 11 (2004) 1011-1021.
3. Y. Itai and etc.: Automatic extraction of abnormal areas on CT images of the lung area, 2005 Inter. Sym. On Advanced Intelligent Systems, volume 1 (2005) 360-392.
4. R. Kimmel.: Fast Edge Integration, Geometric Level Set Methods, Springer (2003) 59-77.
5. J. A. Sethian.: Level Set Methods and Fast Marching Methods, Cambridge university press (2005).
6. M. Sonka and J. M. Fitzpatick.: Handbook of Medical imaging: Volume2 Medical Image Processing and Analysis, SPIE Press (2000).
7. L. A. Vese and T. F. Chan.: A Multiphase Level Set Framework for Image Segmentation Using the Mumford and shah model", International Journal of Computer Vision, Vol. 50 (2002) 271-293.

Dual-Space Pyramid Matching for Medical Image Classification

Yang Hu[1,⋆], Mingjing Li[2], Zhiwei Li[2], and Wei-ying Ma[2]

[1] University of Science and Technology of China, Hefei 230027, China
yanghu@ustc.edu
[2] Microsoft Research Asia, No 49, Zhichun Road, Beijing 100080, China
{mjli,zli,wyma}@microsoft.com

Abstract. With the increasing of medical images that are routinely acquired in clinical practice, automatic medical image classification has become an important research topic recently. In this paper, we propose an efficient medical image classification algorithm, which works by mapping local image patches to multi-resolution histograms built both in feature space and image space and then matching sets of features though weighted histogram intersection. The matching produces a kernel function that satisfies Mercer's condition, and a multi-class SVM classifier is then applied to classify the images. The dual-space pyramid matching scheme explores not only the distribution of local features in feature space but also their spatial layout in the images. Therefore, more accurate implicit correspondence is built between feature sets. We evaluate the proposed algorithm on the dataset for the automatic medical image annotation task of ImageCLEFmed 2005. It outperforms the best result of the campaign as well as the pyramid matchings that only perform in single space.

1 Introduction

Due to the rapid development of biomedical informatics, medical images have become an indispensable investigation tool for medical diagnosis and therapy. A single average size radiology department may produce tens of tera-bytes of data annually. The ever-increasing amount of digitally produced images require efficient methods to archive and access this data. One of the most important issues is to categorize medical images, which is a prerequisite to subsequent processing steps since it allows content-specific selection of appropriate filters or algorithmic parameters [1]. Manual classification of images is time-consuming. Besides, since annotating medical images can only be done by doctors with special expertise, manual classification of medical images should be more expensive than in other image classification problems. Therefore, automatic classification techniques become imperative for a variety of medical systems.

⋆ This work was performed when the first author was a visiting student at Microsoft Research Asia.

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 96–105, 2007.
© Springer-Verlag Berlin Heidelberg 2007

Recently, a class of local descriptor based methods, which represent an image with an collection of local photometric descriptors, have demonstrated impressive level of performance for object recognition and classification. And this kinds of algorithms have also been explored for medical image classification, considering that most information in medical images is local [1]. Unlike global features, local features are always unordered. Different images are represented by different number of local descriptors and the correspondence between the features across different images is unknown. Therefore, it is challenging to apply this kind of representation to discriminative learning, which usually operates on fixed-length vector inputs. Many recent works have devoted to leverage the power of both local descriptor and discriminative learning [2][3][4].

In this work we propose a dual-space pyramid matching kernel for medical image classification, which is inspired by Grauman and Lazebnik's works [2][3][4]. The pyramid match kernel proposed by Grauman and Darrell [2][3] performs multi-resolution matching of local features through weighted histogram intersection. The matching is conducted in feature space and the information about the spatial layout of the features is discarded. Lazebnik's spatial pyramid matching [4], on the other hand, considers rough geometric correspondence between local descriptors and constructs pyramid in image space. However, it loses the multi-level scalability in feature space. Our algorithm integrates these two algorithms in a systematic way: it embeds feature space pyramid matching in multi-level image pyramid, and therefore builds more accurate implicit correspondence between feature sets. We evaluate our algorithm on the dataset for the automatic medical image annotation task of ImageCLEFmed 2005 [6]. It outperforms the best result of the campaign as well as Grauman and Lazebnik's single space pyramid matchings.

The rest of this paper is organized as follows. In Sect.2, we introduce some related works on medical image classification. In Sect.3, we describe the original formulation of pyramid matching in feature space. The spatial pyramid matching and the proposed dual-space pyramid matching algorithm are presented in Sect.4. We report the experiment results in Sect.5. Finally, we conclude in Sect.6.

2 Related Works

While most previous experiments on medical image classification have been restricted to a small number of categories, a great effort has recently been made to evaluate this task on a larger dataset with more predefined categories. ImageCLEF, which conducts evaluation of cross-language image retrieval, has come up with an automatic medical image annotation task in 2005 [6]. It provided a dataset (Fig. 1) consisting of 9000 fully classified radiographs, which were taken randomly from medical routine, for participants to train a classification system. These images were classified into 57 categories according to image modality, body orientation, body region and the biological system examined. 1000 additional radiographs for which classification labels were unavailable to participants

were used to evaluate the performance of various algorithms. In total, 41 runs were submitted by 12 groups last year, with error rates ranging from 12.6% to 73.3% [6].

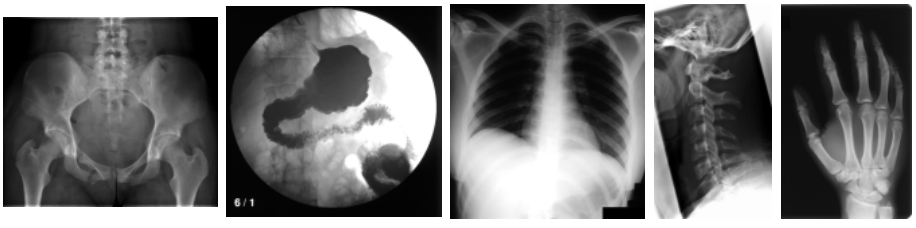


Fig. 1. Example images from the IRMA database[6]

The submission which obtained the minimum error rate applied a two dimensional distortion model to the comparison of medical images [7]. The model matched the local image context of a test image to the best fitting counterpart in a training image. The distance between two images was the accumulation of the differences between the matched pairs. Deselaers et al. [8] used an object recognition and classification approach to classify the test images. It first extracted image patches from interest points and clustered them into groups. Then it trained a discriminative log-linear model for the histograms of the image patches. Similarly, Marée et al. [9] extracted square sub-windows of random sizes at random positions from training images and then built an ensemble of decision trees upon them. The performances of these two methods were almost the same and were approximate to the best one. However, neither of them considered the geometrical relation between the extracted patches, which was valuable for medical image classification. Some other runs used global features, such as texture, edge and shape features, to describe medical images. Their performances were not as good as the previous ones[6].

3 Feature Space Pyramid Matching

The pyramid matching proposed by Grauman and Darrell [2] works by mapping the feature vectors to multi-resolution histograms and then compare the histograms with weighted sum of histogram intersection. To build the multi-resolution histograms, it doubles the number of bins along each dimension of the feature space iteratively, which results in a sequence of increasingly finer grids that are uniformly shaped over feature space (Fig. 2(a)). The idea is simple and intuitive. However, it fails to capture the structure of the feature space as feature vectors are usually distributed non-uniformly. A more reasonable way to build the multi-resolution histograms is through hierarchical clustering, which would be more consistent with the underlying feature distribution. Following the

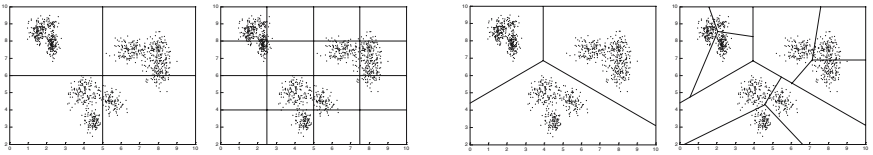


Fig. 2. Two ways to partition the 2-dimensional feature space [3]

idea of "vocabulary tree" proposed by Nistér and Stewénius [10], more flexible multi-resolution histograms can be constructed (Fig. 2(b)) [3].

A subset of feature vectors are randomly selected from training images to learn the hierarchical clusters. We use hierarchical GCS (Growing Cell Structure) neural network [11], which is able to detect high dimensional patterns with any probability distribution and is a high speed algorithm, to build the hierarchical structure. Other hierarchial clustering algorithms are also possible and do not change the overall scheme.

First, c initial cells are learned from the selected training feature vectors. They constitute the top level clusters of the hierarchical structure. The training features are partitioned into c groups by mapping each feature vector to the best matching cell. Then we recursively apply GCS to each group of feature vectors, i.e. partition the vectors belonging to the same group into k sub-groups, where k is a branching factor (number of children of each node). This process repeats $L^F - 1$ times, producing a tree with L^F levels (the superscript F indicates "feature space"). The first level contains c nodes and level l, $l = 2, \ldots, L^F$, contains $ck^{(l-1)}$ nodes. Each level will later produce a histogram whose dimension is the number of nodes at that level.

Once we have built the cluster hierarchy, we can map feature vectors to it. A vector is first mapped to the best matching cell at the top level of the tree, then it is assigned to the closest subcell among the children of the matched cell. This process repeats recursively and propagates the feature vector from the the root to the leaf of the tree. The path down the tree can be recorded by L^F integers (L^F-dimensional path-vector) that indicate which node is chosen at each level of the tree. After a set of feature vectors have been mapped to the cluster hierarchy, we can build the multi-resolution histograms, i.e. the pyramid in feature space, by counting the number of feature vectors belonging to each node of the tree.

Let X and Y be two sets of vectors in feature space. Their histograms at level l are denoted by H_X^l and H_Y^l with $H_X^l(i)$ and $H_Y^l(i)$ indicating the number of feature vectors from X and Y that fall into the ith bin of the histograms. The number of features that match in the ith bin at level l is given by the "overlap" between the two bins [2]:

$$\mathcal{M}\left(H_X^l(i), H_Y^l(i)\right) = \min\left(H_X^l(i), H_Y^l(i)\right) \quad . \tag{1}$$

Due to the hierarchical character of the clusters, matches found in the ith bin at level l also include all the matches found in its child bins at the finer level $l+1$. Therefore, the number of new matches is given by

$$\mathcal{N}\left(H_X^l(i), H_Y^l(i)\right) = \mathcal{M}\left(H_X^l(i), H_Y^l(i)\right) - \sum_{j=1}^{k} \mathcal{M}\left(c_j\left(H_X^l(i)\right), c_j\left(H_Y^l(i)\right)\right) \ , \tag{2}$$

where $c_j\left(H_X^l(i)\right)$ and $c_j\left(H_Y^l(i)\right)$ denote the number of feature vectors that fall into the jth child bins of $H_X^l(i)$ and $H_Y^l(i)$ respectively [3].

The similarity between X and Y is defined as the weighted sum of the number of new matches found at each level of the pyramid [2][3]:

$$\begin{aligned} K^F(X,Y) &= \sum_{l=1}^{L^F} \sum_{i=1}^{ck^{(l-1)}} q_{li}^F \mathcal{N}\left(H_X^l(i), H_Y^l(i)\right) \\ &= \sum_{l=1}^{L^F} \sum_{i=1}^{ck^{(l-1)}} q_{li}^F \left(\mathcal{M}\left(H_X^l(i), H_Y^l(i)\right) - \sum_{j=1}^{k} \mathcal{M}\left(c_j\left(H_X^l(i)\right), c_j\left(H_Y^l(i)\right)\right) \right) \\ &= \sum_{l=1}^{L^F} \sum_{i=1}^{ck^{(l-1)}} (q_{li}^F - p_{li}^F) \mathcal{M}\left(H_X^l(i), H_Y^l(i)\right) \\ &= \sum_{l=1}^{L^F} \sum_{i=1}^{ck^{(l-1)}} w_{li}^F \mathcal{M}\left(H_X^l(i), H_Y^l(i)\right) \ , \end{aligned} \tag{3}$$

where q_{li}^F refers to the weight associated with the ith bin at level l and p_{li}^F refers to the weight for the parent bin of that node. Let $w_{li}^F = q_{li}^F - p_{li}^F$.

Intuitively, matches found in smaller bins are weighted more than those found in larger bins, because the matched pais in smaller bins are more likely to be similar. Besides, in order to make the similarity measure $K^F(X,Y)$ eligible for kernel-based discriminative learning, it must be a positive semi-definite kernel (Mercer kernel). Since the min operation is positive-definite, and since Mercer kernels are closed under addition and scaling by a positive constant [12], we require that $q_{li}^F \geq p_{li}^F$ or the weight of every child bin should be greater than that of its parent bin. We follow the weighting scheme adopted by the original pyramid matching [2] and set the weights for the bins at level l as 2^{l-L^F} ($q_{li}^F = 2^{l-L^F}$), $l = 1, 2, \ldots, L^F$. So we have

$$w_{li}^F = \begin{cases} 2^{l-L^F} & l = 1 \\ 2^{l-L^F-1} & l > 1 \end{cases} \ . \tag{4}$$

Although it might be more reasonable to set the weight inversely proportional to the diameter of the bin, i.e. the maximal pairwise distance between the vectors in that cell [3], it is time-consuming to calculate the diameters when the training

feature set is large, and this measure didn't show any apparent advantage over the former simple one in our informal experiments.

4 Dual-Space Pyramid Matching

Inspired by Grauman and Garrell's work [2], Lazebnik et al. [4] advocates to perform pyramid matching in the two-dimensional image space. They partition the images into increasingly fine sub-regions, then compute and compare histograms of local features found inside each sub-region (Fig. 3). Through incorporating spatial layout of local features into pyramid matching, they achieve significant performance improvement on scene categorization task. The geometric information of local features is extremely valuable for medical images, since the objects are always centered in the images and the spatial layouts of the anatomical structures in the radiographs belonging to the same category are quite similar. Therefore, we can expect promising results using this spatial matching scheme. However, in feature space, the spatial matching simply use non-hierarchical clustering techniques and assume that only features of the same type can be matched to one another. As a result, the scalable property in feature space is discarded. The relations between features of different types are missing, while in feature space pyramid matching, features that are not matched in finer resolution are possible to be matched in coarser scale. Therefore, it should be profitable to combine feature space and image space pyramid matching together, and we regard this unified multi-resolution framework as dual-space pyramid matching.

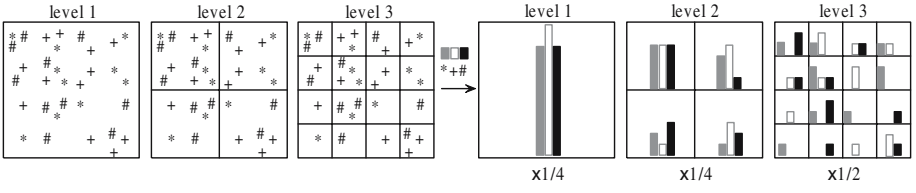


Fig. 3. Toy example of constructing a three-level spatial pyramid. The image has three types of features, indicated by asterisks, crosses and pounds. At the left side, the image is subdivided at three different levels of resolution. At the right, the number of features that fall in each sub-region is counted. The spatial histograms are weighted during matching [4].

Intuitively, the major improvement of dual-space matching over spatial matching is that it applies hierarchial clustering to the feature vectors in each sub-region and single-level histogram intersection in feature space is replaced by pyramid matching. Compared with the feature space matching in Sect.3, the new scheme extends to perform it in multi-resolution sub-regions. More specifically, if the height of the pyramid in image space is L^I (the superscript I indicates

"image space"), the dual-space pyramid matching for feature sets X and Y is given by

$$K^D(X,Y) = \sum_{l_1=1}^{L^I} \sum_{i=1}^{4^{(l_1-1)}} w_{l_1 i}^I \sum_{l_2=1}^{L^F} \sum_{j=1}^{ck^{(l_2-1)}} w_{l_2 j}^F \mathcal{M}\left(H_X^{l_1 l_2}(i,j), H_Y^{l_1 l_2}(i,j)\right)$$
$$= \sum_{l_1=1}^{L^I} \sum_{i=1}^{4^{(l_1-1)}} \sum_{l_2=1}^{L^F} \sum_{j=1}^{ck^{(l_2-1)}} w_{l_1 i}^I w_{l_2 j}^F \mathcal{M}\left(H_X^{l_1 l_2}(i,j), H_Y^{l_1 l_2}(i,j)\right) , \quad (5)$$

where $w_{l_1 i}^I$ refers to the weight for the ith region at level l_1 of the spatial pyramid, and is defined the same as (4). $H_X^{l_1 l_2}(i,j)$ and $H_Y^{l_1 l_2}(i,j)$ indicate the number of feature vectors from X and Y that fall into the ith region at level l_1 of the spatial pyramid and the jth bin at level l_2 of the pyramid in feature space.

For each feature vector, we first obtain its L^F-dimensional path-vector in feature space and the L^I-dimensional path-vector in image space, then we could get the indexes of the $L^F L^I$ bins it belongs to and increase their counts. Afterwards, K^D could be implemented as a single histogram intersection of "long" vectors formed by concatenating the histograms of all resolutions in feature space in all sub-regions of the images. The weight of each bin in the single histogram is the product of the corresponding weights in the two spaces. Although the index of the single histogram may be as high as $\sum_{l_1=1}^{L^I} 4^{l_1-1} \times c \sum_{l_2=1}^{L^F} k^{l_2-1}$, the histogram for each image is quite sparse, because the number of non-zero bins is at most $mL^F L^I$, where m is the number of local features extracted from the image and is far less than the number of clusters in the feature space pyramid. Another implementation issue is normalization. In order not to favor large feature sets, which would always yield high similarity due to the intersection operation, we should normalize the histograms by the total weight of all features in the images before conducting matching.

5 Experiments

In this section, we examine the effectiveness of the dual-space pyramid matching on medical image classification task. It is evaluated on the dataset for the automatic medical image annotation task of ImageCLEFmed 2005, using the same experimental setting, i.e. 9000 medical images that belong to 57 differnet categories are used for training and 1000 additional images are used to test the algorithms [6]. Multi-class classification is done with a "one-against-one" SVM classifier [13] using the dual-space pyramid matching kernel.

Although SIFT descriptor [5] has been proven to work well for common object and nature scene recognition, its power to describe radiographs is somewhat limited. Since the scale and rotation variations in radiographs of the same category are small, the SIFT descriptor can not show its advantage of being scale and rotation invariant for describing radiographs. In previous works, local image patches have shown pleasant performance for medical image retrieval and classification [7][8][9]. Therefore, we utilize local image patches as the local features in

our experiments. Before feature extraction, we resize the images so that the long sides are 200 pixels and their aspect ratios are maintained. The positions of the local patches are determined in two ways. Local patches are first extracted from interest points detected by DoG region detector [5], which are located at local scale-space maxima of the Difference-of-Gaussian. We also extract local patches from a uniform grid spaced at 10×10 pixels. This dense regular description is necessary to capture uniform regions that are prevalent in radiographs. We use 11×11 pixel patches in our experiments, i.e. 121-dimension feature vectors. And about 400 patches are extracted from each image.

We first compare the performance of the pyramid matchings conducted in different spaces (Table 1). For feature space pyramid matching, we build a three-level pyramid in feature space with $c = 100$ and branch factor $k = 5$. No image partition is involved in this case, i.e. the matching of local features is conducted on the whole image. In spatial pyramid matching, the height of the pyramid in image space is also set to 3. We use the highest level (level 3) of the pyramid built in feature space to cluster the local features into 2500 cells, and then perform non-hierarchical matching in each sub-region. The dual-space pyramid matching is a combination of the previous two methods. Three-level pyramids are built in both feature space and image space ($c = 100$ and $k = 5$ in feature space). We conduct 10-fold cross-validation on the training set (9000 images in total) and also examine their performance on the 1000 test images of the campaign. According to Table 1, the spatial pyramid matching is much more effective than feature space pyramid matching, which confirms our observation that the geometric information of the local features is extremely valuable for medical images. The dual-space pyramid matching successfully combined the advantages of the other two algorithms and achieved the best performance. It fully exploited the distribution of local features in both feature space and image space, and thus built more accurate implicit correspondence between feature sets.

Table 1. Performance comparison of pyramid matching kernels on medical image classification

Method	*Error Rate*	
	10-fold Cross-Validation	*Test Set*
Feature Space Pyramid Matching	19.0%	18.2%
Spatial Pyramid Matching	12.4%	12.1%
Dual-Space Pyramid Matching	11.5%	11.2%

We then compare the performance of dual-space pyramid matching with the results obtained by other groups that participated the evaluation in ImageCLEFmed 2005 [6]. As shown in Table 2, the dual-space pyramid matching outperforms the best result of the campaign which applied a two dimensional distortion model to the comparison of medical images [7]. Deselaers et al.'s discriminative training of log-linear models for image patches obtained the third

rank [8]. And Marée et at.'s algorithms [9] that were based on ensemble of decision trees and random sub-windows ranked forth and sixth in the list. All of these algorithms also use local patches to describe the images. The nearest neighbor classifier that compared the images down-scaled to 32×32 pixels using Euclidean distance served as the baseline and resulted with 36.8% error rate. The improvement of the proposed dual-space pyramid matching over other methods is statistically significant, which demonstrates the effectiveness of this algorithm.

Table 2. Resulting error rate on medical image classification [6]

Rank	*Method*	*Error Rate*
-	*Dual-Space Pyramid Matching*	11.2%
1	1NN+IDM [7]	12.6%
2	1NN+CCF+IDM+Tamura	13.3%
3	Discriminative Patches [8]	13.9%
4	Random Subwindows+Tree Boosting [9]	14.1%
5	MI1 Confidence	14.6%
6	Random Subwindows+Extra-Trees [9]	14.7%
7	Gift 5NN8g	20.6%
⋮	⋮	⋮
28	Nearest Neighbor, 32×32, Euclidian	36.8%
⋮	⋮	⋮
42	Texture Directionality	73.3%

6 Conclusions

We have proposed a dual-space pyramid matching kernel that is eligible for discriminative training with sets of local features. It combines the feature space pyramid matching and spatial pyramid matching in a systematic way. It explores the distribution of local features in feature space as well as their geometric information in image space. A more accureate implicit correspondence is built between sets of local features through computing a weighted intersection of multi-resolution histograms that span two spaces. The algorithm is computationally efficient since the match requires time linear in the number of features. We have applied our algorithm to medical image classification and evaluated its performance on the dataset for the automatic medical image annotation task of ImageCLEFmed 2005. It outperforms the best result of the campaign as well as the pyramid matchings that only perform in single space. In our future work, we plan to conduct more experiments to examine the influence of different parameter settings, which would further reveal the interaction between the matchings in the two spaces. Although the algorithm is developed for medical image classification, it is also applicable to other object recognition and classification problems. We will evaluate its performance on other datasets later.

Acknowledgements. We would like to thank Menglei Jia for helpful discussion and for providing the code of GCS clustering. We also thank Dr. Thomas Lehmann et al., Dept. of Medical Informatics, RWTH Aachen, Germany, for providing the IRMA database.

References

1. Lehmann, T.M., Güld, M.O., Deselaers, T., Keysers, D., Schubert, H., Spitzer, K., Ney, H., Wein, B.B.: Automatic Categorization of Medical Images for Content-based Retrieval and Data Mining. Computerized Medical Imaging and Graphics, volume 29, pages 143-155, , 2005.
2. Grauman, K., Darrell, T.: The Pyramid Match Kernel: Discriminative Classification with Sets of Image Features. Proceedings of the IEEE International Conference on Computer Vision (ICCV 2005), Beijing, China, October 2005.
3. Grauman, K., Darrell, T.: Approximate Correspondences in High Dimensions. MIT CSAIL Technical report, MIT-CSAIL-TR-2006-045, June 2006.
4. Lazebnik, S., Schmid, C., Ponce, J.: Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2006), New York, June 2006.
5. Lowe, D.G.: Distinctive Image Features from Scale-Invariant Keypoints. International Journal of Computer Vision, 60, 2 (2004), pp. 91-110.
6. Deselaers, T., Müeller, H., Clough, P., Ney, H., Lehmann, T.M.: The CLEF 2005 Automatic Medical Image Annotation Task. International Journal of Computer Vision, 2006(in press).
7. Keysers, D., Gollan, C., Ney, H.: Classification of Medical Images using Non-linear Distortion Models. Bildverarbeitung für die Medizin 2004 (BVM 2004), Berlin, Germany, pages 366-370, March 2004.
8. Deselaers, T., Keysers, D., Ney, H.: Discriminative Training for Object Recognition Using Image Patches. Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR 2005), San Diego, CA, June 2005.
9. Marée, R., Geurts, P., Piater, J., Wehenkel, L.: Biomedical Image Classification with Random Subwindows and Decision Trees. Proceedings of ICCV workshop on Computer Vision for Biomedical Image Applications (CVIBA 2005), Beijing, China, October 2005.
10. Nistér, D., Stewénius, H.: Scalable Recognition with a Vocabulary Tree. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2006), New York, June 2006.
11. Fritzke, B.:Growing Cell Structures – A Self-Organizing Network in k Dimensions. Artificial Neural Networks II, pages 1051-1056, 1992.
12. Shawe-Taylor, J., Cristianini, N.: Kernel Methods for Pattern Analysis. Cambridge Univ. Press, 2004.
13. Chang, C.-C., Lin, C.-J.: LIBSVM : A Library for Support Vector Machines, 2001. Software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm .

An Image Registration Method Based on the Local and Global Structures

Nan Peng[1], Zhiyong Huang[1,2], and Zujun Hou[2]

[1] School of Computing, National University of Singapore
[2] Institute for Infocomm Research (I^2R), Singapore
huangzy@comp.nus.edu.sg, zyhuang@i2r.a-star.edu.sg

Abstract. We propose a novel feature-based image registration method using both the local and global structures of the feature points. To address various imaging conditions, we improve the local structure matching method. Compared to the conventional feature-based image registration methods, our method is robust by guaranteeing the high reliable feature points to be selected and used in the registration process. We have successfully applied our method to images of different conditions.

Keywords: Multimedia Content Analysis, Multimedia Signal Processing, Image Registration.

1 Introduction

Image registration, an important operation of multimedia systems, is a process of transforming the different images of the same scene taken at different time, from different view points, or by different sensors, into one coordinate system. The current automated registration techniques can be classified into two broad categories: area-based and feature-based [1, 5].

In this paper, we propose and implement a novel image registration method to improve the quality of registration by guaranteeing the high reliable feature points to be selected and used in the registration process. Here we adapt the feature matching method proposed by Jiang and Yau [4]. However, it is mainly for fingerprint image under rotation and translation. We modify it so that we can obtain a set a reliable corresponding feature points for images of various conditions. The major contributions are: (1) we improve the quality of registration by applying a more reliable feature point selection and matching algorithm adapted from finger print matching, (2) we improve the local structure matching method, and (3) we implement the method in a software system and conduct various experiments with good results.

2 Our Work

In this section, we describe how to extract the feature points and estimate their orientation (2.1), find correct matching pairs between two partially overlapping images (2.2), and derive the correct transformation between two images (2.3).

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 106 – 115, 2007.
© Springer-Verlag Berlin Heidelberg 2007

2.1 Feature Point Detection and Orientation Estimation

In our approach, the features are defined as points of large eigenvalues in the image. We employ the OpenCV function GoodFeaturetoTrack [3]. A number of methods have been proposed for orientation estimation of the feature points. We apply the least mean square estimation algorithm. A feature point is eliminated if its reliability of the orientation field is below a threshold.

2.2 Feature Point Matching

There are four major steps in our matching algorithm: an invariant feature descriptor to describe the local positional relations between two feature points (2.2.1), local (2.2.2) and global (2.2.3) structure matching, and cross validation to eliminate the false matching pairs (2.2.4). In (2.2.2), we describe our improvement.

2.2.1 Define a Feature Descriptor

We first represent each feature point i detected by a feature vector f_i as:

$$f_i=(x_i,y_i,\varphi_i), \tag{1}$$

where (x_i,y_i) is its coordinate, φ_i is the orientation. The feature vector f_i represent a feature point's global structure. A feature descriptor F_{ij} is defined to describe the local positional relations between two feature points f_i and f_j by their relative distance d_{ij}, radial angle θ_{ij} and orientation difference φ_{ij} (see Fig. 1) as equation (2):

$$F_{ij}=\begin{bmatrix} d_{ij} \\ \theta_{ij} \\ \varphi_{ij} \end{bmatrix}=\begin{bmatrix} \sqrt{(x_i-x_j)^2+(y_i-y_j)^2} \\ d\phi(\arctan\dfrac{y_i-y_j}{x_i-x_j},\varphi_i) \\ d\phi(\varphi_i,\varphi_j) \end{bmatrix}, \tag{2}$$

where $d\phi(t_1,t_2)$ is a function to compute the difference between two angles t_1 and t_2. All these terms are shown in Fig. 1 for two feature points.

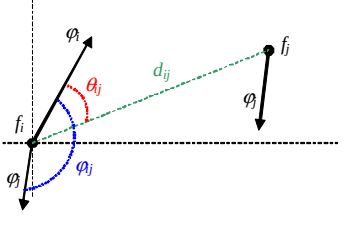


Fig. 1. The local spatial relation between two feature points f_i and f_j

2.2.2 Local Structure Matching

Employing the feature descriptor, for every feature point f_i, a local structure LS_i is formed as the spatial relations between the feature point f_i and its k-nearest neighbors:

$$LS_i=(F_{i1}, F_{i2},\ldots, F_{ik}), \tag{3}$$

where F_{ij} is the feature descriptor consisting of the local positional relations between two minutiae f_i and f_j, as defined in equation (1).

Given two feature sets $\boldsymbol{F}_s=\{f_{s1},\ldots f_{sn}\}$ and $\boldsymbol{F}_t=\{f_{t1},\ldots f_{tm}\}$ respectively, the aim is to find two best-matched local structure pairs $\{f_{sp}\leftrightarrow f_{tq}\}$ and $\{f_{su}\leftrightarrow f_{tv}\}$ to serve as the corresponding reference pair later in the global matching stage.

Now, we start to describe direct local structure matching [4] and complex local structure matching (the proposed improvement).

Direct Local Structure Matching

Suppose LS_i and LS_j are the local structure feature vectors of the feature points i and j from sensed image s and template image t respectively. Their similarity level is:

$$sl(i,j)=\begin{cases} bl-W\left|LS_i-LS_j\right|, & if\ \ W\left|LS_i-LS_j\right|<bl \\ 0, & otherwise \end{cases} \qquad (4)$$

$$W=\{\underbrace{w\ldots w}_{k}\},\ where\ w=(w_d,w_\theta,w_\varphi).$$

where W is a weight vector that specifies the weight associate with each component of the feature vector. The threshold bl can be defined as a function of the number of feature points in a neighborhood. The similarity level $sl(i,j)$, $0\le sl(i,j)\le 1$, describes a matching certain level of a local structure pair. The two best-matched local structure pairs $\{f_{sp}\leftrightarrow f_{tq}\}$ and $\{f_{su}\leftrightarrow f_{tv}\}$ is obtained by maximizing the similarity level [4]. The direct local structure matching method is efficient of $O(k)$, where k is the number of feature points in a neighborhood.

Complex Local Structure Matching

Though the direct local structure matching method is efficient, we found that if there are any dropped or spurious feature points in the neighborhood disturbing the order, the local structure matching will be invalid. We show an example in Fig. 2 to demonstrate this case.

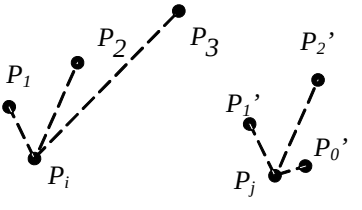


Fig. 2. Illustration of spurious or dropped feature points in the neighborhood

In Fig. 2, p_i in the sensed image s has a neighborhood $\{p_1, p_2, p_3\}$, and p_i's corresponding point p_j in the template image t has a neighborhood $\{p_0', p_2', p_3'\}$, of which $\{p_1\leftrightarrow p_1'\}$ and $\{p_2\leftrightarrow p_2'\}$. Because of the image distortion or scene change, in the neighborhood of p_j, there is no matching feature point for p_1, but a spurious feature point p_0' which does not match to any feature point in the neighborhood of p_j appears. Apply the direct local structure matching, we have $LS_i=\{F_{i1}^T,F_{i2}^T,F_{i3}^T\}$, $LS_j=\{F_{j1}^T,F_{j2}^T,F_{j3}^T\}$. Using equation (4), the similarity level between the local

structures will be very low since their neighbors are mismatched. Thus the similarity level computed by equation (4) is not reliable.

We address the problem by a more complex local structure matching method. First, when we match the two neighbors of two candidate feature points, we consider not only the relative distance but also the radial angle and orientation difference. Second, after we identify those matched neighbors, we will drop the unmatched feature points in the neighborhood in computation of the similarity level of two local structures. In the example shown in Fig. 2, only $\{p_1 \leftrightarrow p_1'\}$ and $\{p_2 \leftrightarrow p_2'\}$ will be considered in the local structure matching.

Suppose we are checking the similarity level between the feature point p and q from the sensed image s and the template image t respectively. Let Knn_p and Knn_q denote the k-nearest neighborhood of the feature point p and q respectively. For every feature point n in Knn_p, we will find its most similar point m in Knn_q. They are qualified as a matching pair if three conditions (equations (5), (6) and (7)) are satisfied:

$$W|F_{pn}-F_{qm}|=min_j W|F_{pn}-F_{qj}| \text{ and } W|F_{pn}-F_{qm}|=min_i W|F_{pi}-F_{qm}|, \quad (5)$$

where $W|F_{pn}-F_{qm}|=w_d|d_{pn}-d_{qm}|+w_\theta|\theta_{pn}-\theta_{qm}|+w_\varphi|\varphi_{pn}-\varphi_{qm}|$. It searches every member in Knn_q and every member in Knn_p.

$$W|F_{pn}-F_{qm}|<T_c, \quad (6)$$

where T_c is threshold value and W is a weight vector same as in equation (4).

$$|\theta_{np}-\theta_{mq}|\leq\pi/4. \quad (7)$$

As we know if both $\{n \leftrightarrow m\}$ and $\{p \leftrightarrow q\}$ are matching pair, the relative orientation difference between θ_{np} and θ_{mq} should be small (equation (7)). Adding this criterion will speed up the search time. If the constraint is not satisfied, it is not necessary to test conditions 1 and 2.

Then the similarity level between the feature points p and q can be computed as

$$sl(p,q)=(bl-nsl(p,q))/bl, \quad (8)$$

where $nsl(p,q)=\Sigma_{n,m} W|F_{pn}-F_{qm}|$, the similarly level only for those matched neighbor pairs from Knn_p and Knn_q according to conditions 1-3. From condition 2, we have $W|F_{pn}-F_{qm}|<T_c$ if point n and point m are matched neighbors. Thus we define threshold bl as T_c times the number of matching neighbor pairs, $bl=T_c|\{n \leftrightarrow m | n \in knn_p, m \in knn_q\}|$, to make sure the similarly level $sl(p,q)$ always greater than zero. The two best-matched local structure pairs $\{f_{sp} \leftrightarrow f_{tq}\}$ and $\{f_{su} \leftrightarrow f_{tv}\}$ are obtained by maximizing the similarity level. Experimental results in Fig. 3 to Fig. 6 confirm the improvement.

2.2.3 Global Structure Matching

There are two limitations in the local structure matching: first, two different feature points from the sensed and template images may have similar local structure. Second, two images from the same scene may have only a small number of well-matched local structures. We need to apply the global structure matching.

Assume that we obtain two best-matched local structure pairs, say (p,q),and (u,v), from the local structuring matching, either one of them can serve as a reliable correspondence of the two feature points' sets. We perform the global structure

matching in two cues for consistence. The best-matched local structure pair (*p, q*) is sent to cue 1 as the corresponding reference to align two feature sets, while another best-matched local structure pair (*u, v*)is sent to cue 2 for the same purpose. In cue1, all feature points will be aligned as follows:

$$GS_i^s = \begin{pmatrix} r_{ip} \\ \theta_{ip} \\ \varphi_{ip} \end{pmatrix} = \begin{pmatrix} \sqrt{(x_i - x_p)^2 + (y_i - y_p)^2} \\ d\phi(\tan^{-1}(\frac{y_i - y_p}{x_i - x_p}), \varphi_p) \\ d\phi(\varphi_i, \varphi_p) \end{pmatrix}, \quad GS_i^t = \begin{pmatrix} r_{iq} \\ \theta_{iq} \\ \varphi_{iq} \end{pmatrix} = \begin{pmatrix} \sqrt{(x_i - x_q)^2 + (y_i - y_q)^2} \\ d\phi(\tan^{-1}(\frac{y_i - y_q}{x_i - x_q}), \varphi_q) \\ d\phi(\varphi_i, \varphi_q) \end{pmatrix}, \tag{9}$$

where GS_i^s and GS_i^t represent the aligned global structure of feature points *i* and *j* in sensed image *s* and template image *t* according to the corresponding reference *p* and *q*, respectively.

Then we define the matching level *ml*(*i,j*) for feature point *i* of the sensed image *s* and feature point *j* from the template image *t* by:

$$ml(i, j) = \begin{cases} 0.5 + 0.5 * w \mid GS_i^s - GS_j^t \mid, & \text{if } \mid GS_i^s - GS_j^t \mid < Bg \\ 0, & \text{otherwise} \end{cases} \tag{10}$$

where *w* is a weight vector and B_g is a 3-D bounding box in the feature space to tolerate the image deformation. We empirically choose $B_g = (10, \pi/4, \pi/4)$.

Thus for an arbitrary feature point *a* in the feature sets $\boldsymbol{F}_s$, we find a feature point *b* in the feature sets $\boldsymbol{F}_t$ such that $ml(a,b)=max_j\ (ml(i,j))$. While for this feature point *b*, we search for a feature point *c* in the feature sets $\boldsymbol{F}_s$ such that $ml(b,c)=max_i\ (ml(i,j))$.

The feature point *a* and the feature point *b* will be recognized as a matching pair if and only if the feature points *c* and *a* are the same point. A matching pair set *MP1* containing all correspondences is generated as the output of cue 1.

In cue 2, we align the two feature sets with respect to another corresponding reference f_u and f_v, and then perform the same global matching as what we did in cue 1 to generate the matching pair set *MP2*. Only those pairs are found in both cues are considered as valid matching pairs. Finally, the matching pair set *MP*, which is the intersection *MP1* and *MP2*, is the result of the global structure matching.

2.2.4 Eliminating the Low-Quality Matching Pairs

We have obtained a number of matching pairs from the global structure matching. Now, we apply the validation step to eliminate those low-quality matching pairs by cross-validation. First of all, we compute the mapping parameters (say, *Map*) from the whole matching pair set. Then in each step, we exclude one pair (say, P_i) from the set of matching pairs and compute the mapping parameters (say, Map_i). If the displacement between Map_i and *Map* is beyond a threshold, the matching pair P_i is identified as a low-quality matching. Eliminating them we get the more reliable matching pair set *MP'*. The experimental results are shown in Fig. 7 and Fig. 8.

2.3 Transformation Model Estimation

Assume that the matching pair set obtained is $\{u_i \leftrightarrow v_i\}_{i=1,2,\ldots,N}$. They should satisfy the relation $v_i = u_i A$, where *A* is the mapping function corresponding to the geometric transformation of the images. We compute it by least-square QR factorization.

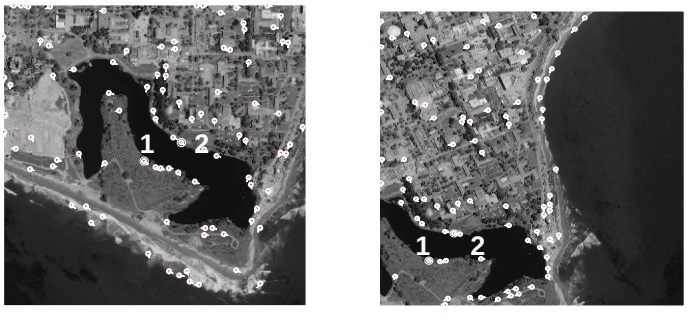
3 Experimental Study

A series of experiments are conducted. The majorities of our testing images are from [2], including optical, radar, multi-sensor, high-resolution and Landsat images. The testing platform is a Pentium 2.20GHz, 512MB RAM PC.

3.1 Results of Local Structure Matching

To demonstrate the improvement of the local structure matching, we ran tests on the following four pairs of images with different types of image variations. The results are shown in Fig. 3 to Fig. 6. For every pair, the dots and arrows indicate the positions and orientations of the feature points, and the two best-matched local structure pairs computed are circled and numbered. The variations between the input and the template image and the number of feature points detected are listed in Table 1. From the results, we see that the best-match local structure pair computed by the improved local structure matching method is more reliable.

To compare the performance of the two local structure matching methods (subsection 2.2.2), we present Table 1 of the experiments results on the four pair of images by both methods. The variation between the input and the template images are shown in the 2nd column. The number of feature points used is listed in the 3rd



(a) template image (b) input image

Fig. 3. An example of local structure matching on images with geometry transformation

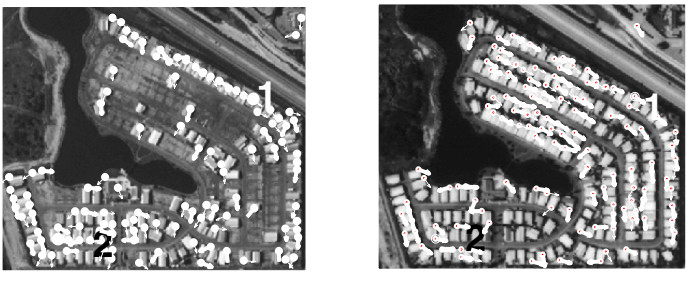


Fig. 4. An example of local structure matching on images with highly temporal changes

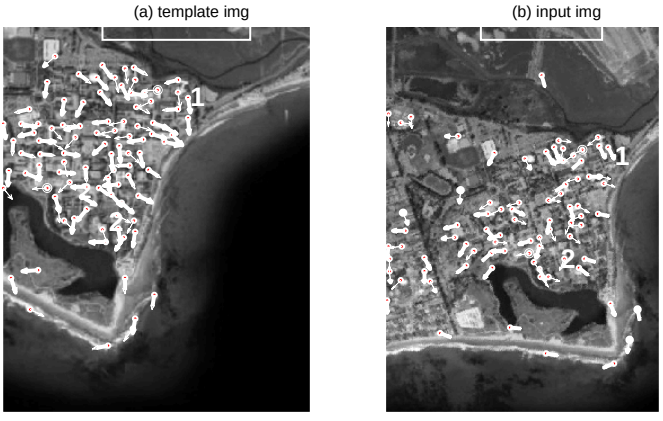


Fig. 5. An example of local structure matching on images with serious deformation

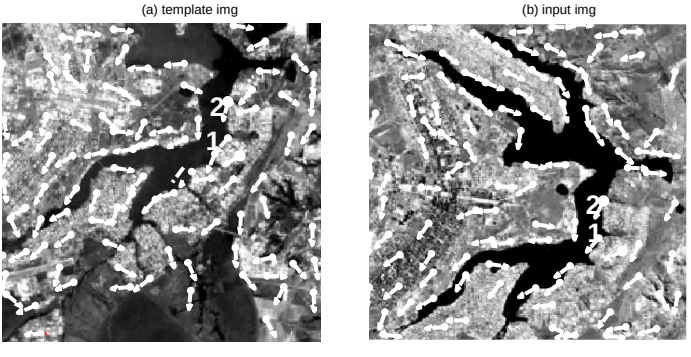


Fig. 6. An example of local structure matching on images from different sensors. In (a) SPOT band 3; (b) TM band 4.

column. In the 4th and 5th column, method 1 is the direct local structure matching and method 2 is the complex local structure matching. For each method the time of computing the best-local structure pair is listed, where × means the corresponding method fails to compute the best-matched local structure pair.

From Table 1, we can see that the direct local structure matching method fails on images with significant scene changes, while the complex local structure matching method is applicable in those cases. However, the direct matching method is more

Table 1. Comparisons of the two local structure matching methods

Testing Images	Image variation type	#Feature points	Method 1	Method 2
Fig. 3	transformation	95 and 86	2.04s	20.34s
Fig. 4	temporal change	114 and139	×	57.80s
Fig. 5	distortion	96 and 81	×	5.25s
Fig. 6	different sensors	97 and 103	×	46.63s

efficient of $O(kmn)$. The computation time of the complex matching method is $O(k^2mn)$, where k is the number of feature points in a neighborhood, m and n are the number of feature points in the input and template images.

3.2 Results of Global Structure Matching

In this subsection, we show how the reliability of the feature points matching is improved by the global structure matching and cross validation. The testing image is pair of urban images from SPOT and TM (Fig. 7), where the two align references pairs are shown in Fig. 6. The final result of global structure matching is shown in Fig. 8.

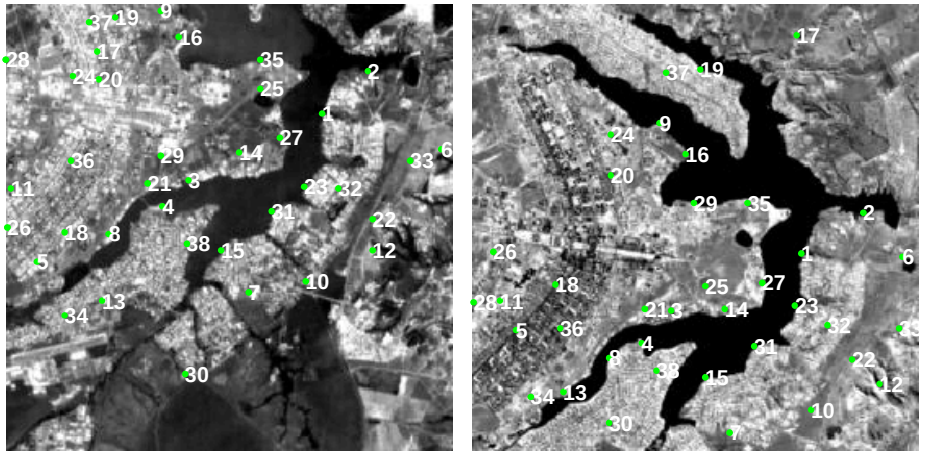


Fig. 7. The matching pairs detected from the global structure matching in cue 1

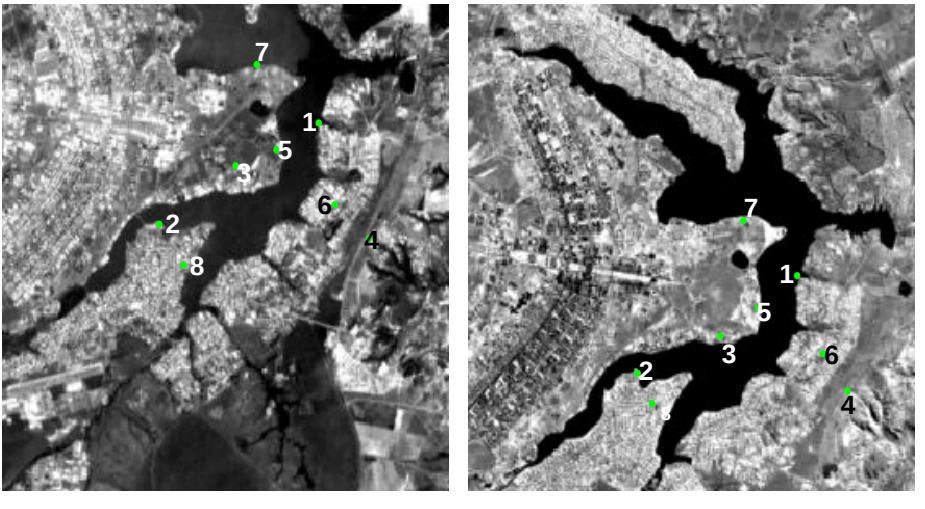


Fig. 8. The final matching pair set after cross-validation

3.3 Using Image Mosaic for the Registration Results

We use image mosaic to show the registration results intuitively. Because of the page limit, only two examples of our test are shown in Fig. 9 and Fig. 10. The correctness of the registration results can be verified visually by checking the continuity of the common edges and regions in the mosaic images.

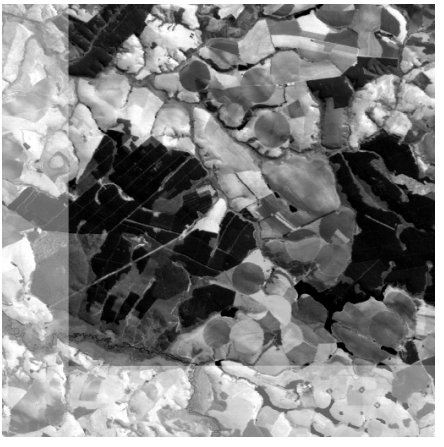


Fig. 9. Registration of Landsat images with four year difference and associated rotation



Fig. 10. Registration of images with serious distortions

We also compared the registration results generated by the UCSB automatic registration system [2]. In Table 2, $[s_1,t_{x1},t_{y1},\theta_1]$ are the registration parameters generated by our method and $[s_2,t_{x2},t_{y2},\theta_2]$ are the results by the UCSB system. In the last two column of Table 2, REMS is the root mean square error at the matching pairs. #MP indicates the number of matching pairs detected for each pair of images. It shows that our method performs better.

Table 2. The registration results on 8 pairs of images

Test cases	Scale: s		Translation t_x		Translation t_y		Rotation: θ		REMS	#MP
	s_1	s_2	t_{x1}	t_{x2}	t_{y1}	t_{y2}	θ_1	θ_2		
1	1.002	1.002	715.1	714.9	-489.66	-490.67	-25.02	-24.98	1.607	211
2	1.012	0.996	87.07	75.06	9.83	9.57	-1.234	-1.098	4.192	8
3	1.042	0.997	21.49	22.35	-8.205	-8.937	-0.668	-0.168	1.498	6
4 (Fig 9)	0.994	0.991	87.88	87.65	-78.98	-79.30	0.125	0.193	1.081	19
5	1.020	0.997	-4.12	0.020	2.064	-0.625	0.562	0.291	11.751	13
6	0.991	0.991	33.57	34.24	-183.43	-186.24	0.984	1.032	9.428	17
7	0.998	0.997	1.44	1.84	-3.17	-0.91	-0.269	-0.047	2.185	8
8 (Fig 10)	0.997	1.004	144.90	144.86	75.33	74.22	-19.90	-20.20	1.611	21

4 Conclusion

Image registration is an important operation in multimedia system. We have presented a feature-based image registration method. Compared to the conventional feature-based image registration methods, our method is robust by guaranteeing the high reliable feature points to be selected and used in the registration process. We have successfully applied our method to images of different conditions.

Acknowledgment

We want to thank Dr. Tong San Koh of NTU for discussions, Dr. Wee Kheng Leow and Dr. Alan Cheng Holun of NUS for comments on Nan Peng's MSc thesis, and comments from MMM 2007 annoymouus reviewers. Nan Peng was supported by NUS scholarship for graduate study.

References

1. L. G. Brown, A survey of Image Registration Techniques, ACM Computing Surveys, vol.24, pp. 326-376, 1992.
2. Dmitry V. Fedorov, Leila M. G. Fonseca, Charles Kenney, and B.S. Manjunath, Automatic image registration and mosaicking system, http://nayana.ece.ucsb.edu/registration/, 2001-2004.
3. Intel Corporation. Open source computer vision library reference manual, December 2000. http://sourceforge.net/projects/opencvlibrary/.
4. X. D. Jiang and W. Y. Yau, Fingerprint Minutiae Matching Based on the Local and Global Structures. 15th International Conference on Pattern Recognition (ICPR'00), vol. 2, pp.1038-1041, 2000.
5. B. Zitova and J. Flusser, Image Registration Methods: a Survey, Image and Vision Computing, vol.21, pp. 977-1000, 2003.

Automated Segmentation of Drosophila RNAi Fluorescence Cellular Images Using Graph Cuts

Cheng Chen[1], Houqiang Li[1], and Xiaobo Zhou[2]

[1] Department of Electronic Engineering and Information Science
University of Science and Technology of China
andychen@mail.ustc.edu.cn, lihq@ustc.edu.cn
[2] Harvard Center for Neurodegeneration and Repair – Center for Bioinformatics
Harvard Medical School
zhou@crystal.harvard.edu

Abstract. Recently, image-based, high throughput genome-wide RNA interference (RNAi) experiments are increasingly carried out to facilitate the understanding of gene functions in intricate biological processes. Effective automated segmentation technique is significant in analysis of RNAi images. Traditional graph cuts based active contours (GCBAC) method is impractical in automated segmentation. Here, we present a modified GCBAC approach to overcome this shortcoming. The whole process is implemented as follows: First, extracted nuclei are used in region-growing algorithm to get the initial contours for segmentation of cytoplasm. Second, constraint factor obtained from rough segmentation is incorporated to improve the performance of segmenting shapes of cytoplasm. Then, control points are searched to correct inaccurate parts of segmentation. Finally, morphological thinning algorithm is implemented to solve the touching problem of clustered cells. Our approach is capable of automatically segmenting clustered cells with low time-consuming. The excellent results verify the effectiveness of the proposed approach.

Keywords: image segmentation, graph cuts based active contours, control points, RNAi, fluorescence microscopy.

1 Introduction

Recently, the understanding of functions of genes in various biological phenomena is becoming more and more important. The functional analysis of genes has been revolutionized by the recent discovery and application of RNAi, which made high-throughput functional genetics a reality. Usually, *Drosophila*, a long-favored model organism for genetic studies, is preferred as a premier cell-based system for such systematic functional genetic analysis.

However, manual analysis of such large-scale datasets produced by RNAi screening is unreasonably time-consuming. Therefore, fully automated techniques are urgently needed in order to analyze RNAi progress in biological research. It should be emphasized that segmentation acts the key role in image analysis process. Various categories of segmentation techniques have been proposed in recent years; however, due to the following three challenging problems: (1) Intensity variations are present

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 116 – 125, 2007.
© Springer-Verlag Berlin Heidelberg 2007

inside cells; (2) Many spiky and ruffly cells are observed; (3) Cells are commonly clustered with weak or even no edges; existing techniques cannot be directly applied to real RNAi datasets to get satisfactory results. Simple thresholding-based method, such as Ostu's method [7], easily causes holes or even division of one cell during segmentation, and they cannot segment clustered cells.

Considering segmentation as an energy minimization problem, active contour model is an effective technique. Active contours can be broadly categorized into two kinds: parametric active contours [2] and geometric active contours [3]. However, both of them are time-consuming and have the shortcoming of local optimization. In contrast, graph cuts method is a global optimization technique for segmentation. Usually, exact solution could be computed in polynomial time. However, the graph cuts method has a bias towards cuts with short boundaries and results in small regions.

Graph cuts based active contours (GCBAC) [4], emerges as an important improvement. With an initial contour, the objective could be achieved by iteratively searching for the closest contour and replacing a contour with a global minimum within the contour neighborhood (CN is defined as a belt-shaped neighborhood region around a contour). This approach overcomes the graph cuts' disadvantage of yielding short boundary, while keeps the advantage of polynomial time-consuming. However, GCBAC could not be applied directly in automated analysis of RNAi images. Manual definition of initial contour hinders the realization of automated segmentation. Furthermore, GCBAC just utilizes gradient information of pixel intensities to set graph's edge weights, thus, complicated variations of intensities inside cells may cause inaccurate results.

In this paper, we present a novel modified GCBAC approach to overcome the aforementioned drawbacks. The whole process is described as follows: First, nuclei are segmented. Region-growing algorithm utilizes each extracted nucleus to get the corresponding initial contour for following segmentation of cytoplasm. Additionally, information of clustered cells' shape obtained from rough thresholding segmentation is considered as a constraint factor and incorporated into GCBAC to update edge weights of the original graph, which largely improves the performance for shapes of cytoplasm. Then, control points on global and local features of the image will be detected by wavelet based salient point detector. By forcing the contour to go through these control points, inaccurate spiky part of the contour could be corrected. Finally, morphological thinning algorithm is implemented to solve the touching problem of clustered cells. Compared with original GCBAC, our approach could get much more accurate results automatically. As a result, the proposed method can potentially serve as the primary tool in automatic biomedical image analysis.

The rest of the paper is organized as follows: Section 2 introduces GCBAC method and wavelet based salient point detector. Section 3 presents our novel modified GCBAC method in details. Section 4 shows experiment results. Finally, Section 5 concludes the paper and discusses the future work.

2 Graph Cuts and Salient Point Detector

2.1 Graph Cuts and GCBAC Method

Graph cuts is an efficient technique with global optimization for image segmentation. The basic theory of GCBAC, $s-t$ min cut, will be introduced first.

Let $G = (V, E)$ (V is the set of vertexes, E is the set of edges between vertexes) be an undirected graph with vertices $v \in V$, and edges of neighboring vertices $(u, v) \in E$. The corresponding edge weight $c(u, v)$ is a non-negative measurement of the similarity between neighboring elements u and v. Also, there are two special nodes called terminals, namely, the source s and the sink t. A cut with source s and sink t is defined as a partition of V into two parts: S and T, $T = V - S$, while the cut's value is the sum of edge weights across the cut.

$$cut(S,T) = \sum_{u \in S, v \in T, (u,v) \in E} c(u,v) \tag{1}$$

Consequently, s-t min cut is to find an optimal cut with the smallest cut value. Also, an important correspondence between flows and cuts in networks is described:

Theorem 1 (Ford-Fulkerson Theorem): *The maximum flow from a vertex s to vertex t, $|f|$, is equal to the value of the capacity $c(s,t)$ of the minimum cut separating s and t.*

According to Ford-Fulkerson Theorem [5], an s-t min cut problem can be converted to an s-t max flow problem, which could be solved by existing algorithms in polynomial time. Graph cuts based active contours model [4], keeps this advantage. In GCBAC, initial contour is manually defined. Contour neighborhood (CN) is obtained by dilating the initial contour with the priori known size and an inner boundary and outer boundary of the CN are extracted. By representing the image within the CN as an adjacency graph, and treating the pixels on the inner boundary and outer boundary as multiple sources and multiple sinks, the problem of finding the global min-cut contour within this CN is formulated as a multi-source, multi-sink s-t min cut problem on this graph. The results on target images in [4] look excellent.

However, GCBAC could not be applied directly in automated RNAi analysis. The manual definition of initial contour hinders the realization of automated segmentation. Furthermore, GCBAC utilizes gradient information of pixel intensities to set edge weights of graph, however, complicated variations of pixel intensities inside cells sometimes cause inaccurate results. Our modified GCBAC method, which overcomes the aforementioned drawbacks, will be presented in Section 3.

2.2 Wavelet Based Salient Point Detector

In GCBAC, inaccurate result is corrected by clicking control points and forcing the contour to pass through these points. In our RNAi dataset, interesting points always stay on where variations occur in the image. Thus, we employ wavelet based salient point detector [6] to implement automatic control point correction. Compared with other feature detectors, this detector aims at getting interesting points related to any visual interesting parts of the image. The algorithm is implemented as follows.

By convoluting the image with the wavelet function dilated at different scales, the wavelet detail image $W_{2^j} f$ is obtained. Further, the wavelet coefficients at the finer scale 2^{j+1} could be learned by computing with the same points as a coefficient $W_{2^j} f(n)$ at the scale 2^j. This set of coefficients is called the children $C(W_{2^j} f(n))$ of the coefficient $W_{2^j} f(n)$, which is defined as follows:

$$C(W_{2^j} f(n)) = \{W_{2^{j+1}} f(k), 2n \le k \le 2n + 2p - 1\} \quad (2)$$

p is the wavelet regularity and $0 \le n < 2^j N$ with N as the length of the signal.

Each wavelet coefficient $W_{2^j} f(n)$ is computed with signal points to represent the variations at the scale 2^j. Its children coefficients give the variations of some particular subsets of these points (with the number of subsets depending on the wavelet). The most salient subset is the one with the max absolute value of wavelet coefficient at the scale 2^{j+1}. In [6], this maximum is considered, and its highest child is looked for. This process is applied recursively to select a coefficient $W_{2^{-1}} f(n)$ at the finer resolution 1/2. Hence, this coefficient represents $2p$ signal points. To select a salient point from this tracking, the one with the highest gradient is selected among these $2p$ points. The saliency value is defined as the sum of the absolute value of the wavelet coefficients in the track:

$$saliency = \sum_{k=1}^{-j} | C^{(k)}(W_{2^j} f(n)) |, -\log_2 N \le j \le -1 \quad (3)$$

The tracked point and its saliency value are computed for every wavelet coefficient, and the point with a high saliency value usually corresponds to a global variation. By thresholding the saliency value, the desired salient points will be selected.

3 Our Modified GCBAC Approach

We develop a modified GCBAC method to segment clustered cells automatically. Our proposed approach consists of following steps: First, region-growing algorithm utilizes extracted nuclei to get the initial contours for the segmentation of cytoplasm. Second, rough segmentation of cytoplasm's shape, considered as a new constraint factor, is incorporated into GCBAC to improve the performance of segmentation. Then, control points are automatically searched to correct inaccurate segmentation result. Finally, morphological thinning algorithm is implemented to solve the touching problem of clustered cells. The details will be described in the following.

3.1 Initial Contour Using Region Growing

Manual definition of initial contour hinders automation of GCBAC. Luckily, nuclei and cytoplasm could be screened by fluorescence microscopy, and only one nucleus exists near the center of corresponding cell generally, therefore, region-growing algorithm could utilize extracted nuclei as seed regions to find initial contours for following segmentation. Thresholding method [7] could easily extract nuclei from background. Here, a modified Ostu's method is implemented, which includes taking into account the max and min values in the image and log-transforming the image prior to calculating the threshold. Sometimes, the threshold may consistently be too stringent or too lenient. Thus, an adjustment factor could be multiplied with the threshold to get a more accurate one. The number 1 means no adjustment, 0 to 1 makes the threshold more lenient and greater than 1 makes the threshold more stringent. Since automated segmentation is required, the fixed universal parameters

are needed. We tried out the parameter adjustment factor within a specified range on an image and the one with the best performance for segmenting nuclei is 1.3.

By observing the feature that pixel intensities inside cells are approximate, region-growing algorithm [1] is considered to find initial contours for cell segmentation. The basic approach of region growing is to start with a set of points as seed region and group their neighbouring pixels into seed region according to predefined criteria. In our experiment, we just define two simple criteria: (1) the grey-level of any pixel should be in a specified range around the average grey-level of pixels in the seed region. If the average grey-level is α, the grey-level of grouping allowed pixel β should satisfy: $\beta \in [\alpha - L, \alpha + L]$, L is the specified range size. Note that too large or small range size setting will cause unsuitable initial contour. Here, the parameter setting L with the best performance is 40, with grey levels of pixels in the image normalized to [0,255]. (2) The pixels must be 8-connected to any pixel in seed region.

With an extracted nucleus as a seed region, we iteratively implement the following operation: Each time, 8-connected neighboring points of seed points on the boundary of seed region, are judged whether could be grouped into seed region; then, morphological operations are implemented to fill holes, break narrow isthmuses and eliminate thin protrusions of new seed region. When no more pixels satisfy the criteria, region growing algorithm stops. Usually, the number of iteration is no more than 10 in our experiment. Note that incomplete nuclei touching the border of image, or object smaller than a specified range, which is likely to be fragments of real nuclei, will be discarded. Results are illustrated in Section 4.

3.2 New Constraint Factor Changing Edge Weights

Here, GCBAC method [4] is implemented after the previous step to get a rough result of segmentation. GCBAC method consists of the following steps:

(1) Represent the image as an adjacency graph G.
(2) Dilate current boundary into its contour's neighborhood with an inner boundary and an outer boundary.
(3) Identify all the vertices corresponding to the inner boundary and outer boundary as a single source s and a single sink t.
(4) Compute the $s-t$ minimum cut to obtain a new boundary that better separates the inner boundary form the outer boundary.
(5) Return to step 2 until the algorithm converges.

In [4], the image is represented as an 8-connectivity graph, which means each vertex, corresponding to a pixel, has edges connecting to its 8 neighboring pixels. Also, the edge weight between vertex i and vertex j is defined as follows.

$$c(i, j) = (g(i, j) + g(j, i))^6 \tag{4}$$

$$g(i, j) = \exp(-grad_{ij}(i) / \max_k(grad_{ij}(k))) \tag{5}$$

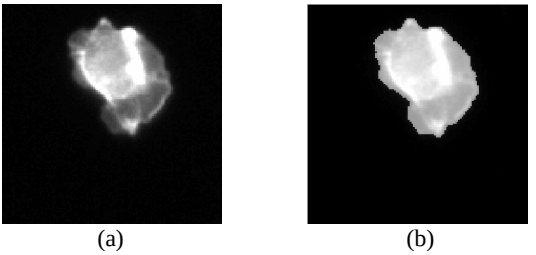
Where $grad_{ij}(k)$ is the image pixel intensity gradient at location k in the direction of $i \to j$. This weight assignment method leads the active contours to high gradient

edges and considers direction of the gradients. GCBAC just utilizes gradient information of pixel intensities in the image, thus, it is good at segmenting objects with high contrast between boundary and background. However, it fails in segmenting real RNAi dataset, because of pixels' intensity variations inside cells. For example, sometimes, a cell's inner part is much brighter than boundary part, causing GCBAC to segment the inner part as the final result.

Idea of constraint factor in [8] could be adopted to solve this problem. Assumed I as the original image, the constraint factor $H(I)$ is got by the modified Ostu's method introduced in Section 3.1. Here, the adjustment factor is 1.0. Ranges of pixels' grey level in both I and $H(I)$ are normalized to [0,255], then a new constraint factor incorporated image $N(I)$ is computed.

$$N(I) = \lambda I + (1-\lambda)H(I) \tag{6}$$

Constant $\lambda(0 < \lambda < 1)$ is a scalar parameter weighting the importance of these two sources of information. If λ is large, we trust more regional information; otherwise, we trust more edge information. In our experiment, λ is selected as 0.5, meaning that the importances of both sources of information are the same. Thus, graph and its edge weights will be built using information of $N(I)$ instead of I .



(a) (b)

Fig. 1. (a) original image I of a RNAi cell. (b) new image N(I) of the same RNAi cell.

3.3 Inaccurate Correction Using Salient Points

Segmentation results obtained by previous two steps are not always satisfactory, due to that the shape of cell is often non-convex. Especially, it is difficult for GCBAC to segment narrow spiky shape successfully. Luckily, control points in [4] could be employed to correct inaccurate segmentation result. In our approach, we choose wavelet based salient point detector to search control points automatically. Wavelet based salient point detector [6] has been introduced in Section 2.2.

All the interest points where variation occurs in the whole image could be detected by wavelet based salient point detector. However, only salient points for the target cell are needed as control points. Therefore, only salient points located in the neighboring area of boundary of rough segmentation obtained in section 3.1 will be selected and other points will be removed. These selected points are used as control points in GCBAC. In our experiment, we find this method very effective in correcting inaccurate spiky shape. Corresponding results are shown in Section 4.

3.4 Clustered Cells Segmentation Using Morphological Thinning Algorithm

Sections 3.1, 3.2, 3.3, mainly discuss the procedure of segmenting single RNAi cell. However, clustered cells are commonly observed in real RNAi images. As a result, only techniques with ability of segmenting clustered cells could be applied in analysis of real RNAi images. Based on the techniques introduced above, we develop a novel algorithm to segment clustered cells. The scheme will be summarized as follows.

First, the modified Ostu's method mentioned above is implemented to segment the nuclei and corresponding cytoplasm. Note that nuclei without corresponding cytoplasm, or cytoplasm without corresponding nuclei, will be regarded as noise and discarded. Additionally, original image will be enhanced before implementation of modified Ostu' method to better segment cytoplasm of spiky shape. Generally, cytoplasm of touching cells is segmented as a whole.

Second, region-growing algorithm utilizes each extracted nuclei as seed region to find corresponding initial contour automatically. Then, our modified GCBAC uses initial contours to segment cytoplasm. Due to the complex intensity variations between touching cells, segmented cell cannot touch with each other ideally. Blank regions are left between segmented objects, as illustrated in Fig. 6 (a).

Third, the blank regions left in the second steps are found to be touching regions of cells, in which weak or even no edges exist. Traditional edge-based techniques can hardly get accurate boundary between cells. To address the problem, these blank regions between segmented objects are extracted. Then, the boundary of segmented cytoplasm obtained in the first step is extracted and incorporated into the extracted blank regions as a whole. Corresponding result is shown in Fig. 6 (c).

Finally, mathematical morphological thinning algorithm [1] has the ability of extracting central line of regions. Thus, it is iteratively implemented on the boundary-incorporated areas to get 'skeleton', which could be regarded as the touching boundary of clustered cells. In this way, touching problem could be solved. In our experiment, the number of iteration is no more than 5. The segmentation result in Fig. 6 (d) seems perfect. Details are described in section 4.

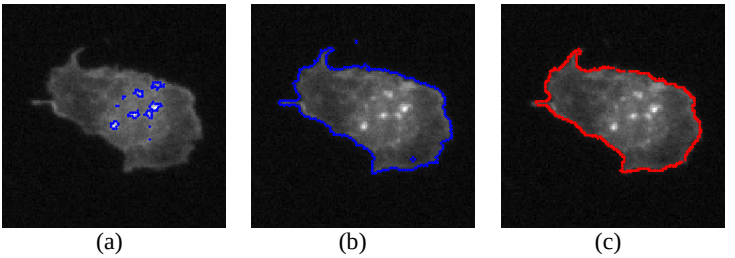
4 Experiment Results

In this section, our approach is applied to real RNAi cells to verify its advantages to handle complicated shapes, interior intensity variation case. Experiment results and corresponding descriptions are shown in the following.

A. *Advantages of our method over thresholding method*

Some may argue that thresholding method, such as Ostu's method, will be good enough for segmenting cells. However, thresholding method is sensitive to noise, causing small stains inside and outside the cytoplasm, or narrow isthmuses and thin protrusions. Additionally, thresholding method cannot segment clustered cells in real RNAi images. Comparatively, our method overcomes this shortcoming. In Fig.2, we illustrate three images of results. Fig.2 (a) is the result obtained by Ostu's method. The threshold value obtained by Ostu's method is quite high, causing inaccurate results. The blue contours in Fig.6 (a) show that only light stains inside the cell are segmented. Fig.6 (b) is the modified Ostu's method's result. From the image, it is

obvious to see noises segmented as fragments, and the boundary has many thin protrusions. Fig.6 (c) is our method's result. An accurate result without noises is segmented, and its smooth boundary is shown in red in the image.

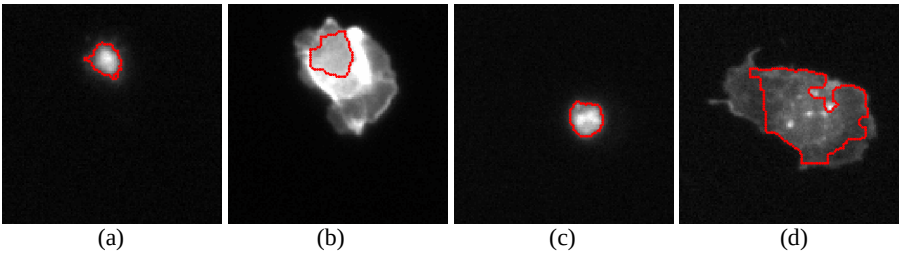


(a) (b) (c)

Fig. 2. Comparison of results using thresholding method and our modified GCBAC (a) Segmentation result using traditional Ostu's method (b) Segmentation result using modified Ostu's method (c) Segmentation result using our approach

B. *Initial contour using region-growing algorithm*

The first step of our method is to find an initial contour. In Fig.3, we illustrate ability of this algorithm in finding initial contours. Two types of target objects are shown, including cells with ruffly and spiky shapes. In Fig.3 (a) and (c), two nuclei are separately segmented using simple thresholding method. Information of nucleus is utilized in region-growing algorithm as seed region to find initial contour for segmentation of corresponding cytoplasm. The results are shown in Fig.3 (b) and (d), respectively. Note that initial contours could depict rough structure inside the cells.

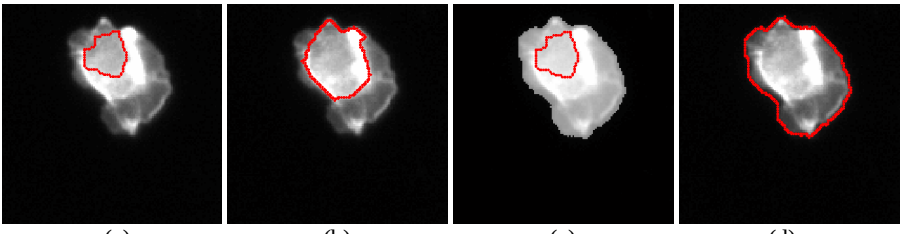


(a) (b) (c) (d)

Fig. 3. Region-growing algorithm is used to find initial contour. (a) Nucleus of a ruffly cell is segmented with boundary marked in red line. (b) Initial contour of a ruffly cell is found and marked in red line. (c) Nucleus of a spiky cell is segmented with boundary marked in red line. (d) Initial contour of a spiky cell is marked in red line.

C. *Constraint factor added GCBAC to handling interior intensity variation*

Our modified GCBAC is able to handle RNAi cells with weak boundary. Fig.4 shows a group of results, including traditional GCBAC's results and modified GCBAC 's results. Fig.4 (a) shows the image I used for segmentation. Fig.4 (b) shows the corrected images $N(I)$ used for segmentation. In Fig.4 (c), traditional GCBAC just segments the bright parts inside the cell. In Fig.4 (d), our approach segments the real cell from the background. The results shown are excellent.

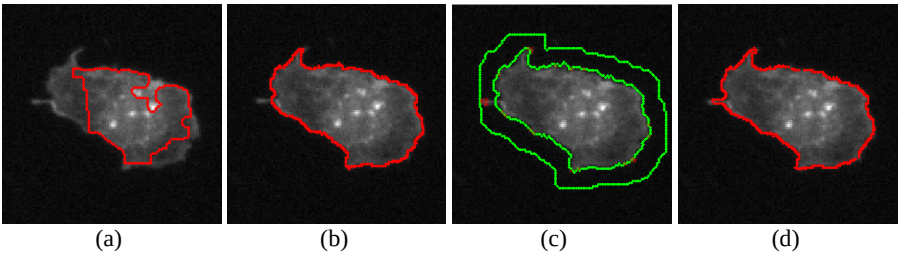


(a) (b) (c) (d)

Fig. 4. Segmentation results of both traditional GCBAC and modified GCBAC. (a) is original image I with initial contour shown in red. (b) is the constraint factor incorporated image N(I) with initial contour shown in red. (c) is the segmentation result of traditional GCBAC. (d) is the segmentation result of our modified GCBAC.

D. *Inaccuracy correction using salient point detector*

GCBAC cannot always get satisfactory result. Control points are used to correct inaccurate part. Fig.5 shows the whole process of segmenting a spiky cell. Fig.5 (a) shows the red initial contour. Then, the previous two steps are implemented to get the result in Fig. 5(b). In the image, one narrow protrusion has not been segmented. Fig.5 (c) shows the control points found in the neighbourhood area of the rough segmented boundary. These points are marked by red cross points. Fig.5 (d) shows the corrected result using these control points. The final result is perfect.



(a) (b) (c) (d)

Fig. 5. Use salient point to correct inaccurate part. (a) Initial contour. (b) Rough segmentation result obtained. (c) Salient points found. (d) Final result with inaccuracy part correction.

E. *Clustered cells segmentation using morphological thinning algorithm*

Morphological thinning algorithm could be employed to segment clustered cells. Fig. 6 demonstrates the whole process. Fig. 6 (a) shows unsatisfactory result of GCBAC. The red boundaries show segmented objects. Due to complex intensity variations between cells, segmented object can hardly touch with each other and their touching regions are left. Fig. 6 (b) shows the touching regions left in a binary image. Our novel algorithm solves this problem by incorporating boundary into touching regions. In Fig. 6 (c), white contour is the boundary of segmented cytoplasm, which the gray areas are touching regions left in GCBAC. Finally, morphological thinning algorithm is iteratively implemented to get the final result. Fig. 6 (d) shows the segmented objects in red. The result is quite satisfactory for biomedical image analysis.

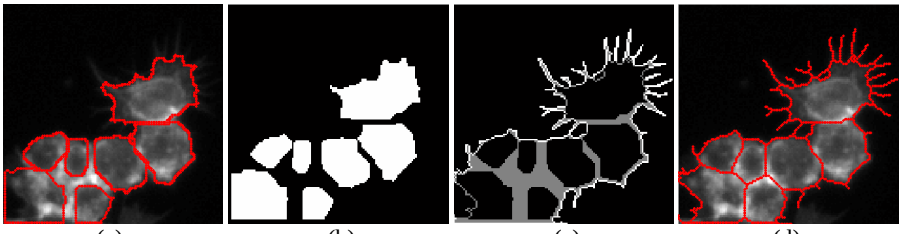


Fig. 6. Clustered cells segmentation. (a) rough segmentation by GCBAC. (b) segmented objects with unsolved touching regions. (c) boundary incorporated into touching areas. (d) final segmentation result of clustered RNAi cells.

5 Conclusion and Discussion

In this paper, a modified GCBAC is presented as a fully automatic segmentation method for RNAi fluorescence images. First, extracted nuclei are used in region-growing algorithm to get the initial contours for the segmentation of cytoplasm; In addition, constraint factor obtained from thresholding method is incorporated into GCBAC to improve the segmentation performance for faint shapes of cytoplasm; then, control points are automatically searched to correct inaccurate part of results caused by spiky shape; finally, morphological thinning algorithm is implemented to solve the touching problem of clustered cells. Compared with traditional GCBAC or other active contour methods, our novel approach has advantages of automatically segmenting RNAi cells in polynomial time. Experiment results verify the effectiveness of our proposed approach. In addition, the approach described in this paper is generic in its nature. And our future work is to extend the segmentation to high throughput imaging RNAi experiments of other cell types.

References

1. Rafael, C. Gonzalez. and Richard, E. Woods.: Digital Image Processing. *Prentice Hall, second edition.* (2002).
2. M, Kass., A, Witkin. and D, Terzopoulos.: Snakes: Active contour models. *Int. J. Comput.* Vis., (1987) vol. 1, pp. 321–331.
3. V, Caselles., R, Kimmel. and G, Sapiro.: Geodesic active contours. *Proc. 5th International conf. on Computer Vision, Boston*: (1995) pp. 694-699.
4. Ning Xu, Ravi Bansal, and Narendra Ahuja: Object segmentation using graph cuts based active contours. *CVPR2001 Technical Sketches*, (2001) pp. 87-90.
5. L, Ford. and D, Fulkerson.: Flows in Networks. *Princeton University Press*. (1962)
6. Loupias, E., Sebe, N., Bres, S., Jolion.: Wavelet-based salient points for image retrieval. *In: Internat. Conf. on Image Processing, J.-M.*: Vol.2, (2000) pp. 518-521
7. Otsu, N.: A threshold selection method from gray level histogram.ïIEEE *Transactions on System man Cybernetics 8*:62-66. (1978)
8. Xiong GL, Zhou XB, Ji L, Bradley, P., Perrimon, N., and Wong STC.: Segmentation of drosophila RNAi fluorescence images using level sets. *International Conference on Image Processing, Altanta, GA, USA.* (2006)

Recommendation of Visual Information by Gaze-Based Implicit Preference Acquisition

Atsuo Yoshitaka, Kouki Wakiyama, and Tsukasa Hirashima

Graduate School of Engineering, Hiroshima University
1-4-1 Kagamiyama, Higashi-Hiroshima, Hiroshima, 739-8527 Japan
{yoshi, wakiyama, tsukasa}@isl.hiroshima-u.ac.jp

Abstract. Collaborative filtering is one of the information filtering techniques, that recommends information referring to the evaluation of others' feedback, where their preferences are similar to the target user to assist. The user's preference is often specified explicitly, and it is often a burden or disturbance in concentrating on his/her primary activity. This issue affects the granularity, i.e., degree of detail of user's feedback, since it largely depends on the easiness of acquisition for the user's preference. In this paper we describe a method of information recommendation based on social filtering, where the preference of user is implicitly acquired by gaze detection. As an example of the application we implemented a system that recommends paintings to a person based on others' attention in appreciating paintings. The evaluation of preference is based not on individual objects (i.e., paintings) but on sub-regions in an object (e.g., a person, a building, an animal, and so on), that is detected by gaze point tracking. The strength of interest is measured as the duration of watching a sub-region in the painting, and each user's interest model is organized based on it. Experimental result showed the sub-region based information recommendation provides us better recommendation compared with object-based recommendation, and the proposed implicit preference acquisition method is comparable to explicit preference specification method.

Keywords: gaze detection, multimedia information recommendation, social filtering.

1 Introduction

Mobile computers and small eyeglass shaped displays are becoming popular and have opened to a variety of applications. Augmented reality systems with mobile computers are studied under an environment in the real world. One of the directions of augmented reality systems is to assist activities of a user in the real world by providing him/her information related to an object that he/she is facing.

As a method of assisting user's activity, information recommendation is one of the effective approaches. Recommendation data is retrieved based on a user's preference or history of his/her activity. In some cases, such as appreciating paintings, photos, or video, information recommendation based on the user's preference is considered to be

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 126 – 137, 2007.
© Springer-Verlag Berlin Heidelberg 2007

effective since evaluating visual information is not an easy task. In recommending information to be provided, it is mandatory to evaluate which information is more worth to access for the user; i.e., how much the recommendation fits to his/her preference in the context of his/her activity. As a solution for this issue, social filtering is one of the promising methods.

Social filtering retrieves data from a database, whose attributes fit to one's preference. Generally, we need to collect other users' preferences for each of the objects to be recommended prior to filtering, since social filtering is performed by evaluating similarity between the preferences of a user for the objects and those of others. In performing social filtering, one of the issues is how we alleviate the burden in collecting users' preferences for target objects. Users' preferences or evaluation for target objects are generally collected by the users' explicit feedback to a system. As far as feedback of preference depends on explicit specification by users, their burden is indispensable and it should be alleviated since it is not their primary task. One of the solutions is to make the feedback into the form of specifying numerical values [1-4]. In social filtering, creating profiles based on unconstrained description in natural language is also adopted [5]. However, explicit feedback, that is often entered manually, may deteriorate the quality of feedback if the number of items for feedback becomes larger. In that sense, it is desirable to collect users' feedback from implicit behavior of the users.

Implicit acquisition of evaluation for information is desirable because it enables to organize user's interest model without disturbing his/her primary task. An example of implicit acquisition of evaluation is studied for function recommendation in software [6]. In recommending functions in the software, frequently applied functions are ranked higher in evaluation, and therefore, such functions are recommended more than less-applied functions. Since the distribution of the frequency of application of functions lies in potential bias, there is an issue that a recommended function does not always fit to the user's purpose of using the software. Other examples of implicit acquisition of user's preference are applications for recommending Web pages [7-8]. A user interest model is organized based on the access history on Web pages. In these studies, since the unit of measuring one's interest is one Web page, it is not possible to detect which portion of the Web pages he/she is interested. This problem may deteriorate the quality of recommendation. In this sense, the granularity of preference acquisition, in other words, interest modeling should be considered. Detecting and identifying object in attention and measuring preference for it is comparatively easy in case where it is displayed on the computer screen. However, it is more difficult and, as far as we know, no study is reported that aims at detecting the visual objects in attention in the real world and measuring preference for them implicitly.

In case where a user's task is to appreciate visual information such as photos, paintings, or scenery, the degree of the user's interest, i.e., preference for the objects, may be revealed in his/her behavior of eye movement; what he/she watches with more time may correspond to the visual information that he/she is more interested. If it is possible to extract a user's preference from his/her eye movement, we can detect his/her preference without his/her explicit feedback of specifying preference for social filtering. Based on this idea, we proposed a framework of social filtering method with implicit acquisition of users' preference based on eye movement.

In this paper we describe a framework of social filtering method that collects users' preference based on gaze detection, assuming an activity of appreciating visual information such as paintings. While users appreciate paintings with carrying the system, each user's target of visual object is detected from gazing point and the gaze duration for the object. Gaze duration for each of the predefined regions in a painting is measured, and the length of duration is regarded as the strength of interest for the region in the painting. Social filtering is performed for information recommendation by referring to other users' preferences, i.e., the pattern of interest, which are similar to the user to assist. Finally, paintings that other users, whose pattern of preference is similar to the user, appreciated with more interest are retrieved and presented to the user as recommendation.

The organization of the paper is as follows: Section 2 describes the method of extracting eye movement and the criterion for detecting user's state of gazing. Section 3 discusses user's model of interest and information recommendation based on social filtering, and the system organization and user interface is described in Section 4. Experimental result of the proposed framework is shown in Section 5. Lastly, concluding remarks are given in Section 6.

2 Measuring Interest for an Object

In this section, we describe the method of extracting the degree of interest for a visual object such as painting. We assume the degree of interest corresponds to the gaze duration for the visual object in case of appreciating visual objects, e.g. appreciating paintings, photos, or sculptures. The validity of this assumption is discussed with experimental result later in this paper. We first describe the method of extracting eye movement and measuring the degree of interest for an object.

2.1 Gaze Detection

Figure 1 shows headset of the prototype system. The headset equips two CCD cameras; the eye camera is a monochrome CCD camera with infrared LEDs for shooting an eye, and the view camera is a color CCD camera for shooting user's view to detect visual axis pointing to a visual object. Since contrast between iris and pupil is low in luminance under visible ray but it becomes higher under infrared, infrared is illuminated to augen to extract the region of pupil.

A viewpoint is calculated based on pre-calculated correspondence of coordinates between the location of the user's pupil and those in a user's view after the extraction of the region of the pupil. The eye camera is placed so that the center of pupil locates at the center of the video frame of the eye camera when one keeps his/her eyes front. In the same way, the view camera is located so that the center of user's view corresponds to that of the video frame of the view camera. View point in the video frame of the view camera is nonlinearly mapped with reference to the position of the centroid of the region of pupil for compensating the lens aberration.

The state of 'gazing visual object' is detected based on the frequency of saccade in an interval and the duration of fixation. According to the result of exploratory

experiment, the duration of one fixation tends to be less than approximately 3sec., and three or more times of fixations, each of which ranges 0.3sec. to 3sec., repeat in case where he/she gazes visual object with interest. Therefore, when fixations whose interval range from 0.3sec. to 3sec. repeat more than two times, we regard the beginning of the first fixation as the beginning of the state of gazing. If there appears a fixation whose duration is more than 3sec., the end of the previous fixation, i.e., the last fixation in a state of gazing visual object, is regarded as the end of the state of gazing visual object. We denote a gazing section as the section between the beginning and the end of gaze, which is determined by following the above-mentioned criterion. Because of the errors in extracting the centroid of the extracted pupil and involuntary eye movement, we regard the movement of centroid that is less than the angle of 2.1 degrees in consecutive 3 frames (i.e., 0.3 sec.) as a fixation, and that more than the angle of 2.1 degrees as a saccade. Note that the video frame rate is 10fps.

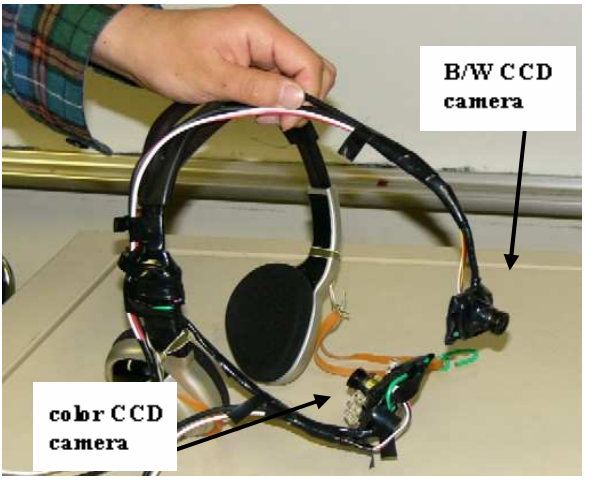
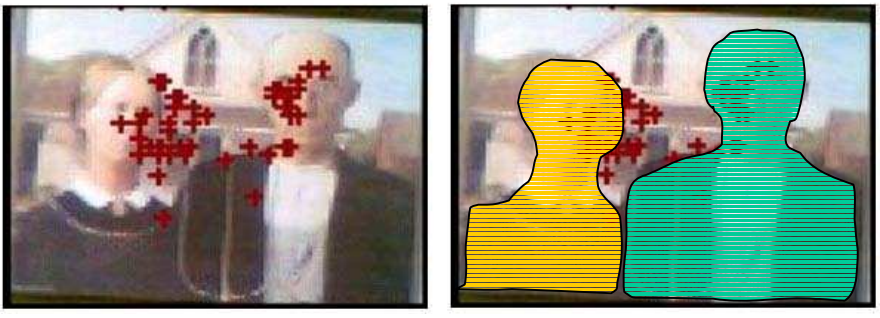


Fig. 1. Head Mounted Cameras

2.2 Gaze Point Distribution and Gaze Duration

We denote gaze point distribution consists of gaze points in one gazing section. Figure 2(a) shows an example of gaze point distribution in one gazing section. In the Figure, the symbol of '+' denotes a view point. A minimum convex polygon whose area is minimum but contains all gaze points in one gazing section corresponds to the region of interest in watching visual object. In order to detect the region of interest, the sub-regions, i.e. component objects, of an object, such as a man, a woman, a house, an animal and so on in a painting, are carved out beforehand. Figure 2(b) shows an example of predefined sub-regions in the painting. In this example, two sub-regions that correspond to a man and a woman are defined as individual sub-regions. The object of interest is extracted by detecting the overlap between the region of interest and the predefined sub-region of the object. The degree of interest for a component object in the painting is obtained as the sum of fixation duration, where

the gazing points in fixations are located in the same sub-region. If fixation points exist over two or more sub-regions in the painting in a single gazing section, the sum of fixation duration is calculated for each of the sub-regions in the painting. As described above, the degree of interest is measured not for the painting as a whole, but for each of the component objects drawn in the painting.



(a) Gaze Point Distribution (b) Predefined Regions of Objects

Fig. 2. An Example of Detecting Region of Interest (‘American Gothic’ by Grant Wood)

3 Interest Model and Information Recommendation by Social Filtering

In this section, we discuss the method of information recommendation based on user interest model. The user interest model is constructed from implicitly detected preference for component objects. The preference is extracted from the behavior of gazing for component objects in a painting. We suppose component object based preference measurement is superior to object, i.e., painting, based preference measurement, since the user's preference based on component object reflects preference for each of the components, but object based preference cannot extract preference for each of the component objects depicted in a painting. We regard the strength of interest corresponds to the total amount of fixation duration for a component object. We evaluated the adequacy of above-mentioned assumption by experiments, which is discussed later in this paper.

We define the interest model of a user as the accumulation of the preferences of component objects for ever-appreciated paintings. The interest models of other users', that are similar to the target user's interest model, are first extracted in order to perform information recommendation. After that, paintings are retrieved, which contain component objects of higher preference in the interest models of other users correlating with that of the target user. Those paintings are regarded as the paintings that fit to his/her preference since they contain component objects of their preference, i.e., they are also regarded as the target user's preference.

In the following subsections, we first define the user interest model, and after that we discuss the information recommendation based on the correlation between user interest models.

3.1 User Interest Model

We assume the strength of interest in a component object depicted in a painting corresponds to the sum of the gaze duration for the component object. Let i_j be the sub-region, i.e., a component object, in the painting i, and $\mathrm{int}_a(i_j)$ be the degree of interest for the sub-region i_j observed for the user a. We define $\mathrm{int}_a(i_j)$ as follows:

$$\mathrm{int}_a(i_j) = \frac{t_a(i_j)}{ave(t_a)} \tag{1}$$

In the above definition, $ave(t_a)$ denotes the average of the sum of fixation duration for a sub-region in paintings which he/she ever appreciated. The reason why $t_a(i_j)$ is normalized by dividing $ave(t_a)$ is to compensate the difference among individuals. For example, when a person watches a certain component object for 10 sec. under the condition where his/her average duration for a sub-region is 20 sec., we should evaluate he/she paid less attention than another person who watched the same component object for 10sec., under his/her average gaze duration is 8 sec. Since the user's gaze point and the duration of fixation is obtained by video processing, it is possible to construct user interest model without user's explicit feedback to the system.

After evaluating the degree of interest for each of component objects in a painting, user interest model is constructed. The user interest model for the user a consists of the values of degree of interest that correspond to ever-appreciated paintings as shown in (2).

$$Ma=(int_a(1_1), .., int_a(1_m),.., int_a(i_1),.., int_a(i_m)) \tag{2}$$

3.2 Evaluating Correlation of the Interest Models

We assume that interest models of other users, who have already appreciated all the visual objects, i.e., paintings, are available. Objects to be recommended are retrieved by referring to other users' interest models, which correlate to the target user to assist. The idea is that if it is possible to find other users whose user models correlated to the target user, we can regard objects of their higher preference are worth recommending. The correlation between the user a who requests information recommendation and one of the other users, the user b, who already appreciated all the visual objects, is calculated as follows. Let *ave(int a)* denote the average strength of interest of user a for component objects in paintings and let *ave(int b)* denote that of user b. The strength of correlation of preference between user a and b is defined as:

$$r_{ab} = \frac{\sum_{i,j} \{\mathrm{int}\, a(i_j) - ave(\mathrm{int}\, a)\}\{\mathrm{int}\, b(i_j) - ave(\mathrm{int}\, b)\}}{\sqrt{\sum_{i,j} (\mathrm{int}\, a(i_j) - ave(\mathrm{int}\, a))^2 \sum_{i,j} (\mathrm{int}\, b(i_j) - ave(\mathrm{int}\, b))^2}} \tag{3}$$

In the above definition, r_{ab} takes [-1,1]. More of its absolute value signifies more strength of positive or negative correlation. If it takes positive value, the preference of user a is similar to that of user b, and therefore, the objects yet to be appreciated by user a, which are in the positive preference of user b, correspond to the objects worth

recommending. In case of visual information, negative preference of a person in negative correlation does not always corresponds to positive correlation. Therefore, we do not refer to the negative preferences of the users in negative correlation for information recommendation, as well as negative preferences of the users in positive correlation.

3.3 Information Recommendation

In the process of extracting objects, i.e., paintings, users whose interest model correlate to that of the target user's is first picked up. After that, the sum of the strength of interest is calculated for each of the component objects in the paintings. That is, assume that there are users $u_k(k=1,2,..,n_u)$ in the set of users O, who have positive high correlation to the target user u_t to assist. The strength of interest $P(i_j)$ for component object i_j in the object i is calculated as follows:

$$P(i_j) = \sum_{u_k \in o} \mathrm{int}_{u_k}(i_j), where \quad r_{u_t u_k} \geq t_c \tag{4}$$

In the above expression, t_c denotes the threshold to extract users in positive high correlation with the target user u_t. Since objects where $P(i_j)$ is high contains component objects of the target user's preference, they are presented to the user as the recommendation worth watching. They are presented in the descendant order of $P(i_j)$.

4 AttentionShare: A Prototype System

4.1 System Organization

The organization of the prototype system is illustrated in Fig. 3. A monochrome CCD camera is located below an eye for observing eye movement, which we call an *eye camera*. It is placed so that the centroid of the pupil is located at the center of the video frame. A color CCD camera, which we call a *view camera*, is placed between the eyebrows, which is aimed at shooting the user's view.

Eye movement is detected by extracting the centroid of a pupil, and the state of gazing is identified by the criterion which we described in section 2.1. The region of convex polygon formed by the coordinates of gazeing points is held for identifying a sub-region, i.e., a component object, in the painting. Coordinates of predefined sub-regions as well as the title, the name of painter, category, period of a painting are stored in the Painting Database. Identification of the painting he/she is currently watching can be performed by image similarity matching based on 2D color space, where the region of painting is segmented by detecting horizontal and vertical lines that correspond to the inner frame of a picture. However, identifying a painting by image similarity measure may imply some errors and they interfere the evaluation of the proposed method. Therefore, we designed the painting in appreciation is explicitly specified by user. Note that image matching for painting identification can be adopted in actual operation of the system.

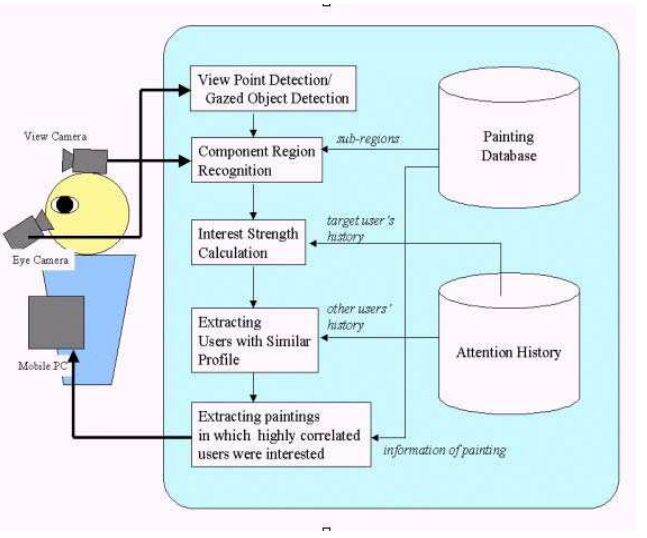


Fig. 3. System Organization

When the state of gazing is detected by eye movement analysis, the sub-region in gazing is extracted and its relative location is measured originating from the upper-left corner of the inner frame of the painting. Next, the degree of overlap is calculated by referring to predefined sub-regions in the painting that is defined in the Painting Database. The most-overlapped sub-region is regarded as the gazing region. The gaze duration for the region is also measured and it is normalized by the average gazing duration for a sub-region of the user. Then, it is regarded as the degree of interest for the sub-region.

After that, user interest models of other users, which are similar to the target user, are extracted as described in section 3.2. Finally, recommended information is extracted by referring to the user interest models of other users in positive high correlation, and it is presented to the target user.

4.2 User Interface

The user interface of the prototype system is shown in Fig. 4(a) and Fig. 4(b). Figure 4(a) shows a user interface for confirming the result of gaze detection and sub-region detection in gazing. The upper left part of the interface marked as '<1>' shows the eye and the extracted pupil with the position of centroid. The upper middle section marked as '<2>' indicates a gazing point on the painting, which is shown as the cross point of a horizontal and a vertical line. The upper right section labeled as '<3>' is to specify a painting that he/she is in appreciation. Eye movement is decomposed into horizontal and vertical displacement, and it is graphically displayed at the lower part of the interface. Note that the primary purpose of implementing the prototype system is to evaluate the effectiveness of the proposed technique. Therefore, the

recommendation data and the sub-regions in a painting are displayed in the LCD display of a mobile computer in order to confirm how the system is working. Recommended information may be presented as a voice annotation or a sound notice with visual annotations so as not to disturb a user's activity of appreciating visual information in the practical operation of the proposed system.

Figure 4(b) shows an example of presenting recommended paintings. The painting placed at the upper-left corner of the interface is the painting that he/she is currently appreciating. Three paintings displayed at the lower of the interface are recommended paintings extracted by evaluating the similarity of user interest model. Descriptions on painter, title, category and the style of drawing are denoted below each painting. Recommended paintings are displayed in the descendant order of the degree of interest measured by referring to the users of similar interest model. Switching to display lower or higher in ranking is performed by clicking buttons on the right of recommended paintings.



(a) Detecting Visual Line and Inner Frame of a Painting

(b) Presenting Recommendation of Paintings

Fig. 4. User Interfaces

5 Experiments

5.1 Quality Evaluation by Object Granularity

We conducted experiments to evaluate the effectiveness of component object based (i.e., sub-region based) information recommendation by comparing it with object based (i.e., painting based) information recommendation. We prepared 21 paintings whose size is normalized to A3 paper size, and defined sub-regions based on depicted component objects in each of the paintings. The number of sub-regions ranged from 2 to 4, depending on the subject. The sum of the dimensions of all the sub-regions in a painting occupies approximately 80% of the depicted area in average. We prepared 20 testing subjects, all of them were graduate and undergraduate students who are unprofessional in art. Each testing subject was

instructed to stand in front of a painting, and the distance between an eye of the user and the picture was kept to be 50cm in each trial. The threshold to classify an interest model into positive high correlation by r_{ab} was set to 0.7, which was determined by pre-experiment. Note that the same condition was applied for conducting experiment described in 5.2 as well.

In the experiments, each subject appreciated 16 paintings first. Gazed sub-regions and the duration of gazing are recorded to construct user interest model based on component object. After that, each user is instructed to appreciate the rest of 5 paintings and rate them in accordance with his/her subjective preference. We regard this order of preference as the ideal result in information recommendation for the user *a*, which we denote as *Ra*. In the same way, we denote the order of recommendation obtained by component object-based interest model as *R'a*, and the order of recommendation obtained by object-based interest model as *R''a*. We measure the difference of the quality of information recommendation by means of *ndpm* (normalized distance-based performance measure)[9]. We evaluate the effectiveness of the component object-based information recommendation by comparative merits and demerits of *ndpm(Ra, R'a)* and *ndpm(Ra, R''a)*.

Here, we denote *n* as the number of objects (i.e., paintings) to be listed in order, and *m* denotes the number of pairs where the recommended objects are in the same rank but are different from each other. The definition of *ndpm* is as follows.

$$ndpm = \frac{m}{{}_nC_2} \tag{5}$$

The value of *ndpm* denotes the difference of permutation of preference between the ideal permutation of preference, *Ra*, and the permutation of preference obtained by information recommendation, *R'a* or *R''a*. The *ndpm* takes a value that ranges from 0 to 1, where 0 means two permutations are the same and 1 means two permutations have no similarity in the order of the elements. Since the value of 0.5 in *ndpm* corresponds to the theoretical figure where recommendation of object is decided randomly, we can argue the recommendation data is determined in desirable sense if *ndpm(Ra, {R'a or R''a})* takes less than 0.5.

First experiment is to evaluate the performance between object-based preference acquisition and component object-based, i.e., sub-region based preference acquisition. As described, the object-based preference is implicitly acquired as the gaze duration for a painting, which is normalized by dividing it by average gaze duration for a painting. Correlation between the interest model of the target user and those of others' are calculated as the same manner as sub-region based interest model correlation.

The result of experiment is shown in the Table 1. According to the result of t-test, significant difference is confirmed at 1% significant level. Therefore, we can conclude that the sub-region based information recommendation outperforms object-based information recommendation. That is, information recommendation based on sub-region based interest model offers better recommendation than that based on component object based interest model.

Table 1. The Result of Comparing Object-based with Component Object-Based Information Recommendation

	ndpm(Ra, R'a)	*ndpm(Ra, R''a)*	random
average	0.27	0.36	0.50
min.-max.	0.20-0.33	0.25-0.40	--

5.2 Comparison Between Implicit and Explicit Interest Model Construction

The second experiment is to evaluate the quality difference between explicitly specified interest model by user's subjective rating and implicitly constructed sub-region based interest model based on gaze detection. In order to equalize the rating in explicit and implicit interest model, both ratings are forced into 5 classes of preference (1 to 5; 5 corresponds to maximum positive preference). In constructing implicit user model, each subject is instructed to appreciate 16 paintings. Then, the degree of interest is measured by the gaze duration for each of sub-regions, and the duration is classified into one of the five degrees of interest. After that process, implicit interest model is constructed.

In constructing explicit interest model, each subject is instructed to appreciate 16 paintings and the strength of interest is manually specified for each of the paintings by 5 degrees of rating. Each subject was free to change once rated evaluation until the end of rating process.

After that, each subject appreciates the rest of the 5 paintings and rates in the order of his/her preference. We regard this rating, r_a, as the desirable recommendation by the user *a*. Here, we denote the order of rating calculated by implicit interest model as r'_a, and that by explicit model by r''_a. The quality of recommendation is evaluated with *ndpm*.

The result of performance comparison between explicit and implicit construction of interest model is shown in Table 2.

Table 2. The Result of Comparing Recommendation by Explicit Interest Model with Implicit Interest Model

	ndpm(r_a , r'_a) (proposed)	*ndpm*(r_a , r''_a) (explicit)
average	0.26	0.23
min.-max.	0.20-0.30	0.20-0.30

As shown in the table, the value of ndpm of explicit interest model is 0.23, and that of implicit interest model is 0.26. According to the result of t-test, there is no significant difference between them. Therefore, we can conclude that implicit construction of sub-region based, implicit interest model based on gaze detection can be the alternative to explicit interest model construction.

6 Conclusion

We proposed the method of implicit construction of interest model of users for information recommendation based on gaze detection. According to the experimental result, sub-region based interest model outperforms object based interest model. In addition, implicit interest modeling by gaze duration is comparable to explicit rating for constructing interest model. We think the result is remarkable because the proposed method keeps the quality of information recommendation without disturbing user's primary activity.

Acknowledgments. This work is partially supported by Grant-in-Aid for Scientific Research.

References

1. J. A. Konstan, B. N. Miller, D.Maltz, J. L. Herlocker, L. R. Gordon, and J. Riedl, "GroupLens: Applying Collaborative Filtering to Usenet News," Communications of the ACM, Vol. 40, No. 3, pp. 77-87, 1997.
2. A. Kohrs and B. Meriald, "Using Category-Based Collaborative Filtering in the Active Web-Museum," Proceedings of IEEE International Conference on Multimedia and Exposition, Vol. 1, pp. 351-354, 2000.
3. P. W. Folts and S. T. Dumais, "Personalized Information Delivery: An Analysis of Information Filtering Methods," Communications of the ACM, Vol. 35, No. 12, pp. 61-70, 1992.
4. M. Balabanovic and Y. Shoham, "Fab: Content Based Collaborative Recommendation," Communications of the ACM, Vol. 40, No. 3, pp. 66-72, 1997.
5. D. Goldberg, D. Nichols, B. M. Oki and D. Terry, "Using Collaborative Filtering to Weave an Information Tapestry," Communications of the ACM, Vol. 35, No. 12, pp. 61-70, 1992.
6. T. Hirashima, K. Hachiya, A. Kashihara and J. Toyoda, "Information Filtering Using User's Context on Browsing in Hypertext," User Modeling and User-Adapted Interaction, Vol. 7, No. 4, pp. 239-256, 1997
7. L. Chen and K. Sycara, "WebMate: A Personal Agent for Browsing and Searching," Proceedings of 2nd International Conference on Autonomous Agent, pp. 132-139, 1998.
8. T. Joachims, D. Freitag and T. Mitchell, "WebWatcher: A Tour Guide for the World Wide Web," Proceedings of IJCAI-97, 1997.
9. Y. Y. Yao, "Measuring Retrieval Effectiveness Based on User Preference of Documents," Journal of the American Society for Information Sciences, Vol. 46, No. 2, pp. 133-145, 1995.

The 3D Sensor Table for Bare Hand Tracking and Posture Recognition

Jaeseon Lee[1], Kyoung Shin Park[2], and Minsoo Hahn[2]

[1] Smart Interface Research Team, Electronics and Telecommunications Research Institute
161 Gajeong-dong, Yuseong-gu, Daejeon, 305-700, S. Korea
jaeseon@etri.re.kr
[2] Digital Media Laboratory, Information and Communications University
517-10 Dogok-dong, Gangnam-gu, Seoul 135-854, S. Korea
{park, mshahn}@icu.ac.kr

Abstract. The 3D Sensor Table system senses the movement of bare-hand and recognizes simple hand postures, such as stretched-hand, fist, and knife-shape hand. This system is designed for user interaction with real-time two- and three-dimensional graphics applications. It uses the electric field sensing technique to track bare-hand movements up to 30cm away from the display surface. This paper describes an overview of the system design, implementation, and algorithm for the 3D hand position and posture recognition.

Keywords: Electric field sensing technique, Bare-hand tracking, Hand-posture recognition, 3D display, Human computer interaction.

1 Introduction

With the technological advancement, computers become more powerful and widely used for various application domains, such as office automation and entertainment, with the video game becoming a huge industry. In particular, virtual reality is a computer technology that provides immersion, interactivity, collaboration, and direct user interface. It is widely used in the area of scientific visualization, education and training, medical rehabilitation, etc. As the computer technology paradigm is shifted to invisible and existed everywhere, the user interface is also developed to go beyond the use of keyboard and mouse. For example, the hand gesture interfaces have also been studied extensively recently since they are natural and familiar methods. Gesture is an importance means of human interaction, and it is an expressive body motion with the intent to convey information or interact with the surrounding environment.

In virtual reality, the most typical 3D user interfaces are wand (like, a three-dimensional mouse) or data glove, which provide accurate tracking and hand shape information but they are too cumbersome for the use over extended periods. Recently, there are some attempts on developing user interfaces for bare-hand human computer interaction. Among them, vision-based methods (i.e., with the use of camera) have been studied most vigorously. However, these systems have many limitations, such as expensive computational cost under lighting conditions, restrictive background, and etc.

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 138 – 146, 2007.
© Springer-Verlag Berlin Heidelberg 2007

Other researchers have explored alternative approaches, such as the electric field sensing technique, to make the recognition of bare-hand movements easier. However, the electric field sensing techniques are mostly developed for two-dimensional or non-contact type applications rather than three-dimensional applications.

In this paper, we present the 3D Sensor Table system which is designed for bare-hand tracking and hand shape recognition on a three-dimensional (3D) stereoscopic display. The aim of this system is to provide natural bare-hand user interaction with the 3D imaginary, while freeing users from wearing specialized input devices. The system uses the electric field sensing technique for recognizing user's bare-hand movements in three-dimensional space (15 to 30 centimeters from the display surface). It can detect multiple bare-hands simultaneously and simple hand postures such as straight, fist, and knife-shape. It can also be used in two-dimensional applications seamlessly with three-dimensional applications.

The paper reviews related works on bare-hand human computer interaction. It will then describe the system architecture of the 3D Sensor Table system and software algorithm for determining bare-hand tracking and hand posture recognition. Finally, it will present the conclusion and future research directions.

2 Related Works

There are many systems developed for bare-hand tracking and gesture recognition using computer vision techniques [1,2,3]. However, they must be restricted due to problems associated with vision-based systems, such as, lighting, complex background processing, and computation power requirement. This includes non real-time calculation [4], use of colored gloves [5], expensive hardware requirements (e.g. 3D-camera or infrared camera) [6][7], restrictive lighting conditions, restrictive background clutter [8], explicit setup stage before starting the tracking [9] and restrictions on the maximum speed of hand movements [9,10,11,12].

The electric field sensing technique computes the capacitance between a hand and an insulated array of metal electrodes. The presence of a hand effectively increases the electrode capacitance to the ground since the capacitance between a conductive hand and an electrode is typically very weak while the capacitance of human body with respect to the earth ground is relatively large. This technique has numerous advantages against vision-based systems. For examples, it is simple, inexpensive and easily scalable in hardware implementation. The better resolution and the larger volume of active tracking can be obtained by simply increasing more number of electrodes.

However, the major disadvantage of this technique is mathematical complexity of inferring the object's position and orientation accurately from the indirect measurement of electrical properties since the coupling between object and electrode is nonlinear. Hence, the object's position and orientation is determined in terms of ambiguity classifications and probability distributions. Another drawback is the type of object being sensed is restricted. That is, tracking different types of object requires the use of different modeling assumption. The modeling of human hand or body is widely used in electric field sensing research. According to Smith [15], real time hand

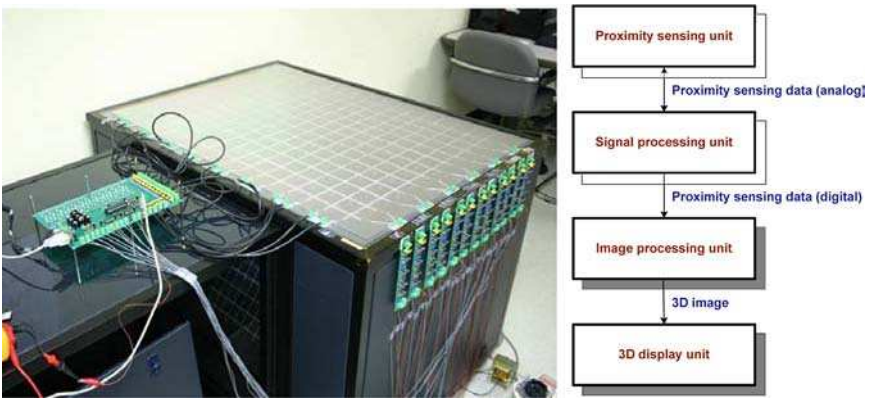


Fig. 1. The 3D sensor table system and the system architecture overview

position tracking using the electric field sensing appears to be feasible, while real time orientation tracking is still remained unknown.

Unfortunately, most near-field capacitive sensing techniques have been used as simple touch switches. While there had been some works done on using an electric field sensing technique for bare-hand position tracking, the tracking range was a few centimeters away from the display surface (and hence, it was not suitable for 3D graphics applications) since they were mainly designed for the 2D graphics applications [13,14].

3 System Design and Implementation

Fig. 1 shows the system architecture of the 3D Sensor Table. The system consists of four main parts: the proximity sensing unit, the signal processing unit, the image processing unit, and the 3D display unit. The proximity sensing unit is made up of sixteen copper-coated TX (Transmit) wires and twelve RX (Receive) wires arranged orthogonally on the transparent acryl board installed on the screen. It measures the proximity values on the crossing nodes of TX and RX wires according to the bare-hand movement over the screen. The signal processing unit transmits proximity values of TX and RX signals to the image processing unit. The image processing unit computes the proximity detection and posture recognition. The 3D display unit is a rear-projection stereoscopic table-type display using two DLP projectors for left and right eye image rendering respectively.

The proximity sensing unit uses the electric field sensing technique derived from Faraday's law for sensing the movement of bare-hand over the screen. It consists of sixteen copper-coated TX (Transmit) wires and twelve RX (Receive) wires that is 0.3mm in diameter orthogonally on transparent acryl board, and the interval between wires is 5cm apart. Fig. 2 illustrates the principle of proximity sensing of a bare hand over one TX line. When a bare-hand is over the crossing point between TX and one of

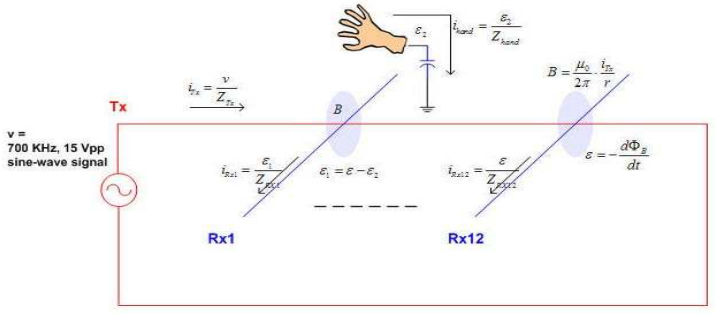


Fig. 2. The principle of sensing the proximity of bare-hand on 1 TX channel

the twelve RX wires, the electric current (i_{TX}) that flows through TX wire is constant in the normal state because it is related to the size of TX signal and the impedance of TX wire. But, the magnitude of the induced EMF (electromotive force) in RX wire changes according to the distance between a bare-hand and the crossing point of TX and RX wire because some of induced EMF (ε_2) flows into the earth through a bare-hand. Since the induced EMF (ε_1) in RX wire decreases as much as it flows into the earth, the induced electric current (i_{RX}) is reduced in RX wire. Therefore, if ε is the induced EMF in the normal state and ε_2 is the induced EMF that flows into the earth when a bare-hand approaches to the crossing point of TX and RX wire, ε_1 (the induced EMF in RX wire) and i_{RX} (the induced electric current) can be obtained from Equation (1):

$$\begin{aligned} \varepsilon_1 &= \varepsilon - \varepsilon_2 \\ i_{RX} &= \frac{\varepsilon_1}{Z_{RX}} = \frac{\varepsilon - \varepsilon_2}{Z_{RX}} \end{aligned} \tag{1}$$

The signal processing unit consists of the master processing unit (that controls TX signals and transmits RX signal converted into digital value to the image processing unit) and the RX processing unit (that converts RX signal to be suitable for A/D conversion). The master processing unit uses Microchip's microcontroller, PIC16F877A. To control TX signal, it uses two Motorola's multiplexers, MC14051B, and it applies TX signal generated from the Agilent's wave-generator, 33250A, to one of sixteen TX channels one by one. When twelve RX signals enter into RX channels, multiplexer connects RX channel to the input terminal of A/D converter one by one. After the A/D conversion of RX signals, the digitized RX values are sent to the image processing unit through RS232 communication. The RX processing unit transforms RX signals from the proximity sensing unit to be suitable for A/D conversion by noise filtering, amplification, and rectification. In the RX processing unit, the input signal is originally 50 ~100 mVpp, sine-wave AC. The transformer and HPF are used to remove other signals except 700 KHz signal. Through this process, 60 Hz noise is removed. The national semiconductor's operational amplifier, LM318N, amplifies

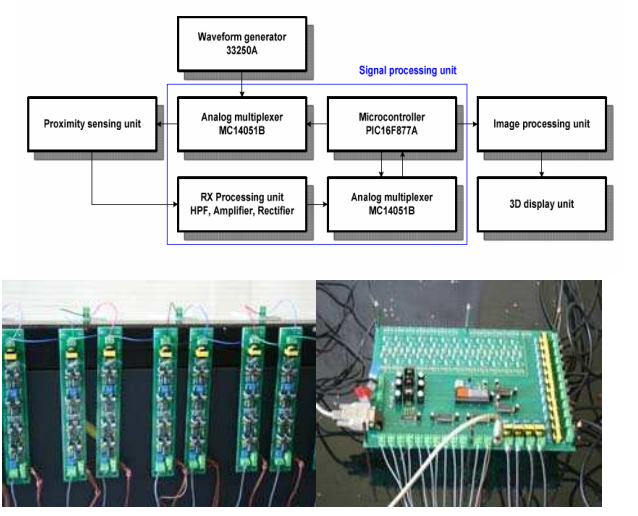


Fig. 3. The master and the 12 RX processing in the signal processing unit

weak RX signals to support high slew rate (70 V/us) that is enough for high precision speed operation. LM318N is used to rectify amplified RX signal in AC to change it to DC signal suitable for A/D conversion. Finally, the rectified RX signal is amplified to have the suitable magnitude (2.5 ~ 5 V) for A/D conversion. Fig. 3 shows the photograph of the signal processing unit.

The image processing unit analyzes the proximity data obtained from the signal processing unit to track the position of bare-hand and recognize the hand posture. The software algorithm for hand position and posture recognition will be explained in the section 4. Finally, the 3D display unit projects the 3D image (created by the image processing unit) on the rear-projection table screen. A stereoscopic 3D image is produced through the dual-video output of graphics card for right and left image respectively, which are connected to two DLP projectors. There are polarized-light filters in front of the projector lens to separate the left and right image. The 3D images are reflected through the mirror to rear project onto the table screen. Users must wear polarized-light filter glasses to see the 3D image.

4 Software Algorithms

The image processing unit analyzes the proximity data obtained from the 192 crossing nodes (of TX and RX wires) every 33 ms in the signal processing unit. It then finds the position of bare-hand and classifies simple hand postures (such as stretched hand, fist, and knife-shaped hand). Fig. 4 shows the process of producing the 3D image interacted with the bare-hand location and hand postures.

In the data acquisition process, it receives 384 bytes raw data every 33 ms through serial communications from the proximity sensing unit. Next, the proximity data

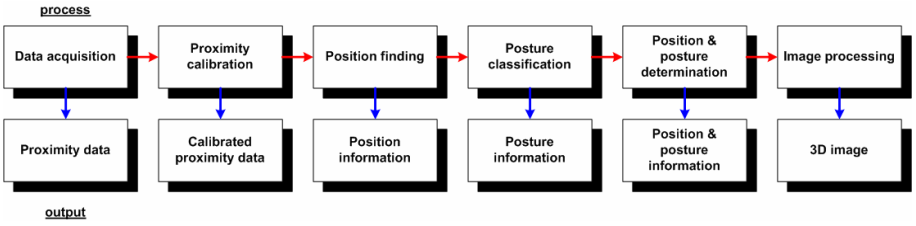


Fig. 4. The proximity calculation and posture recognition process of the image processing unit

modification is done to make raw proximity data to be more consistent – there is no constant initial value in the normal state. Next, the position of bare-hand is detected through finding the peak value. Then, the proximity data of 7x7 neighbor nodes (among 192 crossing node of TX and RX wires) whose center node represents the position of hand is analyzed to classify one of the three simple hand postures, such as stretched hand, fist, and knife-shape hand. However, the proximity values are varied according to the height of hand location from the screen surface as well as the hand shapes. The height of the hand is recalculated from the information of hand posture again. Finally, the 3D image is produced according to the hand position and posture.

The 3D Sensor Table system has 750 (width) x 500 (length) x 200 (height) mm sensing ranges that can detect the movement of bare-hands over the screen. The system coordinate, x and y, is the same as the two-dimensional screen coordinate system, and z represents the height of hand from the screen surface. The 2D hand position is measured by finding the peak value on the 192 TX and RX crossing nodes, and the height is measured by using the relationship between the maximum proximity value and the distance. The resolution of bare-hand tracking on the 2D screen is 5 cm because TX and RX wires are arranged by 5 cm interval. When a bare-hand move over the screen, the proximity value of crossing nodes is changed according to the movement of hand and the crossing node with the maximum (or peak) proximity value is selected as the hand position. However, the maximum proximity value is changed according to the height of hand (i.e., the distance between hand and the screen). In addition, the relationship between the maximum proximity value and the height changes rapidly when the hand is located closely to the screen, and the proximity value changes gently when the hand is located far away from the screen.

The 3D Sensor Table system also detects the simple hand posture, such as the stretched-hand (the basic default hand posture), fist, and knife-shape hand. This posture classification is needed for the proximity value determination because the proximity value is changed according to the height of hand with the different hand-postures. In addition, different hand postures can be used for triggering special events in human computer interaction – for example, the moment that a user changes his/her hand from stretched to fist can be used as a mouse clicking event. The hand posture classification method uses only 7x7 neighbor nodes whose center is the position of hand determined by the peak signal value. In this system configuration, vision-based pattern recognition such as using a predefined template (that simply compares the input image and the stored template to detect shape) cannot be used because the system resolution is only 16x12 (on the TX and RX wire grid). Thus, it uses the

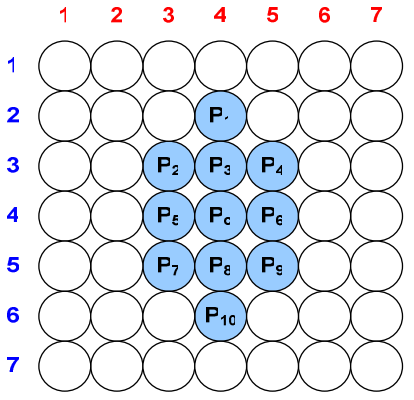


Fig. 5. The eleven crossing nodes used for hand-posture classification in the image processing unit

simple comparison between the central crossing node (that represents the position of hand) and the adjacent 7x7 crossing nodes.

As shown in Fig. 5, the proximity value of eleven crossing nodes ($P_1 \sim P_{10}$) among 7x7 nodes are used for classifying the hand posture. With the straight-hand posture, the distance of tracking bare-hand is about 30 cm away from the screen. On the other hand, the distance of tracking a fist hand or knife hand shape is about 15 cm away from the screen, and the error rate is also increased as the height of hand goes higher than 15 cm. This means that the range of tracking hand position and posture is limited within the distance of 15 cm away from the screen in order to use hand-posture changes for events. In a preliminary user evaluation, the accuracy of hand position detection was about 80 ~ 90 % and the accuracy of classifying the hand postures was about 80 %.

5 Conclusion and Future Works

This paper describes the 3D Sensor Table system designed for the bare-hand human-computer interaction with the stereoscopic three-dimensional display. The system allows users to move their bare-hand over the screen and interact with 3D images without wearing any special input device. This system implements the electric field sensing technique on the sensing board installed on the screen of the 3D display. Also, it can be installed on any display surface, such as front-projection screen, rear-projection screen, or a regular flat screen monitor (e.g. LCD). The system can track bare-hand movements within 15 to 30cm from the display surface depending on the hand shapes (i.e., it detects a stretched-hand nearly 30 cm), and it distinguishes three simple postures, such as stretched-hand, fist, and knife-shape hand posture. This paper presents the system architecture and the software algorithm that detects the proximity value changes according to user's bare-hand movements and simple hand postures over the proximity sensing unit installed on the 3D display surface. It can also be used in non-contact two-dimensional graphics applications.

The system detects the hand movement at the speed of about 30 fps (with the 33 ms of the sensing refresh rate). The proximity sensing unit resolution is 5cm interval (i.e., 5 cm interval between TX and RX crossing nodes in the proximity sensing unit, and 5 cm interval of height range). The 3D Sensor Table system measures the height of hand (i.e., the distance between a hand and the screen) discretely in seven regions, 0, 5, 10, 15, 20, 25, 30cm. The stretched-hand posture can be detected up to 30 cm away from the screen, and the fist posture and knife-hand posture can be sensed up to 15 cm. This system is more robust on the lighting condition or complex background than most vision-based bare-hand tracking systems. It is easily extensible in the resolution or the size of proximity sensing unit by simply adding more electrodes and making the sensing unit more fine grid structure.

In addition, we believe, the system will detect hand shapes more accurately if the TX and RX crossing nodes are rearranged to shorter intervals (such as, 1 cm) and it will also extend the proximity height sensing range further if the magnitude and the frequency of TX signals are increased. We are planning to make the proximity sensing unit wire invisible (i.e., completely embedded into the screen) by using a transparent ITO (Indium Tin Oxide) instead of coated copper TX and RX wires (currently with 0.3 mm diameter). We will improve the current implementation of hand position tracking and posture classification algorithm by adopting neural network pattern recognition techniques. Finally, we need to add the hand orientation detection algorithm for more natural user interaction. For this hand orientation recognition, we will examine the proximity value changes by the presence of bare-hand and arm over the screen.

Acknowledgements. This research was supported by the MIC (Ministry of Information and Communication), Korea and supervised by the IITA (Institute of Information Technology Assessment).

References

1. Christian von Hardenberg, Francois Berard. Bare-Hand Human-Computer Interaction, PUI2001, 2001.
2. Jun Rekimoto, Nobuyuki Matsushita, Perceptual Surfaces: Towards a Human and Object Sensitive Interactive Display, Workshop on Perceptural User Interfaces (PUI'97), 1997.
3. Yasuto Nakanishi, Yoichi Sato, Hideki Koike, EnhancedDesk and EnhancedWall: Augmented Desk and Wall Interfaces with Real-time Tracking of User's Motion, In Proc. of Ubi-Comp'02 Workshop on Collaborations with Interactive, Walls and Tables, pp. 27–30, 2002.
4. Triesch, J. and Malsburg, C. Robust Classification of Hand Postures Against Complex Backgound, International Conference On Automatic Face and Gesture Recognition, Killington, 1996
5. Lien, C. and Huang, C. Model-Based Articulated Hand Motion Tracking For Gesture Recognition, Image and Vision Computing, vol. 16, no. 2, 121-134, February 1998.
6. Rehg, J. and Kanade, T. Digiteyes: Vision-based human hand tracking, Technical Report CMU-CS-93-220, School of Computer Science, Carnegie Mellon University, 1993.

7. Sato, Y., Kobayashi, Y. and Koike, H. Fast Tracking of Hands and Fingertips in Infrared Images for Augmented Desk Interface, International Conference on Automatic Face and Gesture Recognition, Grenoble, 2000
8. Segen, J. GestureVR: Vision-Based 3D Hand Interface for Spatial Interaction, ACM Multimedia Conference, Bristol, 1998.
9. Crowley, J., Berard. F., and Coutaz, J. Finger tracking as an input device for augmented reality, Automatic Face and Gesture Recognition, Zurich, 195-200, 1995.
10. Laptev, I. and Lindeberg, T. Tracking of Multi-State Hand Models Using Particle Filtering and a Hierarchy of Multi-Scale Image Features, Technical report ISRN KTH/NA/P-00/12-SE, September 2000.
11. MacCormick, J.M. and Isard, M. Partioned sampling, articulated objects and interface-quality hand tracking, European Conference on Computer Vision, Dublin, 2000.
12. O'Hagan, R. and Zelinsky, A. Finger Track - A Robust and Real-Time Gesture Interface, Austrlian Joint Conference on Artificial Intelligence, Perth, 1997.
13. Lee, S. K., Buxton, W. and Smith, K. C. A Multi-Touch Three Dimensional Touch-Sensitive Tablet, In Proceedings of CHI '85 (April 1985), ACM/SIGCHI, NY, 1985, pp. 21–25., 1985.
14. Jun Rekimoto. SmartSkin: An Infrastructure for Freehand Manipulation on Interactive Surfaces, CHI 2002, April 20-25, 2002.
15. J.R. Smith, T. White, C. Dodge, J. Paradiso, and N. Gershenfeld. Electric Field Sensing for Graphical Interfaces. IEEE Computer Graphics and Applications, 18(3):54-60, 1998.

Legible Collaboration System Design

Toshiya Fujii, Wonsuk Nam, and Ikuro Choh

Waseda University, Global Information
and Telecommunication Studies
Media Design Laboratory
Taito Designers Village, 2-9-10 Kojima, Taito-ku, Tokyo 111-0056, Japan
Tel.: 81+3-3865-0271
tfujii@asagi.waseda, monsoon@toki.waseda, choh@waseda

Abstract. There is an abundance of idea processing tools available for individual use while, at the same time, there is a shortage of environments supporting collaborative thinking and brainstorming. Additionally, recorded transcripts from meetings, conferences etc. principally consist of letters and symbols to be used as future aid when recalling the proceedings. We argue that this recognition based recording method is limited, since much information cannot be captured in literal data. For example, it is difficult for a non-attendee to recall and experience an accurate reflection of the circumstances surrounding a previous meeting only through these recognition-based recordings. This problem stems not only from the inability to record the other attendant's characteristics and conduct, but also from the difficulty to capture the atmosphere and other surrounding physical information. Thus, there is a need for a system incorporating functions to record physical factors while allowing CSCW (Computer Supported Cooperative Work). In our research, we are developing a table based collaboration support system not strictly limited by time and location. In doing so, we propose an enhanced table with a legible input system supporting direct image manipulation and video recording functions.

Keywords: CSCW, Memory recollection, legibility, recognition, direct manipulation, Association Memory, Information Filing.

1 Introduction

When computers were introduced to offices around the world, the expectations of increased efficiency and the realisation of the paperless office suddenly seemed possible. In reality, however, it cannot be said that these aspirations have been achieved. Moreover, although it can be argued that computers have contributed to quality improvement in design work, they have simultaneously brought extended working hours as a side effect. While it can be recognized that the GUI and the direct operating environment of personal computers has been helpful to a majority of the

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 147 – 155, 2007.
© Springer-Verlag Berlin Heidelberg 2007

users, there is a great discrepancy in the development of computer aided work support environments targeting collaborative office work.

1.1 Our Goal

Difficulties with present teleconference systems often stem from problems related to the lack of ability to accurately read facial expressions combined with sound transmission delays. To combat these problems, research development in broadband technology, multiple screens utilization and work to attain higher resolution is in progress. However, this direction of development does not necessarily improve joint work. Rather, it suggests that there is an urgent need to reconsider what a true collaborative environment really is. In order to deal with these issues we are developing and researching a table type direct manipulation system.

1.2 Related Researches

In his paper, Digital Desk [1], Wellner is experimenting with redefining the desktop metaphor coupled with the personal computer. In his experiment, a projector and a camera was fitted above a desk, projecting images of documents onto the desk that were interactive using image recognition technology. This research was pioneering in the quest to unite the inside of the computer with the physical world. While this technology has been adopted and developed by Kobayashi, Winograd, Rekimoto and others, it can be argued that it has merely led to extending the direct manipulation utilized in the personal computer world. In the area of research targeting joint work between remote locations Wellner´s Double Digital Desk and Yamasaki´s Agora [2] can be mentioned. The former utilizes two or more Digital Desk setups and connects them by video link, the latter aims to depict the physical arrangement surrounding the table and conveying it to a distant location. This research aims to build a teleconference system with presence as the key purpose. Using the current hierarchical folder system, it is impossible to efficiently access information recorded automatically. In My Life Bits [3], a project based on Vannevar Bush´s memex vision from 1945, Bell and others explores the possibility of storing and digitalising all information obtained through normal life into a hard drive. The digital information is then to be processed and sorted to fit in on a timeline, easily accessible and visualised later on. The idea of abolishing the folder structure and progressing towards a time-line based structure is also advocated by Rekimoto in Time Machine Computing [4]

2 System and Theory

At our Laboratory, Legible Design System (Fig. 1) has been a research project for several years, and we have built a support environment for editorial and design purposes.



Fig. 1. Legible Design System



Fig. 2. Legible Collaboration System Prototype

The hardware in this research serves as a fundamental derivation. The table we use measures 700mm. in height, and has a surface area measuring 900mm in depth with a variable length. It is our intention that the table should be in a format allowing not only casual meeting style interaction sitting down, but also standing interaction from all sides of the table (Fig. 2).

The system layout consists of a video camera for wide spread capture, a separate narrow focus camera for capturing objects in a predefined area, a microphone and a display domain equipped with position sensing technology. (Fig. 3)

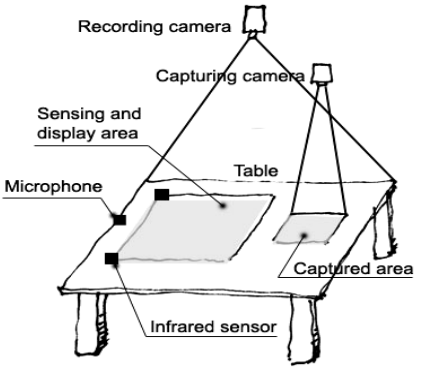


Fig. 3. Conceptual Design

We manufactured act@cubes™ for trial which is based on above compositions. The concrete system premised production. This system proposed several ideas as operation for surface. (Fig. 4) Further, novel design is on progress with new sensing system for embodiment multi-pointing operation. (Fig. 5)



Fig. 4. act@cubes™



Fig. 5. Novel design on progress

2.1 Capturing Objects

While it can argued that we live in the computer age, it does not necessarily mean that all information available is digitalized. Effectively sharing physical information, exchanging opinions and communicating between different locations in different time-zones grow increasingly difficult. The only solution to accomplish this is to digitalize the information. We use the wide angle capture camera for this purpose (Fig. 6). The narrow focus camera is used to capture in high resolution. The capture operation is carried out by placing the object on the capture podium. A pressure sensitive switch underneath the podium serves as the capture control. After an object has been captured and digitalized, it automatically slides into the projected screen area on the table. The captured data is saved in a database as an object, and direct manipulation such as rotation, movement, duplication, deletion and scaling can be



Fig. 6. Image capturing

performed through finger input. If the captured object is not used for a set period of time, it will gradually fade out from the screen in order to keep the table clean and provide space for other interaction. This is an automatic cleanup mechanism for the table, and control of transparency levels can be executed by the user at any time. In addition, the reason for the existence of two separate screen domains is to avoid picture degradation by the video loop between the camera and the screen.

2.2 Circumstance Recording and Linking

Our system is designed to capture information not only based on what is happening on the table, but also information from a close proximity around the table via the wide angle camera. For example, data recording of who is present around the table, who operates what on the table, who is speaking etc. The main objective of the wide angle camera is not to capture high resolution images of the people, but rather to capture and record the atmosphere of the surrounding interaction. In order to catch this data, the camera and the microphone can be set to alternate modes. We are also exploring the possibility to process people as shadows. Our current experiment is designing a trial system utilizing shadows which is an abstract information as a concept when displaying attendance of other participants. Remote users are represented as hand shaped shadows on the surface of a table. (fig.7)



Fig. 7. Shadow form a remote place

2.3 Collaboration in Co-location

Research involving remote collaborate activities is conducted in various fields for various purposes. Most of this research is limited to overcoming physical limitations and aspects of remote collaboration. However, there is a lack of research focusing on co-location. The trial system being developed is not only supporting remote collaboration, it also includes a new style for conducting remote collaborative work. Further, depending on how the system is connected, it can dynamically display how many people are participating simultaneously. (Fig.8)

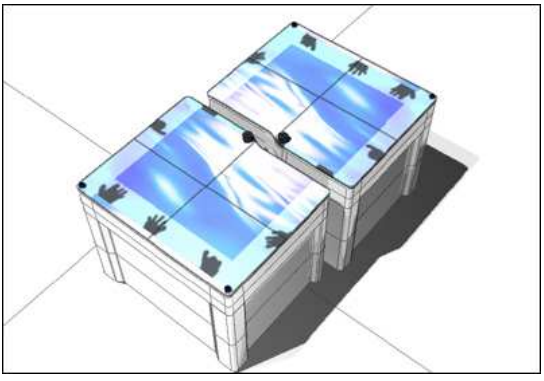


Fig. 8. Direct connection

2.4 Multi-point Sensing and Object-Finger Parallax

It is exhausting operate a machine with a different glance position of the operation and the position of the tool. It is the largest problem of current personal computers GUI, and it is mysterious that most people are using this interface without questioning it. On the other hand, to solve physical position differences, some researchers and developers are trying to utilize touch panel technology on PC's and PDA's. However, it alone cannot totally solve the problem of keyboards that have different insertion point and physical key position. These alternatives do not reach the production efficiency of an existing personal computer by far. We believe that using direct manipulation with an entire table width will become breakthrough, and that multi-user finger recognition technology and specialized tools for operation are important.

In this trial, the table has a non-sensible area around the edge, so participants can safely put their hands or other objects there without affecting the input functions. Also, with this system, it is possible to put thin objects like document paper, pencils and notebooks on the sensible area. This system is using NEXTRAX™ technology [5] which has infrared sensors to recognize multi-point interaction. Therefore, two or more people simultaneously are able to operate this system. Moreover, a pen tool is available for drawing purposes. The drawing is managed as drawing-object in the system as well as capturing-object. The objects gradually fade out from screen, so this system has no need for clearing screen function.

2.5 Timeline Visualization

It is virtually impossible by using the folder concept on a personal computer system to retrieve information that has been automatically recorded. Therefore, we are proposing timeline visualization as a basic axis for solving this problem. There are two modes for this system, one is a recording mode and the other is a memory recall mode. In contrast, the recording mode automatically collect information, the memory

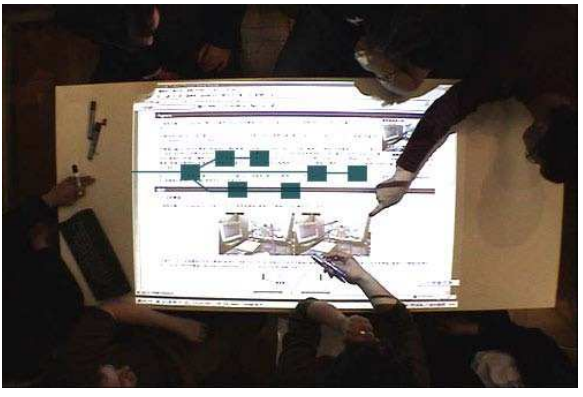


Fig. 9. Time-line Visualization

recall mode can then be used to draw out past information from the data storage. In the memory recall mode, the recorded video image is projected onto the screen overlapping the timeline visualization horizontal axis. (Fig.9)

Additional information like time-stamp, number of participants etc. can also be attached to the timeline. The captured objects and the drawn objects mentioned above are also represented as thumbnails on the timeline. In other words, the act of capturing and drawing are memorized in the system as an event. We are planning to incorporate automatic gesture recognition functions; voice information and specialized participant related features in future work. Moreover, the transition between recording mode and memory recall mode becomes one of the events causing the branch dividing of the timeline (Fig. 10), and it is shown to extend like two dimensional graphs. This may correspond to a thinking simulation or recalling visualization of our brain. The shape of the voice wave is superimposed on the timeline too, as it is convenient to know

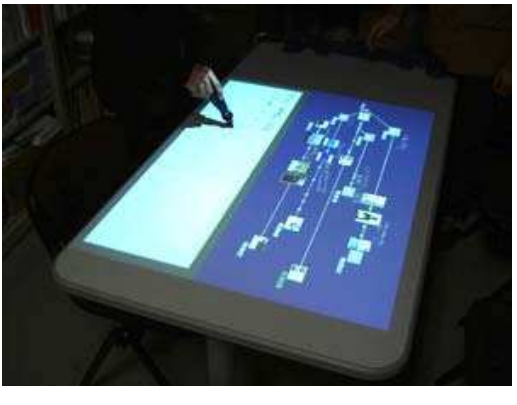


Fig. 10. Thumbnail and Branching

when participants spoke. The displayed thumbnail can be duplicated by simple drag and drop maneuvers and it becomes targeted in the operation again.

In this concept, two timeline operations exist for searching. One is a time shifting slider; the other is a time unit slider. The participants can freely move from one position to another on the timeline like in a time-machine by using the time shifting slider function. Using the time unit slider, users can browse information from a bird's-eye view. Unlike the folder concept were it is not possible to really see what is inside a folder until opening it, this system allows for seamless browsing of complete information while permitting free movement between upper and lower level views. (Fig.11).

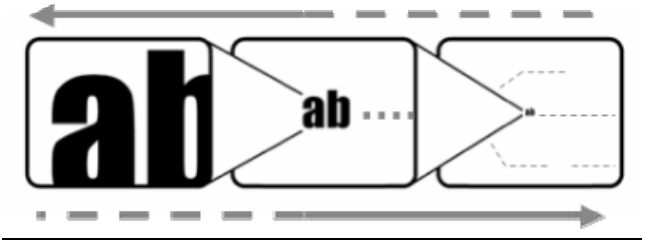


Fig. 11. Timeline Structure

3 Conclusions and Future Study

In this research, Legible Collaboration System was deployed experimentally. We found the system to be easy to use even for participants inexperienced with similar technology. Information sharing between participants and memory recall support functionality works well. At present, we are planning further experiments for data collection. And, we would like to attempt to utilize this system for sharing information with remote places. Other issues currently being worked on include accuracy improvement of the infrared sensor and the expansion of tools which are available in the system.

Acknowledgment

The correct BibTeX entries for the Lecture Notes in Computer Science volumes can be found at the following Website shortly after the publication of the book: This research is a collaborative project with CAD CENTER Corp. and ITOKI Co., Ltd., The infrared sensor named NEXTRAX™ is developed by CAD CENTER Corp. ITOKI Co., Ltd. is taking charge of the production named act@cubes™ and marketing of the system that will be on sale next spring. These trademarks are the registered trademark of CAD CENTER Corp and ITOKI Corp.

References

[1] P. Wellner, 1993, Interacting with paper on the digital desk, Communications of the ACM, vol. 36, No. 7, pp. 87-96.
[2] A. Yamazaki, 1999 Agora: A Remote Collaboration System that Enables Mutual Monitoring, CHI'99 Extended Abstracts, pp.190-191.
[3] J.Gemmell, 2002, MyLifeBits: Fulfilling the Memex Vision, ACM Multimedia '02, Juan-les-Pins, France, pp. 235-238.
[4] J.Rekimoto, 1999, Time-Machine Computing: A Time-centric Approach for the Information Environment, UIST'99, pp.45–54.
[5] NEXTRAX, http://www.nextrax-cadcenter.com/

Presentation of Dynamic Maps by Estimating User Intentions from Operation History

Taro Tezuka and Katsumi Tanaka

Graduate School of Informatics, Kyoto University
Yoshida-honmachi, Sakyo-ku, Kyoto, 606-8501, Japan
{tezuka, tanaka}@dl.kuis.kyoto-u.ac.jp

Abstract. Advances in dynamic map interfaces have turned maps into interactive media. These dynamic interfaces respond to users' operations in real time, and present fully visualized geographic information. However, the current systems have only reacted to explicitly specified user intentions. For example, users have been required to elaborately specify visible layers to fully utilize a map interface. In contrast, we propose a method of adjusting the way a map interface is presented by estimating the users' intentions based on their operation history. By reducing their operations, the system facilitates the use of maps especially for novices. It is especially effective in online or mobile map interfaces, where it is difficult to adjust the presentation of the map interface, due to the limited bandwidth and the size of the interface. This paper specifically focuses on the trajectory, which is a series of panning operations, and discusses our inference of users' implicit intentions.

Keywords: Map interface, user intention, trajectory, oeration history.

1 Introduction

Advances in dynamic map interfaces have turned maps into interactive media. Map interfaces on the Web or mobile devices have recently been dramatically increasing the opportunities for general users to benefit from dynamic maps. However, a map presented on the Web or a mobile device suffers from two limitations. The first is the data transmission bandwidth and the second is the limited size of the presentation area. It is necessary to adjust the content to meet users' intentions in such cases to reduce the amount of information being transmitted or presented.

One way to infer users' intentions is to use their operation history. Conventional map interfaces have only responded to each user's operations, and not to the series of operations. This paper proposes a system that infers users' intention from a series of operations. Here, we do not discuss highly sophisticated functions for special-purpose map interfaces such as those used for spatial analysis by GIS (Geographical Information Systems) specialists. Instead, we cover a basic series of operations done by general users and infer their intentions.

Although actual implementations of such advanced functions may vary, we can have an abstract model. We discuss such a model in this paper. Some of the use case scenarios of a map interface are described below.

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 156–165, 2007.
© Springer-Verlag Berlin Heidelberg 2007

Search task: John wants to visit the new Toni's Restaurant in the downtown area. He types in the address, locates the building, and finds his way from the nearest station.

Browse task: John watches a TV show and wants to know more about the Imam Square in Isfahan, Iran. He types in the place name, and browses around to get better knowledge of the square, such as the names of the buildings surrounding it.

Note that there is a significant difference between two tasks in types of information required by the user, and also in the user's operations involved in performing the tasks. We focus on such differences to optimize the information presented to the user on the map interface.

The rest of the paper is organized as follows. Section 2 describes related work. Section 3 discusses users' operations and their intentions on a dynamic map interface. Section 4 describes an application built over the framework of our model, and discusses our evaluation of its effectiveness. Section 5 is the conclusion.

2 Related Work

Dynamic map interfaces are currently made available in many ways. A number of Web-based dynamic map interfaces are especially prominent. Google's local search service, Google Local, presents search results on a map interface [12]. Yahoo! Local Maps is a map-based local Web-search system provided by the major portal site, Yahoo! [13].

Claypool et al. discussed the estimation of intentions based on user operations on a Web browser [4]. Mouse clicks, mouse movement, scrolling and elapsed time were measured using a special browser, *The Curious Browser*. The system obtains *implicit ratings* of each Web page using such measures. Their method was based on the user's operations on a Web browser, therefore was specific to Web browsing.

Nielsen described a non-command interface as a next generation user interface [5]. Their proposed interfaces involve portable devices that estimate user intentions from the environment that they are involved with. Hijikata extracted user interests from the mouse motion of a Web browser, and used them in relevance feedback to search for similar documents [6]. The system uses only the parts that the user might be interested in, instead of using the entire pages. Text tracing, link pointing, link clicking and text selection were used to determine the part the user is interested in.

Mueller and Lockerd developed a system, Cheese, which records mouse movements and infers the user's interest [7]. Unlike our research which aims to automatically adjust the map features, their goal was to help content providers to increase the effectiveness of their interface design when creating a page manually.

Goecks and Shavlik described a method of obtaining user's interests by measuring the number of hyperlinks clicked or the amount of scrolling performed [8]. After training, the system could predict 1) the number of clicks on hyperlinks, 2) the amount of scrolling, and 3) the amount of mouse movement by analyzing the Web text. Although the system does estimate the user's intention, application of the result was not in the focus of their paper.

Weakliam et al. discussed inference of the users' intention on a map interface from their past behavior stored in a log file. Their research was different from ours in that

their main goal was to find the most relevant geographic features from past operations [1]. There method stores operation history for a long period of time, and gradually adopt to the user's preference. On the other hand, our method reacts almost instantly to the user's operation and dynamicaly changes the map content.

Hiramoto and Sumiya proposed a system of searching Web content based on a series of user operations on a map interface [2]. Their aim was to search relevant Web content to match the user's intention, rather than adjusting the content in the map interface as our system does. Zhang et al. discussed location-based spatial queries for mobile clients moving around in space [9]. Their method employed a validity region to reduce the number of frequent updates from closest neighbors. Tao et al. discussed the assignment of closest neighbors to every point on a line segment [10]. Ishikawa et al. proposed a function to search information for car navigation systems along their trajectories [11]. Their work was based on locating a moving object and establishing its velocity at points along the route.

3 User Operations and Intentions in a Dynamic Map Interface

This section describes our modeling of user interactions with a dynamic map interface. The process, when they interact with a dynamic map interface, involves three layers: semantic intention, action intention, and operation. The distinction between semantic and action intentions has been discussed by Chen et al. [3].

Semantic intention is a long-term intention characterized by meaningful information obtained using the map interface. It is the user's goal in using the interface. Action intention is a short-term intention, characterized by the resulting state of the map interface. Action intentions are planned to achieve semantic intentions. Operations are performed to achieve action intentions.

We will define concepts and discuss them in detail in the sections that follow.

3.1 Definitions

This subsection defines the special terms used in this paper. The elements of a dynamic map interface are defined as follows:

Map interface: A user interface that presents a map and a tool set.

Map-view area: A map presented on the map interface.

Target geographic area: The physical area in a geographic space that is presented on a map-view area.

Area of interest: The physical area that the user has interest on.

Item: An element that could be presented on a map interface. An item generally contains spatial data used to determine where it should be presented on the map interface. Some items are points, while others are lines or polygons.

Layer: A set of items belonging to a specific category.

Response: An activity carried out by the system that is evoked by user operation.

Relevant layer set: A set of layers in a GIS that is relevant to the user's intention.

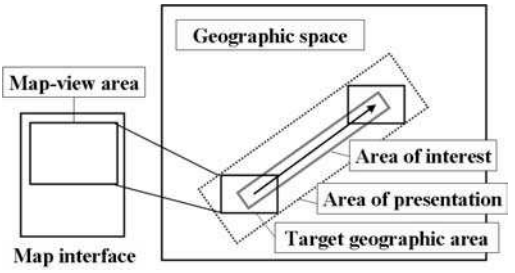


Fig. 1. Definitions of areas on map interface

3.2 Inference of Intention Based on User Operations

This section discusses the inference of user intentions from the feature values of operations. We discuss two ways of distinguishing panning operations. The first is distinguishing between a route-oriented pan and a goal-oriented pan. The second is distinguishing between a reach-to pan and a search-around pan.

Route-oriented pan and goal-oriented pan: There are at least two types of panning operations with different intentions as discussed in Section 3.

The first is a route-oriented pan, where the user is panning to find information along a route. The second is a goal-oriented pan, where the user wants to reach a certain destination, and is not interested in the regions in between. These are the two distinct types of intentions for panning, and must be distinguished from operations. Figure 2 is an illustration of these two types of pans.

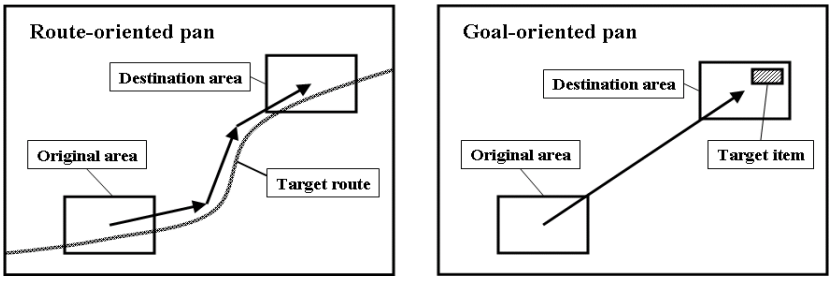


Fig. 2. Route-oriented pan and goal-oriented pan

A series of panning operations results in a trajectory. The system with our method aggregates data concerning the trajectory and estimates the user's intentions. We propose *relative-width* as a measure of representing the level of route-orientedness and goal-orientedness. Relative-width $r(T)$ of trajectory T is defined as

$$r(T) := \frac{\sum_i |\mathbf{x}_i \times \sum_j \mathbf{x}_j|}{|\sum_j \mathbf{x}_j| \sum_j |\mathbf{x}_j|} \qquad (T = \{\mathbf{x}_0, \mathbf{x}_1, \cdots\})$$

$\mathbf{x}_i$ is a series of the user's panning trajectory, T, expressed in vectors, segmented either by a temporal or spatial threshold. $\times$ is the outer product. r is dynamically calculated as panning continues. The pan is straight when r is zero. The pan moves further from being straight as r gets closer to one.

Reach-to pan and search-around pan: There are another two types of pans, i.e., reach-to and search-around pans. A reach-to pan is performed when users know the direction to the target object prior to panning, but not the distance. They therefore drag the map interface until the map-view area reveals the target object. A search-around pan, on the other hand, is performed when users do not know the direction or the distance to the target object. Figure 3 is an illustration of these two types of pans.

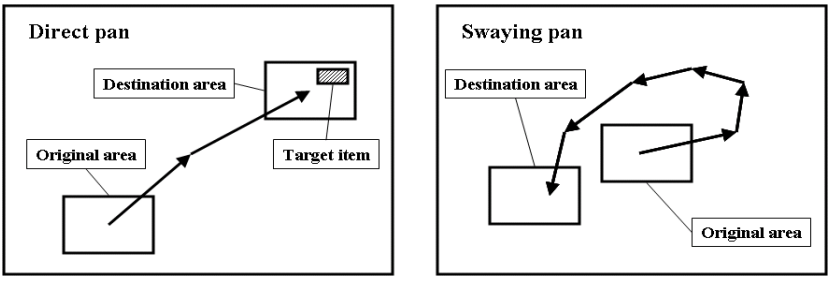


Fig. 3. Reach-to pan and search-around pan

We propose *directness* as a measure representing how direct the pan will be. Directness $d(T)$ of trajectory T is defined as

$$d(T) := \frac{|\sum_i \mathbf{x}_i|}{\sum_i |\mathbf{x}_i|} \qquad (T = \{\mathbf{x}_0, \mathbf{x}_1, \cdots\})$$

$\mathbf{x}_i$ is a series of a user's panning trajectory, T, expressed in vectors, segmented either by the temporal or spatial thresholds. d is dynamically calculated as panning continues. The pan goes straight to the target when d is equal to one. The user returns to the original point when d equals zero. Although there have been cases where users reach search targets and return, these are rare, and more plausible estimates are when they have searched around to find their item.

4 Application

This section describes an actual implementation of our model of inference of intentions based on the operation history.

4.1 Reducing Amount of Data Being Transmitted

Reducing the amount of data is important on a map interface on the Web or on a mobile device. While these map interfaces must access their data at great transmission cost,

users expect their map interface to present the map continuously, even while they are panning the map. To meet this, most map interfaces download data when the map-view area has moved a certain distance. However, there are cases where data downloading is unnecessary. Based on the distinction made in Section 3.2, we propose an advanced map interface that determines whether to update the map-view area or not. If panning is estimated to be route-oriented, the system frequently updates the presentation, since the user wants to see the route in detail. If panning is estimated to be goal-oriented, the system waits until the user ends panning.

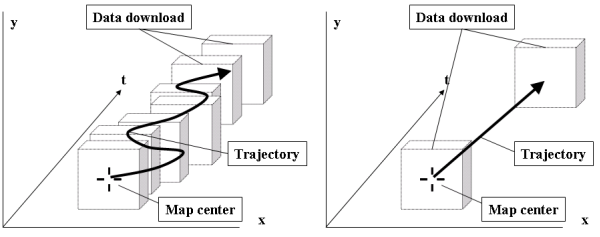


Fig. 4. Reduced downloads by distinguishing panning types

Figure 4 plots the number of occasions data downloads have been reduced by distinguishing between route-oriented and goal-oriented pans. This is performed in the following way.

Parameters: Coordinates of map center, time stamp
Record history: Trajectory T
Calculate measure: Relative-width $r(T)$
Distinguish intentions: Route-oriented pan/goal-oriented pan
Decision: Whether to download new data or not

4.2 Presenting Surrounding Elements

Landmarks are significant geographic objects that are used to determine directions and locations. They are commonly presented on the map-view area of a map interface.

In addition to landmarks within the currently presented geographic area, landmarks outside the area also help users to locate themselves and to determine the directions for panning or expand the geographic area being presented. It is necessary to distinguish whether the surrounding landmarks will satisfy users due to the limited space for presentation.

If they know which direction they want to move their map-view area, presenting surrounding landmarks is redundant. If their intention is to search around the area, on the other hand, presenting surrounding landmarks is often helpful.

Figure 5 is a snapshot of our implementation that presents surrounding landmarks around a map-view area. It shows distant landmarks in the direction they exists.

The system either shows or hides surrounding landmarks based on user intentions in panning. This is perfomed in the following way.

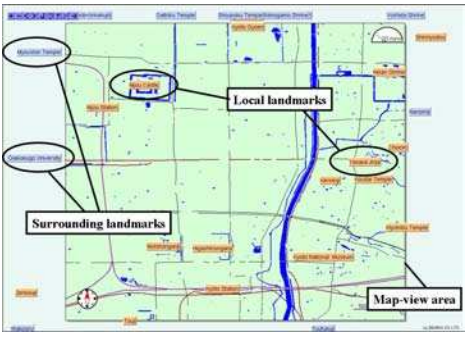


Fig. 5. Presentation of surrounding landmarks

Parameters: Coordinates of map center, time stamp.
Record history: Trajectory T.
Calculate measure: Directness $d(T)$.
Distinguish intentions: Search-around pan/reach-to pan.
Decision: Show or hide surrounding landmarks.

4.3 Evaluation

We did experiments to validate one of our proposed measures, relative-width, which was used to distinguish between route-oriented and goal-oriented pans. We used a map interface implemented on a Java applet for the experiment. The trajectory of the map center was recorded as log data.

We conducted experiments to distinguish between route-oriented and goal-oriented pans. The origin and destinations that we used in the experiments were as follows.

Origin: Kyoto Station.
Destinations: Arashiyama Station, Iwakura Station, Yamashina Station, Uji Station, Katsura Station, Ohyamazaki Station, Kinkakuji Temple, Ginkakuji Temple, Koryuji Temple, and Kyoto University, Chuushojima Station, Kurama Mountain, Matsuo Shrine, Jingoji Temple, Sanjo Bridge, Ryukoku University, Doshisha University, Shijo Bridge, Ritsumeikan University, Kyoto Industrial University.

Users did route-oriented and goal-oriented panning to the same destination, by manually dragging the map interface from the origin to the destination. They dragged the map-view area along the route for route-oriented pans, changing the direction at intersections if necessary. For goal-oriented pans, they dragged straight to the destination, until the map-view area revealed the target item at the center. Directness d and relative-width r defined in Section 3.2 were calculated for each of the trajectories.

Figure 6 and 7 plot the resulting measurements. X-axis indicates entries in decreasing order of relative-width and directness for route-oriented pans, respectively.

They indicate that relative-width is a better measure of distinguishing between route-oriented and goal-oriented pans than directness. The average value of directness for

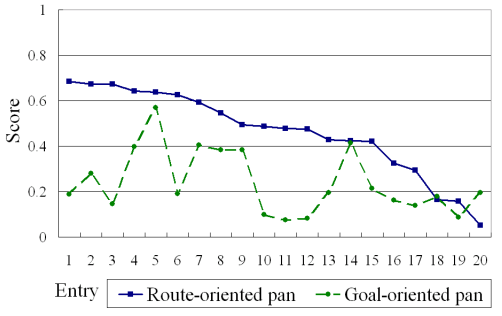


Fig. 6. Relative-width for route-oriented and goal-oriented pans

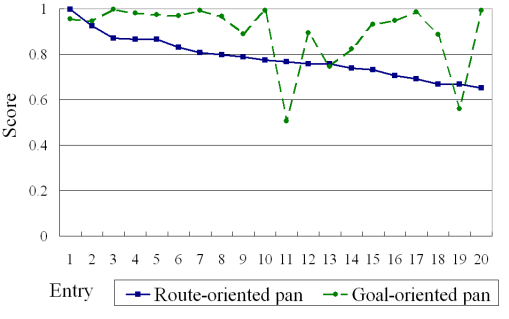


Fig. 7. Directness for route-oriented and goal-oriented pans

route-oriented pans was 0.783. The average was 0.897 for goal-oriented pans. In contrast, the average value of the relative-width for route-oriented pans was 0.464. The average was 0.239 for goal-oriented pans.

We also did experiments on distinguishing between search-around and reach-to pans. The search targets used for the experiments were as follows. Users knew neither the direction nor the distance to the target for search-around pans. They knew the direction to the target, but not the distance, for the reach-to pans.

Destinations: Kiyomizudera Temple, Sanjo Bridge, Kinkakuji Temple, Chuusho-jima Station, Shijo Bridge, Kyoto Industrial University, Seiryouji Temple, Jingoji Temple, Ryouanji Temple, Kouzanji Temple, Sanzenin Temple, Doshisha University, Matsuo Shrine, Ryukoku University, Nijo Station, Kurama Mountain, Togetsukyo Bridge, Ritsumeikan University, Kangetsukyo Bridge, and Kyoto City Hall.

Figure 8 and 9 plot the resulting measurements. X-axis indicates entries in decreasing order of relative-width and for search-around pans and directness for reach-to pans, respectively. They indicate that both the directness and relative-width are potentially useful for distinguishing between search-around from reach-to pans. The average value for directness for search-around pans was 0.495. The average was 0.895 for reach-to

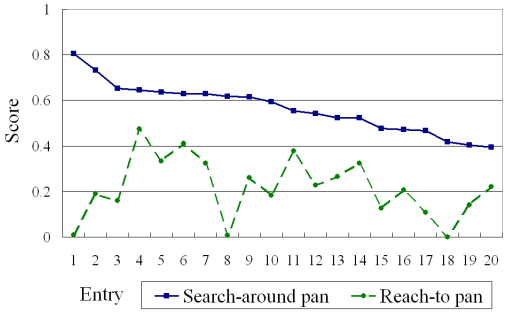


Fig. 8. Relative-width for search-around and reach-to pans

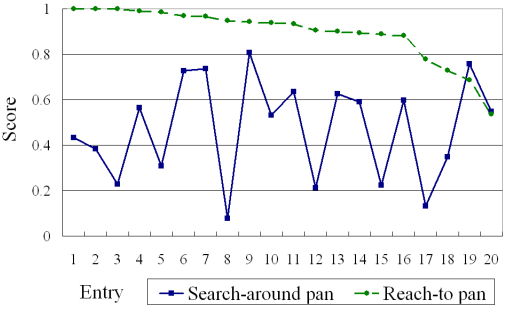


Fig. 9. Directness for search-around and reach-to pans

pans. The average value, on the other hand, for the relative-width for search-around pans was 0.544. The average was 0.218 for reach-to pans.

5 Conclusion

This paper discussed a model for adjusting the presentations of a map interface based on the inference of users' intentions expressed by their operation history. The system uses the trajectories of map movement to estimate their intentions. We conducted experiments to evaluate how effective the methods we propose were. The results revealed our method in an original situation was more effective than when it was not used.

We intend to employ learning mechanisms to distinguish between types of panning as well as the relationships between panning and zooming in future work. Zooming is a fundamental part of the dynamic map interface. Information on the speed of panning will also be considered.

Acknowledgments

This work was supported in part by MEXT Grant for "Development of Fundamental Software Technologies for Digital Archives", Software Technologies for Search and

Integration across Heterogeneous-Media Archives (Project Leader: Katsumi Tanaka), MEXT Grant-in-Aid for Scientific Research on Priority Areas: "Cyber Infrastructure for the Information-explosion Era", Planning Research: "Contents Fusion and Seamless Search for Information Explosion" (Project Leader: Katsumi Tanaka, A01-00-02, Grant#: 18049041), and MEXT Grant-in-Aid for Scientific Research on Priority Areas: "Cyber Infrastructure for the Information-explosion Era", Planning Research: "Design and Development of Advanced IT Research Platform for Information" (Project Leader: Jun Adachi, Y00-01, Grant#: 18049073).

References

1. J. Weakliam, M. Bertolotto and D. Wilson, Implicit Interaction Profiling for Recommending Spatial Content, in Proceedings of the 13th ACM International Symposium on Advances in Geographic Information Systems (ACM-GIS'05), pp. 285-294, Bremen, Germany, 2005.
2. R. Hiramoto and K. Sumiya, A Web Search Method using User Operation on Digital Maps, in Proceedings of the 7th International Conference on Mobile Data Management (MDM'06), pp. 106, Nara, Japan, 2006.
3. Z. Chen, F. Lin, H. Liu, Y. Liu, W. Y. Ma and L. Wenyin, User intention modeling in Web applications using data mining, World Wide Web: Internet and Web Information Systems, Vol. 5, pp. 181-191, Kluwer Academic Publishers, 2002.
4. M. Claypool, P. Le, M. Wased and D. Brown, Implicit interest indicators, Proceedings of 2001 International Conference on Intelligent User Interfaces, pp. 33-40, 2001.
5. J. Nielsen, Noncommand user interfaces, Communications of the ACM, Vol 36, No 4, pp. 83-99, 1993.
6. Y. Hijikata, Implicit user profiling for on demand relevance feedback, Proceedings of 2004 International Conference on Intelligent User Interfaces, pp. 198-205, 2004.
7. F. Mueller and A. Lockerd, Cheese: Tracking mouse movement activity on websites, a tool for user modeling, Proceedings of the 2001 Conference on Human Factors in Computing Systems, pp. 279-280, 2001.
8. J. Goecks and J. Shavlik, Learning user's interests by unobtrusively observing their normal behavior, Proceedings of the 5th International Conference on Intelligent User Interfaces (IUI'00), pp. 129-132, New Orleans, Louisiana, 2000.
9. J. Zhang, M. Zhu, D. Papadias, Y. Tao and D. L. Lee, Location-based spatial queries, Proceedings of the 2003 ACM SIGMOD International Conference on Management of Data, pp. 443-454, San Diego, California, 2003.
10. Y. Tao, D. Papadias and S. Qiongmao, Continuous nearest neighbor search, Proceedings of the 28th International Conference on Very Large Data Bases (VLDB2002), pp. 287-298, Hong Kong, China, 2002.
11. Y. Ishikawa, Y. Tsukamoto and H. Kitagawa, Implementation and evaluation of an adaptive neighborhood information retrieval system for mobile users, Proceedings of the Third International Workshop on Web and Wireless Geographical Information Systems (W2GIS2003), pp. 17-26, Rome, Italy, 2003.
12. Google Local, http://local.google.com/
13. Yahoo! Local Maps, http://maps.yahoo.com/

An Object Tracking Scheme Based on Local Density

Zhuan Qing Huang and Zhuhan Jiang

School of Computing and Mathematics, University of Western Sydney, NSW, 1797, Australia
zhuang@scm.uws.edu.au, zhuhan@scm.uws.edu.au

Abstract. We propose a method for tracking an object from a video sequence of moving background through the use of the proximate distribution densities of the local regions. The discriminating features of the object are extracted from a small neighborhood of the local region containing the tracked object. The object's location probability is estimated in a Bayesian framework with the prior being the approximated probabilities in the previous frame. The proposed method is both practical and general since a great many of video scenes are included in this category. For the case of less-potent features, however, additional information from such as the motion is further integrated to help improving the estimation of location probabilities of the object. The non-statistical location of an object is then derived through thresholding and shape adjustment, as well as being verified by the prior density of the object. The method is effective and robust to occlusion, illumination change, shape change and partial appearance change of the object.

Keywords: Object detection, object tracking, density estimation.

1 Introduction

Object tracking in video sequences is an important task in computer vision, which can be applied to a variety of fields such as surveillance, process control, medical treatment, machine intelligence and object copyright protection. The challenges are, firstly, the appearance of object being continuously changing with the change of pose, location, illumination, occlusion, as well as the internal change of the object, and the object model based on the color or derived features may vary significantly throughout the video sequences. Secondly the computational load is high and therefore critical for real-time applications. Fortunately, by the nature of the tracking process, the objects in many such applications would need neither recognize the object precisely nor identify the object shape accurately. Capturing the whereabouts of the object in the subsequence frames under different conditions is hence the main focus.

Though the appearance of an object may vary quite largely within a video sequence, yet we know it bears only small changes in any two successive frames in most situations. Adaptive methods such as adaptive deformable template approach in the literature were adopted to improve the tracking accuracy by updating the current information for next frame. Features used for capturing an object are normally the

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 166 – 175, 2007.
© Springer-Verlag Berlin Heidelberg 2007

color, edge or shape, and those mathematically derived from them. In terms of color features, the algorithms include template matching, histogram matching and points matching [1,3,14-16]. In term of shape features, there are active contour approaches, edge matching and deformable shape approaches [2,8-12]. Another important piece of information in video sequences is the temporal features. Many algorithms now integrate the motion information to track an object [2,13]. However, purely motion based approaches may exclude the situation where the object becomes static during certain time interval. Paragios *el al* [2] proposed a method by modeling the inter-frame difference data as a mixture density from motion and static component, then integrated the motion information into geodesic active contour functions. The Condensation algorithm in [6] utilized "factored sampling" and learned dynamical models, together with the visual observations, to propagate the probability distribution through the frames. Color approaches from early template match or histogram match to later probability estimation have also been investigated. Comaniciu *et al* [1] regularized the feature histogram-based target representation by spatially masking with an isotropic kernel. The target localization problem becomes finding the local maxima of the similarity function.

To avoid the intensive location searching or exhaustive pixels matching, we propose in this work a method that tracks the object in a video sequence based on the characteristics of the local region density. By approximating the local object and background densities, the object probability (refering to the probability of a pixel belonging to the object) is then obtained within a Bayesian framework. The characteristics of the density features lead to a slight formula adjustment in the object density computation. This paper is organized as follows: Sections 2 introduces the feature selection to be used in the object density estimation later on. Section 3 describes the tracking strategies based on the density feature. Implementation and experimental results are shown in Section 4, and finally Section 5 is the conclusion.

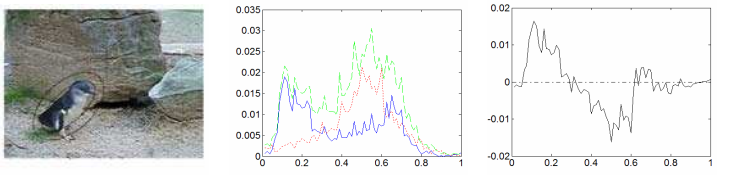
2 Feature Selection

Object features include color, shape, gradient and other derived properties. Some of the features may be more suitable than the others for the tracking purpose depending on the particular individual circumstances. In what follows, we will investigate how to select proper object features to achieve a better tracking performance.

2.1 Discriminating Feature

Significant features such as strong edges, smoothing uniform color or regular texture are known to be good in general for capturing an object. On the other hand, the contrast of the features between the object and the background are also good for tracking an object. It is naturally anticipated that for the effective tracking an object feature that distinguishes itself more from the background is a better choice than those more similar to the background. The object features that are similar to the background would decrease the accuracy of capturing an object, and could even lead to a tracking

failure, whether or not the features are significant on their own. In a dull environment, a bright color of the object is a good choice for tracking, while in a bright environment the dark color of the object is more suitable. In this connection, a segmentation method [5] was proposed to extract bright targets filtered out by wavelet techniques, while a feature selection mechanism [4] was presented to evaluate features used to improve the tracking performance. In our work, we propose to select features for the tracking based on the contrast of probability density between the object and its local background from different feature spaces. The main strategy is to make use of an optimal segment of density distribution from different sources. These density segments sought for such purposes can be used in the later steps, and the density of object in the current frame is calculated with the approximate density from previous frame.



(a) (b) (c)

Fig. 1. (a) Object and local background. (b) Object, local background and mixture density. (c) Difference of densities between object and local background.

In order to examine the local density, we first use a simple shape such as an ellipse to approximate the object, as show in fig. 1(a), (or another shape based on the characteristics of the object so that it better covers the object), while the local background is estimated on an annular area with a larger ellipse. In fig. 1(b), the solid (blue) line is the object density $p_A(x)$ obtained from the smaller ellipse, the dotted (red) line is the local background density $p_B(x)$ obtained from the annular area, the dashed (green) line represents the mixture density $p_M(x)$. The mixture density is the weighted sum of the object density and the local background density via

$$p_M(x) = P_A p_A(x) + P_B p_B(x), \tag{1}$$

where x is the pixel intensity, and P_A and P_B are the weight values with $P_A + P_B=1$. For two densities of the shape in figure 1(b), where object density and local background density are well separated with some overlapping, we propose to use an approximate density for the subsequent frame to obtain the object density, which will be described in detail in the next section. We here describe only the initial processing that determines which density segment is to be selected and how to tune the densities so that the low density in one feature can be complemented by the high density in another feature.

Let use define the difference of the two densities by $f(x)=p_A(x)-p_B(x)$. Then the density segment may be suitable for object tracking if $f(x)$ is sufficiently large there. If both $p_A(x)$ and $p_B(x)$ are very small on a segment, then the segment is not suitable for

detecting the object presence because it does not have enough discriminating power and the result would be unpredictable. We can measure the suitability by the relative value $f(x)/p_A(x)$, where a larger value would correspond to a better performance for the tracking. If the density is approximated by a Gaussian distribution, we can also measure the performance by the model parameters. To reduce the approximation complexity, we can simply use the area centre of the positive difference $f(x)$ as the landmarks. The higher the centre is located, the larger the difference between object and local background.

2.2 Optimal Feature Density

The object densities from different feature spaces may have different characteristics. In most cases, they would not be all the same; otherwise the object and its local background are almost the same. Out goal is to find the ideal candidate density which has maximum difference between object and its local background. In other words, we choose a feature so as to best distinguish the object from its local background. A density may be derived from the RGB space, HSV space or from other properties such as the smoothness. The total difference of object density from local background in a positive portion can be calculated by $\Omega_+ = \int_{f(x)>0} f(x)dx$. We can compare Ω_+ as well as the area centre of the positive $f(x)$ among several different feature spaces, and choose the feature that has larger Ω_+ and higher centre point. For instance, the densities of object and background may overlap too much in RGB space, but may be able to separate in saturation within HSV space, as shown in fig. 3 (b) and (c), or the densities in hue space can separate better than the intensity as shown in fig. 2 (a), (c).

2.3 Complementally Feature Densities

As shown in fig. 1(c), the leftmost part of the object density from intensity is well separated from the local background density, yet the other part on the right will result in weaker object detection or missing area. For better representing the object, we can combine the optimal parts of different densities from different feature spaces, which may complement each other for a fuller object representation.

Though all densities exhibiting larger difference can be used, more densities used may increase the accuracy; it also increases the computation unnecessarily if less is enough for the tracking purpose. One reason for looking into other feature densities is that the selected object density could have a (not small) part overlapping with the background density, and the difference between them is not large, which means there would be some area missing in the object representation. Therefore the size of the density overlap and the extent of the difference can be used to make a decision on whether to utilize an additional feature property. In fig. 1 and fig. 2, for instance, we can observe there is a portion of object density on the right that is overlapped in fig. 1(b), and the difference of this part is small in fig. 1(c). We also observe that the object has another property which is the smoothness, as shown in fig. 2(g). This indicates that we can combine these two set of densities to better represent the object.

3 Proximate Density Approaches

The object tracking is conducted according to the features of the object. For the object with feature density exhibiting discrimination, we propose to use local densities of the object and its surrounding background for its tracking. The method uses the proximate density based on Bayesian rule. For non-discriminating densities, other information such as the motion may be needed to improve the estimation.

3.1 Discriminative Density

Bayesian inference is a method in which latest evidence or observation is utilized to update or to newly infer the probability of a hypothesis being true. Given a complete set of n+1 mutually exclusive hypotheses H_i, the estimated prior probability $p(H_o)$ of a hypothesis H_o can be improved into the posterior probability $p(H_o|D)$, if a set of additionally observed data D is to be used to further improve the probability estimation. More precisely, the posterior probability $p(H_o|D)$ of the hypothesis H_o can be calculated through the use of the prior probability of the hypothesis and the probabilities of the observed data under the different hypotheses:

$$p(H_o \mid D) = \frac{p(D \mid H_o)\, p(H_o)}{\sum_{i=0}^{n} p(D \mid H_i)\, p(H_i)}, \tag{2}$$

where $p(D|H_i)$ is the likelihood of the hypothesis H_i under the observed data D, $p(H_i)$ is the prior probability of the hypothesis H_i. The denominator is essentially a normalization factor. According to the Bayesian rule, the object probability can be expressed as

$$p^t(A \mid x) = \frac{p^t(x \mid A)\, p^t(A)}{p^t(x \mid A)\, p^t(A) + p^t(x \mid B)\, p^t(B)}, \tag{3}$$

where A and B denote that the current pixel of value x belongs to the object and the background respectively, $p(x|A)$ is the likelihood of the current pixel of value x belonging to the object, $p(A)$ and $p(B)$ are the prior probabilities, the probabilities estimated prior to inspecting the actual pixel value, of the pixel belonging to the object and the background respectively. We in general don't know exactly the prior probability of a pixel being on the object or the background; we assume they are constant. We also don't know the exactly object and background density at current time t, but we know the densities, especially the object density, would be very close to the densities in the previous frame at time t-1, which we already knew. So we can use the $p^{t-1}(x|A)$ and $p^{t-1}(x|B)$ in (3) to calculate the approximate object density over the current frame. In fact additional simplification on the formula can be made for the actual calculation. The advantage of this method over the direct density matching is that it requires less iterative searching. In fact, the local object density will in general fall into this catalogue when the local region excludes most background by its border of an ellipse or other shapes.

For the use of multiple feature densities, the object density from the combined features is

$$p(A \mid x) = \sum_{i=1}^{n} \lambda_i p_i(A \mid x), \quad \sum_{i=1}^{n} \lambda_i = 1, \tag{4}$$

where n is the number of feature densities, and $\lambda_i \geq 0$ is the weight factor. This would result in a more complete coverage of the object. For a rough tracking, one distinguishing feature is enough. Nevertheless the multiple densities approach would still increase the robustness of tracking with less impact on the sudden appearance change of the object.

3.2 Non-discriminative Density

When the object and background densities are not well separated from each other, but the difference of main part of density is large, then the method mentioned above is still applicable. If however these densities are flat and/or quite close to each other, in other words, the object is very similar to its local background regarding to the selected feature, the above method alone will not be sufficient to extract the object. One piece of information not being used in the above density approach is the pixel space relationship. Yet if we establish the space relationship based on the object location, it often leads to the iterative location searching. However there is another piece of information that can be made use of, the motion information. For the static background, we can estimate the motion by the difference data from successive frames. We model similar to [2] the difference frame data $D'(s, t)=x(s, t)-x(s, t\text{-}1)$ (s is pixel location) as a mixture of the background density $p_b(d)$ and the motion density $p_m(d)$ similar to (1), and then determine the model parameters by maximizing the joint density using maximum likelihood method. Now we integrate the motion density into the calculation of object density as following:

$$p^t(A \mid x) \approx \frac{p^{t-1}(x \mid A)p(A)p_m(d(x))}{p^{t-1}(x \mid A)p(A)p_m(d(x)) + p^{t-1}(x \mid B)p(B)p_b(d(x))}. \tag{5}$$

The object probability obtained this way would be more distinguishable from the local background than the method without the use of the motion information. For the moving background sequences, the motion information may be obtained by other methods.

3.3 Object Shape Refinement

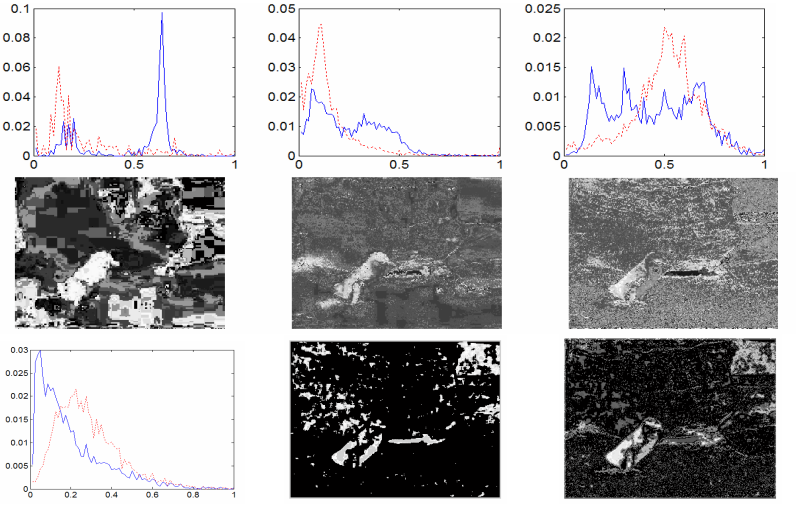
The region an object occupies is obtainable by thresholding the location probability in the local area while ensuring the coherence of its neighbor with such as region growing, and region consolidation method [6]. The probabilities may not always represent the full object shape, as part of object area very close to the background may not be well represented by the probability distribution. Such a problem can be largely alleviated by projecting the object ellipse in the previous frame onto the

current region. Next we compare the object density with the one in previous frame, if the error is smaller than a permitted threshold, the local region is defined. The background density can't play this role since the background may have large variation if motion is large. The ellipse needs to adapt along with the object moving throughout the sequence. The criterion to fit an ellipse onto an object is to ensure that the ellipse differs with the object region as less as possible.

4 Implementation and Experiments

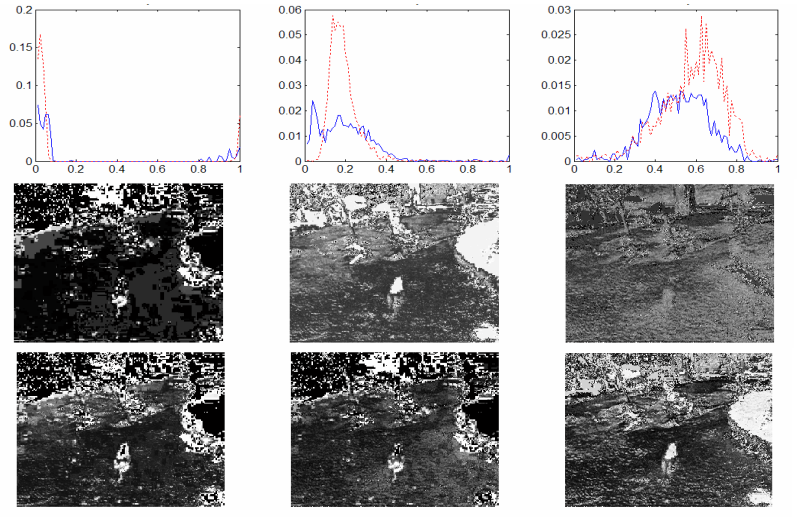
The following experiments are developed to illustrate the effect on tracking an object, due to the density selection, multiple densities, as well as the non-discriminative density. One video sequence is showed in fig. 1(a) and another is showed in fig. 4(a). These results will also be properly explained and discussed.

First we inspect the effect of the density shape difference. For this purpose we examine a penguin sequence also used in fig. 5. We observe that the hue densities in fig. 2(a) are better separated than the saturation and intensity densities depicted in fig. 2(b) and fig. 2(c) respectively, and the resulting object probability is better distinguished from its local background as showed in fig. 2 (d), (e) and (f). Thus the hue is the better feature to use for the local proximate density method to capture the object for this sequence.



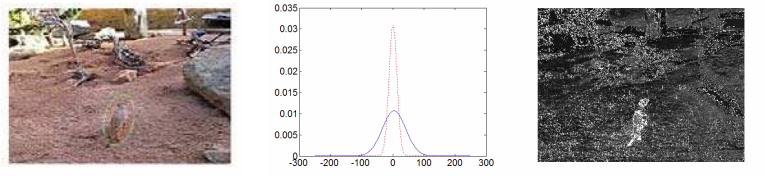
a	b	c
d	e	f
g	h	i

Fig. 2. (a) Hue density. (b) Saturation density. (c) Intensity density. (d) Object probability by hue. (e) Object probability by saturation. (f) Object probability by intensity. (g) Smoothness. (h) Object probability by smoothness. (i) Object probability by intensity and smoothness.



a	b	c
d	e	f
g	h	i

Fig. 3. (a) Hue density. (b) Saturation density. (c) Intensity density. (d) Object probability by hue. (e) Object probability by Saturation. (f) Object probability by Intensity. (g) Object probability by hue and saturation (h) Object probability by hue and intensity. (i) Object probability by saturation and intensity.



(a) (b) (c)

Fig. 4. Non-discriminating density tracking

Additional experiments for combining multiple features are conducted for the second sequence (see fig. 4(a)) as shown in fig. 3. From densities in the fig. 3(a) to (c), we observe that the saturation is a better feature to separate object from its local background than the other two, and the corresponding object probabilities in fig. 3(d) to (f) demonstrate this. The multiple densities approach is shown in fig. 3(g), (h) and (i), where we combined the hue and saturation together via (4), it has improved the results as shown in fig. 3(g). The result manifested in the observation of the density distributions. We note that other features or properties can also play a similar role. For example, in fig.2, the density distribution in fig. 2(g) shows the object is pretty uniform in its color, and when we combine it with the intensity density, it yields a better result as in fig. 2(i) than the intensity itself in fig. 2(f). This shows that when we take into consideration the smoothness (standard variation) of color, the object probability has been enhanced as a result.

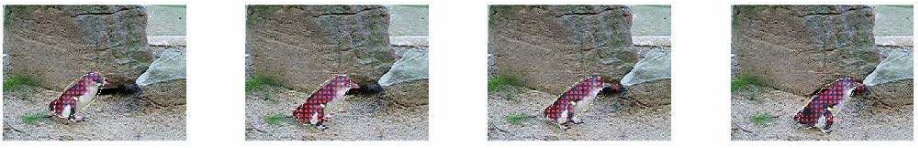


Fig. 5. Tracking penguin



Fig. 6. Tracking a vehicle

We now move to experiment with non-discriminative densities such as in fig. 3(c), where the object and its local background densities do not separate well in terms of the distribution. For this purpose we examine a video sequence starting from image in fig. 4(a). We model the motion density and background density from difference frame data by the Gaussians, and use the Expectation-Maximization algorithm to calculate the model parameters, with the resulting distributions depicted in fig. 4(b). Then use the motion information and apply (5) to calculate the object probability. The result is shown in fig. 4(c). If we compare it with fig. 3(f) that uses only the intensity probability, we see that the additional motion information greatly enhanced the performance for the non-discriminating intensity density.

We now apply our scheme to a complete penguin sequence to see the effectiveness of the penguin tracking. Fig. 5 depicts the tracking through the video sequence, where the red dot on the penguin indicates the tracking results. We note that on black and white images, this reddish shade is not easy to observe. It instead looks like a dot grid on the penguin object. Finally illustrate in fig. 6 that the proposal method is applied to video sequences of a moving background sequence. In the video sequence in fig. 6, the tracked object is the vehicle and the region obtained through the tracking is depicted in green dots, which also fall on the red vehicle as expected.

5 Conclusion

We proposed a fast object tracking method for video sequences based on the local feature densities. The local features are properly selected and the current local densities are typically approximated by the densities in the previous frame. The object's location probability is calculated within a Bayesian framework, and the motion information is used to compensate for the case of non-discriminating features. The method is efficient and robust for most tracking scenarios, and the experiments also demonstrated the efficiency of the method in that it does not involve complicated modeling and heavy computation as many iterative searching algorithms would.

References

1. Comaniciu, D. Ramesh, V. and Meer, P.: Kernel-Based Object Tracking, IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 25, No. 5, (2003) 564-577.
2. Paragios, N. and Deriche, R.: Geodesic Active Contours and Level Sets for the Detection and Tracking of Moving Objects, IEEE Trans. Pattern Analysis and Machine Intelligence, Vol.22, No.3, (2000) 266-280.
3. Sheikh , Y. and Shah, M.: Bayesian Modeling of Dynamic Scenes for Object Detection, IEEE Trans. Pattern Analysis and Machine Intelligence, Vol.27, No. 11, (2005) 1778-1792.
4. Collins, R.T., Liu, Y. and Leordeanu, M.: Online Selection of Discriminative Tracking Features, IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 27, No. 10, (2005) 1631-1642.
5. Zhang, X.P. and Desai, M.D.: Segmentation of Bright Targets Using Wavelets and Adaptive Thresholding. IEEE Trans. on Image Processing, Vol. 10, No. 7,(2001) 1020-1030.
6. Isard, M. and Blake, A.: CONDENSATION-Conditional density propagation for visual tracking, Int. J. Comput. Vis., Vol. 29(1), (1998) 5-28.
7. Mansouri, A.R.: Region tracking via level Set PDEs with Motion Computation, IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 24, No.7, (2002) 947-967.
8. Yusuf, A.S. and Kambhamettu, C.: A Coarse-to-Fine Deformable Contour Optimization Framework, IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 25(2), (2003) 174-186.
9. DeCarlo, D. and Metaxas, D.: Adjusting Shape Parameters Using Model-Based Optical Flow Residuals, IEEE Trans. Pattern Analysis and Machine Intelligence. Vol. 24(6), (2002) 814-823.
10. Shen, D. and Davatzikos, C.: An Adaptive-Focus Deformable Model Using Statistical and Geometric Information, IEEE Trans. Pattern Analysis and Machine Intelligent, Vol.22(8), (2000) 906-913
11. Mukherjee, D.P., Ray, N. and Acton, S.T.: Level Set Analysis for Leukocyte Detection and Tracking, IEEE Trans. On Image Processing, Vol.13(4), (2004) 562-572.
12. Huang, Z.Q. and Jiang, Z.: Tracking Camouflaged Objects with Weighted Region Consolidation, Proceedings of Digital Image Computing: Techniques and Application, (2005) 161-168.
13. Jepson, A. D., Fleet, D. J. and El-Maraghi, T.F.: Robust Online Appearance Modes for Visual Tracking, IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 25, No. 10, (2003) 1296-1311.
14. Elgammal, A. Duraiswami, R. Davis, L.S.: Efficient Kernel Density Estimation Using the Fast Gauss Transform with Applications to Color Modeling and Tracking, IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 25, No. 11, (2003) 1499-1504.
15. Comaniciu, D. and Meer, P.: Mean Shift: A Robust Approach Toward Feature Space Analysis, IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 24, No. 5, (2002) 603-619.
16. Perez, P. Hue, C. Vermaak, J. and Gangnet, M.: Color-Based Probabilistic Tracking, Proc. European Conf. Computer Vision, Vol. I, (2002) 661-675.

Modeling Omni-Directional Video

Shumian He and Katsumi Tanaka

Department of Social Informatics, Graduate School of Informatics, Kyoto University
Yoshida Hommachi, Sakyo-ku, Kyoto 606-8501, Japan
{shumian,tanaka}@dl.kuis.kyoto-u.ac.jp

Abstract. Omni-Directional Video (ODV), which is recorded using an omni-directional camera, has become widely used because of recent advancements in digital video technologies and photographic equipment. In ODV, as multiple subjects are recorded by the camera, keyword annotation is practically impossible. Furthermore, the basic concepts of indexing and retrieval of traditional video databases, such as frame and shot, are not applicable to ODV data. Therefore, the properties of ODV must be discovered and basic concepts must be defined to find out effective ways of indexing and retrieving such video data. We developed a new data model that provides operations for the composition, searching, navigation, and playback of ODV data based on video algebra and spatial properties.

1 Introduction

The increased capacity of data storage devices has allowed large amounts of data to be collected easily. In the field of video databases, even storing all data and selecting specific data for specific applications has become possible. The indexing and retrieval of digital video is an important research area for studies on video databases.

Video data consist of sequences of shots. A single camera shot, usually considered as the basic unit of video to be represented or indexed, consists of one or more frames generated and recorded continuously, representing a continuous series of actions [11]. Temporal segmentation is the problem of detecting boundaries between consecutive camera shots. Over the past several years, substantial progress has been made in detecting shots based on changes of their visual characteristics, as well as in indexing those shots by keywords extracted from text recognition or subtitles. Hence, the main focus of research on conventional video databases is retrieval of shots and frames from stored video data.

Technological advances in omni-directional cameras have inspired many researchers to rethink the way images are captured and analyzed. Omni-Directional Video (ODV) is recorded using an Omni-Directional Camera (ODC), which has a 360º field of vision. A standard TV camera, which has a 30-60º field of vision, can capture only part of scene. As an ODC has a wider field of vision, it can record all objects and events occurring around it. Therefore, data retrieval from the ODV is different from that of conventional video. Because ODV records a 360º scene around the camera, users of the video footage must decide which objects should be noticed and what frames should include those objects, and then create a sequence of these

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 176 – 187, 2007.
© Springer-Verlag Berlin Heidelberg 2007

frames as a shot. Hence, while conventional video retrieval involves detecting segments of shots and selecting needed frames, ODV retrieval involves *creating* frames from video data and *connecting* them to *generate* shots.

We developed a model of ODV that consists of some basic operations for generating basic units, such as frames and shots of ODV. With this model, we unwrap frame sequence of ODV into panoramic images, and then create frames including indicated salient objects. Based on this model, we introduce directional relationships between moving objects and frames, by which a shot that tracks an object can be generated.

The next section of this paper reviews related work on models for video data and video data retrieval. In section 3, we define some conceptual units of ODV. In section 4, we propose several algebraic operations to generate frames. In section 5, we expand the operations to generate shot from sequences of images. In section 6, we describe some applications with ODV in our model. In section 7, we present findings from our studies, and conclude the paper.

2 Related Work

Over the past few years, researchers have developed systems capable of providing database support for video data. Object Video Database (OVID) [4] is an instance-based video database system. In OVID, an arbitrary set of contiguous intervals can be defined as a meaningful entity called a *video object*. Inheritable and non-inheritable attributes can be assigned to such video objects. Inheritable attributes are shared between objects on the basis of an interval-inclusion relationship between them. The basic idea is to let video objects share inheritable attributes with their parent video objects and make the manual annotation process easier.

The *algebraic video data model* [9, 7] is based on the *stratification approach* [6]. Unlike simple stratification, however, in the algebraic video model, the hierarchical relationship between the descriptions that are associated with the same video data is defined. Several algebraic operations, such as union, intersection, and concatenation, are used to define video intervals with descriptions. Parent nodes in the hierarchy represent the context of their child nodes. Using the hierarchy, multiple views with different contexts can be attached to the same video data. Their work is also based on an annotation model rather than a query model. Annotation should be done very carefully to ensure that answers are available for any kind of query. It also puts an extra burden on annotators to define the relationships between descriptions.

All of the above works do not focus on the characteristics of ODV data. In ODV, a needed portion, which is considered as a frame, is generated from a part of a single image in the video. Thus, besides the temporal elements, spatial relationships must be considered. A new model to represent ODV is needed.

3 Conceptual Units of ODV

3.1 Basic Model of Doughnut-Type Video

ODV is recorded by an ODC. An example image captured by an ODC is shown in Fig. 1(a). As the shape of the image is round and each subject is arranged around the



(a) (b) (c)

Fig. 1. A donut-type image (a) captured by ODC and two un-warped images (b) and (c) from image (a) with different baselines

center of the image, we call this type of image a "*donut-type image*". In the concept of ODV, this image is also called a "*donut-type frame*".

The most important characteristic of ODV data is called *continuous data*. Data from ODV is continuous-media data. Consequently, any portion of video data can become a unit for indexing and retrieval.

An *ODV sequence* is a continuous video scene taken by a specific ODC. It consists of a sequence of donut-type frames.

Definition 3.1 (Video object of ODV sequence)
An ODV sequence O is defined as $O = o_1 o_2 \cdots o_n$. Here, each o_i is a donut-type frame, and i is the number of the frame. The timecode function (denoted by $timecode(o)$) is defined for each frame o_i in a video sequence. For example, $timecode(o) = "10:02:03'00"$ indicates that frame o was taken at time $"10:02:03'00"$. o_i and o_{i+1} are adjoining images, and $timecode(o_{i+1}) > timecode(o_i)$.

3.2 Basic Model of Panoramic Video

While a captured donut-type image is warped, it can be un-warped by computer vision techniques [2]. In Fig. 1, two un-warped images from the original donut-type image (a) are (b) and (c). We call the un-warped images "panoramic images".

Definition 3.2 (Panoramic image)
Let o be a donut-type image. The un-warped panoramic image from o is denoted by $p = uw(o)$. Because p is rectangular, it can be represented by two points as $p = ((0,0),(2\pi, L))$, where L is the length of the radius of o , and the radian angle is 2π (Fig. 2).

As shown in Fig. 1, the position of a person in (b) is different from their position in (c) because the baselines of (b) and (c) are different. Hence, the *baseline* is an important element for un-warping an ODV. By indicating the difference of the baselines in two panoramic images un-warped from a donut-type image, a panoramic image can be defined as the result of changing the position of the baseline.

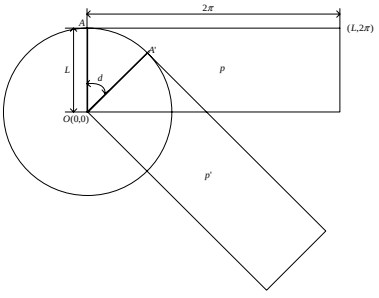


Fig. 2. Panoramic image p un-warped from donut-type image o and another image p' with different baseline from p

For example, as shown in Fig. 2, p and p' are two panoramic images un-warped from a donut-type image o with baselines OA and OA' respectively, and $\angle AOA' = d$. The relationship between p and p' is defined below.

Definition 3.3 (Change of baseline)
Let p be a panoramic image un-warped from donut-type image o. As a result of changing the baseline of p in an angle d, a newly generated panoramic image p' is denoted by $p' = T(p, d)$, and the original p can be denoted by $p = T(p', -d)$. As a special case, $p = T(p, \pm 2n\pi)$.

By un-wrapping each image in an ODV sequence O, we get a new panoramic video sequence P, as defined below.

Definition 3.4 (Video object of panoramic video sequence)
The panoramic video sequence P of an ODV sequence O is denoted as $P(O) = p_1 p_2 \cdots p_n$. Here, each p_i is a panoramic frame un-warped from donut-type frame o_i.

Because video sequence P consists of rectangular panoramic frames, each point in a panoramic frame can be denoted by two-dimensional coordinates (x, y). In the rest of this paper, we use P to represent basic video data instead of the original sequence of donut-type images.

3.3 Description of Salient Object

A salient object is an interesting physical object in a video frame [3]. Each frame usually contains many salient objects. Some of them are moving objects, such as walking people or moving vehicles and others are stationary objects, such as desks or whiteboards in meeting rooms or houses on streets. J. Fan et al. summarized several approaches for modeling and detecting salient objects corresponding to the visually distinguishable image components [1]. These approaches are also adaptable for our ODV model. In many spatial applications, Minimum Bounding Rectangles (MBRs) have been used to approximate objects because they need only two points to represent objects. In our model, we used MBRs for representing salient objects.

Definition 3.5 (MBR of Salient object)
Let salient object A be composed of n points, and each point be denoted by (x_i, y_i), where $1 \le i \le n$. The MBR M of A is defined as $M(A) = ((\min(X), \min(Y)), (\max(X), \max(Y)))$, where $X = \{x_i \mid x_i \in X, 1 \le i \le n\}$, and $Y = \{y_i \mid y_i \in Y, 1 \le i \le n\}$. Here, $(\min(X), \min(Y))$ and $(\max(X), \max(Y))$ are the coordinates of the lower left and upper right corners of the MBR, respectively.

4 Algebraic Operations for Single Frame

The first step of ODV retrieval is creating frames from video data. Because of the wide vision field of ODV, a panoramic image can includes multiple subjects. As shown in Fig. 3, a panoramic image is un-wrapped from a donut-type image. There are many people in the panoramic image, and each single person can be considered as a salient object and can be represented with an MBR. Fig. 3 shows three MBRs with green, red and yellow verges severally.



Fig. 3. Rectangular areas selected from a panoramic image

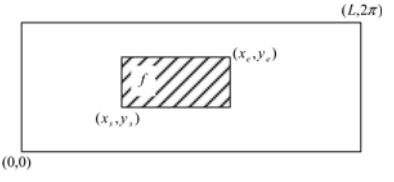


Fig. 4. Projection f in panoramic image p

While in conventional video, each frame is a rectangular area; in ODV, each panoramic image may include many MBRs of salient objects. In this section, we introduce some basic concepts of frame in ODV and propose some definitions and algebraic operations for a single frame of ODV.

4.1 Projection

A panoramic image that contains several meaningful objects can be considered as a scene around a camera. When the camera shoots an object, the object and a rectangular portion of its background are taken in a picture. Thus, the picture is a projection of a portion of the scene. We developed an operation for projection to select a rectangular area from a panoramic image. As shown in Fig. 4, the projection of the MBR of an indicated object is defined below.

Definition 4.1 (Projection of portion in panoramic image)
Let p be a panoramic image, and $((x_s, y_s), (x_e, y_e))$ be an MBR of object A in p. The projection of the MBR is denoted by $f = \Pi_A(p)$. More formally, it is defined as $f = (p, (x_s, y_s), (x_e, y_e))$.

We considered the projection of an area in a panoramic image as a single *frame* of a video. For a special case, when an entire panoramic image is a frame, the frame is denoted by $f = (p,(0,0),(2\pi, L))$, where $p = ((0,0),(2\pi, L))$, as defined by Definition 3.2.

4.2 Composition

Assume there are two objects in a panoramic image. There are two ways to generate a frame that contains both of them. One is to get a projection that contains the MBRs of each object and the area between them; the other is to generate a new frame in which the objects attach to each other. We call the first operation *connection*, and the second *attachment*.

Definition 4.2 (Connection of frames)
For a panoramic image p and two frames f and g that project on p, a *connection* of f and g is defined as a new frame of MBR that includes both f and g and the area between them. That is, the connection of f and g in p is defined as $con(f,g) = (p,(x_{fs}, \min(y_{fs}, y_{gs})),(\max(x_{fe}, x_{ge}), \max(y_{fe}, y_{ge})))$.

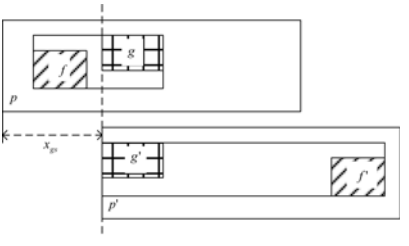


Fig. 5. Connection of two frames

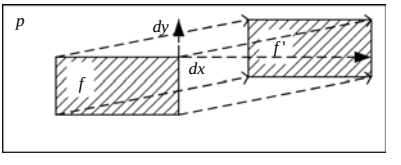


Fig. 6. Frame f in panoramic image p and result f' of moving f by vector (dx, dy)

Notice that in p, f is on the left of g $(x_{fs} \le x_{gs})$. The operation $con(f,g)$ is read as "connect f to g in p". In the case of "connect g to f", because g is on the right of f, the baseline of p must be changed to make g be on the left of f in a new panoramic image p' denoted by $p' = T(p, x_{gs})$. In p', projections of f and g are represented by $f' = (p',(x_{fs} - x_{gs} + 2\pi, y_{fs}),(x_e - x_{gs} + 2\pi, y_{fe}))$ and $g' = (p',(0, y_{fs}),(x_{ge} - x_{gs}, y_{fe}))$, respectively. This guarantees that in p', no objects exist on the left of g', making the operation $con(g', f')$ possible (Fig. 5).

The operation of attachment makes a frame change its position to attach to another one. Hence, before describing the attachment operation, we defined to the *movement* operation of a frame, as shown in Fig. 6.

Definition 4.3 (Movement of frame)
Let f be a frame in panoramic image p, and $\vec{v} = (dx, dy)$ be the movement vector. After f moves, the position of f' is denoted by $f' = mv(f, \vec{v})$ and is defined as.

As shown in Fig. 7, the attachment operation is defined below.

Definition 4.4 (Attachment of two frames)
For a panoramic image p and two disjointed frames f and g in p, an *attachment* of f and g is defined as a new frame of MBR that includes both f and g, and that g moves to the right of f so that no space exists between them. That is, the attachment of f and g in p is defined as

$$attach(f,g) = con(f,g') = (p,(x_{fs},\min(y_{fs},y_{g's})),(\max(x_{fe},x_{g'e}),\max(y_{fe},y_{g'e})))$$

where $g' = mv(g,(x_{fe}-x_{gs},0))$.

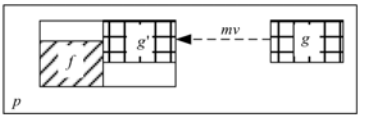


Fig. 7. Attachment of two frames

5 Algebraic Operations for Sequence of Frames

We have introduced some definitions and operations for a single frame of ODV in section 4. In this section, we propose some operations for sequence of frames.

5.1 Projection

With projections in single panoramic images, the MBR of an indicated object or area can be considered as a frame, the basic unit in video data. We expanded the operation to deal with a sequence and defined a *shot* of ODV data.

When using a normal video camera to shoot an object, two types of shots can be generated, those without zooming or panning and those track moving objects. In the ODV, these types of shots can be generated by creating frames that include indicated areas and objects. Frames of a stable area A and a moving object B in a sequence of panoramic images are shown in Fig. 8.

Definition 5.1 (Projection of fixed area in video sequence)
Let $M(A) = ((x_s,y_s),(x_e,y_e))$ be the MBR of a stable area *A*, and $P = p_1 p_2 \cdots p_n$ be a sequence of panoramic images. The sequence of frames that indicate the MBR is represented by $F = \Pi_{(M(A),FIX)}(P)$, where the parameter *FIX* refers to each frame in the sequence showing the same area. More formally, it is defined as $F = \{f_i \mid f_i = (p_i,(x_s,y_s),(x_e,y_e)),1 \le i \le n)\}$.

We named this type of sequence the *fixed shot* because indicating the position and shape of the MBRs in each frame is not necessary. Once the MBR of an area is given, a shot aimed at the area is generated by the operation. In Fig. 8, the frames that contain area *A* generate a fixed shot.

In contrast to the fixed shot, we named a shot that tracks a moving object a "*float shot*". As shown in Fig. 8, a float shot that tracks object *B* consists of frames that contain *B*.

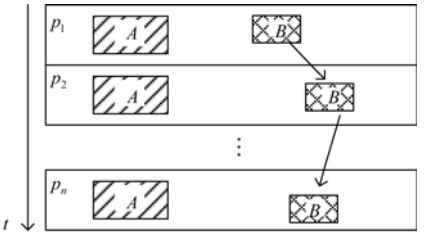


Fig. 8. Sequence of frames displaying stable area *A* and moving object *B*

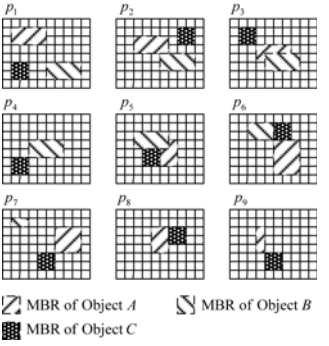


Fig. 9. Sequence of images that contain MBRs of objects A, B, and C

Definition 5.2 (Projection of moving object in video sequence)
Let P be a panoramic image sequence denoted by $P = p_1 p_2 \cdots p_n$. The MBR of a moving object A in p_i is represented by $((x_{si}, y_{si}),(x_{ei}, y_{ei}))$. The sequence of frames that contain the MBR of moving object A is denoted by $F = \Pi_{(M(A),FLOAT)}(P)$, where the parameter *FLOAT* means that calculation of the position information of the MBR of the moving object is necessary in each frame. More formally, it is defined as $F = \{f_i \mid f_i = (p_i,(x_{si}, y_{si}),(x_{ei}, y_{ei})),1 \le i \le n)\}$.

The sequence of frames that tracks a moving object may include several float shots. If the MBRs of the object in two neighboring frames are disjointed, the two frames are considered to be the boundary between two consecutive float shots. Based on this presupposition, the definition of a float shot is as follows.

Definition 5.3 (Float shot of moving object)
Let $F = f_1 f_2 \cdots f_n$ be the projections of the MBRs of a moving object A, with the shots that track A denoted by $S_{M(A)}(P) = \{s_1, s_2, \cdots, s_m\}$, where $s_i = f_a f_{a+1} \cdots f_b$, and $f_j \bigcap f_{j+1} \neq \phi$ $(a \le j \le b-1, \quad 1 \le i \le m)$.

Example: A panoramic image sequence denoted by $P = p_1 p_2 \cdots p_9$ is shown in Fig. 9. In these images, objects *A*, *B*, and *C* change their shape and position from frame to frame. The projections of each object in the sequence are denoted as $\Pi_{(M(A),FLOAT)}(P) = f_1 f_2 f_3 f_5 f_6 f_7 f_8 f_9$, $\Pi_{(M(B),FLOAT)}(P) = f_1' f_2' f_3' f_4' f_5' f_6' f_7'$, and $\Pi_{(M(C),FLOAT)}(P) = f_1'' f_2'' f_3'' f_4'' f_5'' f_6'' f_7'' f_8'' f_9''$.

As $f_3 \bigcap f_4 = f_4 \bigcap f_5 = \phi$, the projection of *A* is discontinued, and the set of shots of *A* of is denoted by $S_{M(A)}(P) = \{f_1 f_2 f_3, f_5 f_6 f_7\}$. $s_1 = f_1 f_2 f_3$ and $s_2 = f_5 f_6 f_7$ are two shots of *A* generated from image sequence P.

The shots of *B* are denoted by $S_{M(B)}(P) = \{f_1' f_2' f_3' f_4' f_5' f_6' f_7'\}$, which includes only one shot represented by $s_1 = f_1' f_2' f_3' f_4' f_5' f_6' f_7'$.

Finally, for each f_i'' and f_{i+1}'' in $\Pi_{(M(C),FLOAT)}(P)$, $f_i'' \bigcap f_{i+1}'' = \phi$, meaning that no shots are generated, hence, $S_{M(C)}(P) = \phi$.

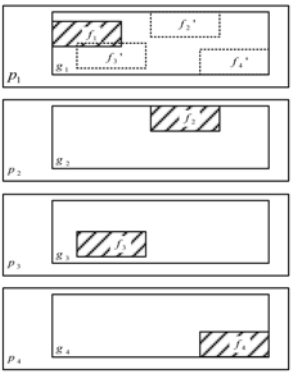


Fig. 10. Generation of overview shot

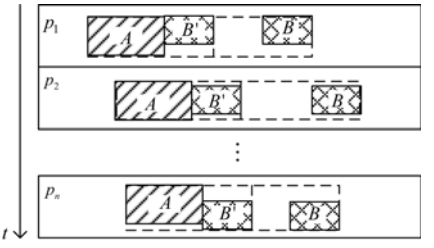


Fig. 11. Connection and attachment operations in video sequence

The float shot tracks a moving object by selecting the MBR of the object in each panoramic image. There could be another shot with a fixed size and position and a wide field of vision. The moving object is shown in each frame of the shot, and the viewer can observe the tracking of the object. In this type of shot, named an "*overview shot*", calculation of the MBRs of moving object a sequence of images is necessary. We developed a new operation that calculates an MBR that indicates the path of a moving object in a sequence of panoramic images. The definition is given as follows.

Definition 5.4 (Broad MBR of moving object)
For $F = f_1 f_2 \cdots f_n$, a projection of an MBR of an object A in a sequence of panoramic images $P = p_1 p_2 \cdots p_n$, and for $f_i = (p_i, (x_{si}, y_{si}), (x_{ei}, y_{ei}))$, $X_s = \{x_{si} \mid 1 \le i \le n\}, Y_s = \{y_{si} \mid 1 \le i \le n\}$, and $X_e = \{x_{ei} \mid 1 \le i \le n\}, Y_e = \{y_{ei} \mid 1 \le i \le n\}$, an MBR that contains all the area covered by the object in the whole sequence, which is defined as $BMBR(A) = ((\min(X_s), \min(Y_s)), (\min(X_e), \min(Y_e)))$.

Based on the concept of the broad MBR, an overview shot is defined as follows.

Definition 5.5 (Overview shot of moving object)
Let $F = f_1 f_2 \cdots f_n$ be a projection of an MBR of a moving object *A*. The overview shot of the object in the sequence of panoramic images $P = p_1 p_2 \cdots p_n$ is denoted by $G = \Pi_{(BMBR(A), OV)}(P)$, where the parameter *OV* means that each frame has the same size and position as the board MBR. More formally, it is defined as $G = \{g_i \mid g_i = (p_i, (x_s, y_s), (x_e, y_e)), 1 \le i \le n)\}$, where $BMBR(A) = ((x_s y_s), (x_e, y_e))$.

An example of the generation of an overview shot of a moving object is shown in Fig. 10. In a sequence of panoramic images $P = p_1 p_2 p_3 p_4$, the projections of the MBRs of a moving object *A* are denoted by $F = f_1 f_2 f_3 f_4$. In the first panoramic image p_1, the frame g_1 is the projection of the broad MBR of F. Hence, the overview shot is denoted by $G = g_1 g_2 g_3 g_4$.

5.2 Composition

A sample of the connection and attachment operations in a sequence of panoramic images is shown in Fig. 11. In each image, new frames are generated by the operations $con(f_{Ai}, f_{Bi})$ and $attach(f_{Ai}, f_{Bi})$ defined in Definitions 4.2 and 4.4. The operations in the video sequence are defined as follows.

Definition 5.6 (Operations of connection and Attachment)
Let each panoramic image in a sequence $P = p_1 p_2 \cdots p_n$ contain two MBRs of moving objects *A* and *B* that are denoted by *M*(*A*) and *M*(*B*). The float shots of each object are represented by $F_A = \Pi_{(M(A),FLOAT)}(P)$, and $F_B = \Pi_{(M(B),FLOAT)}(P)$.

The connection operation is defined as

$$CON\ (F_A, F_B) = con(f_{A1}, f_{B1})con(f_{A2}, f_{B2}) \cdots con(f_{An}, f_{Bn}) = \{ f_i \mid f_i = con(f_{Ai}, f_{Bi}), 1 \le i \le n \}\ .$$

The attachment operation is defined as

$$ATTACH\ (F_A, F_B) = attach(f_{A1}, f_{B1})attach(f_{A2}, f_{B2}) \cdots attach(f_{An}, f_{Bn}) = \{ f_i \mid f_i = attach(f_{Ai}, f_{Bi}), 1 \le i \le n \}\ .$$

6 Practical Applications Using ODV Model

In this section, we introduce two samples of practice application using ODV, and explain how the operations of our model can be used in these applications.

6.1 Video Conference

In the recoding of round table meetings, conventional cameras cannot acquire images that contain the faces of all participants. Y. Rui et al [5] designed a system using an ODC that enable users to view meetings. The system provides several interfaces: the all-up interface which shows appearances of specific participants selected by users, and the overview interface which shows all participants around the table.

The processes for generating video can be described using our model. The participants are labeled *A*, *B*, *C* and *D*, as shown in Fig. 12. The participant who is speaking can be detected with a voice recognizing system. Because the participants sit at the table and do not change their positions, the shot containing each participant can be considered a fixed shot, defined by Definition 5.1. For each two neighboring participants, a shot that contains both of them can be generated with the connection operation defined by Definition 5.6. *A* and *C* are on the opposite sides of each other, with either *B* or *D* between. A shot containing only *A* and *C* can be generated by attaching the frame of *C* to that of *A* in each image. This is the attachment operation defined by Definition 5.6.

In Fig. 12, (a) indicates the position of participants around the table and the panoramic image. (b) indicates a fixed shot of each participant. (c) and (d) indicate shots generated by the connection operation. (e) indicates shots of each two participants on opposite sides of each other generated by the attachment operation.

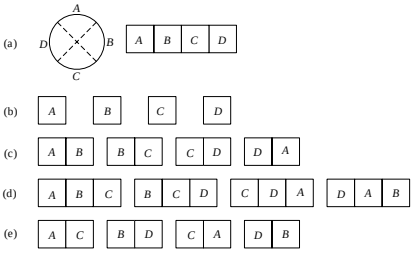


Fig. 12. Possible shots generated in video conferences



Fig. 13. Interface of Retrax System

6.2 Moving Object Tracking

The Retrax System [10] creates video-based virtual spaces with multiple video streams captured using an ODC. In this system, with several ODCs, complete sequences of activities, such as meetings or parties, can be recorded and stored. The interface of the system is shown in Fig. 13. The position and shape of each moving object is detected with image analysis technologies [8]. After the user indicates a human subject by changing the viewpoint, the system generates projections of the MBR of the target using the operation defined by Definition 5.2. If the user indicates multiple human subjects to be displayed in the main video window, the system is also able to generate shots containing those subjects using the connection operation defined by Definition 5.6. The system is also able to generate overview shots, enabling users to watch the tracks of indicated human subjects with a wide field of vision.

7 Conclusion

In this paper, we developed a new model for algebraic retrieval of ODV. ODV is recorded by an ODC, which has a vision field of 360º, much wider than conventional cameras. Because of the wide field of vision, it can record all objects and events occurring around the camera. Hence, when we observe a specific object in ODV, we need to create frames from video data which include the object.

We determined the characteristics of ODV data and defined conceptual units, such as frames, shots, and video sequences. We defined the semantic relationships to interrelate the conceptual units with each other. Based on the semantic relationships, we developed several algebraic operations for generating those conceptual units. Finally, we described two sample applications that our model can be used for based on the operations we developed.

Acknowledgements. This work was supported in part by MEXT The 21st Century COE (Center of Excellence) Program "Informatics Research Center for Developmentt of Knowledge Society Infrastructure" (Leader: Katsumi Tanaka, 2002-2006), MEXT Grant-in-Aid for Scientific Research on Priority Areas: "Cyber Infrastructure for the Information-explosion Era", Planning Research: "Contents Fusion and Seamless

Search for Information Explosion" (Project Leader: Katsumi Tanaka, A01-00-02, Grant#: 18049041), and MEXT Grant-in-Aid for Scientific Research on Priority Areas: "Cyber Infrastructure for the Information-explosion Era", Planning Research: "Design and Development of Advanced IT Research Platform for Information" (Project Leader: Jun Adachi, Y00-01, Grant#: 18049073).

References

[1] Fan, J., Gao, Y., and Luo, H. *Multi-Level Annotation of Natural Scenes Using Dominant Image Components and Semantic Concepts.* In Proceedings of the 12th annual ACM international conference on Multimedia, pp. 540-547, 2004.

[2] Kang, S. B. *Catadioptric Self-Calibration*. In Proceedings of IEEE CVPR, pp. 201-208, June 12, 2000.

[3] Li, J. Z., Özsu, M. T., and Szafron, D. *Modeling of Moving Objects in a Video Database*. In Proceedings of 1997 International Conference on Multimedia Computing and Systems (ICMCS '97), pp. 336-343, 1997.

[4] Oomoto, E. and Tanaka, K. *OVID: Design and Implementation of a Video-Object Database System*. IEEE Trans. on Knowledge and Data Engineering, vol. 5, no. 4, pp. 629-643, August 1993.

[5] Rui, Y., Gupta, A., and Cadiz, J. J. *Viewing Meetings Captured by an Omni-Directional Camera*. In Proceedings of ACM CHI 2001, pp. 450-457, Seattle, March 31- April 4, 2001.

[6] Smith, T. G. A., and Davenport, G. *The Stratification System: A Design Environment for Random Access Video*. In Proceedings of 3rd Int'l Workshop on Network and Operating System Support for digital Audio and Video, pp. 250-261, 1992.

[7] Tanaka, K., Tajima, K., Sogo, T., and Pradhan, S. *Algebraic Retrieval of Fragmentarily Indexed Video*. New Generation Computing. Vol. 18, No. 4, pp. 359-374, 2000.

[8] Urano, T., Matui, T., Nakata, T., and Mizoguchi, H. *Human Pose Recognition by Memory-Based Hierarchical Feature Matching*. In Proceedngs of IEEE SMC'2004, pp. 6412-6416, Oct. 2004.

[9] Weise, R., Duda, A., and Gifford, D. "Content-based access to algebraic video", In Proceedings of IEEE First International Conference on Multimedia Computing and Systems, pp. 140-151, Boson, MA, May 1994.

[10] Yokota, Y., He, S., and Kambayashi, Y. *Querying Multiple Video Streams and Hypermedia Objects of a Video-Based Virtual Space System*. Digital Cities III, Information Technologies for Social Capital - a Cross-Cultural Perspective, Lecture Notes in Computer Science, 3081, pp. 299-309, Springer-Verlag, 2005.

[11] Zhang, H. J., Low, C. Y., Smoliar, S. W., and Wu, J. H. *Video Parsing, Retrieval and Browsing: An Integrated and Content-Based Solution*. In Proceedings of ACM Multimedia 95, San Francisco, CA, pp. 15-24, Nov. 1995.

Temporally Integrated Pedestrian Detection from Non-stationary Video

Chi-Jiunn Wu and Shang-Hong Lai

Department of Computer Science, National Tsing Hua University, Hsinchu, Taiwan
lai@cs.nthu.edu.tw

Abstract. In this paper, we propose a novel approach for detecting pedestrians from video sequence acquired with non-static camera. The proposed algorithm consists of three major components, including global motion estimation with motion-compensated frame subtraction, AdaBoost pedestrian detection, and temporal integration. The global motion estimation with frame subtraction can reduce the influence of the background pixels and improve the detection accuracy and efficiency. The simplified affine model is used to fit the global motion model from some reliable blocks by using the RANSAC robust estimation algorithm. After motion-compensated frame subtraction, the AdaBoost classifier is employed to detection pedestrians in a single frame. At last, the graph structure is applied to model the relationship of different detection windows in the temporal domain. Similar detected windows are grouped as the same clusters by using the optimal linking algorithm. The missed detection windows will be recovered from the object clustering results. Finally, we show the experimental results by using the proposed pedestrian detection algorithm on some real video sequences to demonstrate its high detection accuracy and low false alarm rate.

Keywords: non-static camera, global motion, AdaBoost, affine motion model, RANSAC robust estimation, graph structure, optimal temporal linking.

1 Introduction

In the past decade, object detection has been a major focus of research in computer vision, including face detection and car detection. Previous researches usually focus on the static object detection from a single image. The detection of articulated objects is a very challenging topic in this research field. Especially, the human detection has attracted many researchers recently. This problem is also closely related to the image and video retrieval. For example, an intelligence home video retrieval system may need to search videos containing persons with specified features under certain environment.

Before recognizing a person's identity or activity from video, it is necessary to search where the person is as the first step. Many researchers have proposed different approaches for human detection or pedestrian detection in a single image or from a video sequence. The previous methods can be roughly divided into two major approaches: one is extracting robust features to represent a human and the other is

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 188 – 197, 2007.
© Springer-Verlag Berlin Heidelberg 2007

partitioning the human into several parts for recognition. The former used templates to represent the human model or extracted the motion pattern of pedestrian in the temporal domain, followed by applying some machine learning technique to detect the pedestrian. The main idea of the former approach is to model the contours of the human or the moving trajectory of the pedestrian. Since human is an articulated object, it is not easy to model the human of different poses and different views. The main idea of the latter approach is to partition the human into several rigid segments. Recently, many researches represent the human contour as a graph with each segment being a node in the graph and perform statistical learning on the graph to be able to describe articulated human motion. However, this approach is usually very time-consuming.

Human detection can be roughly divided into two major approaches: image-based human detection and video-based pedestrian detection. Papageorgiou et al. [13][12] proposed a pedestrian detection system based on the Harr-like wavelet descriptor and employ the polynomial SVM learning algorithm to discriminate the human and non-human images. Depoortere et al. [14] presented an improved version of it. Gavrila and Philomen [6][7] gave a more direct approach of human similarity measure. Dalal et al. [4] extended this idea by grids of histograms of oriented gradient (HOG) descriptors. Another idea is to partition a person into several components and classify each of them. Mohan et al. [11] proposed the component-based human detection system. Leibe et al. [9] proposed an Implicit Shape Model (ISM) for object detection. Most video-based pedestrian detection methods are applied to the surveillance system to detect the human or pedestrian. In order to increase the accuracy, the background subtraction has been employed to find the moving object and detect the human as the foreground region. Haritaoglu et al. proposes the W^4 visual surveillance system [8] to analyze the behavior of the detected people. A statistical field model [18][17] is applied for pedestrian detection. The Markov network was employed to model the deformable objects, such as pedestrian. Another approach analyzes the motion pattern to detect the pedestrian in the video sequence. Avidan [1][2] incorporated the optical flow information into SVM classifier and called it Support Vector Tracking for tracking learned objects. The other approach used the motion pattern between two consecutive frames as proposed in [16]. In this work, the appearance model is established via the wavelet descriptor and its motion pattern is extracted by subtracting the current image from the shifted versions of the reference images.

The existing approaches focus on the human detection from a single image or pedestrian detection in a surveillance system. As the video camcorder is more and more popular, the video database is also increasing very rapidly. The appearance of pedestrians in video provides important information for the retrieval and categorization of video sequences. In this paper, we focus on the pedestrian detection from a non-stationary camera via temporal integration to increase the detection accuracy and reduce the false alarm rate.

2 The Proposed Pedestrian Detection Algorithm

Many approaches have been presented for human or pedestrian detection. In the single image based human detection scheme, it can achieve high detection accuracy

with considerable false positive rate. In the video based pedestrian detection scheme, the false positive rate can be reduced with the background modeling but the limitation is that the system should be setup with a static camera. By generalizing the above two different approaches, we propose a video based pedestrian detection system for video captured from non-static camera. The system can be divided into three major steps: global motion estimation and background removal, pedestrian detection, and temporal integration. Firstly, the global motion between adjacent frames is estimated to remove the background region, thus reducing the false positive rate. Secondly, the AdaBoost classifier is trained and applied to determine the candidate regions for pedestrian detection. Finally, a temporal integration strategy is used to remove the false positive windows and recover the missed detection regions. We will discuss these components in details in the following sections.

2.1 Global Motion Estimation and Background Removal

Because we assume the video sequence is acquired from a non-static camera, it is difficult to model the background image based on several frames containing moving foreground and background regions. We can estimate the global motion from two consecutive frames in order to eliminate the background region. First of all, we adopt the diamond search motion estimation to compute the motion field. Secondly, the motion fields of the uncertain blocks, such as homogeneous regions, are removed because they are not the real motion vectors and may distract the global motion estimation. Finally, for the remaining motion fields, including the foreground and background regions altogether, the RANSAC robust estimator is applied to determine all inliers as the background regions by estimating a simplified affine model as the global motion model.

2.1.1 Rejection of Uncertain Regions

The motion vectors estimated from the block matching may contain the ambiguous problem. These blocks are called the uncertain blocks, such as homogeneous or 1D structure blocks. In order to remove these blocks, the relationship between pixels in the same block should be considered based on the gradient covariance matrix, which is given as follows:

$$C(x,y) = \sum_x \sum_y \nabla I(x,y) \nabla I(x,y)^T = \begin{bmatrix} \sum_{p=i}^{i+n} \sum_{q=j}^{j+n} I_x(p,q) I_x(p,q) & \sum_{p=i}^{i+n} \sum_{q=j}^{j+n} I_x(p,q) I_y(p,q) \\ \sum_{p=i}^{i+n} \sum_{q=j}^{j+n} I_x(p,q) I_y(p,q) & \sum_{p=i}^{i+n} \sum_{q=j}^{j+n} I_y(p,q) I_y(p,q) \end{bmatrix} \quad (1)$$

The block structure can be estimated by the eigenvalue decomposition. If it is a homogeneous block, both of its two eigenvalues are close to zero. If the block contains a 1D structure, then one eigenvalue is large and the other is close to zero. Otherwise, both eigenvalues are large. The results are obtained by computing the

gradient covariance matrix in each block and taking the singular value decomposition on this 2-by-2 matrix. If both eigenvalues of the gradient covariance matrix for a block are large enough, this block is not treated as an uncertain block. Otherwise, it is treated as an uncertain block and removed from the global motion estimation.

2.1.2 Simplified Affine Model Fitting

After removing the uncertain blocks, the remaining blocks can be used to establish the global motion model. The following simplified affine model is used as the global motion model in this paper.

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a & b \\ -b & a \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} \tag{2}$$

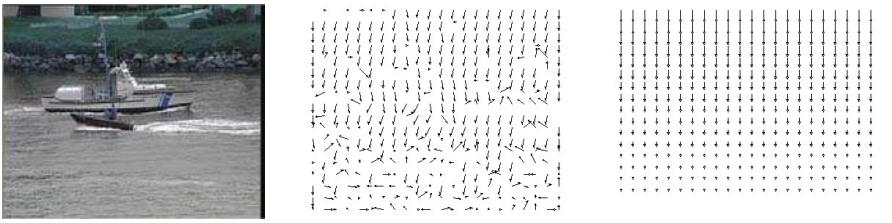
where the parameters a and b control the scaling and rotational motion, $(\Delta x, \Delta y)$ is the translation vector, and (x', y') and (x, y) are the corresponding points in the current frame and reference frame, respectively. The correspondence between the two consecutive frames is obtained by the motion vectors computed by the diamond search motion estimation. The estimation of the affine parameters can be solved by using the least-square method. However, the least-square estimation is sensitive to outliers, which are inevitable in the global motion estimation when there is foreground motion in addition to the camera motion.

2.1.3 RANSAC Robust Estimation

Although standard least-squares method can solve the over-constrained linear system, the problem with the least-square solution is that it is sensitive to the outliers. The RANdom SAmple Consensus [5] is a technique to estimate the model from a set of data containing outliers. The main idea is to randomly select *n* data to generate a model hypothesis and check how well this model fits to all the data. This hypothesis-and-testing process is repeated for a sufficient number of times. Finally, we select the best hypothesis that fits to the most number of data points as the final solution. Figure 1 shows an example of the global motion estimation on the coastguard sequence.

2.2 Pedestrian Detection

Extracting the representative features to describe a human object is not easy because of the illumination variations, variations in human shapes, different backgrounds, a wide variety of clothes, and different postures of pedestrians. The features used for the pedestrian detector is similar to the histogram of oriented gradient descriptor (HOG) [4], which is a suitable method to represent a human silhouette. After extracting the features for each image, the AdaBoost learning algorithm is employed to classify the human and nonhuman images. Figure 2 shows examples of image subtraction after the global motion compensated image pairs. The HOG features are computed based on the subtracted images.

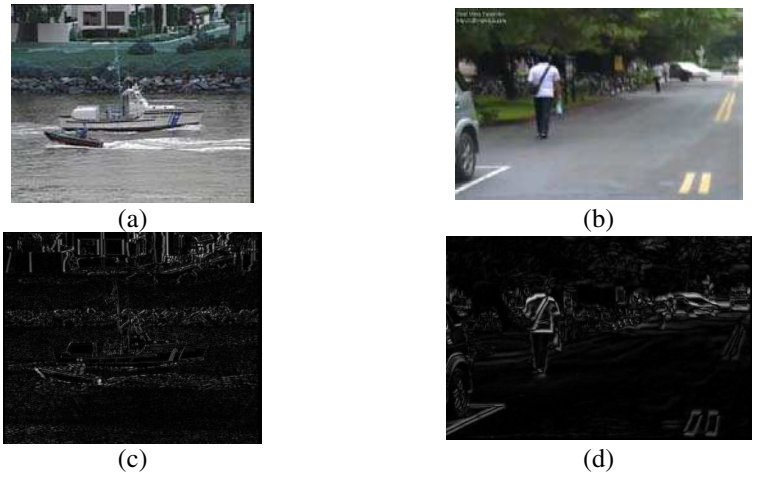


(a) (b) (c)

Fig. 1. Example of the global motion estimation: (a) the original image, (b) the results of diamond motion estimation, (c) the robust global motion estimation results

2.2.1 Adaboost Classifier

The Adaboost learning algorithm is a well-known classifier used for several applications [10][15]. Its idea is to select a subset of complementary features via re-weighting the training examples based on the classification results for the current classifier in each iteration. The conventional AdaBoost algorithm is to find the decision boundary from the distribution of positive and negative data sets. A problem may occur when both distributions are mixed together. To overcome this problem, we apply the histogram equalization to quantize the feature space into several bins for the positive training data. In addition, the Bayesian decision rule is employed to compute the probability of each weak classifier, given as follows:



(a) (b)

(c) (d)

Fig. 2. Examples of the foreground energy: (a) and (b) are the original images. (c) and (d) are the foreground energy with respect to the global motion model.

$$B(x_i^j) = \frac{p(yi = 1 \mid x_i^j \in Q_j(x_i^j))}{p(y_i = 0 \mid x_i^j \in Q_j(x_i^j)) + p(y_i = 1 \mid x_i^j \in Q_j(x_i^j))} \tag{3}$$

where x_i^j denotes the *j-th* feature of the *i-th* image and Q_j is the *j-th* quantization bin.

2.3 Temporal Integration

The temporal information is used to further reduce the false positive rate and recover the missed detection as well. For the temporal integration, a graph model is established to model the detection windows throughout the whole video shot. Next, an optimal linking algorithm is employed to group the detection windows of the same persons from a subset of the shot. Finally, the missed detection windows are recovered and false positive windows are removed.

2.3.1 Graph Model

The graph model is used to describe the relationship between the detection windows across the nearby frames. We treat each detection window as a node in the graph model. Ideally, there is an edge linked for every two nodes between two consecutive frames. Unfortunately, there may be missed detection windows even if a person appears throughout the video sequence. In this way, the links should be established between the nearby *W* frames.

In order to determine the similarity between two nodes, the color histogram has been computed as the features for each node. The following two equations are used to compute the weight between two nodes *x* and *y*:

$$L(x,y)=\begin{cases}\infty & ,if\begin{cases}D(x,y)>T_{dist}\\ S(x,y)>T_{size}\end{cases}\\ \psi(x,y) & ,otherwise\end{cases} \tag{4}$$

$$\psi(x,y)=\sum_{i=1}^{n}|\hbar(x_i)-\hbar(y_i)| \tag{5}$$

where $\hbar$ is the distribution of color histogram, *n* is the total number of quantized bins, *D(x,y)* is the distance between the centers corresponding to the two nodes *x* and *y*, *S(x,y)* is the difference of the sizes of the two nodes, and T_{dist} and T_{Size} are the two thresholds for the distance and size difference measures.

2.3.2 Optimal Linking

The solution to finding the optimal path in the graph model has been proposed in several previous works [3], such as the depth first search algorithm and maximum-flow/minimum-cut algorithm. The depth first search may lead to the wrong path because it can not guarantee the link is minimal in the whole graph model. The maximum-flow/minimum-cut algorithm needs the starting and ending nodes from prior knowledge. In this case, each frame may contain the situation that a person appears and leaves. It is difficult to determine which nodes are the starting and ending nodes automatically. To avoid this problem, we regard this temporal linking problem as a segmentation problem in the graphical model framework. The nodes are combined as the same group, which means the same person is detected in this subset of sequence.

Therefore the optimal linking algorithm is employed to solve this problem. The idea is to find the minimum link through the whole graph model in each iteration. Once the minimum link has been found, the two nodes or two groups will be treated

as the same group if all nodes in two groups exist in different frames. The following equation is to determine if the two different groups contain the same frame number.

$$\Gamma(G_A, G_B) = \begin{cases} 0 & ,if\ F(a_i) = F(b_j) \text{ for all pair}(a_i, b_j), a_i \in A, b_j \in B \\ 1 & ,otherswise \end{cases} \quad (6)$$

where A and B are two different groups containing several nodes a_i and b_j, and $F(.)$ returns the frame number of the corresponding node. For each iteration, the minimum link will be chosen and determine if two groups with this link contain the same frame number. If they have the same frame, the link will be skipped and the next minimum link is checked. Otherwise, the link is established and the two groups will be grouped together. Finally, there are several groups in the graph model, the smaller groups will be treated as false detections and the rest corresponds to detected persons.

2.3.3 Recovering Missed Detection

There are some missed detection windows in the group containing large nodes. The recovery procedure is simply to find the highest detection probability inside the windows determined from the nearest two detection nodes in the lost frames. The position and the size of the detected windows are used to determine the positions and window sizes to be searched again in the recovery process.

3 Experimental Results

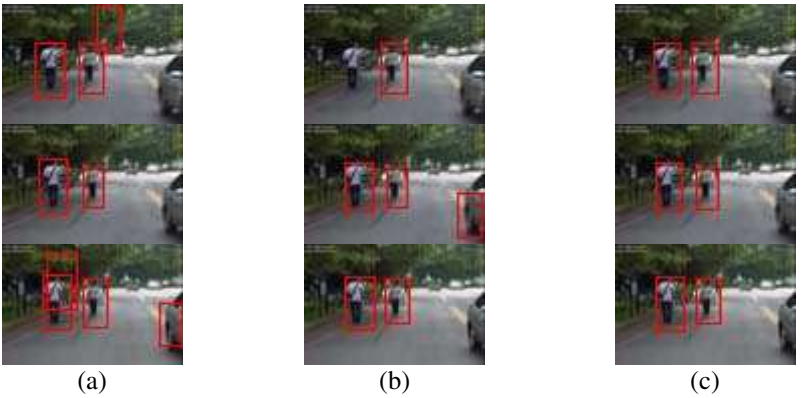
In our proposed method, there are three major components combined in our system, including the global motion elimination, pedestrian detection, and temporal integration. In this section, the results show that the performance is improved in each stage. About the training database of the Adaboost algorithm, we take the positive human images from the MIT and INRIA human databases and negative images from a set of natural scene images. Totally, there are about 2235 positive images, including 1018 images from MIT database and 1227 images from INRIA database, and 6085 negative images trained in our pedestrian detection system. About the testing sequences, three video sequences are used to evaluate the performance of our pedestrian detection system.

We show some experimental results of applying the proposed pedestrian detection algorithm on some real video sequences acquired on campus. Fig. 3 shows the experimental results in the 107^{th}~109^{th} frames of sequence 1 with different combinations of the components in our system. Similarly, Figure 4 shows the experimental comparison on the 448^{th}~450^{th} frames of sequence 1. It is evident from the results that the temporal integration and the subtraction of the global motion compensated consecutive frames helps to reduce the false alarms. The temporal integration scheme can also recover the missed detection windows as shown in Fig. 4(c). Fig. 5 and Fig. 6 show some more experimental results in other sequences by using our proposed method.



(a) (b) (c)

Fig. 3. The detection results for frame 107^{th}~109^{th} of Sequence 1 by using (a) AdaBoost detection algorithm only, (b) AdaBoost detection + Background removal, and (c) the proposed pedestrian detection method



(a) (b) (c)

Fig. 4. The detection for frame 448^{th}~450^{th} in results of Sequence 1 by using (a) AdaBoost detection algorithm only, (b) AdaBoost detection + Background removal, and (c) the proposed pedestrian detection method

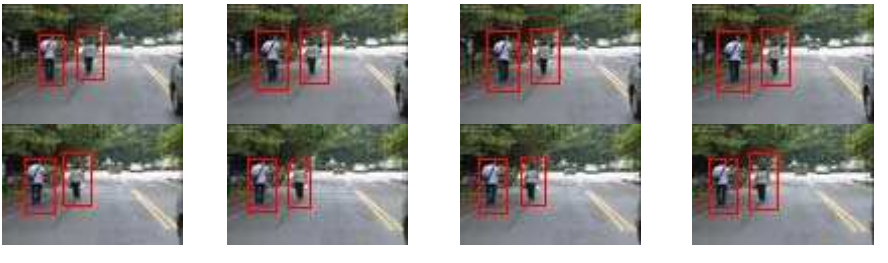


Fig. 5. The pedestrian detection results of applying the proposed algorithm to sequence 1

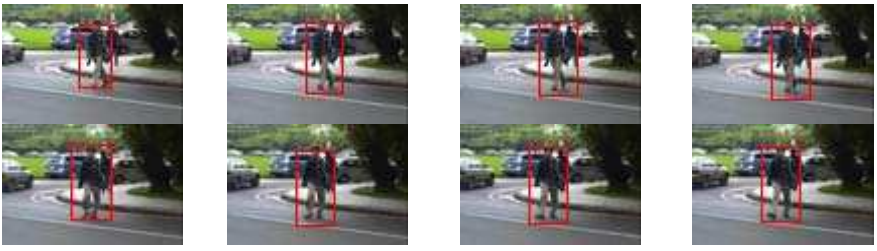


Fig. 6. The pedestrian detection results of applying the proposed algorithm to sequence 2

4 Conclusion

In this paper, we proposed an integrated approach that can detect pedestrian from video acquired with non-static camera. The proposed system consists of three components; namely, global motion estimation with motion compensated consecutive frame subtraction, AdaBoost pedestrian detection, and temporal integration. The global motion estimation with background subtraction can reduce the influence of background pixels and increase the detection accuracy. It discards the uncertain blocks and fits a simplified affine model by using the RANSAC robust estimation algorithm. After motion-compensated image subtraction, the AdaBoost algorithm is applied to detect the pedestrian in a single frame. Finally, the graph structure is used to model the relationship of detection window in the temporal domain. Similar nodes in the graph model are grouped together as the same clusters by using an optimal linking algorithm and the missed detection windows can be recovered in the clusters with a large number of nodes. Experimental results on real video sequences are shown to demonstrate its improved pedestrian detection accuracy.

Acknowledgements

This research work was supported in part by ITRI EOL as well as the MOEA research project under the grant 94-EC-17-A-01-S1-034.

References

1. S. Avidan.: Support vector tracking. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, volume 1 (2001) 184-191
2. S. Avidan.: Support vector tracking. IEEE Trans. on Pattern Analysis and Machine Intelligence (2004) 1064-1072
3. T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein.: Introduction to Algorithms. 2001
4. N. Dalal and B. Triggs.: Histograms of oriented gradients for human detection. In IEEE Computer Society Conference on Computer Vision and Pattern Recognition, volume 1 (2005) 886-893
5. M. A. Fischler and R. C. Bolles.: Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM (1981) 381-395

6. D. Gavrila.: Pedestrian detection from a moving vehicle. In Proceedings of the 6th European Conference on Computer Vision, (2000) 37-49
7. D. M. Gavrila and V. Philomin.: Real-time object detection for smart vehicles. In IEEE Computer Society Conference on Computer Vision and Pattern Recognition (1999) 87-93
8. I. Haritaoglu, D. Harwood, and L. S. Davis.: W^4: Real time surveillance of people and their activities. IEEE Trans. on Pattern Analysis and Machine Intelligence (2000) 809-830
9. B. Leibe, E. Seemann, and B. Schiele.: Pedestrian detection in crowded scenes. In IEEE Computer Society Conference on Computer Vision and Pattern Recognition, volume 1 (2005) 878-885
10. S. Z. Li and Z. Zhang.: Floatboost learning and statistical face detection. IEEE Trans. on Pattern Analysis and Machine Intelligence (2004) 1112-1123
11. A. Mohan, C. Papageorgiou, and T. Poggio.: Example-based object detection in images by components. IEEE Trans. on Pattern Analysis and Machine Intelligence (2001) 349-361
12. C. Papageorgiou, M. Oren, and T. Poggio.: A general framework for object detection. In Proceedings of the IEEE International Conference on Computer Vision (1998) 555-562
13. C. Papageorgiou and T. Poggio.: A trainable system for object detection. International Journal of Computer Vision (2000) 15-33
14. V. de Poortere, J. Cant, B. Van den Bosch, J. dePrins, F. Fransens, and L. Van Gool.: Efficient pedestrian detection: a test case for svm based categorization. In Workshop on Cognitive Vision (2002)
15. P. Viola and M. J. Jones.: Robust real-time face detection. International Journal of Computer Vision (2004) 137-154
16. P. Viola, M. J. Jones, and D. Snow.: Detecting pedestrians patterns of motion and appearance. In Proceedings of IEEE International Conference on Computer Vision, volume 2 (2003) 734-741
17. Y. Wu and T. Yu.: A field model for human detection and tracking. IEEE Trans. on Pattern Analysis and Machine Intelligence (2006) 753-765
18. Y. Wu, T. Yu, and G. Hua.: A statistical field model for pedestrian detection. In Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, volume 1 (2005) 1023-1030

Visual Perception Theory Guided Depth Motion Estimation

Li Bing[1,2], Xu De[2], Feng Songhe[2], and Wang Fangshi[2]

[1] Beijing Key Lab of Intelligent Telecommunications Software and Multimedia
Beijing University of Posts and Communications, Beijing, China 100876
[2] Institute of Computer Science, Beijing Jiaotong University, Beijing, China, 100044
binggege@people.com.cn, xd@computer.njtu.edu.cn

Abstract. Motion estimation is an important and computationally intensive task in video coding and video analysis. But existent motion estimation algorithms mainly focus on 2-D image plane motion and neglect the motion in depth direction, which we call it depth motion in this paper. There are even few researches on the depth motion, their methods are complex and most of them need binocular images. In this work, visual perception theory is used to estimate the depth motion. A novel depth motion estimate method is proposed base on visual perception theory and it can estimate the depth motion from just monocular video. Experimental results show that our model is simple, effective and corresponds to the human perception.

1 Introduction

Motion estimation is an important task in video coding and video analysis. So far various methods for motion estimation are proposed. But most of them mainly estimate the motion vector on the 2-D image plane and ignore the motion in depth direction, which is called depth motion (DM) in this paper. 2-D motion estimation methods mainly include block-matching based method [1, 2, 3] and optic flow based method [4]. These methods will generate errors, when face to the DM. Just like what is in shown Fig.1, the motion vectors in Fig.1(C) are all wrong.

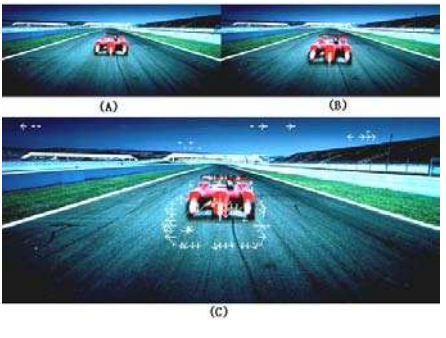


Fig. 1. (A) and (B) are successive frames in a video. The car in the frames is running toward the observer (or camera). The white arrows in (C) are motion vector computed by block-matching based algorithm.

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 198 – 206, 2007.
© Springer-Verlag Berlin Heidelberg 2007

Existent researches about depth motion or affine motion [5,6] mostly need binocular or more images. And they all based on physical model and ignore the human perception, which is most important in visual system. In this paper, introducing human perception into depth motion estimation, we establish a simple model to estimate the DM and our method just need monocular video. Because the expansion and contraction of object may affect the perception of DM, we just research on the rigid objects' depth motion.

This paper is organized as follows. In Section 2, perceptual size constancy theory and depth motion perception are described. Then we propose the computation of the DM and design a computation algorithm in section 3. The experimental results using this method are presented in Section 4. Section 5 concludes this paper.

2 Some Visual Perception Theories About Depth Motion

The visual psychological explanation of the depth motion is base on the perceptual size constancy, which is a famous and admitted theory in the psychology. Consequently, perceptual size constancy theory is introduced firstly in the follow section.

2.1 Perceptual Size Constancy

Our perception of objects is far more constant or stable than our retinal images. Retinal images change with the movement of the eyes, the head and our position, together with changing light. If we relied only on retinal images for visual perception we would always be conscious of people growing physically bigger when they came closer, objects changing their shapes whenever we moved, and colors changing with every shift in lighting conditions. Counteracting the chaos of constant change in retinal images, the visual properties of objects tend to remain constant in consciousness. This phenomenon is called perceptual constancy in visual psychology theory. Psychologists classified the perceptual constancy into four categories: color constancy, brightness constancy, size constancy and shape constancy [7,8].

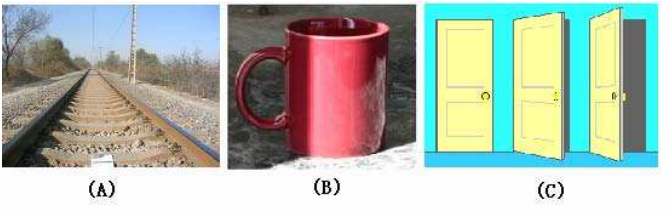


Fig. 2. Examples of perceptual constancy: Image(A) shows the size constancy, the near point of railway is wider than farther points in the image, but the perceptual width of the railway is the same . The color and brightness constancy are illustrated in image(B), the cup is illuminated by different brightness lights. The color of the cup in the brighter part is different from the color in the dark part, but human perceives that all the cup is red, just one color. The shape constancy is described in image(C). The same door has different images from different angle, but the shape, which is perceived by human, is still rectangle.

The size constancy is playing a key role in human vision in recognizing objects. Psychologists have discovered computation theory of size constancy [9]. They have got an expression to compute the perceptual size as:

$$PS = k \times A \times D \tag{1}$$

Where *PS* is the object's perceptual size, *A* is the angle of view, *D* is the perceptual depth of object, k is the zoom coefficient of human eyes or camera, and it keeps invariable in a certain imaging process. The angle of view A can be represented by the size of object in the image [9]. So we use the object's image size *I* instead of the angle of view *A*, it can also be expressed as follow:

$$PS = k \times I \times D \tag{2}$$

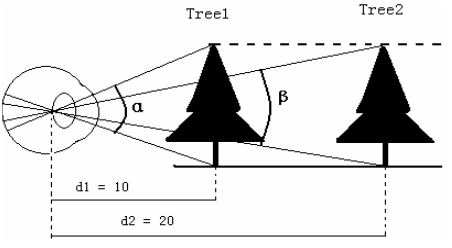


Fig. 3. Example of computation of size constancy: The images of two trees in human retina have different heights. Tree 1's image is higher than tree 2's in the retina, but the heights which human perceive are same.

Fig. 3 shows an example of the computation theory of the size constancy. The view angles of the two trees are α and β. The view angle can be represented by the size of object in the image. We define the sizes of the trees in the image as S_1 and S_2. In addition, according to the pinhole imaging model, the size of the object in image is in inverse proportion to the distance between the object and the observer, that is $S_1/S_2 = d_2/d_1$. From equation (1) and (2), we can compute the perceptual size of the two trees, we define the perceptual size of the two trees are PS_1 and PS_2, the relationship between PS_1 and PS_2 is as follow

$$\frac{PS_1}{PS_2} = \frac{k \times \alpha \times d_1}{k \times \beta \times d_2} = \frac{S_1}{S_2} \times \frac{d_1}{d_2} = 1 \tag{3}$$

From the computation of the size constancy, we find the perceptual size of tree1 is equal to the perceptual size of tree2. Although they have different size images on human retina, the sizes perceived by human are same.

2.2 Perception of Depth Motion

In perception, a characteristic feature of depth motion stimulation is that a moving, rigid object generates a changing retinal image characterized by looming, that is, by a

shrinking or expanding two-dimensional motion pattern over the receptor surface of the eye. It has long been known that such a stimulus of expansion or contraction along with perceptual size constancy generates the perception of motion in depth. Under these conditions of optic flow, it has repeatedly been found that the perceived size constancy is perfect or very high. Consequently, the concept of perceived size constancy plays an important role in several current theoretical analyses of motion perception [10].

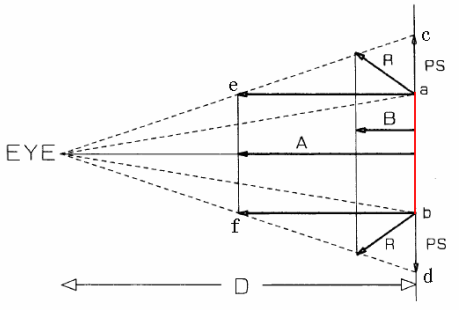


Fig. 4. A model of the perception of depth motion: *ab* is a certain object, when its retinal image changes to as same as the retinal image of *cd*, human visual system extracts this spatial change and perceives three different motion perception

In Fig.4 (the figure is from [10]), it illustrates a model of depth motion perception. The retinal image of the object changes from *ab* to *cd*, this proximal 2-dimensional change can be perceived by human visual system and may be transformed into three different possible motions: (1) All the changes are transformed into motion in depth together with perfect size constancy. The motion vector is A, which implies a depth motion toward the eye to the position *ef*. According to the size constancy theory and equation (2), we can get a expression as follow:

$$D = \frac{PS}{k \times I} \tag{4}$$

The retinal image of a certain object, that is *I* in the expression above, become larger, and the perceptual size, that is *PS*, keep fixed, so the depth *D* decreases. That means the object have a motion in the depth and the direction is toward to the observer. (2) All the changes are transferred to the 2-dimensional elasticity and no depth motion is generated. In this case, the object has only a shape transformation; it expands from *ab* to *cd*, the vector is *ps* in the Fig.4. (3) A combination of category (1) and category (2) implying that the proximal change is transferred into a depth motion together with object's itself expansion. This case is shown by the vector *R* in the Fig.4. In this paper, we just consider the rigid object situation. So the category (2) and (3) can not exist. These spatial changes are only caused by depth motion. In the next section, we will discuss how to compute the depth motion vector.

3 Depth Motion Estimation

In this section, we study on the computation of the depth motion based on the visual perception theory. And then we design an algorithm to implement this depth motion computation.

3.1 Computation of Depth Motion

From the equation (4), we can get the depth motion vector dD, but the computation just from 1-dimensional information is not accurate and sensitive to the error. So we use the 2-dimensional information to compute the depth motion. From the equation (2), the follow two expressions can be got:

$$PW = k \times W \times D \tag{5}$$

$$PH = k \times H \times D \tag{6}$$

Where *k* is zoom coefficient, which is invariable without zooming or dollying operations of the camera; the *W*,*H* are the width and the height of the object in the image plane, *PW*, *PH* respectively are the perceptual width and the perceptual height of the object by human beings. Ignoring the depth of the object itself, the *D* in the (5) and (6) are the same to a certain object. Make equation (5) multiply equation (6), we can get:

$$PA = k^2 \times A \times D^2 \tag{7}$$

Where $PA = PW \times PH$, is the perceptual area of the object, and $A = W \times H$, is the object's area in the 2-dimension image plane. The depth *D* can be computed as follow:

$$D = \sqrt{\frac{PA}{k^2 \times A}} \tag{8}$$

To a certain object in a certain image, the *PA* and *k* are fixed. We define $\varepsilon = \sqrt{\frac{PA}{K^2}}$, and get the differential coefficient of *D* as:

$$DM = dD = -\frac{\varepsilon}{2\sqrt{A^3}} dA \tag{9}$$

Where dD is just the motion vector in depth, which is depth motion *DM*. The positive direction of the *DM* is the direction away from the observer. So we can quantitatively analyze the depth motion in the monocular 2-D video sequences according to the equation (9).

3.2 Algorithm of Depth Motion Estimation

In this section, we design an algorithm to analysis the depth motion in the video. The extraction of video object plane (VOP) is a difficult problem in video analysis; consequently, we only use the video sequence, which has static background, to validate our depth motion computation. It implies that the camera in this video sequence is fixed and has no zooming operation. The flowchart of the algorithm is as follow:

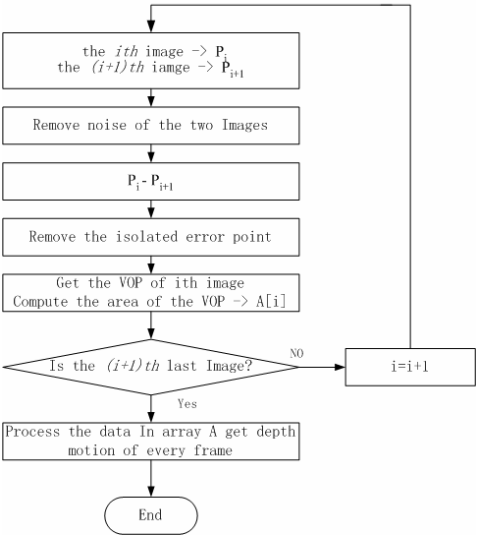


Fig. 5. The flowchart of the depth motion estimation algorithm

We firstly make the successive two frames minus and get the result, after that, we remove the isolated error points from the result above. At last, the moving object can be extracted and the area of object is computed simultaneity. After all the frames of the video processed, the areas of every frame will be processed according to the equation (9). Consequently, the depth motion vectors are gotten.

4 Experimental Results

We use a video sequence from an American film, whose name is “Michel Vaillant”, to experiment. The numbers of the frames are from 1 to 12, according to the order from right to left, then from top to bottom in the Fig. 6. This video sequence displays that a car runs toward the observer (or camera).

The algorithm in section 3.2 is used to extract the moving object (car) in the sequence. The result of moving object is shown in Fig.7. And the data of the car’s area is computed at the same time. Table 1 shows the data of car’s area and the results of the depth motion vector.

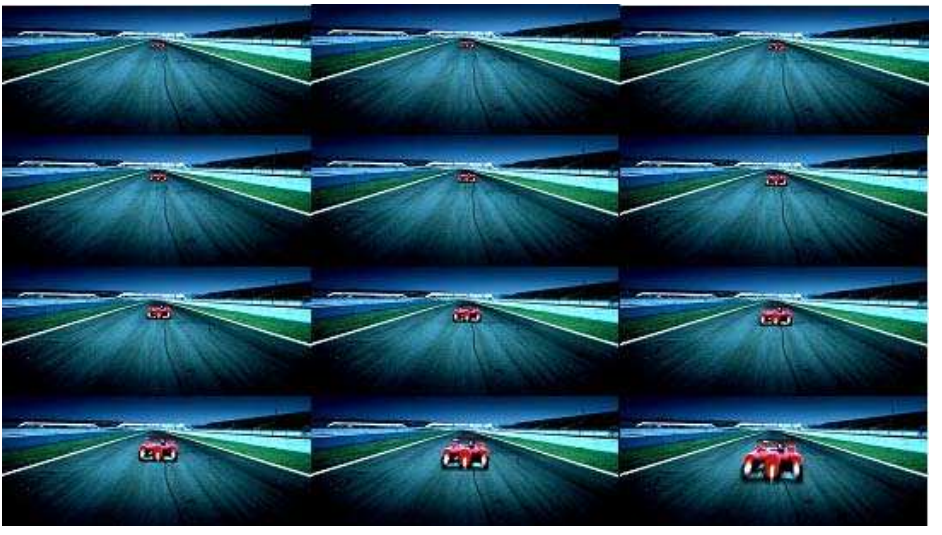


Fig. 6. The experimental video sequence of the car: There are 12 successive frames from the video. The car is running toward the observer. The areas of the car in the images become bigger and bigger.

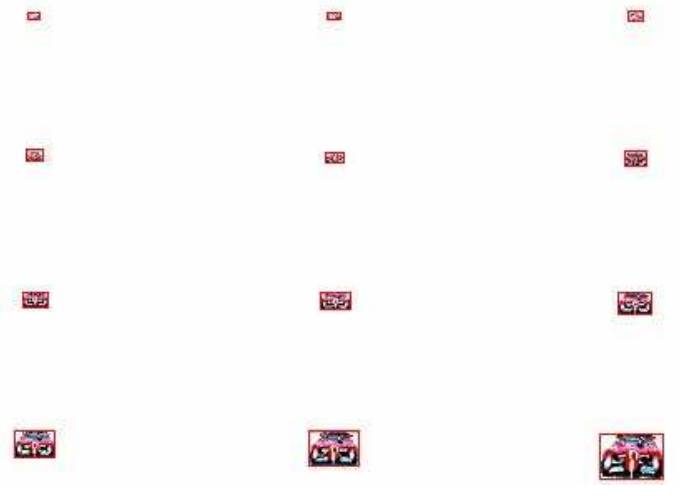


Fig. 7. The moving object (car) in every frame is extracted

From the table 1, we can find that the depth motion vectors are all minus, which implies that the car is running toward to the observer. And the value of the DM vector becomes smaller and smaller. We can draw a conclusion that the velocity of the car is decreasing in this moving process. The trend of motion is shown in Fig.8.

Table 1. A is the area of the moving object (car). $dA = A_i - A_{i-1}$. DM is the vector of the depth motion. In order to compute simply, we make $\varepsilon = 2$ in the expression (9).

Frame No.	A	dA	$\sqrt{A^3}$	DM
0*th*	335	---	---	---
1*st*	406	71	8180.673	- 0.008679
2*nd*	496	90	11046.44	- 0.008147
3*rd*	635	139	16001.5	- 0.008687
4*th*	819	184	23438.29	- 0.00785
5*th*	1086	267	35788.6	- 0.00746
6*th*	1512	426	58793.28	- 0.007246
7*th*	2120	608	97612.13	- 0.006229
8*th*	3190	1070	180171.5	- 0.005939
9*th*	5174	1984	372168.5	- 0.005331
10*th*	8844	3670	831712.3	- 0.004413
11*th*	17095	8251	2235134	- 0.003692
12*th*	37313	20218	7207593	- 0.002805

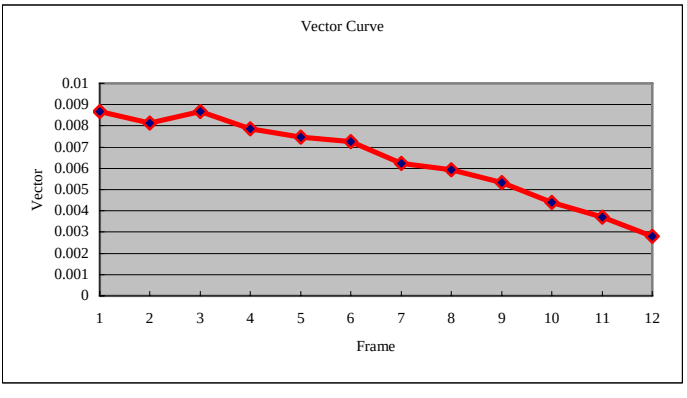


Fig. 8. The change curves of the depth motion vector of the car

From the experiments and analysis, we can find that our method is effective in depth motion estimation. The depth motion can be simply and correctly estimate from the monocular video using this method. And we can quantitatively analyze the depth motion in video.

5 Conclusion

Motion is one of the most important cues in video. And depth motion is very important in motion. In this paper, based on the human visual perception, we propose a new depth motion estimation method. This method is simple, effective and it accord with human visual perception. The novelty is that this method can directly estimate the

DM from the monocular video. In this paper, we tentatively introduce the perceptual constancy theory and depth motion perception theory into the depth motion estimation and achieve a good result. It may be a novel effective way to solve some problems in computer vision with visual perception theory.

References

1. K. R. Namuduri: Motion Estimation Using Spatio-Temporal Contextual Information. IEEE Trans. on Circuits and Systems Video Technology (2004) 1111– 1115.
2. H. So, J. Kim, W.-K. Cho and Y.-S. Kim: Fast motion estimation using modified diamond search patterns. ELECTRONICS LETTERS (2005) 62-63.
3. Ce Zhu, Xiao Lin, and Lap-Pui Cha: Hexagon-based search pattern for fast block motion estimation. IEEE Trans. on Circuits and Systems Video Technology (2002) 349-355.
4. Yan Huang, Xinhua Zhuang: Optic Flow Field Segmentation and Motion Estimation Using a Robust Genetic Partitioning Algorithm.IEEE Trans. On Pattern Analysis and Machine Intelligence (1995) 1177-1190.
5. Zarian Myles, Niels da Vitoria Lobo: Recovering Affine Motion an Defocus Blur Simultanteously. IEEE Trans. On Pattern Analysis and Machine Intelligence (1998) 652-658.
6. Antonis A. Argyros, Stelios C. Orphanoudakis: Independent 3D Motion Detection Based on Depth Elimination in Normal Flow Fields. IEEE Conf. on Computer Vision and Pattern Recognition (1997) 672-677.
7. Qigang Gao: A Computation Model for Understanding Three-Dimensional View Space. IEEE conf. on ICSMC(1996) 941-946.
8. Qigang Gao, Andrew K. C. Wong, Shang-Hua Wang: Estimating Face-pose Consistency Based on Synthetic View Space. IEEE Trans. on system, man, cybernetics(1998) 1191-1199.
9. I. Rock: Perception. Scientific American Books, Inc. (1984).
10. Sture Eriksson: Depth motion sensitivity functions. Psychological Research (2000) 41-68.

Adaptive Data Retrieval for Load Sharing in Clustered Video Servers

Minseok Song

School of Computer Science and Engineering,
Inha University, Korea
mssong@inha.ac.kr

Abstract. Increasing the number of concurrent streams while guaranteeing jitter-free operation is a primary issue for video servers. Disks storing popular videos tend to become overloaded, preventing the server accommodating more clients due to the unbalanced use of bandwidth. We propose an adaptive data retrieval scheme for load sharing in clustered video servers. We analyze how the data retrieval period affects the utilization of disk bandwidth and buffer space, and then develop a robust period management policy to satisfy the real-time requirements of video streams. We go on to propose a new data retrieval scheme in which the period can be dynamically adjusted so as to increase the disk bandwidth capacity of heavily loaded clusters and increase the number of clients admitted. Simulations demonstrate that our scheme is able to cope effectively with dynamically changing workloads and enables the server to admit many more clients.

1 Introduction

Recent advances in multimedia and network technologies make it possible to provide video-on-demand (VOD) services to clients. VOD services require video servers capable of transmitting videos to thousands of clients. Due to the high bandwidth and large storage requirements of video data, video servers are typically built on disk arrays which may consist of hundreds of disks. A key concern in the design of servers is how to place and retrieve video data so that the number of concurrent streams is maximized.

To support multiple clients, round-based scheduling is generally used for data retrieval: time is divided into equal-sized periods, called rounds, and each admitted client is served once in each round [8]. To guarantee continuous playback, the data required for current playback is read from the disk in the previous round, while the data to be played in the next round is read during the current round. Otherwise, playback will be distorted or there will be a pause due to the violation of the timing constraints of the video data [8]. We refer to this phenomenon as jitter.

To utilize disk bandwidth effectively, a video object is usually partitioned into segments and distributed over multiple disks. We will refer to this scheme as striping, and a segment is the maximum amount of contiguous data that is stored on a single disk. Striping over a large number of disks involves additional complexity and can lead to reliability problems [6]. Therefore, it is practical to limit the scope of striping by partitioning a disk array into several clusters, each of which independently forms a striping group, and each video is then striped within a single cluster [2,3].

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 207–216, 2007.
© Springer-Verlag Berlin Heidelberg 2007

Typically clients' requests tend to be highly skewed to a small popular videos, so the clusters containing copies of popular videos receive more requests than other clusters and are thus more likely to be overloaded [10]. Since it is difficult accurately to predict the access probabilities of videos requested by future clients, it is not possible to place video data in a balanced way. To make matters worse, request patterns may change on an hourly basis [6]. For instance, children's videos are likely to be popular early in the evening, but are seldom requested late at night.

To achieve load balancing in clustered video servers, existing approaches use dynamic data migration or replication [4,6,10]. The main idea of this is to move or copy video data among clusters so as to achieve a certain degree of load balancing. But these schemes require more resource such as additional disk bandwidth and storage overhead for migration or replication, which may cause downtime or have an impact on the performance of applications accessing the VOD system. In addition, such migration needs to be based on estimates of clients' future requests, and incorrect predictions may greatly degrade the system performance. These approaches are therefore less than adequate for vide servers.

We propose an adaptive data retrieval (ADR) scheme to achieve load sharing among clusters without data movement or replication. We will first analyze how the round length affects disk bandwidth and buffer utilization. We will then examine the condition for jitter-free adjustment of round length. We will go on to propose a new data retrieval scheme in which the round length assigned to each cluster is automatically reconfigured in such a way as to reduce the disk bandwidth utilization of heavily loaded clusters with the aim of increasing the number of concurrent clients.

The rest of this paper is organized as follows: We explain the system model in Section 2. We propose an adaptive data retrieval scheme in Section 3. We validate the proposed scheme through simulations in Section 4, and conclude the paper in Section 5.

2 System Model

Our server partitions a disk array into clusters, each of which independently forms a striping group [3]. Suppose that each cluster consists of Q homogeneous disks and that the number of clusters is C where D_k^i denotes the i^{th} disk of cluster k. Figure 1 shows an example of the server with $Q = 4$ and $C = 2$; video V_i is divided into a finite number of sequential segments ($S_{i,1}$, $S_{i,2}$, ...). In Fig. 1, V_1 is stored in cluster 1 and V_2 in cluster 2.

cluster 1				cluster 2			
D_1^1	D_1^2	D_1^3	D_1^4	D_2^1	D_2^2	D_2^3	D_2^4
$S_{1,1}$	$S_{1,2}$	$S_{1,3}$	$S_{1,4}$	$S_{2,1}$	$S_{2,2}$	$S_{2,3}$	$S_{2,4}$
$S_{1,5}$	$S_{1,6}$	$S_{1,7}$	$S_{1,8}$	$S_{2,5}$	$S_{2,6}$	$S_{2,7}$	$S_{2,8}$
...	...	...	...	...	...	...	...

Fig. 1. An example of data placement in a clustered video server with $Q = 4$ and $C = 2$

When a client requests a video stream, the server allocates a certain amount of buffer space to store a portion of the video. The server also allocates disk bandwidth for each stream to retrieve video data from disks continuously. To reduce the inevitable overhead of disk seek time, we use SCAN scheduling, in which the disk head scans back and forth across the surface of the disk and retrieves blocks as they are passed [1].

3 Adaptive Data Retrieval

3.1 Main Idea

The choice of round length is important because it effectively determines the maximum number of concurrent users [5,9]. A long round improves the efficiency of disk utilization because retrieving a large amount of data during a single round reduces the impact of disk latency; but a long round increases buffer requirements because a lot of buffer space is needed to store the data retrieved [9].

In an ADR scheme, each cluster has its own round length and buffer space. To overcome the stress on disk bandwidth caused by a heavily loaded cluster, ADR increases its round length. This increases the requirement for buffer space, but we can steal buffer space from under-utilized clusters. However, the total buffer space is a limited resource, and, in addition, the process of changing the round length may produce jitter. In order to perform this balancing act effectively, we first need to know how resource utilization varies with round length, which is the topic of the next subsection.

3.2 Variation of Resource Utilization with Round Length

Changing the round length may incur an additional seek overhead because the data retrieved during the new round may not be stored contiguously. To remedy this, we split each data segment $S_{i,m}$ into NS sub-segments $ss^n_{i,m}$ $(n = 1, ..., NS)$, where the size of each sub-segment corresponds to the data retrieved during a basic round of length BR. The NS sub-segments are stored contiguously to make up a segment, and segments are placed in round-robin fashion, as depicted in Fig. 1.

We will use dv_j to denote the j^{th} divisor of NS $(j = 1, ..., ND)$, where ND is the number of divisors of NS, and the set of feasible round lengths FS is $\{fr_j | fr_j = BR \times dv_j\}$. We will assume that the elements of FS are sorted in ascending order. For example, if $NS = 6$, then FS is $\{fr_1 = BR, fr_2 = 2BR, fr_3 = 3BR, fr_4 = 6BR\}$. Round lengths outside FS are not allowed because they might require two seeks for one read. For instance, suppose we were to select $4BR$ as the round length. and that our server then accesses D^3_1 to retrieve $S_{1,3}$ using the placement in Fig. 1. If $NS = 6$ and the 6 sub-segments from $ss^1_{1,3}$ to $ss^6_{1,3}$ constitute a segment of $S_{1,3}$, then the server would retrieve the four contiguous sub-segments, $ss^1_{1,3}$, $ss^2_{1,3}$, $ss^3_{1,3}$ and $ss^4_{1,3}$, from D^3_1 during one round. But, during the next round, the server would need to access the two disks D^3_1 and D^4_1, because $ss^5_{1,3}$ and $ss^6_{1,3}$ are stored on D^3_1, while $ss^1_{1,4}$ and $ss^2_{1,4}$ are stored on D^4_1.

Let us see how the buffer and disk bandwidth requirements for a video V_i with a data rate of dr_i bits/sec, depends on the round length, fr_j, $(j = 1, ..., ND)$. We will

use a typical seek time model [1] in which a constant seeking overhead (seek time + rotational delay) of T_s is required for one read of contiguous data. Retrieving a video stream V_i incurs an overhead of T_s and a reading time of $fr_j \times \frac{dr_i}{tr}$, where tr is the data transfer rate of the disk. As a consequence, servicing a video stream V_i increases the service time of $T_s + fr_j \times \frac{dr_i}{tr}$. Suppose that a client CL_i^m requests a video stream V_i. We can then partition the clients into C client groups (CG_1,...,CG_C) where the clients in group CG_k receive streams from cluster k. If the round length is fr_j, then we can obtain the total service time $ST_k(j)$ for cluster k, as follows:

$$ST_k(j) = \sum_{CL_i^m \in CG_k} (T_s + fr_j \times \frac{dr_i}{tr}). \tag{1}$$

For ease of exposition, we are assuming that disk loads are evenly distributed across disks in the same cluster. Such load balancing can easily be achieved by delaying the admission of clients [9]. The bandwidth utilization for a disk is usually defined as the ratio of the total service time to the round length [1]. Since there are Q disks in a cluster, we can now determine the disk bandwidth utilization $DS_k(j)$ for cluster k, as follows:

$$DS_k(j) = \frac{ST_k(j)}{fr_j \times Q}. \tag{2}$$

Let B be the total buffer size. Since double buffering is used for SCAN scheduling [1], servicing a video stream V_i increases the buffer utilization by $\frac{2 \times fr_j \times dr_i}{B}$. We can now obtain the buffer utilization $BS_k(j)$ for cluster k, as follows:

$$BS_k(j) = \sum_{CL_i^m \in CG_k} \frac{2 \times fr_j \times dr_i}{B}. \tag{3}$$

3.3 Conditions for Jitter-Free Round Adjustment

Increasing the round length may lead to data blocks failing to arrive on time, which will cause jitter in some video streams. This occurs because too few data blocks have been read to support the new round length. Fig. 2 shows the retrieval and playback of streams if $Q = 1$, $fr_{(j+1)} = 2 \times fr_j$, and the round length changes from fr_j to fr_{j+1} at $(m + 2) \times fr_j$. To maintain a jitter-free service, the data retrieved during a round should become available for playback during the next round. For example, in Fig. 2 (a), every item of data consumed between the times $m \times fr_j$ and $(m + 1) \times fr_j$ needs to be available in the buffer at $m \times fr_j$. Fig. 2 (a) illustrates how lengthening the round produces jitter in ongoing streams. We observe from the figure that the ongoing streams are starved of data from $(m + 3) \times fr_j$ to $(m + 4) \times fr_j$ because the blocks required by streams S3 and S4 are not available at $(m + 3) \times fr_j$.

Fig. 2 (b) illustrates a situation in which the round extension does not cause jitter. This is because every item of data that needs to be played between $(m + 3) \times fr_j$ and $(m+5) \times fr_j$ is available at $(m+3) \times fr_j$. From these observations, we can easily see that, in a cluster composed of a single disk, the service time for the new round length fr_{j+1} must not exceed fr_j to ensure jitter-free operation. Since there are Q disks, and disk

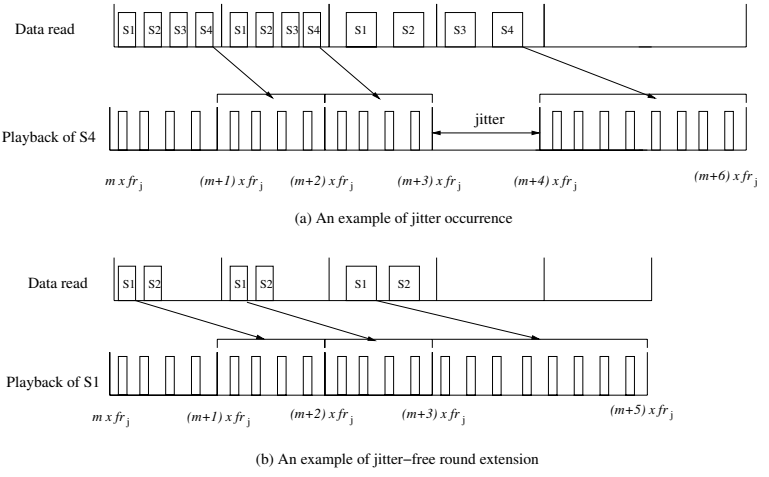


Fig. 2. Relationahip between round extension and jitter

loads are evenly distributed across disks in the same cluster, we obtain the following condition for jitter-free round extension from fr_j to fr_{j+1}, $(j = 1, ..., ND - 1)$:

$$ST_k(j+1) \leq fr_j \times Q. \tag{4}$$

Using Equation (2), this inequality can be rewritten as:

$$DS_k(j+1) \leq \frac{fr_j}{fr_{j+1}}. \tag{5}$$

The server may need to decrease the round length in order to acquire buffer space. Since buffer space is acquired at the cost of disk bandwidth, the server needs to check the disk bandwidth constraint: decreasing the round length from fr_{j+1} to fr_j should satisfy $DS_k(j) \leq 1$ $(j = 1, ..., ND - 1)$. Fig. 3 illustrates data retrieval and playback assuming that $Q = 1$, $fr_{(j+1)} = 2 \times fr_j$ and that the round length decreases from fr_{j+1} to fr_j at $(m+2) \times fr_j$. From the figures, we observe that: (1) data retrieval for the reduced round length fr_j starts at $(m+3) \times fr_j$ and (2) decreasing the round length from fr_{j+1} to fr_j does not cause jitter because all the data played between $(m+4) \times fr_j$ and $(m+5) \times fr_j$ is available at $(m+4) \times fr_j$.

3.4 An Adaptive Data Retrieval Scheme

Let SL_k be a selection parameter indicating that the SL_kth element of FS, fr_{SL_k}, is selected as the round length for cluster k. For example, if $FS = \{BR, 2BR, 3BR, 6BR\}$ and $SL_k = 2$, then $2BR$ is selected as the round length. From Equations (1), (2) and (3), we can easily see that higher values of fr_j decrease $DS_k(j)$ but increase $BS_k(j)$. From these observations, an ADR scheme adjusts the round length in the following way: (1) To cope with the requirement of heavily loaded clusters for disk bandwidth, relatively high values of SL_k are assigned to them. (2) To acquire buffer space to increase the

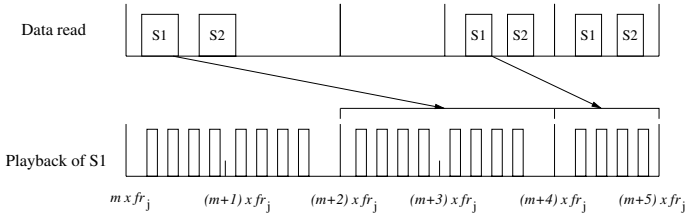


Fig. 3. Data retrieval and playback when the round length decreases

round length of heavily loaded clusters, relatively low values of SL_k are assigned to the lightly loaded clusters. The difficulty here is to decide exactly when the round lengths can be changed in a jitter-free way. We use a threshold-based approach in which changes of round length are triggered by changes in the disk bandwidth utilization. In effect, this means that the round length may be changed when a client requests or closes a video stream. We now explain what happens in each case.

When a client requests a video stream. The server maintains the set of utilizations $DS_k(j)$ and $BS_k(j)$ for every cluster k, $(k = 1, ..., C)$. When a client requests a video stream stored in cluster p, the server recalculates $DS_p(j)$ and $BS_p(j)$, $(j = 1, ..., ND)$, assuming that the new client will be admitted. The server then extends the round length if necessary, and goes on to check whether there is sufficient disk bandwidth and buffer space for the new client. The admission decision procedure is given as Algorithm 1.

For ease of exposition, we will assume that $ND \geq 2$. First it must verify that SL_p is less than ND, or the round length cannot be escalated. If the disk bandwidth utilization of a cluster exceeds a threshold value, then the server extends the round length to reflect the increased load. For this purpose, the server checks whether $\frac{fr_{SL_p}}{fr_{SL_p+1}} - \frac{\epsilon}{fr_{SL_p}} < DS_p(SL_p + 1) \leq \frac{fr_{SL_p}}{fr_{SL_p+1}}$ where ϵ represents the maximum service time for a client $CL_i^m \in CG_p$ (i.e. $\max_{CL_i^m \in CG_p}(T_s + fr_{SL_p} \times \frac{dr_i}{tr})$) (line 5). From Inequality (5), we can see that this condition guarantees jitter-free round extension. If the condition is satisfied, then the server checks whether enough buffer space is available for the round extension (line 6). If the buffer space is sufficient, then the server increases the value of SL_p by 1 (line 7). Otherwise, the value of SL_p does not change.

Next the server checks whether the new client can be admitted, although this is unnecessary if the round length has been extended; in that case, the admission criteria are already satisfied because $DS_p(SL_p + 1) \leq \frac{fr_{SL_p}}{fr_{SL_p+1}}$ and $\sum_{k=1}^{C} BS_k(SL_k) \leq 1$. To decide on admission, the server first checks whether $DS_p(SL_p) \leq 1$ (line 10). If this condition is violated, then the new client is rejected due to a lack of disk bandwidth. Otherwise, the server checks whether $\sum_{k=1}^{C} BS_k(SL_k) \leq 1$ (line 11). If this condition is satisfied, then the new client is admitted. Otherwise, the server may acquire buffer space for the new client by decreasing the round length of a cluster. The ADR scheme examines the clusters with the largest rounds (i.e. $SL_k = ND$), because they use the most buffer space. Let SM be the set of disk utilizations $DS_m(ND)$, $(m = 1, ..., C)$ whose SL_m is ND. If $SM = 0$, then the new client is rejected due to lack of buffer

Algorithm 1. Admission Decision Procedure

1: Set of utilizations of $DS_k(j)$ and $BS_k(j)$, $(j = 1, ..., ND)$
2: A client requests a stream from cluster p, and the server recalculates $DS_p(j)$ and $BS_p(j)$;
3: A set of disk utilizations: $SM = \{DS_m(ND) | m = 1, ..., C \text{ and } SL_m = ND\}$;
4: BOOLEAN: FLAG ← FALSE;
5: **if** $SL_p \leq ND - 1$ and $\frac{fr_{SL_p}}{fr_{SL_p+1}} - \frac{\epsilon}{fr_{SL_p}} < DS_p(SL_p + 1) \leq \frac{fr_{SL_p}}{fr_{SL_p+1}}$ **then**
6: **if** $\sum_{k=1}^{C} BS_k(SL_k) + BS_p(SL_p + 1) - BS_p(SL_p) \leq 1$ **then**
7: $SL_p \leftarrow SL_p + 1$; { Round length increases }
8: **end if**
9: **end if**
10: **if** $DS_p(SL_p) \leq 1$ **then**
11: **if** $\sum_{k=1}^{C} BS_k(SL_k) \leq 1$ **then**
12: FLAG ← TRUE;
13: **else**
14: **while** FLAG = FALSE and $SM \neq \phi$ **do**
15: Find the lowest value, $DS_l(ND) \in SM$;
16: $SM \leftarrow SM - \{DS_l(ND)\}$;
17: **if** $DS_l(ND - 1) \leq 1$ **then**
18: $SL_l \leftarrow ND - 1$;
19: FLAG ← TRUE;
20: **end if**
21: **end while**
22: **end if**
23: **end if**
24: **if** FLAG = TRUE **then**
25: The new client passes the admission test;
26: **else**
27: The new client is rejected;
28: **end if**

space. Otherwise, to seek the most lightly loaded cluster in SM, the server finds the smallest value of $DS_l(ND)$ in SM and removes it from SM (lines 15-16). If that does not violate the disk bandwidth constraint (line 17), then the server reduces the value of SL_l from ND to $ND - 1$ (lines 18-19). This one reduction in round length will accommodate the new client.

When a client closes a video stream. By decreasing the round lengths in a timely manner, the server is able to obtain buffer space that can subsequently be used to extend the round length of heavily loaded clusters. For this purpose, when a client closes the video stream stored in cluster k, the server recalculates $DS_k(j)$ and $BS_k(j)$, $(j = 1, ..., ND)$, and then checks whether $DS_k(SL_k) \leq \frac{fr_{SL_k-1}}{fr_{SL_k}} - \frac{\epsilon}{fr_{SL_k-1}}$, $(SL_k = 2, ..., ND)$. If this condition is satisfied, then the server checks whether decreasing the round length would violate the disk bandwidth constraint (i.e. $DS_k(SL_k - 1) \leq 1$). If this constraint can be maintained, then the server reduces the value of SL_k to $SL_k - 1$.

4 Experimental Results

To evaluate the effectiveness of the ADR scheme, we simulated a server with 40 IBM Ultrastar36Z15 disks whose tr and T_s are 55 MB/s and 11.2 ms, respectively [7]. The server is divided into 10 clusters, each of which composed of 4 disks. The arrival of client requests is assumed to follow a Poisson distribution. We also assume that all

videos are 90 minutes long, and have the same bit-rate of 1.5Mb/sec, which is typical of MPEG 1 playback. NS is set at 6 and FS is $\{BR = 0.5, 2BR = 1, 3BR = 1.5, 6BR = 3\}$. The total buffer size B is 2GB. The cluster location of each movie is chosen randomly. Let $p_i(t)$ be the access probability of video V_i $(i = 1, ..., NV)$ at time t, where NV is the number of movies and $\sum_{i=1}^{NV} p_i(t) = 1$. We consider two service scenarios:

1. Gradual change of popularities (GCP): The access probability follows a Zipf distribution, where $p_i(0) = \frac{1}{\sum_{m=1}^{NV} \frac{1}{m^{(1-\theta)}}} \times \frac{1}{i^{1-\theta}}$. We set $\theta = 0.0$, which corresponds to the real measurement value for a real VOD application [2]. Initially, $p_1(0) > ... > p_{NV}(0)$. At time τ, $p_{(i+1) \bmod NV}(\tau) = p_i(0)$. The popularities of the movies rotate over time with a period of τ so that, for example, $p_{(i+1) \bmod NV}((j+1) \times \tau) = p_i(j \times \tau)$. This pattern might correspond to changing types of audience at different times of the day [6].
2. Drastic change of popularities (DCP): In this case, the popularity of one movie drastically increases to α during every period of τ. We set $p_1(0) = \alpha$. Then the popularities of movies rotate with a period of τ. For example, at time τ, $p_2(\tau)$ increases to α. This scenario corresponds to the periodic release of a new movie or drama [6].

We will now compare the ADR scheme with three conventional methods, CV1, CV2 and CV3, that do not allow adaptive round length adjustment. The round lengths of CV1, CV2 and CV3 are assumed to be 1, 1.5 and 3 seconds respectively. We then investigated the number of clients admitted over 20 hours under the GCP and DCP scenarios. The value of τ is assumed to be 30 minutes. Fig. 4 shows how the proportion of clients, the admission ratio, depends on the arrival rate of client requests under GCP. In this case, the ADR scheme outperforms the conventional schemes under all workloads, admitting between 1% and 28% more clients.

Figs. 5 and 6 show how the admission ratio depends on the value of α under the DCP scenario when the arrival rates of 20 and 30 requests/minute. The figures show that ADR

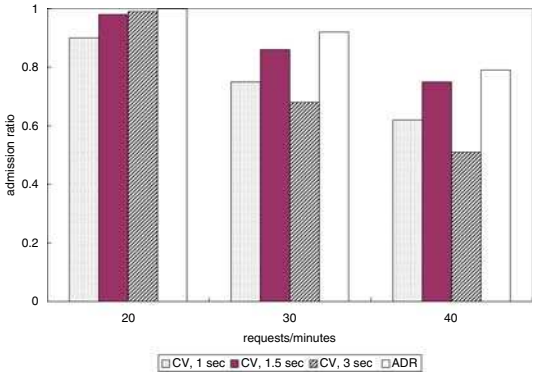


Fig. 4. Admission ratio against arrival rate

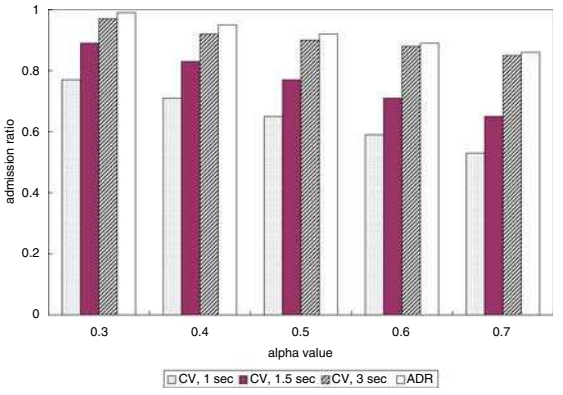


Fig. 5. Admission ratio against α, (arrival rate = 20 requests/minute)

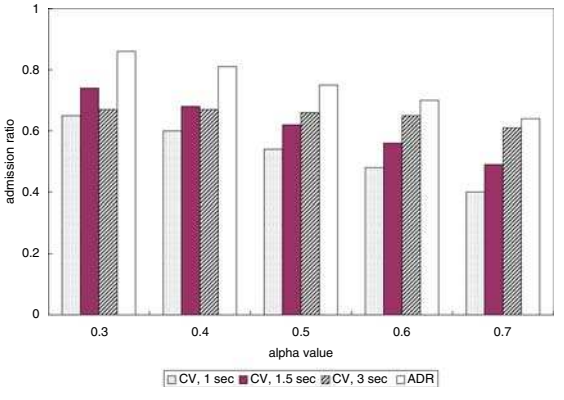


Fig. 6. Admission ratio against α, (arrival rate = 30 requests/minute)

exhibits the best performance under all workloads. It admits between 21% and 33% more clients than CV1, between 10% and 21% more than CV2, and between 1% to 19% more than CV3. From the figures, we observe that the performance gap between ADR and CV2 increases as the value of α increases, and this can be explained as follows: As the value of α increases, the requests become concentrated on to popular clusters so that the disk bandwidth becomes saturated within a shorter time. We also observe that CV3 performs up to 3% worse than ADR when the arrival rate is 20 requests/minute, but up to 21% worse when the arrival rate is 30 requests/minute. This can be explained as follows: When the arrival rate is 20 requests/minute, the buffer resource does not matter in most cases, so CV3 only performs slightly worse than ADR. But, as the arrival rate increases, the server needs more buffer space to accommodate more clients, but this effect is more pronounced with the CV3 scheme, which leads to the increased performance gap.

5 Conclusions

We have proposed a new data retrieval scheme for load sharing in clustered video servers. We analyzed how the round length influences the utilization of disk bandwidth and buffers and provided conditions for jitter-free round adjustment. We then went on to propose an adaptive data retrieval scheme in which the data retrieval period can be changed in a jitter-free way so as to give more disk bandwidth to heavily loaded clusters with the aim of increasing the total number of clients admitted. Experimental results show that our scheme enables the server to admit a much larger number of clients under dynamically changing workloads because it adaptively assigns a round length to each cluster, which leads to disk bandwidth and buffer space being utilized more effectively.

Acknowledgements

This work was supported by IITA through IT Leading R&D Support Project.

References

1. E. Chang. *Storage and Retrieval of Compressed Video*. PhD thesis, University of California at Berkeley, 1996.
2. C. Chou, L. Golubchik, and J. Lui. Striping doesn't scale: how to achieve scalability for continuous media servers with replication. In *Proceedings of the IEEE International Conference on Distributed Computing Systems*, pages 64–71, April 2000.
3. L. Golubchik, J. Lui, and M. Papadopouli. A survey of approaches to fault tolerant vod storage servers: Techniques, analysis, and comparison. *Parallel Computing*, 24(1):123–155, January 1998.
4. Y. Huang and C. Fang. Load balancing for clusters of vod servers. *Information Science*, 161(1-4):113–138, August 2004.
5. K. Lee and H. Yeom. A dynamic scheduling algorithm for large scale multimedia server. *Information Processing Letters*, 68(5):235–240, March 1998.
6. P. Lie, J. Lui, and L. Golubchik. Threshold-based dynamic replication in large-scale video-on-demand systems. *Multimedia Tools and Applications Journal*, 21(1):35–62, May 2000.
7. L. Reuther and M. Pohlack. Rotational-position-aware real-time disk scheduling using a dynamic active subset (das). In *Proceedings of IEEE RTSS*, pages 374–385, December 2003.
8. I. Shin, K. Koh, and Y. Won. Practical issues related to disk scheduling for video-on-demand services. *IEICE Transactions on Communications*, 88(5):2156–2164, May 2005.
9. M. Song and H. Shin. Replication and retrieval strategies for resource-effective admission control in multi-resolution video servers. *Multimedia Tools and Applications Journal*, 28(3):89–114, March 2006.
10. Y. Zhao and C. Kuo. Video server scheduling using random early request migration. *ACM/Springer Multimedia Systems Journal*, 10(4):302–316, April 2005.

A User-Friendly News Contents Adaptation for Mobile Terminals

Youn-Sik Hong, Ji-Hong Kim, Yong-Hyun Kim, and Mee-Young Sung

University of Incheon, Dept. of Computer Science and Eng.,
177 Dowha-dong Nam-gu,
402-749, Incheon, Korea
{yshong, riot999, yh-kim, mysung}@incheon.ac.kr

Abstract. We present a system that transforms web contents in the internet effectively to the corresponding mobile contents adapted to a mobile terminal such as a PDA or a mobile phone. The primary goal of this research is to reuse web contents in wireless internet environments without additional efforts of rebuilding them at scratch for contents adaptation to reduce costs and efforts needed to develop such wireless contents for mobile user. The secondary goal is to develop more convenient user interfaces to read mobile contents easily with a mobile terminal. To do this, we propose a technique, called *page splitting*, to navigate pages with button controls instead of conventional scroll up/down controls. The proposed system has been well operated for both well-known domestic and international news portal sites.

Keywords: contents adaptation, PDA, user interface, mobile web page, page splitting.

1 Introduction

With an explosion in use of the internet, there is enormous number of web pages and user often retrieves useful information in it. In nowadays, a number of people who prefer to use a mobile terminal (MT) such as a mobile phone or a PDA (Personal Digital Assistant) to access web contents over wireless internet have been increased rapidly. However, to this time, there are some restrictions on the use of such a terminal to access web contents. First of all, most of the contents have been built for desktop or laptop PC users. So it should be necessary to transform for contents adaptation or to rebuild its contents at scratch for mobile users. Second, a MT furnishes a poor user interface (narrow screen size, stylus pen, etc) compared to PC [1], [2], [3], [4], [5], [6], [7]. Thus, it should be needed to devise a mechanism to provide a more convenient user interface for it.

In this paper, we present a system called Pocket News that transforms web content in the internet effectively to the corresponding mobile content dedicated to a MT, particularly a PDA. The primary goal of this research is to reuse web contents in a wireless internet environment without any additional effort of rebuilding them at

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 217 – 226, 2007.
© Springer-Verlag Berlin Heidelberg 2007

scratch for contents adaptation. We confine our research scope to news contents, whose content is frequently added and updated. To process them in real-time, all of the pages extracted from a target web site are stored into a web cache as an intermediate storage for bookkeeping operation. Typically news content is not abruptly changed, but updated gradually. Thus, if we found a new web page by checking all of the hyperlinked pages of an index page for a target web site, then we can easily add it in a web cache, whereas the other pages remain unchanged.

The secondary goal is to develop more convenient user interfaces to read mobile contents easily with a MT. To do this, non-textual information of a web page is treated independently. It consists of a distinct mobile page hyperlinked to the mobile page which has textual information only. A text-only page is divided into shorter sub-pages depending on the screen size of a MT. A full text of a sub-page can be viewed at once in a displayable screen of a MT. Then we can navigate these sub-pages with button controls instead of complex scroll up/down controls.

The structure of this paper is as follows: In Section 2, we discuss about related works. The overall structure of the Pocket News is explained in Section 3. In Section 4, the snapshots of the running examples are shown. In addition, the comparison results between two distinct styles of user controls when to read mobile contents are given. Finally, we conclude this paper and discuss about further works.

2 Related Works

Several studies have been made on transforming web contents in the internet to the mobile contents adapted to a MT, with emphasis on mobile phone. Most of the works [9], [10], [11], [12] have been used XML as an intermediate language to generate mobile web pages. A typical process of transformation is depicted in Fig. 1.

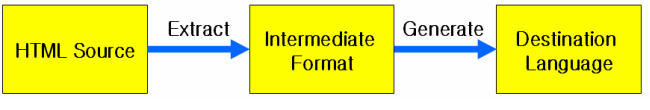


Fig. 1. A typical process of transformation through intermediate language

The implementation methods of such works have been classified into three categories: full-automatic, semi-automatic and manual. A semi-automatic method allows user interaction to improve the results of transformation. For example, user can make a user profile to give a tip while interpreting a meaning of HTML tag used in a web page. In Table 1, WebViews [4] and Web-based interactive document translator systems [12] from the related works are compared with the system to be proposed in this paper.

Table 1. The characteristics of the systems

System	Intermediate Format	Automatic/Manual Method	Target Markup Language
WebViews[4]	use (XML)	Semi-automatic	WML, VoiceXML
interactive system [12]	use (XML)	Manual	WML
The proposed system	not use	Full automatic	WML, mHTML

The most difficult thing for contents adaptation is that each of web sites adheres to its own presentation style. For example, to output its message to web browser, one uses <P> tag, whereas the other uses <TR> tag. Thus, it is possible for a strictly restricted web site to transform such a meaningless HTML tag into its structured XML tag [12] by just defining their relation manually. In other words, it is not worth for most of the web sites in the internet to transform its contents to their corresponding XML documents.

Thus, in this paper, our proposed system will try to transform a web page to the corresponding mobile page directly without going through any intermediate language like XML. With our approach, it is best fit for the frequently changed web sites, like news contents. Thus, we confined our research scope to real-time contents for contents adaptation.

Also some of the works have been studied about tag conversion between web pages in the internet and the corresponding mobile pages [8]. That is, it takes a markup language as an input and produces an output in either XHTML Basic (recently XHTML MP) or WML (or mHTML). It keeps the rules necessary for tag conversion. Each conversion rule has been already made for the corresponding tag. The advantage of the above system is to add conversion rules easily for a new markup language. However, it dose not guarantee the accuracy of transformation.

3 The Overall Structure of the Pocket News

The Pocket News system is an infrastructure network that integrates a wireless LAN (WLAN) based on the IEEE 802.11b/g standard. A PDA as a MT with a PCMCIA network interface card is connected to the WLAN through a base station (BS). In our test-bed network, BS is simply Access Point (AP) for simplicity.

In a standpoint of software implementation, it is running on two different sides: *server side* and *client side*. The Pocket News running on the server side loads and updates web contents for a target site periodically in a web cache. It transforms them into the corresponding mobile pages adapted to a MT. Then it transfers such pages to the terminal on receiving the request of a MT. It consists of the loader module and the parsing components as shown in Fig 2. The client-side Pocket News runs a user-interface program to read the mobile pages generated by the server-side Pocket News.

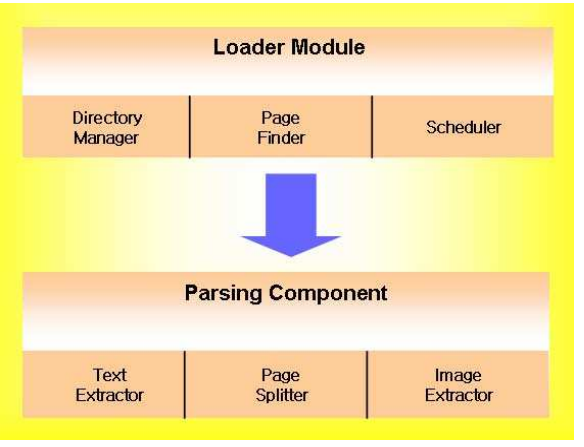


Fig. 2. The structure of the server side Pocket News

3.1 The Loader Module

If we specify the URL of a target web site (*i.e.*, the index page) to be transformed, it extracts all of the pages which have been hyperlinked to the index page and saves them in a web cache. Only the page stored in a web cache will be transformed into a corresponding mobile page. In the following, we will briefly explain each of the elements.

3.1.1 The Directory Manager

Basically, it creates a new directory (*i.e.*, web cache) whose name is the same as the URL of the web site to be transformed. In addition, it renames a relative URL of hyperlinked page (or hyperlinked image) used in a web page into the corresponding absolute URL. In that case of saving a web page in the directory, the unacceptable characters such as \, /, :, *, ?, <, >, | used in the URL of a page should be replaced with the underscore(_) for proper naming.

3.1.2 The Page Finder

First of all, it saves an index page into the directory which was created by the directory manager. Then, it retrieves all of the pages which are hyperlinked to the index page and saves them into the same directory.

3.1.3 The Scheduler

Because the contents of a target web site will be updated frequently, it visits the site periodically to check whether its contents are changed or not. If it happens, it overwrites the page just updated, whereas the other pages remain unchanged.

3.2 The Parsing Components

If a mobile page contains non-textual information like static images, it may not be possible to view a full-page at once with a MT. In that case, one manipulates scroll up/down controls to view the rest of it. This is a tedious work due to many user interactions. Besides, the size of an original image should be reduced for fitting the screen

size of a MT. To solve these, non-textual and textual information are extracted separately and each of them is stored into a distinct file. They can be connected implicitly with each other by sharing the name of the original page with them. In addition, to minimize user interactions, the full text extracted is divided into shorter sub-texts which are best fit into a displayable screen of a MT. Therefore, navigation between such pages can be done with simple button controls. Notice that during the stage of preprocessing unnecessary information, for example, frame, banner menu, advertisement, and style-sheet from each of the pages stored in the web cache is eliminated.

3.2.1 The Text Extractor

We assume that a web page is written in HTML without loss of generality. It extracts textual information only from a web page in HTML and saves them in a temporal file for splitting sub-pages. Note that tabular information still keeps its style after extraction. However, a tag described an image is replaced with a hyperlink tag to the page to be generated by the image extractor.

3.2.2 The Image Extractor

It extracts a static image from a given page and saves it in a distinct mobile page. A typical resolution of PDA (based on Compaq iPAQ3660) is 150×150. So, an original image should be reduced to fit this size. Thus, every image with its caption can be viewed at once in a displayable screen of a MT. The textual description about this image can be retrieved by clicking a hyperlink embedded in the image tag.

3.2.3 The Page Splitter

It splits a transformed page (text only) into two or more sub-pages depending on the screen size of a MT. When the font size of a PDA is 9, each sub-page generated by the page splitter has less than 380 characters except the first one. Because the first sub-page requires additional spaces to include the title of a page, it has less than 200 characters. With the font size of 10, the number of characters for the first and the rest sub-page does not exceed 180 and 350, respectively. Notice that these figures are obtained from our empirical results.

4 The Experimental Results

4.1 The Experimental Environments

Basically, the Pocket News system in the server side is running on Windows 2000 Server, where the system in the client side is running on Windows CE 3.0 (or higher). In the server side, it has two major modules as explained in Section 3: the loader module and the parsing components. The former has been implemented using Win32 API and the latter has been designed and implemented using c# and ASP.NET running on .NET Framework. MMIT (Microsoft Mobile Internet Toolkit) has been used to implement the client-side user-interface.

We have used Compaq iPAQ 3660/3850/5450 and Casio Cassiopeia E-125. A testbed wireless internet system has been built and tested with utilization of AP and



Fig. 3. The Sample News Contents (from the New York Times)

WLAN cards based on the 802.11b/g standard. A mobile phone as another MT has been tested with both OpenWave SDK 6.2.2 and Microsoft's Mobile Explorer.

4.2 The Snapshots of the Pocket News in the Server Side

4.2.1 The Results of the Loader module

Fig 3 shows the technology section of the New-York Times web site. It is selected as typical new contents for the demonstration of our proposed system. The loader module extracts all of the pages which are hyperlinked to the index page of the web site and saves them in a web cache.

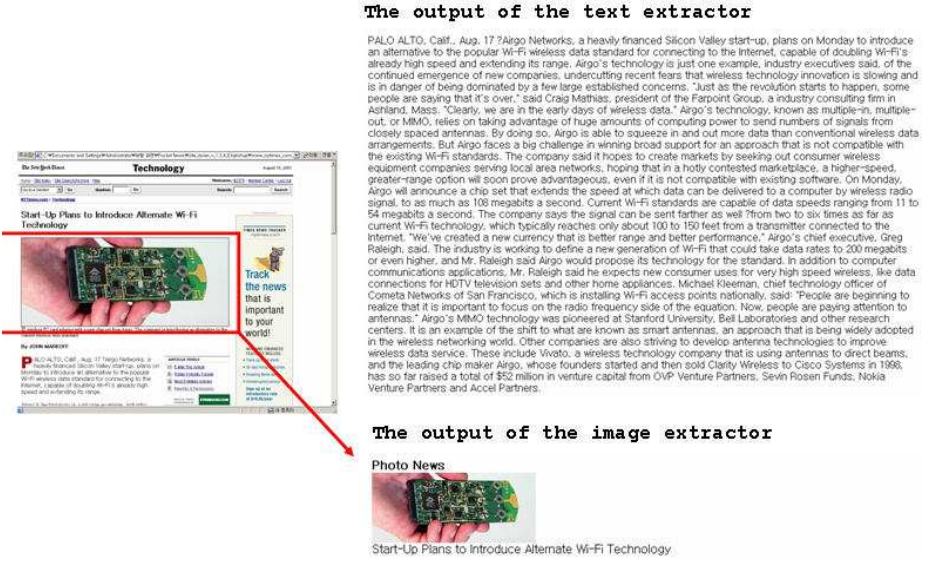


Fig. 4. The outputs of the parsing components



Fig. 5. The outputs of the page splitter

4.2.2 The Results of the Parsing Components

Fig 4 shows the two distinct outputs (*right-hand side*) from the original web page (*left-hand side*); one is the output of the text extractor which consists of text only and the other is the output of the image extractor which has the image and its caption. They can be referred to each other by assigning the name of the original page to their name. Notice that the transformed image will be slightly changed compared to the original image. The reason is that it should be adjusted to fit it into a single page, while considering the resolution of a given MT.

With the page splitter of the parsing components, the transformed text-only page is divided into two or more sub-pages depending on the screen size of a MT. Fig.5 shows the first mobile page after splitting (*bottom-side*) from the extracted page (*top-side*).

4.3 The Snapshots of the Pocket News in the Client Side

If we choose the first sub-title of the index page of the Technology section in the New York Times web site as shown in Fig.6, the first sub-page will become as shown in Fig. 6(b). Notice that it has two navigation buttons, instead of scroll bar on the right-most trail. If we click the next button, we'll see the 2nd sub-page.

It can be possible for the Pocket News to transform a web page into the corresponding mobile pages in mHTML or WML which is suitable for mobile phone.

4.4 Performance Assessment

It is obvious that there is a significant difference in the time of loading pages for a MT between with and without contents adaptation. With contents adaptation, a time elapsed to load a single mobile page generated for a PDA is 65 ms in average.

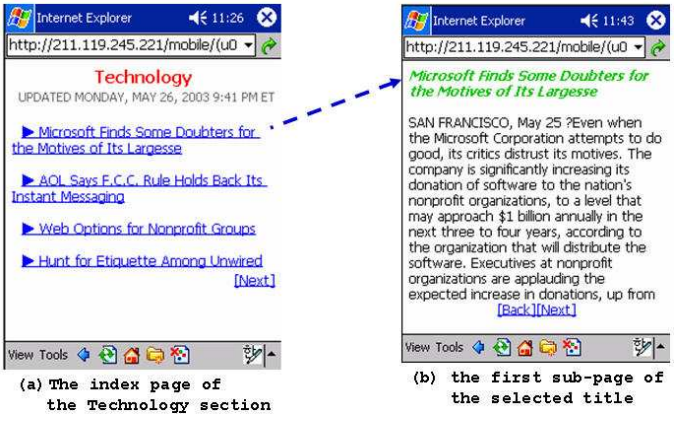


Fig. 6. The index page and its sub-page of the Pocket News with PDA

However, without adaptation it takes 3 to 5 times longer. We omit a detailed discussion about the page loading time due to the page limit.

One of the most distinguishable features of the Pocket News is to present more convenient user interfaces to read mobile contents easily with a MT. To do this, non-textual and textual information in a web page are treated separately and each of them is stored into a distinct file. To minimize user interactions, the full text extracted is divided into shorter sub-texts which are best fit into a single screen of MT. We have two great advantages by doing such a page splitting. First, a full-text of such a sub-page can be viewed at once without additional user interaction. Second, once a static image can be adjusted to fit into a displayable screen of a specific MT, no additional effort for an image adjustment will be needed.

Because of the narrow window for a MT and difficulty in controlling scroll bar by using stylus pen in a PDA or arrow key pads in a mobile phone we can say that using scroll bar control is inconvenient compared to navigation button controls. To verify this, we have made a series of tests to see which is better to read mobile contents with PDA.

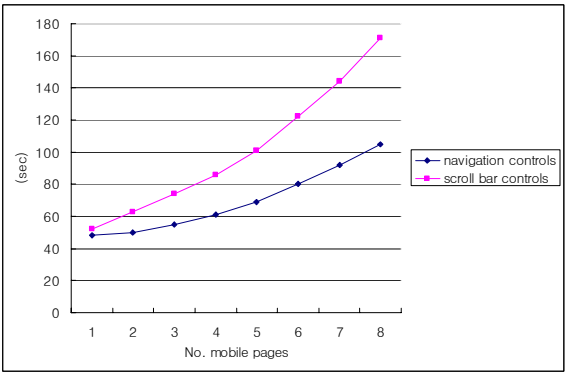


Fig. 7. The comparison of user manipulation time

In this experiment, each of 10 testers used his (her) own PDAs to read mobile contents. Two different news contents, entertainment news and sports news, are selected for this experiment. Note that the size of a page generated by the page splitter is about 1 Kbytes.

We have summarized the experimental results as shown in Fig 7. The data is obtained from the average of the elapsed time (in second) for the 10 trials. With the aid of the page splitting, manipulation of navigation button controls to read mobile pages with PDA is faster than with scroll bar controls by 8 to 39%. It becomes more efficient as the size of a mobile content becomes larger.

5 Conclusion and Further Works

We have presented a system called Pocket News that transforms web contents in the internet effectively to the mobile contents adapted to a mobile terminal such as a PDA or a mobile phone. We proved that it is feasible for this system to reuse web contents in a wireless internet environment without any additional effort of rebuilding them at scratch for contents adaptation. It enables us to reduce costs and efforts needed to develop wireless contents for mobile user. Our proposed system is adequate for frequently changed web sites, especially News contents. A series of experiments show that the proposed system has been well operated for well-known domestic and international new portal sites, including the New York Times.

We also proposed a technique, called page splitting, to navigate pages with button controls instead of scroll up/down controls to read mobile pages easily with a mobile terminal. With adaptation of the page splitting technique, manipulation of navigation button controls to read mobile pages is faster than with scroll bar controls by 8 to 39%. It becomes more efficient as the size of a mobile content becomes larger.

In the further works, we will add user profiles to the text extractor of the parsing components for improving its performance. It will keep heuristic rules for frequently used HTML tags to direct the way of translation.

Acknowledgement

This work was supported by the MOICE and ITEP through the Multimedia Research Center and partially supported by University of Incheon.

References

1. Bum-Ho Kim, Pyung-Soo Ma: A method to extract indexes that transform web contents for mobile terminals, Proceeding of the 29th KISS Fall Conference (domestic), Vol. 15, No. 4, (2002).
2. Corin R. Anderson, Pedro Domingos, and Daniel S. Weld: Personalizing Web Sites for Mobile Users, ACM, (2001).
3. Eun-Jeong Choi, Ji-Yeon Son and Dong-Won Han : Design of Multi-document Parsing System for Mobile Device, Proceeding of The 29th KISS Fall Conference (domestic), Vol. 15, No. 4, (2002).

4. In-Sook Park, et al.:A Design and Implementation of Web-Based Interactive Document Translator Systems, Proceeding of HCI 2002(domestic), (2002).
5. Juliana Freire, Bharat Kumar and Daniel Lieuwen, WebViews: Accessing Personalized Web Content and Services, WWW10, May 1-5, (2001), ACM.
6. K.Henricksen, J.Indilska: Adapting the Web Interface: An Adaptive Web Browser, IEEE, (2001).
7. Mi-Young Lee, el al.: A Design and Implementation of A Tag-Converter of the Wired/Wireless Document Conversion System, Proceeding of the 29th KISS Fall Conference(domestic), Vol. 15, No. 4, (2002).
8. Mike Perkowitz and Oren Etzioni, Adaptive Web Sites, Vol. 43 No. 8, Communications of the ACM, August (2000).
9. M. Perkowitz and Etzioni: Towards adaptive web sites: Conceptual framework and case study, Artificial Intelligence Journal, 118(1,2), (2000).
10. T.W.Bickmore and B.N.Schilit.Digestor: Device-independent Access to the World Wide Web, In Proceedings of the Sixth International World Wide Web Conference, (1997).
11. http://www.w3.org/Style/xsl
12. http://www.w3.org/TR/xslt11
13. http://www.w3.org/TR/2001/WD-di-princ-20010918/

An Efficient Predictive Coding of Integers with Real-Domain Predictions Using Distributed Source Coding Techniques*

Mortuza Ali and Manzur Murshed

Gippsland School of Information Technology, Monash University,
Churchill, Victoria 3842, Australia
{Mortuza.Ali, Manzur.Murshed}@infotech.monash.edu.au

Abstract. By exploiting the commonly observed Laplacian probability distribution of audio, image, and video prediction residuals, many researchers proposed low complexity prefix codes to compress integer residual data. All these techniques treated predictions as integers despite being drawn from the real domain in lossless compression. Among these, Golomb coding is widely used for being optimal with non-negative integers that follow geometric distribution, a two-sided extension of which is the discrete analogue of Laplacian distribution. This paper for the first time presents a novel predictive codec which treats real-domain predictions without rounding to the nearest integers and thus avoids any coding loss due to rounding. The proposed codec innovatively uses the concept of distributed source coding by replacing the reminder part of Golomb code with the index of the coset containing the actual value.

Keywords: Predictive coding, Prediction residual, Lossless coding, Laplacian distribution, Distributed source coding, and Coset.

1 Introduction

Predictive coding techniques [1] have found widespread applications in lossless coding of digital audio [2], image [3], and video [4] data. In linear predictive coding of an integer-valued source X, after having observed the past data sequence $x^{t-1} = (x_1, x_2, \ldots, x_{t-1})$, the value of x_t is predicted as a linear combination of the previous p values as

$$\hat{x}_t = \sum_{k=1}^{p} a_k x_{t-k} , \tag{1}$$

where a_k, $k = 1, \ldots, p$, are the predictor coefficients. Given a data sequence $x^n = (x_1, x_2, \ldots, x_n)$ and a fixed prediction order p, the most common way of

* This research was partially supported by Australian Research Council's Discovery scheme (Project Number: DP0666456).

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 227 – 236, 2007.
© Springer-Verlag Berlin Heidelberg 2007

selecting the predictor coefficients is to minimize the total squared prediction error such that

$$(a_1, a_2, \cdots, a_p) = \underset{(b_1, b_2, \cdots, b_p) \in \mathbb{R}^p}{\arg\min} \sum_{t=1}^{n} (x_t - \sum_{k=1}^{p} b_k x_{t-k})^2 . \tag{2}$$

Observe that even if the source values are integers, the predictor coefficients are drawn from the real domain, leading to continuous valued predictions. In non-linear prediction, if the predictions are constrained to integers, a real valued DC offset is typically present in the prediction residuals [3].

Real valued prediction residuals in audio, image, and video coding are commonly observed following Laplacian distribution [5] where the pdf of the residual ε can be modeled as

$$f_\lambda(\varepsilon) = \frac{\lambda}{2} e^{-\lambda|\varepsilon|}, \ \lambda > 0 . \tag{3}$$

Here, the parameter λ controls the two sided exponential decay rate. Substituting $e^{-\lambda}$ with θ in (3) yields

$$f_\theta(\varepsilon) = -\frac{\ln\theta}{2} \theta^{|\varepsilon|}, \ 0 < \theta < 1 . \tag{4}$$

Many researchers exploited this distribution to develop coding techniques that can encode and decode residuals very fast without using any temporary memory. All these codes, however, can handle only discrete values and consequently their application in lossless predictive coding demands rounding of the predictions to the nearest discrete value. In doing so, the prediction residuals are modeled by the two sided geometric distribution, which is analogous to Laplacian distribution in the discrete domain. Among these, Golomb codes [6] are optimal [7] for one sided geometric distribution of non-negative integers. Given a parameter *m*, Golomb code of a non-negative integer residual ε has two parts: quotient of ε / m in unary representation and the reminder of that division in minimal binary representation. Popular lossless codecs, e.g., JPEG-LS [3], use Golomb codes for compressing integer prediction residuals by mapping them into non-negative integers using the following *overlap and interleave* scheme prior to encoding [8]:

$$M(\varepsilon) = \begin{cases} 2\varepsilon, & \varepsilon \geq 0; \\ 2|\varepsilon| - 1, & \text{otherwise.} \end{cases} \tag{5}$$

In this scheme, the symmetry inherent in the original distribution is no longer handled properly as equally probable opposite signed integers get codes of different lengths. Moreover, Golomb codes being optimal for non-negative integers does not preclude improving predictive compression gain further if the residuals could be handled in real domain and thus avoiding any loss due to rounding.

This paper presents a lossless predictive prefix codec of integers where the Golomb codec is modified so that the otherwise independent prediction mechanism is infused

into both the encoding and decoding stages. This modification innovatively applies distributed source coding concept for the first time in such application by first partitioning the integer domain into *m* distinct cosets, each having *m* distance between successive members, and then replacing the reminder part of Golomb code with the index of the coset having the actual integer to be coded. Unlike Golomb codes, this technique thus treats two residuals of opposite sign alike as these are always coded by the same coset. Consequently, the mapping function in (5) is effectively transformed into $M(\varepsilon) = 2|\varepsilon|$ for any real valued residual ε.

The organization of the rest of the paper is as follows. In section II we present the concept of distributed source coding which form the basis of our algorithm presented in section III. Experimental results are presented in section IV. We conclude the paper in section V.

2 Distributed Source Coding Principle

Distributed Source Coding (DSC) refers to the compression of correlated sources which are not co-located, i.e., the encoders for the sources are independent and cannot communicate with each other to exploit the redundancy. The encoded bit-stream from each source is sent to a single decoder which operates on all incoming bit-streams and decodes the sources. To be more precise, let consider the distributed source coding problem for two discrete-alphabet sources *X* and *Y*. With separate encoders and decoders (see Fig. 1(a)) one can transmit *X* at a rate $R_X \geq H(X)$ and *Y* at a rate $R_Y \geq H(Y)$, where $H(X)$ and $H(Y)$ are entropies of *X* and *Y* respectively. However, with joint decoding (see Fig. 1(b)) we can do better than this. The information theoretic bound for this problem has been established by Slepian and Wolf in [9] which states that if the joint distribution of *X* and *Y* is known the achievable rate region is as shown in Fig. 1(c). It is surprising that we can have both $R_X < H(X)$ and $R_Y < H(Y)$, and the sum of the rates $R_X + R_Y$ can be equal to the joint entropy $H(X, Y)$ even though each encoder has access to its own source only.

Although the Slepian-Wolf theorem states the fundamental limits on communication efficiency, it is silent about how this could be achieved. However, Wyner indicated in [10] that coding of correlated sources with separate encoders but

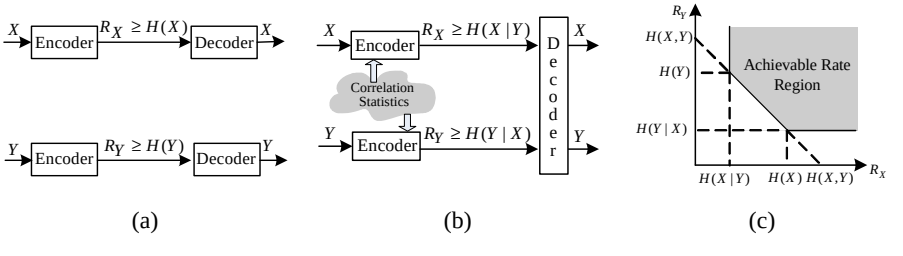


Fig. 1. (a) Separate encoders and separate decoders; (b) separate encoders and joint decoder; and (c) achievable rate region for Slepian Wolf coding

joint decoder can be approached using channel coding concepts. The main idea is to divide the source data space into a finite number of cosets using channel coding techniques, where the distance between any pair of elements in each coset is maintained greater than twice of the correlation noise in the data set. Compression of the scheme stems from transmitting only the coset index instead of the actual value. The decoder is then able to extract the actual value from the given coset as long as some form of side information is already available at the decoder such that the distance between the actual value and the side information is less than half of the minimum distance between any pair of elements in the coset.

Although distributed source coding refers to the compression of correlated sources which are not co-located, the same coset based technique can be used to compress correlated as well as co-located sources. In order to clarify concept of DSC techniques in coding correlated as well as co-located sources let consider the predictive coding technique which rounds the prediction $\hat{x}$, available both at the encoder and the decoder, to the nearest integer $\tilde{x}$. Let $e = x - \tilde{x}$ and $|e| \leq 1$ represents the underlying correlation statistics. Then $E = \{-1,0,1\}$ is the finite set of possible residual values. If these possible values are considered equally-likely, the entropy encoder takes $\lceil \log_2 3 \rceil$ bits to encode the residual. The value of x is then decoded by simply adding the residual to the rounded predicted value since $x = \tilde{x} + e$. Now assume that the universe of integers $\mathbf{Z}$ is divided into three cosets $\Psi_0 = \{0,3,6,\cdots\} = 3\mathbf{Z}$, $\Psi_1 = \{1,4,7,\cdots\} = 3\mathbf{Z}+1$, and $\Psi_2 = \{2,5,8,\cdots\} = 3\mathbf{Z}+2$. Instead of coding the residual, DSC encodes the value of x by simply coding the index of the coset containing x, which also takes $\lceil \log_2 3 \rceil$ bits. Here it may be noted that the encoder need not to know the predicted value $\hat{x}$. As the minimum distance between any pair of integers in any of the three cosets is $d_{\min} = 3$, the value of x can then be decoded by retrieving the value in the indexed coset nearest to $\tilde{x}$, since $|e| \leq 1 < (d_{\min}/2)$ (see Fig. 2).

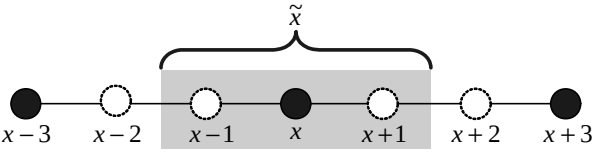


Fig. 2. x is nearest to $\tilde{x}$ in the indexed coset

Although the conventional entropy coder has to round $\hat{x}$ to the nearest integer $\tilde{x}$, DSC techniques can potentially work without rounding as explained in the following example and in doing so can avoid compression efficiency loss due to rounding. Let $\varepsilon = x - \hat{x}$ and the underlying correlation be $|\varepsilon| < 1$. Since conventional coding technique first rounds $\hat{x}$ to the nearest integer, in this case also $E = \{-1,0,1\}$ is the

finite set of possible residual values and if all the residual values are equally likely, conventional entropy coder takes $\lceil \log_2 3 \rceil$ bits to encode. However, DSC techniques based coder partitions $\mathbf{Z}$ into two cosets $\Psi_0 = \{0,2,4,\cdots\} = 2\mathbf{Z}$ and $\Psi_1 = \{1,3,5,\cdots\} = 2\mathbf{Z}+1$ with $d_{\min} = 2$, and encodes x by simply encoding the index of the coset containing x which requires $\lceil \log_2 2 \rceil$ bit. The value of x can be decoded by retrieving the value nearest to $\hat{x}$ in the indexed coset, since $|\varepsilon| < 1 = d_{\min}/2$. Thus, in this case gain due to avoiding round is $\lceil \log_2 3 \rceil / \lceil \log_2 2 \rceil$.

3 DSC Based Residual Coding

Although both the conventional coding techniques and the DSC techniques theoretically have the same compression efficiency, in practice one coding technique may have advantage over the other. In contrast to the conventional coding techniques, we present in this section a DSC based technique to code integers with Laplacian prediction residuals that has the advantage of working without rounding.

3.1 Encoding

For a given positive integer parameter *m*, consider the coding of x with prediction $\hat{x}$ available both at the encoder and the decoder. Let partition the set of integers $\mathbf{Z}$ into *m* cosets $C(i)$, $0 \le i \le m-1$, each coset being an equivalence class modulo *m*, i.e., $C(i) = \{z : z \in \mathbf{Z} \wedge z \equiv i (\mathrm{mod}\ m)\}$. The minimum distance between any pair of integers in each coset is at least *m*. Now let $\varepsilon = x - \hat{x}$, $d = |\varepsilon|$, and $h = m/2$. If $\hat{x}$ is at a distance less than *h* from $x \in C(i)$, i.e., if $d < h$ the decoder can recover x by finding the integer nearest to $\hat{x}$ in $C(i)$. However, to handle arbitrary value of *d*, we define another index *j* such that $hj \le d < h(j+1)$ or equivalently $j = \lfloor d/h \rfloor$. In the next section it will be shown that these two indices (*i*, *j*) are sufficient for decoding x if $hj < d < h(j+1)$ and the case $d = hj$ can easily be handled with one additional bit. However, for real valued random variable the probability of a particular value is zero and thus this extra bit overhead does not affect the average code length.

In this scheme the index *j* is coded in unary and the coset index *i* is coded in minimal binary. In minimal binary coding of an non-negative integer *i* from the alphabet {0, 1, …, *m*-1}, $\lfloor \log_2 m \rfloor$ bits are used to code *i* when $i < 2^{\lceil \log_2 m \rceil} - m$ and $\lceil \log_2 m \rceil$ bits are used otherwise. Here shorter codes are given to the integers at the beginning of the alphabet assuming that probabilities of the integers are non-increasing. Since, according to Laplacian distribution the probabilities of the integers nearer to $\hat{x}$ are higher, the probabilities of the coset indices of the integers nearer to $\hat{x}$ are also higher. Thus for better compression efficiency the coset indices are

ordered according to their distances from $\hat{x}$ and instead of coding the coset index i its position index k in the ordered set is coded in minimal binary. Now the encoding algorithm can summarized as follows.

Algorithm 3.1 (k, j, b) = **ENCODE**$(m, x, \hat{x})$

1. $d = |x - \hat{x}|$
2. $h = m/2$
3. $j = \lfloor d/h \rfloor$
4. Encode j in unary
5. Find the k-th nearest integer y to $\hat{x}$ such that $x \equiv y \pmod{m}$
6. Encode k in minimal binary.
7. IF $d = hj$
   Append bit b
   IF $x < \hat{x}$ Set $b = 0$ ELSE Set $b = 1$

3.2 Decoding

Given the positive integer parameter m, the prediction $\hat{x}$, and the index pair (j, k), the decoder first finds the k-th nearest integer y to $\hat{x}$. Then $i = y \bmod m$ is the coset index of x, i.e., x belongs to the coset $C(i)$. The value of j indicates that $hj \le d < h(j+1)$ which means that either $x \le \hat{x} - hj$ or $x \ge \hat{x} + hj$. Moreover, it also indicates that $x > \hat{x} - h(j+1)$ and $x < \hat{x} + h(j+1)$. Let $\hat{x} - hj = L$ and $\hat{x} + hj = R$. The decoder then finds in $C(i)$ the greatest integer x_f such that $x_f \le L$ and the least integer x_c such that $x_c \ge L$. Now x must be either of these two values since any other integer in $C(i)$ violates the constraints imposed by j (see Fig. 3). If $0 < L - x_f < h$ then $x_c - R > h > 0$, i.e, $x_c > R + h = \hat{x} + h(j+1)$ and thus x must be x_f. Similarly if $0 < x_c - R < h$ then x_c must be x. However, if $L - x_f = 0$ then also $x_c - R = 0$ and an ambiguity arises which can be solved from the extra bit appended with the code for this case. The decoding algorithm is summarized in Algorithm 3.2.

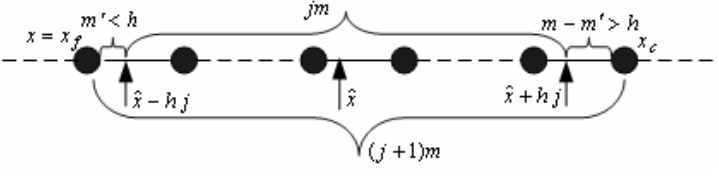


Fig. 3. DSC based decoding

Algorithm 3.2 (x) = **DECODE**$(m,\ \hat{x},\ (k, j, b))$

1. Find the k-th nearest integer y to $\hat{x}$
2. $i = y \bmod m$
3. $h = m/2,\quad L = \hat{x} - hj,\quad R = \hat{x} + hj$
4. Find the greatest integer x in $C(i)$ such that $x_f \le L$
5. Find the least integer x_c in $C(i)$ such that $x_c \ge R$
6. IF $0 < L - x_f < h$ THEN $x = x_f$
   ELSEIF $0 < x_c - R < h$ THEN $x = x_c$
   ELSE
   Get next bit b
   IF b = 0 THEN $x = x_f$ ELSE $x = x_c$

4 Experimental Results

To demonstrate the effectiveness of the proposed technique, we applied it in coding both the random integer sources with Laplacian prediction residuals and QCIF standard test video sequences. We considered the predictive video scheme in [4] as a representative predictive coding scheme since it uses motion compensation as well as linear prediction and the prediction residuals obtained by this scheme when applied on test video sequences closely matched the Laplacian distributions. The performance of this new technique was also compared against widely used technique of coding the prediction residuals, i.e., rounding the real valued residuals to integers, mapping the integers into non-negative integers using equation (5), and then encoding the non-negative integers using Golomb codes.

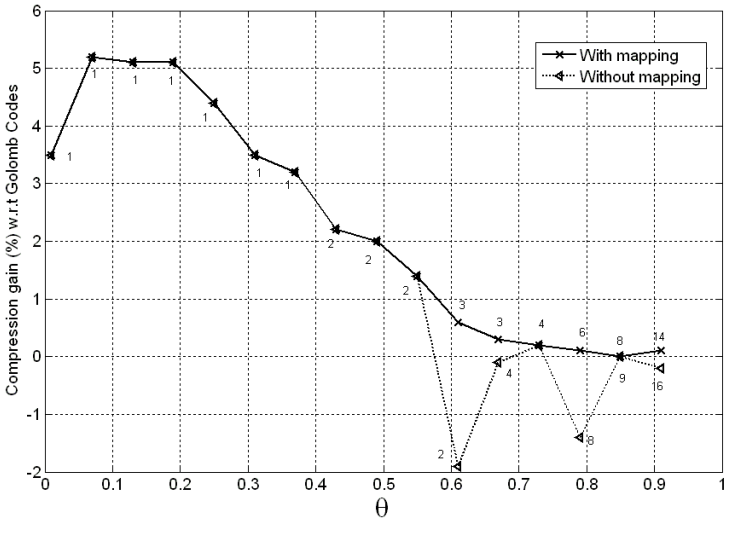


Fig. 4. Performance comparison against Golomb codes. Optimal values for the parameter m are shown on the curves

In the ideal case we encoded uniformly distributed integers in the range [1, 256] with real valued predictions such that the distributions of the residuals were Laplacian. We performed the experiments for different values of the parameter θ, $0 < \theta < 1$. For each value of θ, we determined the optimal value of the parameter m for both the Golomb codes and the proposed codes by exhaustive search.

Fig. 4 compares the performance of the proposed technique against conventional technique, both without mapping of the coset indices and with mapping of the coset indices according to their probability. It is clear from the figure that the proposed

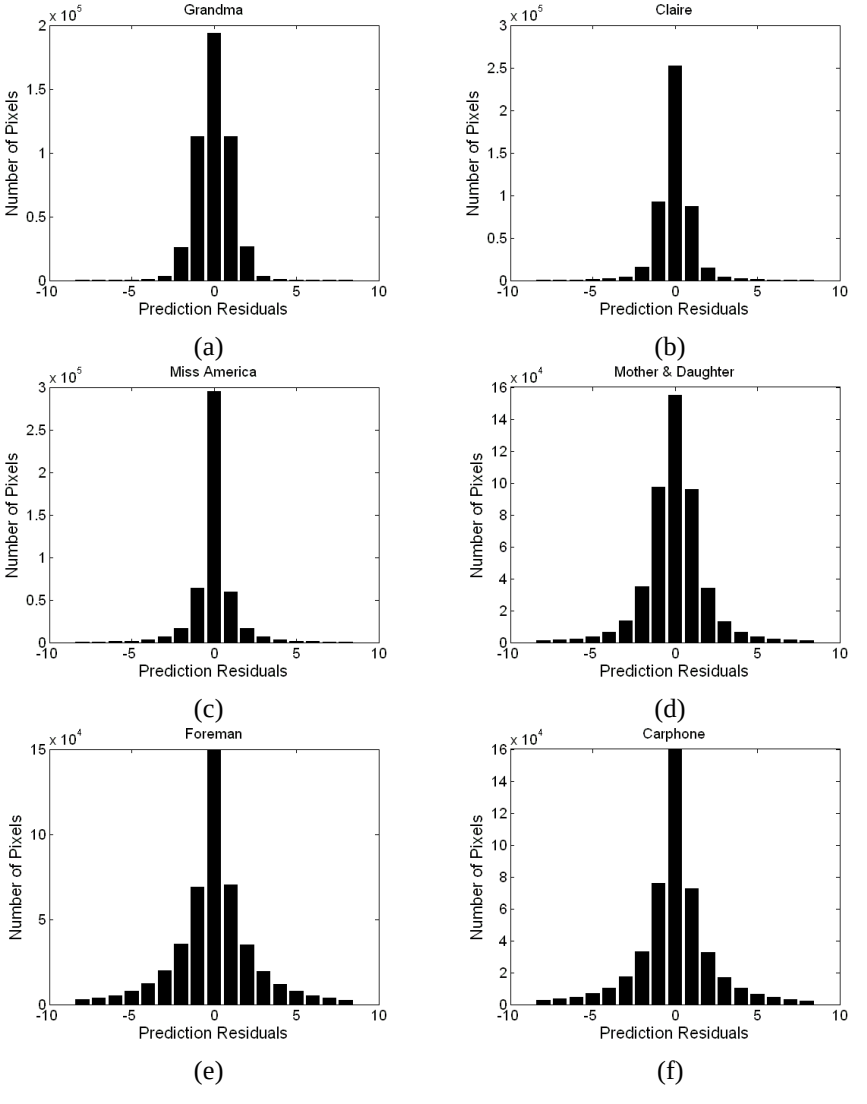


Fig. 5. Distribution of prediction residuals in standard test video sequences (a) *Grandma*; (b) *Claire*; (c) *Miss America*; (d) *Mother & Daughter*; (e) *Foreman*; and (f) *Carphone*

scheme with mapping of coset indices performed better for all values of θ than the conventional scheme. The performance gain is more pronounced for $\theta < 0.6$ and in practical applications if the predictions are good enough in capturing the underlying correlation, the pdf of prediction residuals are highly peaked at zero with smaller values of θ. It is also evident form the figure that the performance of the new technique without mapping is worse than that of with mapping as expected. In the rest of the paper we have used the proposed technique with mapping of coset indices.

Now let evaluate the effectiveness of the proposed technique in coding empirical prediction residuals. We applied our technique in coding video coding residuals obtained by applying the predictive video coding scheme proposed in [4] on first twenty frames of six QCIF standard test video sequences, namely, *Grandma*, *Claire*, *Miss America*, *Mother & Daughter*, *Foreman*, and *Carphone*. The histograms of the prediction residuals for each of the six video sequences are shown in Fig. 5.

After prediction each frame was encoded with both the conventional scheme and the proposed scheme. In both of the schemes each frame was divided in to 8X8 non overlapping blocks. For each block the value of the coding parameter *m* was optimized and this value was used to encode each of the pixels of the block. However, for sequential decoding we need to send these optimal values of parameter *m* to the decoder in advance. Table I summarizes the compression gain of the proposed scheme over the conventional scheme in coding the residuals. Higher compression gain for the video sequences *Grandma*, *Claire*, and *Miss America* can be attributed to the fact that histograms of prediction residuals for these sequences are heavily peaked at zero and decays rapidly on both sides. Thus, the distributions of their prediction residuals can be modeled by Laplacian distributions with smaller values of θ. On the other hand, histograms for *Mother & Daughter*, *Foreman*, and *Carphone* decays slowly on both sides, thus have larger values of θ, and consequently have less compression gain.

Table 1. Compression gain of the proposed scheme against Golomb code based conventional scheme

Video Sequence	Compression Gain (%)
Grandma	3.3
Claire	3.6
Miss America	2.3
Mother & Daughter	1.7
Foreman	0.8
Carphone	1.0

5 Conclusion

An efficient technique to encode integers with real-domain prediction under the distributed source coding paradigm has been presented in this paper. Experimental results on both predictive coding of random sources with Laplacian residuals and predictive coding of standard test video sequences have shown the superiority of this

new technique over Golomb codes based conventional technique. Our future works aim at improving the performance of the proposed technique by more efficiently encoding the coset indices.

References

1. Jayant, N. S. and Noll, P.: Digital Coding of Waveforms, Principles and Applications to Speech and Video. Prentice-Hall, Englewood Cliffs, New Jersey (1984)
2. Liebchen, T., Moriya, T., Harada, N., Kamamoto, Y., and Reznik, Y. A.: The MPEG-4 Audio Lossless Coding Standard–Technology and Applications.119 AES Convention, 2005.
3. Weinberger, M. J., Seroussi, G., and Saporo, G.: The LOCO – I Lossless Image Compression Algorithm: Principles and Standardization into JPEG-LS, *IEEE Transactions on Image Processing*, Vol. 8, No. 9, (2000) 1309–1324
4. Brunello, D., Calvagno, G., Mian, G. A., and Rinaldo, R.: Lossless Compression of Video Using Temporal Information, *IEEE Transactions on Image Processing*, Vol. 12, No. 2, 2003
5. O'Neal, J. B.: Entropy Coding in Speech and Television Differential PCM Systems, *IEEE Transactions on Information Theory*, (1971) 758–760
6. Golomb, S. W.: Run-Length Encodings. *IEEE Trans. Information Theory*, (1966) 399–401
7. Gallager, R. and Voorhis, D. V.: Optimal Source Codes for Geometrically Distributed Integer Alphabets. *IEEE Transactions on Information Theory*, Vol. IT 21 (1975) 228-230.
8. Rice, R. F.: Some Practical Noiseless Coding Techniques – Parts I-III. Tech. Rep. JPL-79-22, JPL-83-17, and JPL-91-3, Jet Propulsion Laboratory.
9. Slepian, D. and Wolf, J. k.: Noiseless Coding of Correlated Information Sources. *IEEE Transactions Information Theory*, Vol. 19, No. 4 (1973) 471–480
10. Wyner, A.: Recent Results in the Shannon Theory. *IEEE Transactions Information Theory*, Vol. 20, No. 1 (1974) 2–10

A Distributed Video Coding Scheme Based on Denoising Techniques

Guiguang Ding[1] and Feng Yang[2]

[1] School of Software, Tsinghua Universtiy,
100084 Beijing, China
dinggg@tsinghua.edu.cn
[2] Department of Automation, Tsinghua Universtiy,
100084 Beijing, China
yfeng00@mails.tsinghua.edu.cn

Abstract. Distributed source coding is a new paradigm for source compression, based on the information-theoretic results built upon by Slepian and Wolf, and Wyner and Ziv from the 1970s. Recently some practical applications of distributed source coding to video compression have been studied due to its advantage of lower encoding complexity over conventional video coding standards. In this paper, we proposed a new distributed source coding framework based on signal denoising techniques. To apply the proposed framework in video coding systems, we give a novel distributed video coding scheme. Our experimental results show that the proposed scheme can achieve better coding efficiency while keeping the simple encoding property.

Keywords: distributed source coding, video coding, denoising techniques.

1 Introduction

With the development of network and wireless communication, more and more applications about wireless video and sensor networks are emerging. In these applications, video coding has to be performed in small, power-constrained, and computationally-limited low-cost devices. In response to the increasing demand on these applications, a simple video codec is eagerly needed. Moreover, the coding rate should not be compromised because this directly impacts the amount of power consumed in transmission [1]. Video codec based on the principles of distributed source coding is just the required video coding technology that can be adaptive to these applications.

The distributed source coding theorems had been established in the 1970s by Slepian and Wolf [2] for distributed lossless coding, and by Wyner and Ziv [3] for lossy coding with decoder side information. Coding algorithms that build upon these results are generally referred to as distributed source coding. These coding theories give us the surprising insight that efficient data compression can also be achieved by exploiting source statistics-partially or wholly-at the decoder only. One of the most important applications of the distributed coding algorithm is the distributed video

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 237 – 246, 2007.
© Springer-Verlag Berlin Heidelberg 2007

system that exploits temporal (interframe) and spatial (intraframe) correlation in the video stream should be performed by the decoder. This facilitates the design of a simple video encoder at the cost of increased complexity at the decoder.

In 1999, Prandhan and Ramchandran [4] introduced a practical framework for Slepian Wolf problem based on channel coding principles. In their method, called Distributed Source Coding Using Syndrome (DISCUS), each output sequence is divided into a coset and the source transmitted the syndrome of the coset instead of the whole sequence. Since the syndrome is shorter than the original sequence, compression is achieved. The decoder reconstructs the original sequence using the syndrome and side information. After that, S.D.Servetto [5] uses lattice quantization to carry out Wyner Ziv codec. Z.Xiong, etc. [6] and D.Rebollo-Monedero, etc. [7] propose Wyner Ziv codec by adding a quantizer in Slepian Wolf codec and get better results. For distributed video coding, many different schemes have been proposed, such as Aaron et al's "intraframe encoding + interframe decoding "system [8], Puri and Ram-chandran's PRISM system [9], Xu and Xiong's layered Wyner Ziv coding[10] and Sehgal's state-free video coding system[11]. These schemes show the rate-distortion performance of distributed video coding outperforms conventional intraframe video coder by a substantial margin, however it does not yet reach the performance a conventional interframe video coder. So, the new techniques have to be developed to improve the performance of the distributed video coding scheme.

Prior work on this topic has been restricted to a basic framework based on algebraic channel coding principles. In this work, we propose instead a framework based on signal denoising approaches. A simple correlation structure between the source and the side information is first given. And then a new distributed source coding framework is proposed. To apply the proposed framework to video coding systems, we give a novel distributed video coding scheme.

The rest of our paper organized as follows. In Section 2, we review the theoretical background of source coding with side information. In Section 3, we describe the basic philosophy and architecture of the propose scheme and illustrate the key intuition behind framework. In Section 4, a specific implementation of the framework to video coding is described. Section 5 details the simulation results and Section 6 concludes the paper.

2 Theoretical Background

In 1973, Slepian and Wolf presented a surprising result to the source coding (compression) community [2]. The result states that if two discrete alphabet random variables X and Y are correlated according to some arbitrary probability distribution, then X can be compressed without access to Y without losing any compression performance with respect to the case where the encoder of X does have access to Y. More formally, without having access to Y, X can be compressed using $H(X \mid Y)$ bits. This is the same compression performance that would be achieved if the encoder of X had access to Y. In [2], Slepian and Wolf established the rate regions that are needed to represent X and Y. The rate regions are represented as a

graph in Fig.1. From the graph, we can see that the sum of rates $R(X)+R(Y)$ can achieve the joint entropy $H(X,Y)$, just as for joint encoding of X and Y, despite separate encoders for X and Y.

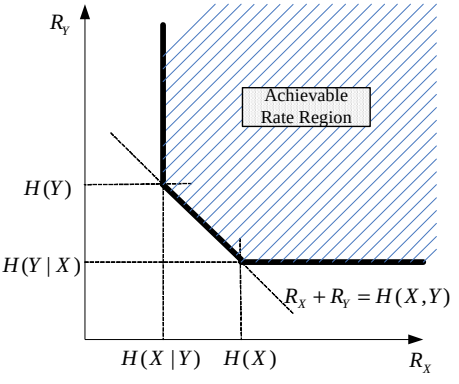


Fig. 1. Slephian-Wolf Theorem: Achievable rate regions for distributed compression of two statistically dependent i.i.d. source

The above results were established only for lossless compression of discrete random variables. In 1976, Wyner and Ziv extended this work to establish information theoretic bound for lossy compression with side information at the decoder. Wyner and Ziv proved that, unsurprisingly, a rate loss is incurred when the encoder does not have access to the side information. However, they also showed that there are no performance degradations in the case of Gaussian memoryless sources and mean-squared error distortion [12].

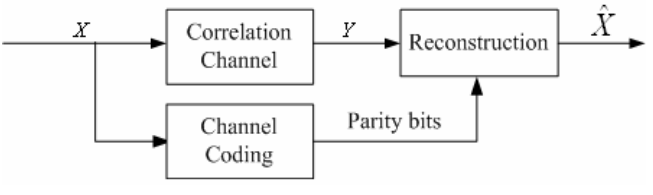


Fig. 2. The interpretation of DSC based on channel codes

They only give us the theoretical results, however, they do not provide intuition how one might achieve the predicted theoretical bounds practically. The first constructive framework for the distributed source coding problem, named DISCUS was proposed in 1999, where a coding construction based on trellis codes was presented. Subsequently, more powerful code constructions for distributed source coding problem have been present, such as turbo codes, low density parity check codes (LDPC) and irregular repteat-accumulate codes (IRA), etc. The reason that

channel coding method can be used for practicing Slepian-Wolf coding is interpreted as follows. In Fig.2, X is transmitted through a correlation channel and the result is Y. To protect X from channel errors, we can use channel coding method to some generate parity bits to reconstruct X from Y.

Although the framework of DSC base on channel codes is very successful to approach Slepian-Wolf bound, it isn't always efficient for the practice application, especially for video coding application. The reason is the ability of correct-error of channel codes is limited. When the correlation between source and side information is weak, this kind of framework is inefficient. Therefore, it is needed to investigate new techniques to solve this problem.

3 Philosophy of the Proposed Framework

So far, all practical Wyner-Ziv video codecs are based on the channel codes. In this paper, we proposed a different implementation method based on signal denoising method, called DISCOID. We consider the system of Fig.3 with the following assumptions:

- X and Y are correlated.
- $X = R + n_1$, $Y = R + n_2$, where R is correlation information, n_1 and n_2 are the noise information.
- Y is available at the joint decoder and we try to compress X as efficiently as possible.

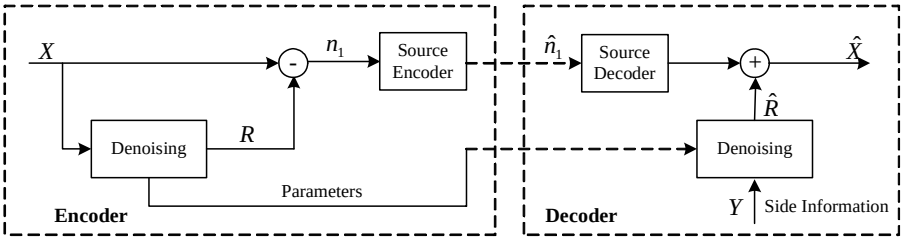


Fig. 3. System for compression based on signal denoising method

With the three assumptions above, in order to allow the use of signal denoising method, at the encoder, X will be fed into the denoising module and its output is R as shown in Fig.3. Then the noise n_1 equals to the difference between X and R, which is intracoded. At the decoder side, Y as side information is processed similar to X according to the denoising parameters from the encoder to abtain the correlation information $\hat{R}$. Then the reconstruction data $\hat{X}$ is calculated by the sum of the correlation information $\hat{R}$ and the noise $\hat{n}_1$.

4 A Distributed Video Codec Based on the Proposed Framework

According to the Wyner-Ziv theorem on source coding with side information and the above implementation method, a simple video codec is proposed. Fig.4 and Fig.5 depict the block diagram of our proposed encoder and decoder, respectively.

4.1 Encoding

The video frame to be encoded is first partitioned into 16×16 blocks. Then each block is classified into one of three block types—Skip blocks, Intra blocks, and Inter blocks. The classification is based on the sum of the absolute difference (SAD) between the current block and the co-located block in the previous frame. If the SAD is small, the block is a Skip block. If the SAD is large, the block is classified as an Intra block. Otherwise, the block type is set to Inter block. Preset thresholds determine this block-type classification, and the block type for each block is passed to the decoder in the bitstream.

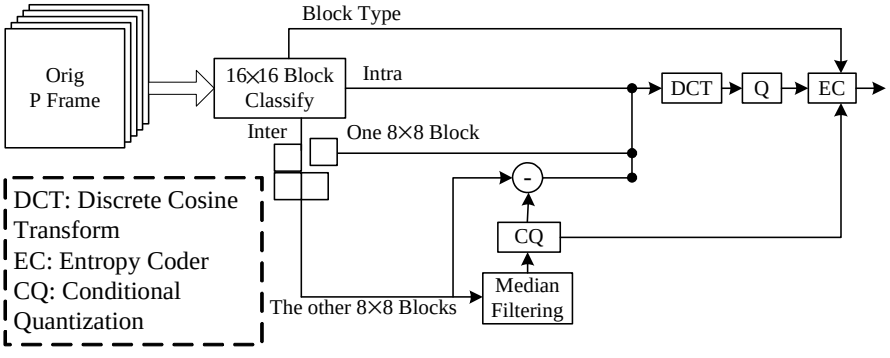


Fig. 4. The proposed distributed video encoder

For Skip blocks, none of bits of the block are coded. On the other hand, for Intra blocks, the coefficients are coded by traditional intra-codec (DCT, Quantization, and Entropy Coding). Finally, for Inter blocks, the coefficients of the block are partitioned Intra coefficients (the size of 8×8) and Inter coefficients (the other coefficients) as illustrated in Fig.4. The Intra coefficients are encoded by the traditional intra-codec. The Inter coefficients are first filtered by median filters, and then quantized by conditional quantization (the conditional quantization consists of the quantized and dequantized processing, the quantization stepsize is in proportion with SAD). The difference of between the original coefficient and the quantized coefficient is the noise $N_1^{'}$ as illustrated Fig.3, which are encoded by traditional intra-codec. The quantization stepsize for each block is passed to the decoder in the bitstream, which are the denoising parameters as illustrated Fig.3.

4.2 Decoding

The block diagram of the proposed decoder is shown in Fig.5. The decoding processing of the bitstream is as follows. The Skip blocks are reconstructed by copying the co-located block in the previous frame. The Intra blocks are decoded by Intra decoder. During decoding the Inter blocks, the 8×8 Intra blocks is firstly intradecoded, then the motion vectors (MVs) are obtained by performing the motion estimation in the previous frame. The motion compensated results of the other 8×8 blocks according to the MVs are used as side information. The side information are first filtered by median filters, and then quantized by conditional quantization (the quantization stepsize is transmitted by the encoder) to obtain the correlation information $\hat{S}$ as illustrated in Fig.3. The reconstructed block is the sum of the noise information from the encoder and the correlation information from the side information.

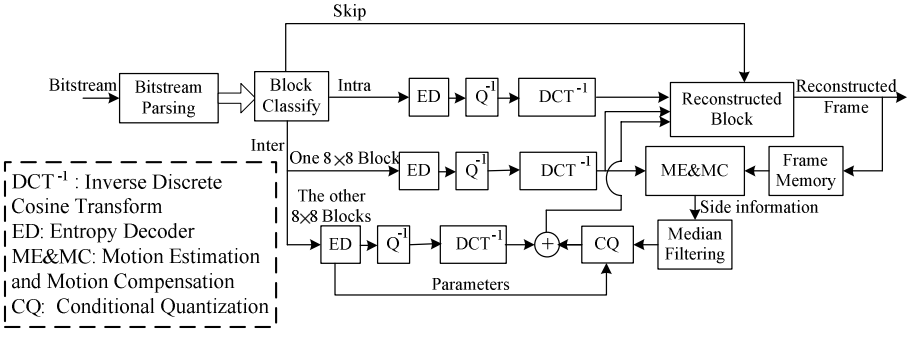


Fig. 5. The proposed distributed video decoder

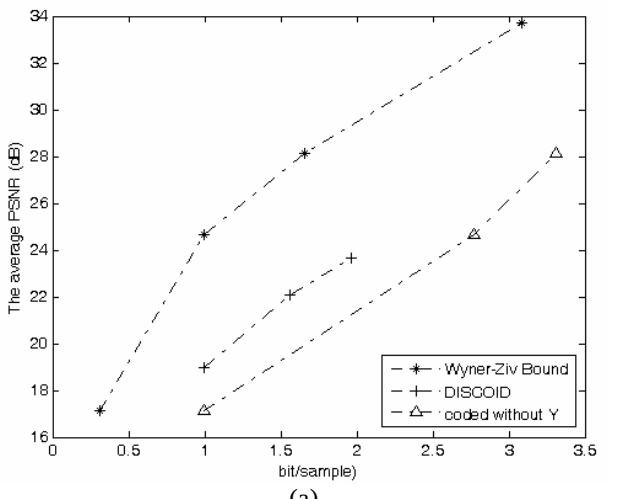
5 Experiment Results

In this section, we present some simulation results that illustrate the performance of the DISCOID framework and video codec based on it.

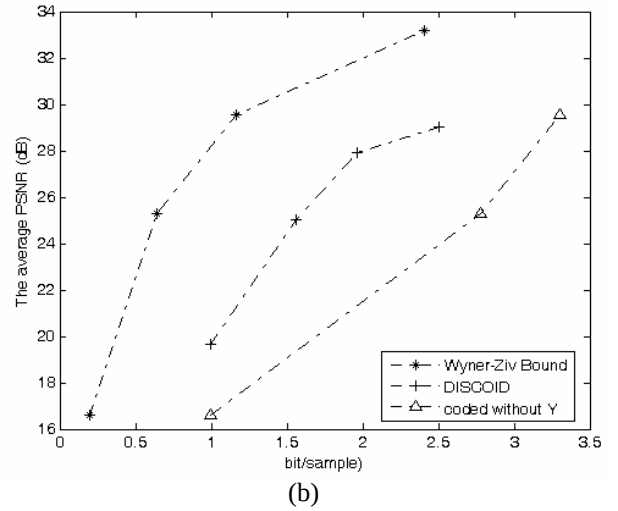
5.1 The Performance of the Proposed Framework

Consider the source model as given in Section 3. We use Lloyd quantizer as denoising method to obtain S through X and Y. The bitrate can be controlled by adjusting the length of the codebook. Initial experiments have been performed to compare the proposed scheme with Wyner-Ziv bound, and independent encoding X without Y. The correlation-SNR (which is the ratio of variance of S and N) is set as 18 and 24, respectively. The PSNR of $\hat{X}$ is plotted versus bit per sample in Fig.6. As can be note from Fig.6, we are about 3 to 5 dB from the Wyner-Ziv bound at

correlation-SNR of 18 and 24, respectively, with gains of about 1 to 4 dB over the independent encoding X without Y. It can also be seen that the performance of the proposed method for higher correlation-SNR (24) is better than that of lower correlation-SNR (18). The reason is that the denoising method is simpler and this makes the denoising less efficient. To improve the performance of the proposed scheme, the more efficient denoising method need be designed, which is part of our ongoing work.



(a)

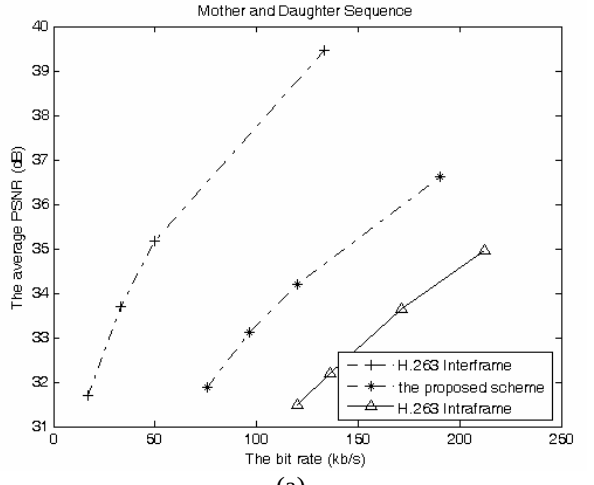


(b)

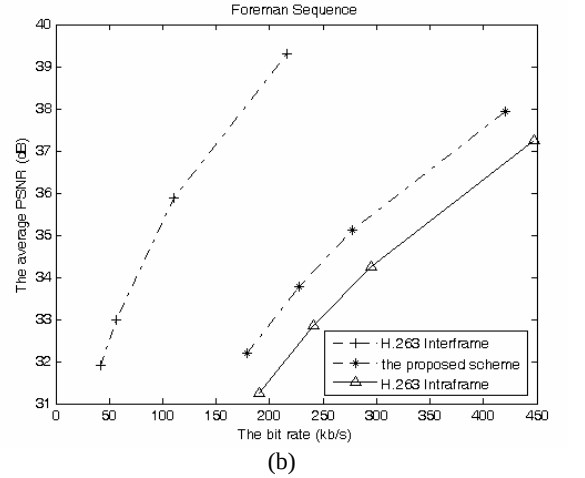
Fig. 6. Rate and PSNR comparision of three schemes: (a) Correlation-SNR is 18 dB, (b) Correlation-SNR is 24 dB

5.2 The Performance of Video Codec Base on Proposed Framework

To evaluate the coding efficiency of the proposed approach to video sequence, we implemented the video codec given in Section 4. The results for the first 30 frames of Mother-Daughter and Foreman QCIF sequences are shown in Fig 7. And only the rate and distortion of the luminance is plotted. The frame rate is 10 frames per second. The results are also compared with H.263 I-P-P-P and H.263 Intraframe coding. From the plots we can see that the rate-distortion performance of our proposed methods lies between H.263 interframe and intraframe coding. For Mother-Daughter sequence the plots show that the proposed scheme can achieve 2 to 3 dB improvement. For



(a)



(b)

Fig. 7. Rate and PSNR comparison of three schemes: (a) Mother and Daughter Sequence, (b) Foreman Sequence

Foreman sequence, which has high motion throughout the frame, we observe less improvement over Intraframe coding. The proposed scheme attains about 1 dB improvements in PSNR. The proposed scheme has no advantage of compression efficiency when compared to the distributed video coding based on channel codes [4,5]. According to the statistic property of video sequence, how to design an efficient denoising method is also dealt with in our following work.

6 Conclusions

A distributed source coding framework based on signal denoising techniques is proposed in this paper. Based on this framework, we give a simple implementation to video coding. The experiment results show that the performance of our system is better than H.263 intraframe coding and the gap between our scheme and H.263 interframe coding is still large. But with more complex and accurate side information estimation and denoising methods, the performance will be better. This proposed framework gives us another selectable method to implement distributed source coding except to channel codes.

Acknowledgments. The authors acknowledge the support received from the National Natural Science Foundation of China (Project 60502014).

References

1. A. Avudainayagam, J. M. Shea, and D. Wu, “A Hyper-Trellis based Turbo Decoder for Wyner-Ziv Video Coding”, IEEE Globecom 2005, St. Louis, MO, USA (2005)
2. D. Slepian and J. Wolf, "Noiseless coding of correlated information sources," IEEE Transactions on Information Theory, vol. 19 (1973) 471-480
3. A. Wyner and J. Ziv, "The rate-distortion function for source coding with side information at the decoder," IEEE Transactions on Information Theory, vol. 22 (,1976)1-10
4. S.S.Pradhan and K.Ramchandran,"Distributed source coding using syndromes (DISCUS): design and construction," in Proc. IEEE Data Compression Conference, Snowbird, UT (1999)158-167
5. D.Servetto,"Lattice quantization with side information,"in Proc. IEEE Data Compression Conference,Snowbird,UT (2000) 510-519
6. Z.Xiong,A.Liveris,S.Cheng and Z.liu, "Nested quantization and Slepian-Wolf coding: A Wyner-Ziv coding paradigm for i.i.d. sources,"in Proc. IEEE Workshop on Statistical Signal Processing, St.Louis,MO (2003)
7. D.Rebollo-Monedero,R.Zhang and B.Girod, "Design of optimal quantizers for distributed source coding, "in Proc.IEEE Data Compre -ssion Conferecne (DCC), Snowbird, UT (2003) 13-22
8. A. Aaron, E. Setton, R. Zhang, and B. Girod, "Wyner-ziv coding for video: applications to compression and error resilience," in Proc. IEEE Data Compression Conference, DCC’03, Snowbird, Ut, (2003)93-102
9. R. Puri and K. Ramchandran, “PRISM: a “reversed” multimedia coding paradigm”, Proc. IEEE International Conference on Image Processing, ICIP2003, Barcelona, Spain (2003)

10. Q. Xu and Z. Xiong, “Layer Wyner-Ziv video coding”, Proc. Visual Communications and Image Processing, VCIP’04, San Jose, USA (2004)
11. A. Sehgal, A. Jagmohan, and N. Ahuja, “ A state-free causal video encoding paradigm,” in Proc. IEEE International Conference on Image Processing ICIP2003, Barcelona, Spain (2003)
12. Girod, B., Aaron, A.M., Rane, S., Rebollo-Monedero, D, “Distributed Video Coding,” Proceedings of the IEEE, Volume 93, Issue 1 (2005)71 – 83

Fusion of Region and Image-Based Techniques for Automatic Image Annotation

Yang Xiao[1], Tat-Seng Chua[1], and Chin-Hui Lee[2]

[1] School of Computing, National University of Singapore, Singapore, 117543
[2] School of Electrical and Computer Engineering Georgia Institute of Technology, Atlanta, GA. 30332, USA
xiaoy@comp.nus.edu.sg, chuats@comp.nus.edu.sg, chl@ece.gatech.edu

Abstract. We propose a concept-centered approach that combines region- and image-level analysis for automatic image annotation (AIA). At the region level, we group regions into separate concept groups and perform concept-centered region clustering separately. The key idea is that we make use of the inter- and intra-concept region distribution to eliminate unreliable region clusters and identify the main region clusters for each concept. We then derive the correspondence between the image region clusters and concepts. To further enhance the accuracy of AIA task, we employ a multi-stage kNN classification using the global features at the image level. Finally, we perform fusion of region- and image-level analysis to obtain the final annotations. Our results have been found to improve the performance significantly, with gains of 18.5% in recall and 8.3% in "number of concepts detected", as compared to the best reported AIA results for the Corel image data set.

Keywords: Automatic Image Annotation, multi-stage kNN, Kullback-Leibler divergence.

1 Introduction

Conventional content-based image retrieval (CBIR) systems require users to retrieve images based on low-level content attributes. Ideally, the users would prefer to query an image database by issuing text-based semantic queries. To facilitate text-based retrieval of images, the images must be annotated with a set of concepts. The automatic image annotation (AIA) involves the analysis of low-level content features of images at the regions/blocks or image level to infer the presence of semantic concepts.

AIA has received extensive attention recently. Starting from a training set of annotated images, many statistical learning models have been proposed in the literature to associate low-level visual features with semantic concepts [1,3,5,17,18,19]. The methods can be divided into two groups: the image-based vs. the region-based methods. The image-based methods [1] attempt to directly label images with concepts based on the selection of low level global features. These methods result in low-cost frameworks for feature extraction and image classification. But using only global visual properties limit their effectiveness to mostly scene-type

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 247 – 258, 2007.
© Springer-Verlag Berlin Heidelberg 2007

concepts and are not effective for object-type concepts. The second group is the region-based methods [3,5,9,10,11,12,17,18,19] that are based on the idea of first dividing the images into regions or fixed-sized blocks. A statistical model is then learnt from the annotated training images to link image regions directly to concepts and use this as the basis to annotate testing images. Most existing region-based methods adopt the discrete approach by tackling the problem in two steps: (1) clustering all image regions to region clusters; and (2) finding joint probability of region clusters and concepts. The performance of region-based methods is strongly influenced by the quality of clustering and consequently the linking of region clusters and concepts, both of which are unsatisfactory.

One of the problems of current AIA systems is that the analysis is carried out at the region or image level. The region level analysis is limited by the accuracy of clustering, and is able to capture mostly object level information. On the other hand, image level analysis is simple but is able to capture only global scene level contents. To overcome the problems of both techniques and to enhance the overall AIA performance, we need to analyze image semantics at multiple levels, the content (region) and concept (image) levels. Thus in this research, we propose a novel concept-centered framework to facilitate effective multi-level annotation of images at region and image levels. The main techniques and contributions of our work include: (1) We propose a novel concept-centered region-based clustering method to tackle the correspondence between the concepts and regions. The process utilizes intra- and inter-concept region distributions to automatically identify the main region clusters (blobs) for each concept, obtain the representative region clusters and typical features for each concept, and use the information to annotate the regions. (2) We perform multi-level annotation by fusing the results of region-level and image-level annotations.

The rest of the paper is organized as follows. Section 2 presents a brief overview of the design of the system. Section 3 discusses the region-based concept-centered technique. Section 4 describes the image-based multi-stage kNN classifier. In Section 5, the image- and region-level results are fused in two stages to produce the multi-level semantics for the testing images, along with results and discussions. Finally Section 6 concludes the paper.

2 System Design

To address the limitations of current AIA systems, our concept-centered AIA system aims to solve the correspondence between image regions and concepts at region-level analysis, and then combine region- and image-level analysis for automatic image annotation, which produces multi-level (both concept level and content level) semantics of images. The overall system consists of 4 main modules as shown in Figure 1.

As with most research in AIA, we consider the case where the concepts are annotated at the image level. Hence each segmented region within an image will inherit all the concepts annotated for that image. As only one or two concepts are likely to be relevant to a particular region, the problem then becomes one of identifying the main concept associated with each region, while eliminating the rest of co-occurring concepts at the image level. To tackle the problem, Module 1 performs

concept-centered region clustering to identify the main region clusters for each concept by taking into consideration the inter- and intra-region distributions. The main region clusters for each concept are used later as the basis to associate regions to concepts. To incorporate image level AIA, Module 2 performs multi-stage kNN classification at the image level to deduce the most similar images. This is based on the assumption that images with same semantic contents usually share some common low-level features. The kNN of similar images are used for refining region-level candidates and performing image-level annotation. Next, we perform the fusion of region- and image-level results in two stages. Module 3 performs an essentially region-based AIA that uses multi-stage kNN to constrain the results. We expect the outcome to be high precision annotation at the region-level. Module 4 fuses the AIA results of image-level and region- level method using Bayesian method. We expect the eventual results at image-level to have high recall while maintaining the precision of the region based method.

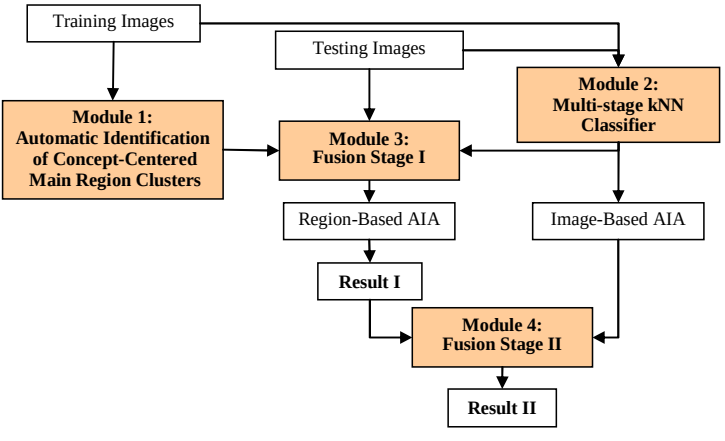


Fig. 1. Concept-centered AIA system workflow

3 Concept-Centered Region Clustering

3.1 Overview of Concept-Centered Region-Based Clustering

At the region level, we first perform the segmentation of training images into regions and merge the smaller regions into modified regions using the k-Means method. As we do not know which specific concept is relevant to which region, we simply associate all annotation concepts for the training image to all its regions. The existing methods treat an image as consisting of a set of region clusters and analyze the semantic concept of each region cluster to build a vocabulary of concepts to represent the whole image. Two difficulties arising from this approach are: (1) how to generate the region clusters of the whole image set; and (2) how to analyze the semantic contents of each region cluster with respect to a set of pre-defined concepts. To overcome the first problem, instead of performing clustering of all regions across all concepts as is done in most current approaches, we group regions into separate

concept groups based on the concepts that they have inherited. By specifically focus on the regions that have the possibility of representing this concept, we hope to minimize the noise resulting from clustering of heterogeneous regions across all concepts using low-level features. At the concept level, we perform clustering of the regions from different images using the k-Means clustering and Davies-Bouldin validation method to group similar regions to clusters. Optimal k for k-Means is decided by the following steps: We run the k-Means on the given dataset multiple times for different k, and the best of these is selected based on sum of squared errors. Finally, the Davies-Bouldin index

$$DB = \frac{1}{k}\sum_{i=1}^{k}\max_{i\neq j}\left\{\frac{\Delta(X_i)+\Delta(X_j)}{\delta(X_i,X_j)}\right\} \tag{1}$$

is calculated for each clustering [15], where $\delta(X_i, X_j)$ defines the intercluster distance between clusters X_i and X_j; $D(X_i)$ represents the intracluster distance of cluster X_i, and k is the number of clusters. Small index values correspond to good clusters, that is to say, the clusters are compact and their centers are far away from each other. Therefore, $argmin_k(DB)$ is chosen as the optimal number of clusters, k. Consequently, we obtain several clusters under each concept.

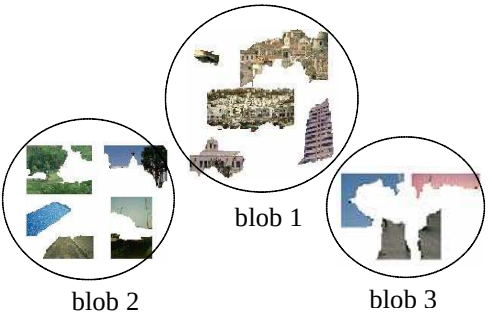


Fig. 2. Example of region clusters for the concept "Building"

Figure 2 shows an example of the region clusters generated for the concept "Building". As can be seen, blob 1 (or region cluster 1) is composed of the representative regions for the "Building" concept, while the others blobs may include regions for co-occurring concepts or a mixture of them. We call blob 1 the "main blob" of concept "Building". In this research, we intend to automatically identify the main blobs of an individual concept. The main blobs found can then be used as the basis for region annotation, image annotation, and even image retrieval. The identification process involves two stages. First we eliminate the unreliable clusters, which are those that clearly do not represent the current (base) concept. Their elimination reduces the possible clusters for main cluster identification. Second, we identify the main blobs, which are the most representative of the base concept. The following sections describe the details of these two identification processes.

3.2 Unreliable Blob Identification

We aim to utilize the concept co-occurrence and the relationship of intra-concept region clusters to find the most unreliable region blobs, i^{un}, under the base concept *T*. Let *W(T)* represent the related (co-occurring) concepts with *T*, including *T* itself. The algorithm is as follows:

First, we cluster regions of training image set *I(T)* into *L* blobs $R(I(T)_i)$, $i=1,\ldots,L$.

Second, given an training image set *I(G)* where $G \in W(T) \backslash T$, we remove part of the images in *I(G)* that correspond to any concepts in $W(T) \backslash G$. The remaining image set is:

$$S_G = I(G) \backslash \bigcup_{x \in W(T) \backslash G} (I(x) \cap I(G)) \tag{2}$$

Here, *G* is the only shared concept between *I(T)* and S_G. This means that we have eliminated the probabilities that images in *I(T)* would be similar with images in S_G due to other concepts beside *G*.

Third, we cluster the regions of S_G into optimal number of clusters $R(S_{G_j})$, $j=1,\ldots,J$, and compute the Euclidean distance of intra-clusters:

$$\Lambda(i, j) = dist(R(I(T)_i), R(S_{G_j})) \cdot \tag{3}$$

At $\arg\min_i(\Lambda)$, that region blob *i* under *I(T)* is most similar to certain blob under S_G. We increment V_i at $\arg\min_i(\Lambda)$, where V_i measures the degree of unreliability of blob *i*.

Fourth, we repeat the second and third steps on all related concepts $W(T) \backslash T$. The result

$$i^{un} = \arg\max_i(V_i) \tag{4}$$

is the most unreliable blob for the base concept *T*.

3.3 Main Blob Identification

Next we aim to identify the main blob i^*, which best represent the semantic meaning within the blobs of the base concept:

$$i^* = \arg\max_i P(concept \mid blob_i), \; i=1,\ldots,L. \tag{5}$$

First we investigate two properties of the distributions of regions in blobs under the base concept and all related concepts: (1) the representative regions are compactly-clustered under the base concept; and (2) they are dispersed under other related concepts. Figure 3 presents an example of region data projected in 2-D space to explain these properties. We assume that there are four kinds of region data, shown in different symbols, representing concepts A, B, C and D. Figures 3(a) and 3(b) respectively show the region distributions under concepts A and B. The ellipses represent the region blobs. It is observed that the representative regions for concept A are compactly-clustered under concept A, while dispersed under concept B, and vice versa. Also the representative regions for concept B in the blob of concept A are only

part of the regions in the main blob under B. From the 1st property, regions of main blob under the base concept are distributed to more region blobs under related concepts than the non-representative regions. From the 2nd property, regions of non-main region blobs under the base concept are distributed to only one or few region blobs under their correspondent concepts.

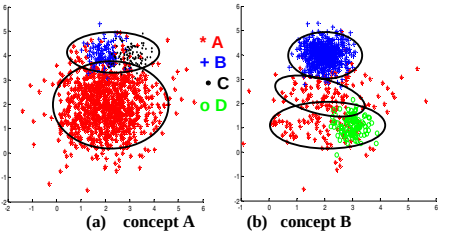


Fig. 3. The distribution of regions under concepts A and B

Given the base concept *T*, after the elimination of unreliable blob i^{un}, the remaining L' region blobs are $R(I(T)_i)$, $i = 1,...,L'$. Also, for every related concept of *T*, $B \in W(T) \backslash T$, we group and cluster the regions under related concept *B* into *J* region blobs $R(I(B)_j)$, $j=1,\ldots,J$. Then we build two functions, *f* and *g*, which focus on the relationship of region distribution by exploiting the above two properties to decide the main blobs. $f(i)$ makes use of Kullback-Leibler (K-L) divergence [13] to measure how well the distribution in blob set of related concepts matches the distribution in certain blob *i* of the base concept. On the other hand, $g(i)$ uses the distribution factor to measure the degree of distribution diversity of the image regions from blob *i* of base concept to the blobs of related concepts.

In probability theory and information theory, the K-L divergence is a natural distance measure from a "true" probability distribution *p* to an "arbitrary" probability distribution *q*. $f(i)$ is defined by the sum of all the related concepts on the mean K-L divergence between a certain blob *i* in the base concept *T* and the blob set *blobs(B)* in a related concept *B*:

$$f(i) = \sum_{B \in W(T) \backslash T} \left(\frac{1}{\| blobs(B) \|} * \sum_{j \in blobs(B)} D_{KL}(p_j \| q_i) \right), \quad i = 1,..., L'. \tag{6}$$

where q_i is the distribution of $R(I(T)_i)$, p_j is the distribution of $R(I(B)_j)$, and $\|blobs(B)\|$ is the number of blobs in concept *B*.

For probability distributions *p* and *q* of a discrete variable the K-L divergence between *p* and *q* with respect to *p* is defined to be:

$$D_{KL}(p \| q) = \sum_k p(k) \log \left(\frac{p(k)}{q(k)} \right). \tag{7}$$

The K-L divergence is the expected amount of information that a sample from *p* gives of the fact that the sample is not from distribution *q*. From the above

distribution property, the regions in the main blob of base concept, comparing with the regions in the other blobs, should be distributed more universally in the blobs of all the related concepts. So the main blob should get the minimization of *f(i)*, which means:

$$i^* = \arg\min_i f(i), i = 1,..., L'. \tag{8}$$

On the other hand, for every other concept *B*, we record how the shared regions between *T* and *B* are distributed under each concept. To do this, we first compute *V(i,j)*, which is set to one if the region cluster *j* of concept *B* has share regions with region cluster *i* of base concept *T*. Otherwise *V(i,j)* is set to zero. We then compute the distribution parameter $N_{T,i,B}$, which is the number of region clusters in *B* that has shared regions with cluster *i* of base concept *T*, as follows:

$$N_{T,i,B} = \sum_{j \in blobs(B)} V(i, j). \tag{9}$$

After analyzing all the related concepts, the region clusters that achieve the maximum of that sum of $N_{T,i,B}$ on all related concepts *B* will be considered as the main blob of the base concept *T*:

$$i^* = \arg\max_i g(i) = \arg\max_i \left(\sum_{B \in W(T) \setminus T} N_{T,i,B} \right), \quad i = 1,..., L'. \tag{10}$$

Finally, we fuse the results for main blob derived from the two functions:

$$i^* = F\left(\arg\min_i f(i), \arg\max_i g(i) \right), \quad i = 1,..., L'. \tag{11}$$

where *F(·)* is simply an union operation in our test.

After we obtain the representative regions and typical features from the main blobs for each concept, we could use the information to annotate the regions and images. It will be discussed in Section 5.

4 Image-Based Multi-stage kNN Classifier

Beside the region-level analysis, we perform image-level analysis using a multi-stage kNN technique. Since images with same semantic meaning usually share some common low-level feature, the multi-stage kNN can be used to perform image matching for annotation at the image level.

As illustrated in Figure 4, the multi-stage system can be viewed as a series of classifiers, each of which provides increased accuracy on a correspondingly smaller set of entities, at a constant classification cost per stage. It can exceed the performance of any of its individual layers only if the classifiers appearing at different layers employ different feature spaces [7]. For effectiveness of multi-stage kNN classifier, we arrange the features in the order that make the classifier at the 1st stage to have high sensitivity (few false negatives), while the classifier at the 2nd stage to have high specificity (few false positives) but less sensitivity. As compared to color histogram, the auto-correlogram is more stable to changes in color, large appearance, contrast

and brightness. It thus serves as a good 1st stage feature to avoid removing too many false negatives, paving the way for the use of simple edge and color histogram features in the 2nd stage. So for the 1st stage, we adopt HSV auto-correlogram; while for 2nd stage, we use the HSV histogram combining with edge histogram. More specifically, we select the top 100 kNN images for the 1st stage and 4 nearest images for the 2nd stage.

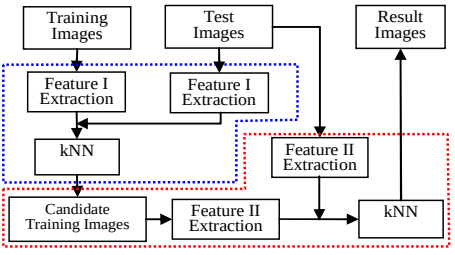


Fig. 4. Image-based multi-stage kNN classifier

5 Two-Stage Multi-level Fusion and Results

We fuse the region- and image-based results in two stages to perform automatic image annotation.

5.1 Fusion Stage I–Fusion of Region-Based Methods with Multi-stage kNN

The main objective of region-level analysis is to enhance the ability of capturing as well as representing the focus of user's perceptions to local image content. We have obtained the main region blobs of each concept for the training images in Section 3.3. The explicit concept of each training region can be determined from which main region blobs that it belongs to. During testing, in order to refine the possible concept range of the test images, we first apply the multi-stage kNN classifier as described in Section 4 to find several most similar training images for each test image. After that, for each region in the test image, kNN is again applied at the local region feature level to find the nearest 2 regions from among the regions of the most similar training images. The concepts of these two nearest training regions are assigned as annotated concepts of the test image region.

One advantage of region-based method is that it provides annotation at the region level. It allows us to pin-point the location of region representing each concept. It thus provides information beyond what is provided by most image-level annotation methods. The use of kNN to narrow the search range further enhances the precision. We thus expect the overall fusion to have good precision.

As with all the other experiments [3,5,6], we use the Corel data set that has 374 concepts with 4,500 images for training and 500 for testing. Images are segmented using the Normalized Cut [16] and each region is represented as a 30 dimensional vector, including region color, average orientation energy and so on [3]. The results are presented based on the 260 concepts which appear in both the training and test sets. Annotation results for several test images are showed in Figure 5. The concepts

shown in the rectangles are the results of region annotation. Ground truth is shown under each image for comparison. The results show that our region-based technique could provide correct annotation at the region level in most cases.

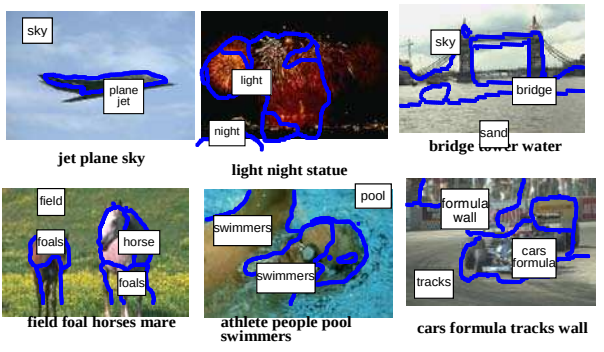


Fig. 5. Region annotation for images

Measuring the performance of methods that predict a specific correspondence between regions and concepts is difficult. One solution, applicable to a large number of images, is to predict the concepts for the entire images and use the image level annotation performance as a proxy. For each test image, we derive its annotation concepts by combining the concepts of each region that it contains and use this as the basis to compute the precision and recall. The number of concepts for each image is not restricted here. Table 1 shows the results of image level annotation in terms of average precision (P), recall (R), and F_1 over all the concepts, and the number of concepts detected (# of Det.), i.e. concepts with recall > 0. The results show that our region-based techniques could achieve an average F_1 measure of 0.20, with 114 detected concepts that have at least one correct answer.

Table 1. Result of region-based AIA

P	R	F_1	# of Det.
0.19	0.21	0.20	114

In comparison with the state-of-the-arts systems listed in Table 3, the performance of the region-based method is better than most except the top two systems. It should be noted that our region-based method provides annotation at region level as shown in Figure 5 instead of just at image level without location information. To enhance the annotation performance at the expense of location, we explore an image-based AIA approach in next Section.

5.2 Fusion Stage II–Fusion of Region-Based AIA and Image-Based AIA

At the image level, we first perform the multi-stage kNN to obtain several nearest training images for each test image. We sum up the concepts of these training images to arrive at a frequency measure for each available concept. To annotate the test

image, we choose the highest frequency concepts until the desired number of concepts is reached. For those concepts with equal frequency, we give priority to those belonging to the annotation of the nearer image.

Test Image			
Ground-truth	deer forest tree white-tailed	caribou grass herd tundra	birds fly nest
Image-based	zebra herd plane grass **tree**	**grass** flowers	**birds** flowers
Region-based	water **deer white-tailed** giraffe	**tundra** flowers **herd**	tree **birds** flowers **nest fly**
Fusion	zebra herd plane grass **tree** water **deer white-tailed**	**grass** flowers **tundra herd**	**birds** flowers tree **nest fly**
Test Image			
Ground-truth	buildings clothes shops street	frost fruit ice	food market people shops
Image-based	**clothes street** museum fountain	**frost ice** spider	buildings tree farms
Region-based	**buildings shops street** people	water **ice fruit**	clouds **people** house **shops**
Fusion	**clothes street** museum **buildings shops** people	**frost ice** spider water **fruit**	buildings tree farms clouds **people** house **shops**

Fig. 6. Annotation results of image- and region-based methods

To illustrate the results of image-based method against that of the region-based method obtained in Section 5.1, Figure 6 shows some automatic annotation results of both methods. Under each image, the ground truth are shown at the top line, followed by the annotation results of the image-based method in the middle line, with the results of region-based method at the third line. Concepts in bold correspond to correct matches. It can be seen that global feature-based results at image level are more concerned with abstract background and frequently occurring concepts, while local region based results are more concerned with specific object-type concepts. It is clear that both methods produce different results, and we should be able to improve the results further by combining both.

Thus, in order to improve the recalls of the overall performance, we employ Bayesian fusion method [4] to perform the fusion. We expect the final results to have better recall while maintaining high precision.

We use the same Corel data set as described in Section 5.1. Table 2 shows the results of AIA for region-based (R_B), image-based (I_B), and fusion (R+I) methods. The desired number of concepts for each test image is set to 8. We can see from Table

2 that the fusion improves the overall performance, with the F_1 measure improve steadily from 0.20 (region-based method) to 0.24 (image-based method) and then to 0.26 (fusion of both). The number of detected concepts reaches 144 for the fusion approach. It is clear that fusion improves the performance for either the region-based AIA or image-based AIA. Figure 6 gives examples of the concepts annotated using the fusion approach (shown in line 4 under each image). It can be seen from the examples that our proposed method is able to infer more correct annotations.

Table 2. Results of fusing region and image-based AIA

	P	R	F_1	# of Det.
R_B	0.19	0.21	0.20	114
I_B	0.23	0.26	0.24	122
R+I	0.23	0.32	0.26	144

Comparison with published results for same data set is listed in Table 3. The results show that our proposed method outperforms the continuous relevance model and other models on the Corel data set. It achieves the best average recall and best number of detected concepts. At the same time, our precision is not too bad. Overall, it improves the performance significantly by 18.5% in recall and 8.3% in the "number of concepts detected", as compared to the best result that has been reported.

Table 3. Comparison with other results

Method	P	R	# of Det.
TM [3]	0.06	0.04	49
CMRM [17]	0.10	0.09	66
ME [18]	0.09	0.12	N.A.
CRM [19]	0.16	0.19	107
MBRM [5]	0.24	0.25	122
MFoM [6]	0.25	0.27	133
Proposed	**0.23**	**0.32**	**144**

6 Conclusion

In this paper, we proposed a novel concept-centered region-based approach for correlating the image regions with the concepts, and combining region- and image-level analysis for multi-level image annotation. At the region level, we employed a novel region-based AIA framework that centers on regions under a specific concept to derive region semantics. Our system aims for automatic identification of the main region blob under each concept by using inter- and intra-concept region distribution. The main region blobs found are then used to determine the explicit correspondence of region to concept. At the image level, we applied a multi-stage kNN classifier based on global features to help region-level AIA. Finally, we performed the fusion of region- and image-based AIA. The results have been found to outperform previously reported AIA results for the Corel dataset.

For future work, we plan to further explore the integration of region- and image-based techniques for image/video classification and retrieval.

References

1. A. Oliva and A. Torralba. Modeling the shape of the scene: a holistic representation of the spatial envelope. *International Journal of Computer Vision*, 42:145-175, 2001.
2. A.Yavlinsky, E. Schofield and S. Rüger. Automated Image Annotation Using Global Features and Robust Nonparametric Density Estimation. *Int'l Conf on Image and Video Retrieval* (CIVR), 507-517, Springer LNCS 3568, Singapore, July 2005.
3. P. Duygulu, K. Barnard, N. de Fretias, and D. Forsyth. Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. In *Proceedings of the European Conference on Computer Vision*, 97-112, 2002.
4. Richard O. Duda, Peter E. Hart, David G. Stork. Pattern Classification, 2nd Edition. Nov. 2000.
5. S. Feng, R. Manmatha, and V. Lavrenko. Multiple bernoulli relevance models for image and video annotation. In *the Proceedings of the International Conference on Pattern Recognition* (CVPR), volume 2, 1002-1009, 2004.
6. S. Gao, D.-H. Wang, and C.-H. Lee. Automatic Image Annotation through Multi-Topic Text Categorization. To appear in *IEEE International Conference on Acoustics, Speech, and Signal Processing* (ICASSP), Toulouse, France, May 14-19 2006.
7. T. E. Senator. Multi-Stage Classification. In *Proceedings of the Fifth IEEE International Conference on Data Mining* (ICDM'05), 2005.
8. T. -S. Chua, S.-Y. Neo, H.-K. Goh, M. Zhao, Y. Xiao, G. Wang. TRECVID 2005 by NUS PRIS in TRECVID 2005, NIST, Gaithersburg, Maryland, USA, Nov 14-15 2005.
9. K. Barnard and D. Forsyth. Learning the semantics of words and pictures. In *International Conference on Computer Vision*, 2: 408-415, 2001.
10. P. Brown, S. D. Pietra, V. D. Pietra, and R. Mercer. The mathematics of statistical machine translation: Parameter estimation. In *Computational Linguistics*, 19(2):263-311, 1993.
11. Y. Mori, H. Takahashi, and R. Oka. Image-to-word transformation based on dividing and vector quantizing images with words. In *MISRM'99 First International Workshop on Multimedia Intelligent Storage and Retrieval Management*, 1999.
12. D. Blei and M. I. Jordan. Modeling annotated data. In *26th Annual International ACM SIGIR Conference*, 127-134, Toronto, Canada, July 28-August 1 2003.
13. S. Kullback and R. A. Leibler. On information and sufficiency. *The Annals of Mathematical Statistics*, 22, 1951.
14. J.-Y. Pan, H.-J. Yang, C. Faloutsos, and P. Duygulu. Automatic multimedia cross-modal correlation discovery. In *10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (KDD2004), Seattle, WA. August 22-25, 2004.
15. D.L.Davies, D.W.Bouldin. A cluster separation measure. *IEEE Transactions on Pattern Recognition and Machine Intelligence*, vol. 1, No. 2, pp. 224-227, 1979.
16. J. Shi and J. Malik, Normalized cuts and image segmentation. *Proc. of IEEE CVPR 97*, 1997.
17. J. Jeon, V. Lavrenko and R. Manmatha. Automatic image annotation and retrieval using cross-media relevance models, In *Proc. of 26th Intl. ACM SIGIR Conf.*, pp. 119–126, 2003.
18. J. Jeon and R. Manmatha. Using maximum entropy for automatic image annotation. In *Proc. of the Int'l Conf on Image and Video Retrieval (CIVR 2004)*, 24-32. 2004.
19. V. Lavrenko, R. Manmatha and J. Jeon, A model for learning the semantics of pictures, *Proc. of NIPS'03*, 2003.

Automatic Refinement of Keyword Annotations for Web Image Search

Bin Wang[1], Zhiwei Li[2], and Mingjing Li[2]

[1] University of Science and Technology of China, Hefei, China
[2] Microsoft Research Asia, No. 49 Zhichun Road, Beijing, China
binwang@ustc.edu, {zli,mjli}@microsoft.com

Abstract. Automatic image annotation is fundamental for effective image browsing and search. With the increasing size of image collections such as web images, it is infeasible to manually label large numbers of images. Meanwhile, the textual information contained in the hosting web pages can be used as approximate image description. However, such information is not accurate enough. In this paper, we propose a framework to utilize the visual content, the textual context, and the semantic relations between keywords to refine the image annotation. The hypergraph is used to model the textual information and the semantic relation is deduced by WordNet. Experiments on large-scale dataset demonstrate the effectiveness of the proposed method.

Keywords: Web image search, multiple modality, hypergraph, image annotation refinement.

1 Introduction

Image is one of the most popular media in humans' lives. With the popularity of digital cameras, the number of online images and personal image collections increase quickly in recent years. To efficiently manage, browse and search images, many methods and systems have been developed. The keyword-based indexing and search method is proven to be the most natural and successful one, while it requires high quality textual annotation.

Because the manually annotation of large number of images is too expensive, automatic image annotation is desired. [5] deems the image annotation as a task to translate the visual words ("blobs") to text words, and proposes the translation model. [9] builds the joint distribution of words and blobs with a cross-media relevance model (CMRM). [10] and [6] improve CMRM using continuous visual features. [1] proposes several generative models. In [4], a hierarchical model and the multiple instance learning are exploited. [7] applied manifold learning in image retrieval. Because of the well-known "semantic gap" [17], some researches focus on leveraging words' semantics. [2] incorporates statistical natural language processing (NLP) in semantic learning and WordNet is used to provide grouping information. [211] utilizes the semantic relation from the relevance feedback of image retrieval. [9] combines multiple semantic measures in deducing word relation.

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 259 – 268, 2007.
© Springer-Verlag Berlin Heidelberg 2007

Traditional image annotation methods require well-annotated training set, which restricts their applications. In the Internet era, web images have been a major part of image collections. Instead of assigning keywords to images in traditional image annotation, we need to refine the rough annotations of web images, which can be extracted from hosting web pages (e.g. anchor text, URL, ALT tags, and surrounding text). Although the annotations are "low-quality", lots of useful information is contained. A similar application is the management of personal/professional image collection. It is infeasible for users to thoroughly and precisely annotate each photo. If we can annotate the images based on some initial incomplete/imprecise annotations, the further indexing, search and browsing services will be greatly facilitated.

In this paper, we propose to jointly utilize multiple modalities including visual content, textual context and words semantic relation to refine images' annotations. The original rough annotation is propagated using the similarities deduced from both visual content and textual context. The textual context is modeled by hypergraph, in which each node represents an image and each hyper-edge represents a word. After such propagation, many keywords can be annotated to an image. It is necessary to extract the most representative ones and prune other words. Thus we employ the words' semantic relation to adjust the weight of candidate words. Experiments show the proposed method is able to refine the annotation effectively.

The rest of this paper is organized as follows. Section 2 presents the related research and background. In section 3, the annotation propagation through content and textual similarity is presented. Section 4 discusses the extraction of most representative words using semantic relations. Experimental results are presented in section 5. Finally, conclusion and future work are presented in section 6.

2 Related Work

2.1 Manifold Learning

Learning on data manifold is an effective semi-supervised learning method when the number of data is large but only few are labeled [19]. Manifold learning assumes the distribution of classes should be smooth (i.e., nearby data tends to have similar class labels) as well as the deviance from original label should be minimized.

Suppose the original label information can be denoted as a $n*c$ matrix Y_{n*c}, where n is the number of data points and c is the number of classes. Each element $Y(i,j)$ represents whether the point x_i is labeled to be in class c_j. Let F denote a label result, the manifold learning is to minimize the overall cost of

$$Q(F) = \frac{1}{2}\left(\sum_{i,j=1}^{n} W_{i,j}\left\|\frac{F_i}{\sqrt{D_{i,i}}} - \frac{F_j}{\sqrt{D_{j,j}}}\right\|^2 + \mu\sum_{i=1}^{n}\|F_i - Y_i\|^2\right) \quad (1)$$

where W_{ij} is the similarity of sample i and j, D_{ii} is the sum of all W_{ij} for a given sample x_i. The first term is the smoothness measure between adjacent points while the second term is the deviation from original labels. μ is a parameter to tradeoff between two factors. The iterative implementation of manifold learning can be stated as figure 1. $\alpha=1/(1+\mu)$ determines the propagation speed.

1. Calculate the affinity matrix W of the data set
2. Normalize W as $S=D^{-1/2}WD^{-1/2}$, where D is the diagonal matrix with $D(i,i)$ is the sum of i-th row of W
3. Iteratively propagate the label information as $F^{(t+1)}=\alpha SF^{(t)}+(1-\alpha)Y$

   where t represent the number of iterations, $\alpha \in [0,1]$, and $F^{(0)}=Y$
4. With the final result of F^*, label each data point to appropriate classes.

Fig. 1. Manifold learning algorithm

2.2 Semantic Similarities

The NLP research suggests the word semantic disambiguation usually needs to calculate the distance between a word and its surrounding words. To calculate the distance, a simple method is to make statistics on large corpus, and use features such as co-occurrence. Another type of methods uses a pre-defined lexicon for semantic deduction. A well-known electrical lexicon is WordNet[11], which collects huge number of words (over 110,000 in ver. 2.1). Each word is clear explained. The whole lexicon is organized as several semantic trees, so it's suitable in deducing words' semantic relation. The nouns and verbs are organized through *is-a* relation. Other semantic relations, such as *has-part, is-made-of* and *is-an-attribute-of*, are provided. All these information can be exploited for calculating semantic relation.

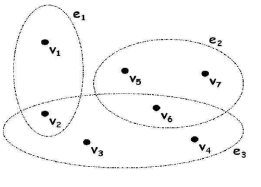
[15] introduces the concept of "information content"(IC) and propose a similarity measure based on the WordNet hierarchical structure. There are many other measures and their performances are evaluated by [3][14].

2.3 Hypergraph Model

Hypergraphs can be deemed as an extension to pair-wise graphs [200] while they can represent more information. Suppose $G = (V, E)$ is a hypergraph, where V is the set of all nodes, and E is the set of all hyper-edges. Each hyper-edge has a weight $w(e)$ and connects multiple nodes. We can think each hyper-edge defines a subset of the whole graph. Let $|S|$ denote the cardinality of a set S. It can be seen that when $|e|=2$ for all $e \in E$, the hypergraph regresses to pair-wise graph model. Then we can define the degree of a node v as:

$$d(v) = \sum_{\{e \in E | v \in e\}} w(e) \tag{2}$$

If we define an incidence matrix to represent the relation between nodes and hyper-edges, it is a $|V|*|E|$ matrix where



	e_1	e_2	e_3
v_1	1	0	0
v_2	1	0	1
v_3	0	0	1
v_4	0	0	1
v_5	0	1	0
v_6	0	1	1
v_7	0	1	0

Fig. 2. An illustration of hypergraph model [20]

$$h(v,e)=\begin{cases}1 & if\ v\ and\ e\ is\ incident\\ 0 & otherwise\end{cases} \tag{3}$$

The degree of a hyperedge e is $d(e)=\sum_{\{v\in e\}} h(v,e)$. Similar to the case of pair-wise graph, the learning on hyper-graph is to minimize the overall cost function

$$Q(F)=\frac{1}{2}\left(\sum_{e}^{n}\frac{1}{d(e)}\sum_{\{u,v\}\subseteq e} w(e)\left\|\frac{F_u}{\sqrt{D(u)}}-\frac{F_v}{\sqrt{D(v)}}\right\|^2+\mu\sum_{i=1}^{n}\left\|F_i-Y_i\right\|^2\right) \tag{4}$$

where μ is to tradeoff between smoothness and deviation from original labels, and $D(u)$ is the sum of weights of all hyperedges the node u belongs to. For the purpose to propagation, the original relation in hypergraph is embedded into pair-wise similarities as equation 7.

The hypergraph model has been widely applied in integrated circuits design since 1970's. Recently, it began to be applied in computer vision and machine learning area. [20] proposes a regularization framework for hypergraphs. [16] applies the hypergraph model for image segmentation and human face clustering.

3 Image Annotation Refinement Using Multiple Modalities

Based on the characteristics of available approximate web image annotations, we propose a framework to refine the annotations. In this framework, images' visual content, textual context, and the semantic relations among words, are all exploited. Figure 3 illustrates the proposed framework. The relations between images from both visual content and textual information are combined. The textual information is modeled by hypergraph and embedded into pair-wise relation for propagation. After the inter-image propagation, each image can receive many words as its annotation. To extract the most relevant words, words semantic relations are leveraged, which will be presented in section 4. It is shown that whether the semantic propagation or content/textual-based propagation is performed first doesn't matter. In this section we discuss the visual and textual based propagation.

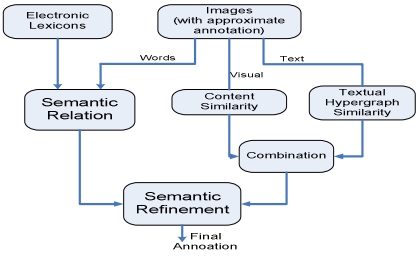


Fig. 3. The framework of annotation refinement

3.1 Visual Content-Based Relation

Image is a visual medium, so it is natural to propagate metadata between visually similar images. For two images I_i and I_j, their content similarity Sim_c (I_i, I_j) can be

calculated using low-level content features, such as color histogram, color moment or others. The visual similarity matrix can be obtained as $V_c=\{Sim_c\ (I_i,I_j)\}$, and normalized to S_c for propagation,. Suppose the original annotation is a matrix T^0 (each row corresponds to an image and each column corresponds to a word), the original annotation can be propagated via visual similarity as

$$T_c^{(t)}=\alpha S_c T_c^{(t-1)}+(1-\alpha)T^0 \tag{5}$$

where the superscript t represents the number of iterations, and $0<\alpha<1$ is a predefined parameter. If the propagation is repeated, because $0<\alpha<1$ and S_c is a normalized affinity matrix whose all eigenvalues are in [-1, 1], equation (5) converges to

$$T_c^{(t)} = (1-\alpha)(I-\alpha S_c)^{-1}T^0 \tag{6}$$

3.2 Textual Context-Based Relation

The textual information is usually represented using vector space model. Thus the similarity can be calculated using the inner-product of two vectors. This pair-wise similarity misses much information. For example, the pair-wise similarity cannot reflect the relationship that three images have a common annotation words,

Given a set of images with their annotation, we propose to model the textual information with a weighted hypergraph G= (V, E), where V denotes the set of nodes representing images, and E is the set of hyperedges representing words. The incidence matrix H represents the relationships between words and images $0\ h(v,e)\ 1$. The continuous weight reflects the relations between words and web images that a word can appear multiple times in the host web page, e.g. title, URL or ALT tag, and with different textual attributions such as bold. For each hyperedge (word) e, its degree is $d(e)=\sum_{\{v:\ e\ in\ v's\ annoation\}} h(v,e)$, and needs not be same with its weight. The degree of each image is calculated as $d(v)=\sum_{\{e\in E\,|\,v\in e\}} w(e)$. For the propagation, the original hypergraph must be embedded into a similarity matrix. The embedded similarity matrix S_t is computed as

$$S_t = D_v^{-1/2}HWD_e^{-1}H^TD_v^{-1/2} \tag{7}$$

where D_e and D_v are the diagonal matrix of node weights and hyperedge weights.

Similar to visual content based propagation, with the original annotation T^0, the iterative propagation process is $(0<\beta<1)$

$$T_t^{(t)}=\beta S_t T_t^{(t-1)}+(1-\beta)T^0 \tag{8}$$

4 Annotation Refinement Using Semantic Relation

After the propagation in section 3, many keywords can be annotated to an image with different weights. It is necessary to extract the most representative ones and prune the others. To utilize the semantic relation between keywords, we perform another round of propagation at the word level. Meanwhile, the semantically related words can aggregate together to form a strong keyword annotation for the image.

Words are directly associated with semantics. Even when the textual information is in low quality, the context information will be useful. Words have many kinds of relationship, such as synonyms, antonyms and so on. All those relations can be exploited to improve the annotation qualities. When several words are labeled for an image, the words semantic relation can help judge if a word is appropriate. Given an image and its approximate annotation words $C = \{c_1, c_2,\ldots,c_n\}$, each word has its initial weight w_i^0. Suppose $Sim(c_1,c_2)$ is the semantic similarity between word c_1 and c_2, the final weight w_i for word c_i can be determined by

$$w_i = \sum_{j \neq i} w_j^0 Sim(c_j, c_i) \tag{9}$$

Or it can be written as $W = S_w W^0$, where W^0 and W are initial and final weight vector, and S_w is a semantic similarity matrix. The word semantic relation is calculated from both corpus statistics and electronic lexicons. For the corpus statistics, we use the co-occurrence $C_o = T^0(T^0)^T$ and C_o is normalized to S_0.

4.1 Semantic Relation Using WordNet

WordNet is adopted because of its huge number of words, good hierarchical structure and many synthetic links between words. [8] proposes a semantic relation measure *jcn* using both node-based and path-based information. For each concept, its information content (IC) is obtained from some semantically labeled corpus:

$$IC(c) = -\log(\frac{Freq(c)}{N}) \tag{10}$$

where $Freq(c)$ is the frequency of concept c, and N is the total size of the samples. The similarity between two concepts c_1 and c_2 is calculated as

$$sim_{jcn}(c_1,c_2) = \frac{1}{IC(c_1)+IC(c_2)-2*IC(lcs(c_1,c_2))} \tag{11}$$

lcs (least common subsumer) is the most specific common parent node of c_1 and c_2, and it reflects the largest common information c_1 and c_2 have. The evaluations in [14] show "*jcn*" is by far one of the best measure among all proposed methods.

One word can have multiple meanings, and thus belongs to several concepts in WordNet. We define the semantic relation between two words w_1 and w_2 as

$$Sim(w_1, w_2) = \max_{c_1, c_2}(Sim_{jcn}(c_1, c_2)) \tag{12}$$

where c_1 and c_2 are the concepts w_1 and w_2 belongs to . So, the semantic relation matrix obtained from *jcn* algorithm is $V_j = \{ Sim(w_1,w_2) \}$, and V_j needs to be normalized as S_j for propagation.

4.2 Semantic Propagation

Since both S_j and S_o are both valid propagation matrices, their linear combination is also valid for the propagation, and we can write the iterative propagation and final convergent state as

$$S_w = \gamma S_j + (1-\gamma)\, S_o$$
$$T_t^{(t)} = (\delta S_w T_t^{(t-1)T} + (1-\delta)(T^0)^T)^T \qquad 0<\delta<1 \tag{13}$$
$$T_w = (1-\delta)\, T^0 (I-\delta S_w)^{-1}$$

4.3 Overall Combination

From above sections, it's shown different types of relations can be used for image annotation. [18] presents several ways to combine S_c and S_t. When S_w is introduced, the word semantic relation can also be incorporated into the process. The simple deduction reveals the final state can be written as

$$T_F = (1-\varepsilon)\,(I-\varepsilon S_d)^{-1} T^0 (I-\delta S_w)^{-1}(1-\delta) \tag{14}$$

where S_d is the combined relation matrix between images as in [18]. It can be seen that the order of whether semantic relation is applied first is not critical.

5 Experiments

The method proposed in this paper is to refine the annotation of images from web or large image collections. To present the performance comparable to other literatures, we conduct experiments on Corel image set from [5]. It contains 5,000 images. Each image is annotated by 1 to 5 keywords. Altogether, there are 371 words. The images are segmented into regions. For each region, 36 dimensions visual feature is extracted. We build two datasets for the experiment. First, two words are randomly selected from each image's annotation to form set 1. Then, we build a noisy set 2. For each image, two irrelevant words (randomly selected from the words not in original annotation) are added, which introduces equally number of noise words. This is to simulate the situation that noisy data on the Internet.

We also use a large image data set downloaded from the web (Microsoft Office Online, http://office.microsoft.com). This dataset contains 34,172 clip art images with large content variance, and 17,194 words and phrases are used for image annotation. We use porter stemming [13] to remove the word variances, and 8985 words remain. Similar to Corel image set 2, we build a noisy training data. The visual feature is 64-dimesion color histogram in HSV color space. In the calculation of the semantic relation between words, *jcn* from WordNet::Similarity [12] is used.

In our experiments, the annotation length is fixed to 5 words per image. We use the measures of precision and recall as in [5]. The mean precision and recall are averaged among all words appeared in test set. We also report the number of words that has recall larger than 0.

5.1 Hypergraph vs. Pair-Wise Models

We first compare the performances of hypergraph model and TFIDF-based pair-wise model. Table 1 shows the results on two Corel datasets. From the table, we can see hypergraph model achieves higher precision with little sacrifice in the recall for Corel set 1. The performance of two models on Corel set 2 is similar. It suggests the hypergraph model can be utilized to refine the annotation for web images.

Table 1. Performance of hypergraph (H) and pair-wise (P) models

	Corel Set 1		Corel Set 2	
	H	P	H	P
Precision	0.844	0.784	0.462	0.464
Recall	0.712	0.732	0.540	0.534

Table 2. Performance of proposed method (Propose) and baseline (CRM-like)

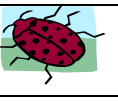
	Corel Set 1		Corel Set 2	
	Propose	CRM-like	Propose	CRM-like
# of correct annotated words	3041	2025	2279	1540
Precision	0.835	0. 434	0.461	0.279
Recall	0.716	0.470	0.555	0.351

5.2 Overall Combination

Because there is little previous work on image annotation refinement, we implement a baseline similar to CRM method in [10]. The results by combining multiple relations including visual content, textual context and word relation are presented in Table 2. It



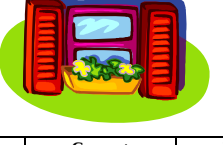
	Correct	Wrong
Train	Hardware, screwdrivers	Lighter, mausoleums
Final	Industrial, household. hardware, screwdrivers	lighter
Groundt ruth	Industrial, household. hardware. Screwdrivers, tool	



	Correct	Wrong
Train	Insect, wildlife	Gambia, Uranus
Final	Animal, nature, insect, wildlife	Arrive
Groundtruth	Animal, nature, insect, wildlife, creature, ladybug, beetles	



	Correct	Wrong
Train	Register, tills	Percentages, precautions
Final	Business, cashiers, Register, tills	precautions
Ground truth	Business, cashiers, Register, tills	



	Correct	Wrong
Train	Flower, planter	Puff, Nairobi
Final	Flower, nature, plant, planter	Nairobi
Groundtruth	Box, flower, nature, plant, build, window, planter, shutter	

Fig. 4. The annotation refinement results for clip art image set

can be seen that the performance of proposed method is much better than CRM-like method. After the propagation using multimodalities, much more correct words are annotated. The reason is CRM-like method doesn't consider the situation in which some annotation words has been available and there is noise in training data.

Figure 4 shows some images with original and refined annotations from the clip image set. The original annotations have two correct words and two noise words. Some wrong words in training data can be properly pruned. These examples illustrate the proposed method can refine the rough annotation for the image index and search.

6 Conclusion

In this paper, we address the problem of refining web images annotation. The approximate annotations are common for web images and image collections. We propose a framework to refine the web image annotations by jointly utilizing multiple modalities including visual content, textual context and word semantics. The approximate annotation can be first propagated to other images by either content similarity or text similarity. Such propagation between images may annotate many words to an image with different weights. Then, we propose to further utilize words semantic similarity to extract representative and related words while pruning irrelevant ones. The experiments illustrates the proposed method is effective.

We will continue our work by verifying our algorithm on actual large web image datasets. Besides, the similarity between images can be better estimated if image segmentations are used. In addition, the semantic relation in this paper is symmetric, while there are asymmetric relations between words. Therefore we may further improve our method about directed semantic word relation.

References

1. Blei D., and Jordan M., "Modeling Annotated Data". In 26th International Conference on Research and Development in Information Retrieval, New York, 2003. ACM Press
2. Barnard, K, Duygulu P. and Forsyth D., "Clustering Art", 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Volume 2 p. 434, 2001
3. Budanitsky A. and Hirst G. "Semantic distance in WordNet: An experimental, Application-oriented Evaluation of Five measure", In Workshop on WordNet and Other Lexical Resources, the North American Chapter of the ACL, Pittsburgh, 2001
4. Carneiro G., Vasconcelos N., "Formulating Semantic Image Annotation as a Supervised Learning Problem", CVPR 05, Washington, USA, pp. 163-168, 2005
5. Duygulu P., Barnard K., Freitas J., and Forsyth D., "Object Recognition as Machine Translation: Learning a Lexicon for a Fixed Image Vocabulary", Proceedings of the 7th European Conference on Computer Vision, London, UK, pp. 97 – 112, 2002
6. Feng S., Manmatha R., and Laverenko V., "Multiple Bernoulli Relevance Models for Image and Video Annotation", CVPR04, pp. 1002-1009, 2004
7. He J., Li M., Zhang H.-J., Tong, H., and Zhang, C., "Manifold-ranking based Image Retrieval", in Proceedings of ACM Multimedia 2004, pp.9-16, 2004
8. Jiang J. and Conrath D., "Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy", in Proceedings of Intl. Conf. Research on Computational Linguistics, 1997

9. Jin Y., Khan L., Wang L. and Awad M., “Image Annotations By Combining Multiple Evidence & WordNet”, ACM Multimedia 2005, pp. 706-715, Singapore, 2005
10. Lavrenko V., Manmatha R., and Jeon J., “A Model for Learning the Semantics of Pictures”, Proceedings of Advance in Neutral Information Processing, 2003
11. Miller, G.A. “WordNet: A lexical database for English”. Communication of ACM, 38, 11 (Nov. 1995), 39–4, 1995
12. Perdersen T., Patwardhan S., and Michelizzi J. “WordNet::Similarity – Measuring the Relatedness of Concepts”, North American Chapter of ACL , May 3-5, 2004, Boston
13. Porter, M.F., “An Algorithm for Suffix Stripping”, Program, 14(3) :130-137, 1980
14. Pucher M. “Performance Evaluation of WordNet-based Semantic Relatedness Measures for Word Prediction in Conversational Speech”, In Sixth International Workshop on Computational Semantics, Tilburg, Netherlands, 2005
15. Resnik, P. “Using Information Content to Evaluate Semantic Similarity in a taxonomy”, In Proceedings of International Joint Conference on Artificial Intelligence, pp. 448-453, 1995
16. Shashua A.,Zass R., and Hazan T.. “Multi-way Clustering Using Super-symmetric Non-negative Tensor Factorization”. ECCV 2006, May 2006, Graz, Austria
17. Smeulders A. W. M., Worring M., Santini S., Gupta A., and Jain R., “Content-based image retrieval at the end of the early years”. IEEE PAMI, 22(12):1349¨C1380, 2000
18. Tong H., He J., Li M., Zhang C., and Ma W.-Y., “Graph Based Multi-Modality Learning”, in Proceedings of ACM Multimedia 2005, pp. 862-871, 2005
19. Zhou, D., Bousquet, O., Lal, T.N., Weston, J., and Schölkopf, B. “Learning with Local and Global Consistency”. NIPS, pp. 237-244, 2003
20. Zhou, D., J. Huang and Schölkopf B., “Beyond Pairwise Classification and Clustering Using Hypergraphs.” MPI Technical Report (143), Tübingen, Germany 2005
21. Zhou X., and Huang. T., “Unifying keywords and Visual Contents in Image Retrieval”, In IEEE Multimedia Magazine, April-June Issue, pp. 23-33, 2002

Mining Multiple Visual Appearances of Semantics for Image Annotation

Hung-Khoon Tan and Chong-Wah Ngo

Department of Computer Science,
City University of Hong Kong,
Kowloon, Hong Kong
{hktan,cwngo}@cityu.edu.hk

Abstract. This paper investigates the problem of learning the visual semantics of keyword categories for automatic image annotation. Supervised learning algorithms which learn only a single concept point of a category are limited in their effectiveness for image annotation. We propose to use data mining techniques to mine multiple concepts, where each concept may consist of one or more visual parts, to capture the diverse visual appearances of a single keyword category. For training, we use the *Apriori* principle to efficiently mine a set of frequent *blob-sets* to capture the semantics of a rich and diverse visual category. Each concept is ranked based on a discriminative or diverse density measure. For testing, we propose a level-sensitive matching to rank words given an unannotated image. Our approach is effective, scales better during training and testing, and is efficient in terms of learning and annotation.

Keywords: Image Annotation, Multiple-Instance Learning, Apriori.

1 Introduction

Content-based image indexing and retrieval is becoming a subject of significant importance. The earlier retrieval systems use only low-level features and the results are unsatisfactory because the semantic image contents are not well captured. An intuitive way is to manually annotate images with captions, and the users retrieve the relevant multimedia documents by typing in keywords at the system. Although low-tech, this approach is effective. However, as the size of the multimedia database explodes over years, such technique is no longer deemed feasible. Automatic annotation of images is becoming increasingly important and has since become an active area of research.

Image annotation systems could be broadly classified into unsupervised learning [1,2,3,4,5,6] and supervised learning [7,8,9,10] problems. The unsupervised learning approaches strive to learn the hidden states of concept, particularly the joint distribution between keywords and multiple visual features. Mori *et al.* [1] proposed the co-occurrence model to collect the co-occurrences between words and image features and used them to predict annotated words for images.

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 269–278, 2007.
© Springer-Verlag Berlin Heidelberg 2007

Duygulu *et al.* [2] proposed a machine translation approach to learn a lexicon which maps a set of keywords to the set of regions of an image. In [3,4,5], relevance models were proposed to find the joint probability of observing a set of image regions together with another set of annotation words. As opposed to [2], the relevance models does not assume an underlying one-to-one alignment between the regions and words in an image and only assume that a set of keywords is related to a set of objects represented by regions. In [6], the Correlation LDA model is proposed to relate the keyword and the image.

The supervised learning approaches use generative or discriminative classifiers from a binary set of visual features with (positive) and without (negative) the semantic of interest. The classifier treats each annotated word as an independent class and a different image classification model is learnt for every semantic category. Recently, weakly supervised method, particularly multiple-instance learning (MIL) [12,13] is becoming a more attractive alternative for learning the semantics of images because of its less stringent requirement on manual labelling. In a MIL setting, we are aware of the presence of the object of interest in the image but which regions correspond to the object of interest is unknown. There are several drawbacks of supervised algorithms that have yet to be addressed before it could be effectively used for image annotation. Some algorithms, particularly MIL, learn a single concept (a point or region in a feature space). Learning multiple concepts for a single keyword category is crucial to the success of the adaptation of supervised approaches for large scale image annotation because (a) there are viewpoint, scale and lighting variations, and more seriously (b) some keyword categories are normally holistic or functional in nature, resulting in rich varieties of visual appearances. Second, the feature vectors generated by the segmentation algorithms are still far from desirable. Often, the object of interest is segmented into different parts. Therefore, it is interesting to investigate how useful modelling a concept with multiple visual parts is for image annotation.

In this paper, we address the fundamental issues of utilizing multi-facet visual concept points to characterize keyword categories. For clarity, we term each keyword as a category and each category is basically formed by multiple concepts. Every concept point can further contain one or several visual parts. The highlights of this work are as follows. First, data mining technique is proposed to learn multiple concepts to effectively handle multi-facet keyword categories. Each concept is composed of several visual parts to characterize its appearance. The learnt concepts are further ranked either by a discriminative measure or a diverse density measure. Second, region independence is not assumed in our approach. Most approaches [3,4,5] assume the process that generates the regions b_i are independent where $P(b_1...b_n) = \prod_{i=1}^{n} P(b_i)$ and neglect the correlation among visual parts. Our approach avoids this drawback by processing groups of visual parts. Third, the proposed technique is computationally efficient and scales well with data size compared to methods such cross-media relevance model (CMRM) [4] that does not scale well with the training set size.

2 Multi-facet Visual Appearance Model

We model the appearance model of a keyword category as a lattice structure shown in Figure 1. The lattice captures the multi-facet concepts while presenting them at different levels of visual granularities. In this structure, each node represents a concept which captures one or several visual parts. Basically the nodes at a higher level carry more specific, and thus more discriminant, categorical information for image annotation. In this section, we first propose techniques to mine, while simplifying, the lattice structure. Two novel measures, from the perspectives of discriminativeness and diverse density, are presented to encode the usefulness of each concept in a probabilistic manner. Image annotation is then performed by capitalizing on the level-sensitive information provided by the hierarchical structure of lattice representation.

2.1 Apriori Based Concept Mining

The different visual appearances of a keyword category can be effectively modelled by a lattice of visual part groups. To generate the structure, all the visual parts in the images of the same category are extracted and then used to create all permutations of visual part groups hierarchically. Modelling each category with a full lattice structure is inefficient because clearly a portion of the nodes in the lattice is uninteresting and does not correspond to the semantics of the keyword category. The extraction of the interesting subset of the lattice structure is posed as a data mining problem [11] where the *Apriori* algorithm can be used to mine for the significant sub-structure of lattice which contains frequent, and thus likely to be more interesting, concepts.

We use a discrete image representation as in [1,2,3]. Regions are extracted from the images using a general purpose segmentation algorithm. Features such as color, texture, position and shape information are computed for these regions and K-means is applied on the collection of all features to form clusters of features known as *blobs*. A training image is represented by a set of blobs $B_I = \{b_1...b_m\}$ and a word list $W_I = \{w_1...w_n\}$. To be consistent with data mining terminologies, we term a candidate concept (a collection of one or more blobs) as a *blobset* and a positive training image B_I as a *transaction* (of blobs). A n-blobset is a set of n blobs. The *level* of a blobset is the number of items in the blobset, which is n. One property of a blobset is its *support count*, which refers to the cardinality of a blobset in the set of all transactions, $T = \{B_1...B_M\}$. A *frequent* blobset is a blobset which satisfies a minimum support count *min_sup_count*.

We are interested to mine all frequent blobsets from the transactions of positive training set. Initially, all frequent 1-blobsets are extracted from T. Then, all the subsequent n-blobsets are recursively generated from the initial list. A data set that contains k 1-blobsets can potentially generate up to $2^k - 1$ frequent blobsets, resulting in a lattice model as shown in Figure 1. We use the well-known Apriori principle to generate only frequent blobsets.

Theorem 1 (*Apriori* Principle). *If an itemset is frequent, then all of its subsets must also be frequent.*

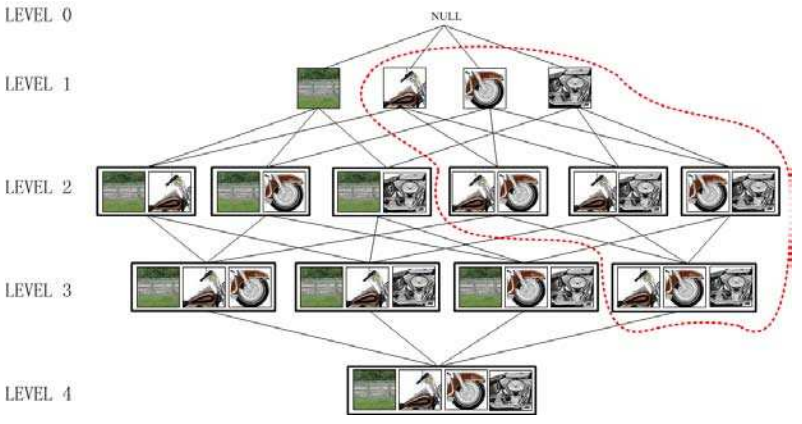


Fig. 1. A simplified lattice structure for representing the multiple visual appearance of a keyword category 'motorcycle' with a four 1-blobset b_i. Apriori algorithm extracts the visually significant blobsets (dotted red line) from the full lattice-set.

As illustrated in Figure 1, if the blobset b_1b_2 is infrequent, then all its supersets such as $b_1b_2b_3$, $b_1b_2b_4$ and $b_1b_2b_3b_4$ are also infrequent. Thus we can utilize the anti-monotonic property of the support count of blobs to prune the generation of uninteresting blobset, resulting in a set of frequent concepts as bounded by the dotted lines in Figure 1. Although *min_sup_count* is determined empirically, it is not critical and is more for the purpose of speed optimization.

The frequent blobset generation has its significance in several aspects. First, the Apriori algorithm mines for a compact and succinct set of concepts to fully model the different visual appearances of a keyword category. Similar to [1], we are using the co-occurrence between words and blobs. However, the co-occurrence model does not explicitly learn a class model for each word and only the co-occurrence between a blob and a word is considered. The model is incomplete in the sense that it does not model all the variations of visual appearances of a keyword category. Second, blobsets of different levels describe a concept with multiple visual parts. In this aspect, we also avoid the assumption that blobs are mutually independent of each other. The underlying assumptions by the relevance models [3,4,5] that the process that generates each blob is independent become invalid when some blobs which are correlated with some other blobs. Third, we can interpret the subset of a blobset as the incomplete or occluded version of the blobset. For example, the blobset $b1b2$ could be interpreted as the occluded version of its superset blobset $b1b2b3$ where the blob $b3$ is occluded. Thus, the Apriori algorithm creates a succinct model to represent a complete description of a keyword category.

2.2 Characterizing Concept Uniqueness

The lattice structure provides a platform to effectively model the multi-facet appearances of a keyword category. However, how do we determine the ownership

of a blobset? Apparently, the frequency of a blobset in the positive examples of a keyword is not a reliable cue if it is common across the whole training set. This scenario is analogous to the use of *idf* in text document retrieval. A reliable measure for characterizing the uniqueness of concept is "discriminativeness" where a blobset is unique if it is only frequent in positive images but rare, as a whole, in the training samples. However, a low discriminative measure does not necessarily rule out the usefulness of blobsets in describing a keyword. This is true for keyword categories that always co-exist together such as the keyword category "plane" is semantically related to "sky" and they share some similar visual content which could still be rare in other keyword categories. For such cases, additional information such as the ratio of the blobset in the positive examples, negative examples and total examples is useful. Therefore, we propose another measure "diverse density", inspired by the fundamental of multiple instance learning (MIL) [12], to handle such cases.

Discriminative Measure. The first measure $conf$ is a measure of how discriminative the concept is in describing a keyword category w_i. It assigns higher confidence values to concepts which appear only frequently in the positive images with respect to the whole training set. It is an asymmetric measure where only the presence of a blob is regarded as important. It ignores the blob's relative size with respect to the positive, negative and the whole training set. The $conf$ of c, a candidate concept for the keyword w_i, could be formulated as

$$conf(c, w_i) = \frac{|c|_+}{|c|_J} \tag{1}$$

where $|\bullet|_S$ denotes the cardinality of $\bullet$ in a set S. $+$ and $-$ denote the positive and negative training set for the keyword category w_i and J denotes the whole training set.

Diverse Density Measure. Motivated by the classical MIL diverse density algorithm [12], the second measure dd assigns the similarity based on how frequent a concept is in positive images and how infrequent it is in negative images. It takes into consideration how discriminative the concept is ($conf$ measure) with respect to the sizes of the positive, negative and the whole training sets. For instance, a keyword category with a higher number of training examples is assigned a higher dd value because there are more examples to support the presence of the category. The diverse density dd of a concept c of the keyword w_i is defined as

$$dd(c, w_i) = P(c|w_i, +)P(c|w_i, -)P(w_i)conf(c, w_i) \tag{2}$$

where $P(c|w_i, +)$ is the ratio of the number of concept c in all positive images, $P(c|w_i, -)$ is the ratio of images not containing the concept c in all negative images and $P(w_i)$ is the the ratio of the training images for keyword w_i over all images.

Both the $conf$ and dd measures have their own strength. The $conf$ measure looks into the exclusiveness of a blobset where higher values are assigned to blobsets which are mainly found in only one particular keyword category regardless

of the cardinality of training set. In other words, it ignores the global statistics of the blobset in the training set. In this aspect, it is similar to the CMRM platform [3] and is well-suited for keyword categories with diverse visual features, such as "boat" and "house". Since visual correlation is expected to be weak for such keyword categories, $conf$ is a better measure by highlighting the visual parts which are unique to the keyword only. The dd measure emphasizes on the global statistics of the candidate concept. In this aspect, it is similar to the MIL platform [12]. It is more useful for keyword categories with prominent visual features, such as "tiger" and "forest", where strong visual correlation exists among the positive examples. Besides, the measure is better positioned to handle semantically related keywords with overlapping visual parts such as "plane" and "sky". The localized overlapping of visual parts have a negative effect on $conf$ but less profound impact on the dd since the visual parts would still be rare statistically in the set of all negative examples as defined in $P(c|w_i, -)$.

2.3 Level-Sensitive Annotation

As we move down in the lattice level as shown in Figure 1, concepts become rarer, more discriminative and have less chance of happening by chance. Basically concepts at higher-level are capable of eliciting more evidence in terms of the number of visual parts to support their keyword category. We thus tap into this implicit feature of the lattice and propose a novel *level-sensitive* annotation. The approach strives to prioritize the scores of a concept according to its level, or more specifically the number of visual parts residing in a concept.

To determine the conditional probability of a keyword category w_i given an unannotated image I, $P(w_i|I)$, we select the best concept from the pool of candidate concepts of the category that matches the unannotated image. A concept matches the unannotated image when all the blobs in a concept are present in the unannotated image. Depending on which measure being used, we embed the notion of level-sensitivity into $P(w_i|I)$ using the following formulations

$$P(w_i|I) = \max_{c \in \mathbf{C}_{w_i}} \{dd(c, w_i) + L(c)\} \tag{3}$$

or

$$P(w_i|I) = \max_{c \in \mathbf{C}_{w_i}} \{conf(c, w_i) + L(c)\} \tag{4}$$

where $\mathbf{C}_{w_i}$ is the set of all frequent visual concepts of the word w_i learnt during the Apriori step. $L(c)$, representing the level of concept c, is aimed for assigning higher score to c at higher level. Then, annotation is performed based on a maximum a posteriori (MAP) criterion as follows

$$\hat{w} = \arg \max_{w_i \in V} P(w_i|I) \tag{5}$$

where the keyword category with the highest conditional probability is assigned to the unannotated image. For multiple annotations, the top-N keywords are selected.

3 Experiment and Results

3.1 Data Set and Evaluation

To evaluate the effectiveness of our approach, we use the data set provided by Duygulu *et al.* [2]. A total of 4,500 images is used as training set and the remaining 500 images as testing set. Each image is annotated with 1-5 keywords with a total vocabulary of 371 keywords. Images are segmented into 5-10 regions using normalized cut [14]. A 36-dimensional feature vector, which is composed of color, texture, mean oriented energy and other features, is extracted for each region. The set of all feature vectors is then quantized by K-means into 500 blobs. Details of the feature extraction process can be found in [2]. We follow the experimental methodology used by [2,3]. Given an unannotated image I from the test set, we use Equation 3 or 4 to arrive at the conditional probability $P(w_i|I)$. We perform a ranking and select the top 5 words as an annotation of image I using the recall and precision measure. Recall is the number correctly annotated images divided by the number of relevant images in the ground truth. Precision is the number of correctly annotated images divided by the total number of images annotated with that particular word. Recall and precision are then averaged over the word set. As in [3], we report the results on two sets of words, the subset of 49 best words and the complete set of all 260 words in the testing set.

3.2 Performance Comparison

We compare our proposed approach with the Co-occurrence (CO) [1], Machine Translation (MT) [2] and Cross-Media Relevance Model (CMRM) [3]. Our approach is named separately as APR_CF and APR_DD which uses the $conf$ and dd measure, respectively. During learning, we learn an average of 100 concepts up to a maximum level of 5 for each keyword category. The performance of the five tested approaches are summarized in Table 1 and illustrated in Figure 2. Our approach, although using co-occurrence between blobs and words, has significantly better performance compared to the CO and MT models. Both APR_CF and APR_DD are comparable to CMRM. They perform better in terms of average recall, and with a higher number of images with at least one correct annotation (i.e., recall > 0). Our approach, however, has a lower precision, partly because the Corel training set of different keyword categories is not well-balanced in terms of number of training examples. We notice that the keyword categories with too few training examples (some as few as 1) end up with a trivial lattice structure, impacting the precision performance. It is also observed that there are no notable differences in performance between the $conf$ and dd measures. We investigate the results and find that this is attributed to the level-sensitive matching scheme which filters out the noisier lower-level matchings.

Sensitivity of lattice height. The height (number of levels) of a lattice formed by the frequent concepts indeed impacts the performance of annotation. Here, we define "height" as the maximum level in a lattice that the frequent concepts of its category reach. The higher the lattice of a category, the more discriminant

Table 1. Performance of our approach (APR_DD and APR_CF) with Co-occurrence (CO), Machine Translation(MT) and Cross-Media Relevance Model(CMRM)

	All 260 Words		Best 49 Words		Recall>0
	Avg. Re.	Avg. Pr.	Avg. Re.	Avg.Pr.	
CO	0.02	0.03	-	-	19
MT	0.04	0.06	0.34	0.20	49
CMRM	0.09	0.10	0.48	0.40	66
APR_DD	0.11	0.07	0.50	0.26	75
APR_CF	0.11	0.07	0.50	0.27	76

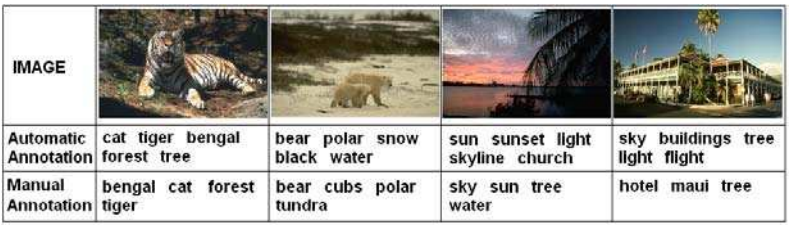


Fig. 2. Some annotation results of our approach compared to manual annotations

the concepts being learnt, and thus leads to a more reliable annotation. The lattice height of different keywords varies depending on the visual appearances of the keyword images, and also partly the number of training examples. In this experiment, we group the keyword categories according to the height of their lattice models. For each group, we compute their average recall, precision and percentage of words > 0. The results are shown in Table 2. Apparently, all the performance measures improve with the increase of the height of the lattice model. This shows that the height of lattice, which translates to the number of visual parts in a concept of a keyword category, is useful in describing the semantics of an image.

Table 2. Performance of APR_DD and APR_CF for keyword groups based on the height of their lattice model

Height of lattice model	APR_DD					APR_CF				
	1	2	3	4	5	1	2	3	4	5
#keywords in group	34	50	99	54	23	34	50	99	54	23
%words with recall>0	0	0.22	0.25	0.46	0.60	0	0.22	0.26	0.46	0.61
Average recall	0	0.05	0.08	0.16	0.40	0	0.05	0.08	0.16	0.41
Average precision	0	0.07	0.05	0.09	0.17	0	0.09	0.05	0.09	0.16

Effectiveness of Level-Sensitive Annotation. To assess the performance improvement due to level-sensitive annotation, we compare the cases with and without the level-sensitive matching. When the level-sensitive matching is disabled, the annotation is performed purely on the dd or $conf$ measure. The result

shown in Table 3 clearly indicates that the performance decreases without level sensitivity matching. Compared to the lower level blobsets, higher level blobsets are more discriminant and thus provides more visual evidence to support the presence of a keyword category. In addition, the $conf$ measure is found to be less sensitive to the concept level compared to dd. We believe it is because most of the keyword categories in the Corel data set have diverse range of visual appearances. As discussed in Section 2.2, the $conf$ measure is more robust to such data set than the dd measure. Level-sensitive matching is able to reduce this gap and improve the performance of both measures through selective matching.

Table 3. Performance of APR_DD and APR_CF *without* level-sensitive matching

	All 260 Words		Best 49 Words		Re.>0
	Avg. Re.	Avg. Pr.	Avg. Re.	Avg.Pr.	
APR_DD	0.04	0.03	0.32	0.25	38
APR_CF	0.06	0.04	0.47	0.23	60

Speed Efficiency. The complexity of our approach is $O(W \times N \times C)$ per image, where W is number of words in the vocabulary, N is the number of visual parts in a concept and C is the average number of concept points per word category. As a comparison, the CMRM has a time complexity of $O(W \times R \times J)$, where J is the average training sample of keywords and R is the number of regions in the data set. Obviously, $R > N$. For the Corel data set, $R = 9$ and $N = 3$. Besides, notice that $J > C$ in general since J is required to be large for reliable learning. In the case of Corel data set, the training data of keyword categories varies a lot, $1 \leq J \leq 1004$. In our current implementation, $C = 100$ on average. Our approach requires only 32.48 seconds for training and 1.61 seconds for annotating all the 500 testing images on a Pentium 4 3GHz and 512MB of memory.

4 Conclusions

In this paper, we propose a new approach for image annotation by learning the multiple concept points of keyword categories. Each concept is supported by one or more visual parts and mined using the Apriori principle. Under the guidance of lattice structure, the level-sensitive selection of concepts based on the discriminative and diverse density measure is exploited for effective image annotation. Experiment results show that learning multi visual parts in a model like lattice structure is useful in capturing the semantics of keyword categories.

Acknowledgments. The work described in this paper was fully supported by two grants from the Research Grants Council of the Hong Kong Special Administrative Region, China (CityU 118905 and CityU 118906).

References

1. Mori, Y., Takahashi, H., Oka, R.: Image-to-word transformation based on dividing and vector quantizing images with words. MIRSM. (1999)
2. Duygulu, D., Barnard, K., Freitas, N. de, Forsyth, D.: Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. ECCV. (2002) 97–112
3. Jeon, J., Lavrenko, V., Manmatha, R.: Automatic image annotation and retrieval using cross-media relevance models. SIGIR. (2003) 119–126
4. Lavrenko, V., Manmatha, R., Jeon, J.: A model for learning the semantics of pictures. NIPS. (2003)
5. Feng, S. L., Lavrenko, V., Manmatha, R.: Multiple Bernoulli relevance models for image and video annotation. CVPR. (2004) 1002–1009
6. Blei, D., Jordan, M. I.: Modeling annotated data. SIGIR. (2003) 127–134
7. Carneiro, G., Vasconcelos, N.: Formulating semantic image annotation as a supervised learning problem. CVPR. **2**(2005) 163–168
8. Ghoshal, A., Ircing, P., Khudanpur, S.: Hidden Markov Models for Automatic Annotation and Content-Based Retrieval of Images and Video. SIGIR. (2005) 544–551
9. Szummer, M., Picard, R.: Indoor-Outdoor Image Classification. Workshop in Content-based Access to Image and Video Databases. (1998)
10. Shi, R., Chua, T. S., Lee, C. H., Gao, S.: Bayesian Learning of Hierarchical Multinomial Mixture Models of Concepts for Automatic Image Annotation. CIVR. (2006) 102-112.
11. Tan, P., Steinbach, M., Kumar, V.: Introduction to Data Mining. Addison Wesley. (2006)
12. Maron, O., Raton, A. L.: Multiple-Instance Learning for Natural Scene Classification. ICML. (1998) 341–349
13. Zhang, Q., Yu, W., Goldman, S. A., Fritts, J. E.: Content-Based Retrieval Using Multiple-Instance Learning. IMCL. (2002) 682–689
14. Shi, Y. Malik, J.: Normalized cuts and image segmentation. CVPR (1997) 731–737

Automatic Video Annotation and Retrieval Based on Bayesian Inference

Fangshi Wang[1,2], De Xu[1], Wei Lu[2], and Weixin Wu[1]

[1] School of Computer &Information Technology, Beijing Jiaotong University
[2] School of Software, Beijing Jiaotong University, Beijing China 100044
{fshwang,dxu,wlu}@bjtu.edu.cn, bjbluesky@126.com

Abstract. Retrieving videos by key words requires semantic knowledge of the videos. However, manual video annotation is very costly and time consuming. Most works reported in literatures focus on annotating a video shot with either only one semantic concept or a fixed number of words. In this paper, we propose a new approach to automatically annotate a video shot with a non-fixed number of semantic concepts and to retrieve videos based on text queries. First, a simple but efficient method is presented to automatically extract Semantic Candidate Set (SCS) for a video shot based on visual features. Then, the final annotation set is obtained from SCS by Bayesian Inference. Finally, a new way is proposed to rank the retrieved key frames according to the probabilities obtained during Bayesian Inference. Experiments show that our method is useful in automatically annotating video shots and retrieving videos by key words.

1 Introduction

Labeling the semantic content of videos with a set of keywords is known as video annotation. Annotation is used primarily for video database management, especially for video retrieval. Annotated videos can usually be found using keyword-based search, while non-annotated videos can be extremely difficult to find in large databases. Since content-based video retrieval is still not very accurate or robust, most people would prefer to pose text queries and find videos relevant to those queries. For example, one should be able to pose a query like “cars on a road”. This needs to bridge the semantic gap between the low-level feature descriptions and the semantic descriptions of multimedia. The traditional “low-tech” solution to this problem is to annotate each image manually with keywords and then search on those keywords using a conventional text search engine. The main disadvantage with manual annotations is the cost and difficulty of scaling it to large numbers of images. Automatically annotating images/videos would solve this problem while still retaining the advantages of a semantic search [1].

In most previous research works, the task of annotating a non-annotated video can be viewed formally as a classification problem, we must make a yes/no decision for each word in the vocabulary [2][3][4]. Experts or users specify several classes, system can construct one or several classifiers through learning from the training set which is built manually by users. The visual features of a new video are extracted automatically and input into the well-trained classifiers. Then the result of the classification is

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 279 – 288, 2007.
© Springer-Verlag Berlin Heidelberg 2007

the semantic annotation of the new shot. The classes defined in such method are mutually exclusive, so each video shot can have only one semantic concept.

In fact, one concept is not enough to fully summarize a video shot with rich contents. A video shot could include more than one concept. For example, we can see “land”, “sky” and “cloud” in the first picture of Figure 1. Three concepts are needed to describe the shot. It should not alone belong to any one of the three classes. Obviously, semantic concepts do not occur independently or are not isolate from each other, and the mutual information between them should be taken into account in order to make the annotation complete.



Fig. 1. Examples of selected key frames with complete annotation in the training set

Intuitively it is clear that the presence of a certain concept suggests a high possibility of detecting certain other concepts. Similarly some concepts are less likely to occur in the presence of others. The detection of “car” boosts the chances of detecting “road”, and reduces the chances of detecting “waterfall”. It might also be possible to detect some concepts and infer more complex concepts based on their relation with the detected ones. Naphade [5] proposed the MultiNet as a way to represent higher level probabilistic dependencies between concepts. However, both the classes and structure of the classification frameworks were either decided by experts or specified by users. Moreover, the structure will become very large with the increase of the class number. If there are n classes, there will be n variable nodes and $n(n\text{-}1)/2$ function nodes and $n(n\text{-}1)$ edges in the MultiNet.

More general approaches attempt to annotate new key frames with concepts in the annotations of training set. MediaNet[6] can automatically select the salient classes from annotated images and discover the relationship between concepts by using external knowledge resources from WordNet. However, the relationships between concepts in MediaNet are too complex. There are not only perceptual relationships such as "equivalent", "specializes", "co-occurs", and "overlaps", but also semantic relationships such as “Synonymy / Antonymy“, “Hypernymy / Hyponymy”, “Meronymy / Holonymy”, “Troponymy”, “Entailment“, which are summarized into a small subclass of all these relationships by clustering subsequently. In his summarized MultiNet, there is only "specializes" relationship such as “man” is a subtype of “hominid”. His main attention is on analyzing the sense of a word and generating the "specializes" relationship and so on.

Jeon *et al*. [7] proposed a cross-media relevance model (CMRM) to learn the joint distribution of blobs and words in order to perform both image annotation and ranked retrieval. Lavrenko *et al*. [8] proposed a Continuous-space Relevance Model (CRM) to computes a joint probability of image features over different regions in an image using a training set and uses this joint probability to annotate and retrieve images. Feng *et al*. [1] learned a statistical generative model called Multiple-Bernoulli Relevance Model (MBRM) using a set of annotated training images to automatically annotating and retrieving images/videos. Words are modeled using a multiple Bernoulli process and images modeled using a kernel density estimate. But all annotation obtained by the three models has the fixed number of words. The length of the annotation is determined by users and has a direct influence on the recall and precision. In addition, it is not reasonable to label every shot with a fixed number of concepts, no matter whether the shot content is rich or not.

In the application of annotating the video, we only concern about whether concept B is also present in the same frame if concept A is present. So we want to discover the co-occurrence relationship among multiple concepts.

In order to overcome the above shortcomings, a new method is proposed to annotate a video shot with a varied number of concepts. This paper is organized as follows. A simple but efficient approach is proposed to extract Semantic Candidate Set (SCS) in section 2. Section 3 describes a way to select the final annotation set from the SCS by Bayesian Inference and automatically annotate a video shot with a varied number of concepts. Section 4 introduces a probabilistic method to rank the retrieved frames based on Bayesian Inference. The experimental results are given in section 5. Finally, section 6 concludes the paper.

2 Extracting the Semantic Candidate Set for a New Shot

The Semantic Candidate Set (SCS) is a set of *N* most probable concepts according to the visual features. It is supposed that all concepts actually annotated for a testing frame are in its SCS, then the actual concepts are chosen from the SCS by Bayesian Inference.

A training set is constructed by manually annotating the key frames of videos shots and regarded as Ground Truth (GT). There is an annotation with from 1 to 4 concepts for each key frame in the training set. Examples of selected key frames with completed annotation are shown in Figure 1**.** Each concept is considered a semantic class. Without loss of generality, suppose there are *n* semantic concepts in the training set. A simple method is introduced to obtain the SCS.

First, the center of each semantic class is calculated as follows.

$$C_{S,k} = \frac{1}{|T_S|} \sum_{f_j \in T_S} f_{j,k} \qquad (k=0,\ldots,\dim-1) \tag{1}$$

where *dim* is the dimension of the visual feature vector, T_S is the set of the samples with concept S in their semantic annotation, $|T_S|$ is the number of the samples in T_S, f_j is the *j*th frame in T_S , $f_{j,k}$ is the *k*th visual feature element of frame f_j and $C_{S,k}$ the *k*th visual feature element of the class center of concept S.

Then, formula (2) is used to compute the distances between the key frame F of a new shot and every semantic class center, which are denoted as $Dist[1]$, $Dist[2],\ldots,Dist[n]$.

$$Dist[i] = \sqrt{\sum_{k=0}^{\dim-1} (F_k - C_{i,k})^2} \quad (i = 1, ..., n) \tag{2}$$

where F_k is the kth visual feature element of the new key frame.

Finally, $Dist[1,\ldots,n]$ is sorted from small to large. N most probable concepts, denoted as S1,…, SN corresponding the N smallest distances, consist of SCS. In our experiment, the result is the best for $N = 4$. S1 is regarded as the first concept of the new shot. This process of extracting SCS is named Semantic Class Centre Method (SCCM).

3 Obtaining the Final Annotation Set Using Bayesian Inference

Having obtained the semantic candidate set (SCS) and the first concept S1 of the new shot, we should have a way to determine which of the others in SCS are also present in the same shot. Bayesian inference is used to calculate the conditional probabilities of the other concepts given S1. Suppose that the initial current evidence set is CE={S1=1} (1 means presence, 0 means absence). If $P(S2=1|CE) > \sigma$, then S2 will be assigned to the new shot; If $P(S3=1|CE) > \sigma$, If $P(S4=1|CE) > \sigma$, then S4 will be assigned to the new shot, and so on. We obtain the final annotation set of the shot by dynastic Bayesian inference because CE varies during inferring.

Bayesian Network (BN), also known as Belief Network, is a graphical model that efficiently encodes the joint probability distribution over a set of random variables. Bayesian Network is selected to construct the Semantic Network, in which a node represents a semantic concept and an edge represents the dependency relationship between two concepts.

Two reasons prompted the selection of Bayesian networks for learning statistical dependencies between concepts. First, there are algorithms to learn both the parameters and the topology of a Bayesian network. If the nodes in a Bayesian network represent concepts, then the algorithms are actually learning statistical relationships among the concepts. Second, once built, the Bayesian network can answer arbitrary probabilistic questions about the concepts (e.g., joint probability for the values of any two nodes).

Learning of Bayesian network includes two parts: learning the structure and learning the parameters given a structure. A three-phase construction mechanism is representative of Dependency Analysis Based Method, abbreviated to TPDA algorithm, and is used to construct Bayesian Network [9]. Having constructed the semantic network, the parameters, i.e. the conditional probability of each node, are learned by standard statistic method given the BN structure.

First, the directed graph of Bayesian Network is moralized and triangulated into a chordal graph. Then a join tree is built from the chordal graph, which consists of cliques node and separator sets (abbreviated as sepset). The belief potentials of each clique and sepset are initialized according to the conditional probabilities of the nodes

in Bayesian Network [10]. Suppose that the initialized join tree is denoted as *JT*. The procedure of obtaining the final annotation set (FAS) is proposed as follows.

Procedure Get_FAS (*JT*)

```
{ NE ={S1};  CE= ∅;  SCS ={S1,…,SN}
  While (NE is not empty) do
   { Input NE into JT and modify the potentials in the JT;
     Perform Global propagation to make the potentials
      of JT locally consistent;
     CE = CE∪NE;  NE = ∅;
     for (each concept Si in SCS) do
      if ((Si not in CE )and (P(Si=1|CE)>σ )) then
        { NE = NE ∪{Si};
          SP[f][Si]= P(Si=1|CE);
        }
   }
}
```

where *CE* is used to store the current evidence, and it is the final annotation set (FAS) of a shot after the procedure stops. SP[*f*][Si] is used to store the probability of annotating the frame *f* with concept Si.

The difference between **Get_FAS** and Huang's algorithm [10] lies in introducing a variable *NE* used to store the newly generated evidence after inferring every time in order to judge when to stop the inference procedure. Huang [10] did not give the stopping condition and it is determined by human. We give an automatic control mechanism that can stop the inferring procedure when generating no more new evidence. Initially, *NE* only includes one concept S1. After *NE* is input into *JT* and incorporated to *CE*, it is set to empty subsequently. After inferring given *CE* each time, the concept whose conditional probability is larger than σ becomes new evidence and is put into *NE*. Repeat the above process until there is no more new evidence generated after inference.

Also, Huang [10] did not give a way to determine the threshold σ. In our experiment, the threshold σ is determined as follows.

$$\sigma = \frac{1}{|\mathrm{TS}|} \sum_{k=1}^{|\mathrm{TS}|} \sum_{i=2}^{\#C_k} P(S_i^k \mid S_1^k, ..., S_{i-1}^k) \tag{3}$$

where |TS| is the number of samples in training set, $\#C_k$ is the number of actual concepts in ground-truth (GT) annotation of the *k*th sample, S_i^k is the *i*th actual concept of the *k*th sample, $P(S_i^k \mid S_1^k, ..., S_{i-1}^k)$ is the conditional probability of S_i^k given $S_1^k, ..., S_{i-1}^k$. In a word, σ is the average conditional probability of all concepts of Ground Truth over all samples in the training set. The threshold can be calculated automatically and adaptively for different data sets, avoiding setting the threshold manually.

4 Ranked Retrieval Based on Annotation

The task of video retrieval is similar to the general ad-hoc retrieval problem. We are given a text query $Q=\{w_1, w_2, \ldots, w_k\}$ and a collection V of key frames of videos. The goal is to retrieve the video key frames that contain concept set Q in V.

A simple approach to retrieving videos is to annotate each key frame in V using the techniques proposed in section 2 and section 3 with a small number of concepts. We could then index the annotations and perform text retrieval in the usual manner. This approach is very straightforward. However, it has a disadvantage that does not allow us to perform ranked retrieval. This is due to the binary nature of concept occurrence in automatic annotations: a concept either is or is not assigned to the key frame. When the annotations of many retrieved frames contain the same number of concepts, document-length normalization will not differentiate between these key frames. As a result, all frames containing the same number of concepts are likely to receive the same score.

Probabilistic annotation can be used to rank the relevant key frames (here 'relevant key frames' are the ones that contain all query concepts in their ground-truth annotation). In section 3 we developed a technique that assigns a probability $P(S=1|CE)$ to every concept S in the annotation. We score the key frames by the probability that a query would be observed. Given the query $Q=\{w_1, w_2, \ldots, w_k\}$ and a frames $f \in V$, the probability of containing Q in frame f is:

$$P(Q|f)=\prod_{j=1}^{k} P(w_j|f) \tag{4}$$

where $P(w_j \mid f)$ has already been computed and stored in $SP[f][w_j]$ in section 3. all retrieved frames are ranked according to $P(Q|f)$ from large to small.

5 Experimental Results

We have chosen videos of different genres including landscape, city and animal from website www.open-video.org to create a database of a few hours of videos. Data from 35 video clips has been used for the experiments. All algorithms were implemented by VC++ on a PC machine with AMD Athlon 2500+ CPU, 256M memory and Windows XP environment.

First, we use the tool VideoAnnEx [11] developed by IBM to partition every video clip into several shots and to manually annotate the shots. The key frames of each shot are extracted automatically using the method in [12] to form the samples set. The perceptual features such as HSV accumulated Histogram and Edge Histogram are extracted automatically from the sample set and stored into a video database after being normalized. In our experiments, there are 544 key frames for training and 263 key frames for testing. Examples of selected key frames with complete annotation are shown in Figure 1. 14 different concepts are extracted automatically from the ground truth annotation of the training set built manually, which are shown in table 1. So our system can also work if the data set varies and more concepts are added.

Table 1. The presence frequency of each concept in training set (%)

concept	**car**	**road**	**bridge**	**building**	**waterfall**	**water**	**boat**
percentage	3.08	6.84	2.66	19.39	1.14	42.97	6.08
concept	**cloud**	**sky**	**snow**	**mountain**	**greenery**	**land**	**animal**
percentage	13.31	44.87	6.84	14.45	32.7	25.48	25.48

5.1 Results: Automatic Videoe Annotation

In this section we evaluate the performance of our method on the task of automatic video annotation. We are given an unannotated key frame *f* and are asked to automatically produce an annotation. The automatic annotation is then compared to the manual ground-truth annotation (GT).

It is said in [6] that different classifiers were evaluated including k-nearest neighbors, one-layer neural network, and mixture of experts, of which k-nearest neighbor was shown to outperform the rest. So we compare our method of selecting the semantic candidate set described in section 2 with K Nearest Neighbor (KNN) and Naïve Bayesian (NB) classifier. Four most probable concepts are also chosen for KNN and NB to build their SCSs. We use three standards to measure the performance of the three methods as follows.

$$S1_\ first = \frac{N_{correct_first}}{N_{first}} \quad (5) \qquad precision = \frac{N_{correct}}{N_{label}} \quad (6) \qquad recall = \frac{N_{correct}}{N_{ground_truth}} \quad (7)$$

where N_{first} is the number of samples with concept S in the first position in GT, $N_{correst_first}$ is the number of the samples whose first concept in SCS is the same as that in GT, i.e. S, $N_{correct}$ is the number of samples having a given concept S in its SCS correctly, N_{label} is the number of samples having that concept in SCS, N_{ground_truth} is the number of samples having that concept in GT.

Table 2 shows the mean $S1_\ first$ (MF), the mean precision (MP) and the mean recall (MR) over all concepts in SCS. KNN consistently outperforms NB, which conforms to the conclusion drawn by Benítez [6]. It also indicates that all metrics of SCCM are the biggest among the three methods, especially MR and MF of SCCM are much bigger than those of NB and KNN methods. So we have two reasons to expect that the annotation performance obtained by SCCM could be the best among the three methods. First, the larger the recall is, the more the SCS covers the correct concepts. Second, the first concept in SCS is the first evidence during Bayesian inference and the accuracy of the evidence is very crucial to the annotating results.

After obtaining the SCS and the first concept of a testing frame, the system obtains the other concepts coexisting in the same frame from SCS by Bayesian Inference detailed in section 3. We also measure the performance of annotating by formula (6) and (7), where $N_{correct}$ is the number of the correct concepts that the system automatically annotated for a testing frame, N_{label} is the total number of the concepts that the system automatically annotates for a testing frame, N_{ground_truth} is the number of the actual concepts of a testing frame in GT.

Table 3 shows the average results of annotating using different methods of extracting SCS. First, it indicates that the inference performance on SCS obtained using

SCCM is significantly higher than that using NB and KNN algorithm. Second, it shows that the recall values of all concepts in SCCM are larger than 0, and that of some concepts with low presence frequency in NB and KNN are zero. This is because KNN and NB algorithm are sensitive to the distribution of each concept in the training set. SCCM has not such problem because it is not sensitive to the presence frequency of each concept. Even a concept with low presence frequency can be assigned to a video shot only if its semantic class centre is close to the shot based on visual features. It indicates that SCCM is more robust than NB algorithm and KNN algorithm.

Table 2. The mean precision (MP), mean recall (MR) and mean $S1_first$ (MF) of semantic candidate set before inference

Method	MP	MR	MF	# concepts with recall>0	concepts with recall=0
NB	0.274	0.444	0.137	13	waterfall
KNN	**0.378**	0.490	0.182	12	waterfall , bridge
SCCM	**0.378**	**0.779**	**0.392**	**14**	\

Table 3. The result of automatic annotation. # con is the number of concepts with recall>0.

Method	MP	MR	# con	concepts with recall=0
NB	0.277	0.414	13	waterfall
KNN	0.355	0.470	11	car, waterfall , bridge
SCCM	**0.406**	**0.731**	14	\

Figure 2 shows the automatic annotation of several testing samples using three methods. We can see that there is a main concept in each frame. For example, the main concept in Figure 2(b) is 'animal', that in Figure 2(c) is 'bridge'. These concepts have lower presence frequency in training set. SCCM can correctly capture these concepts, but NB and KNN algorithm can not. SCCM can correctly annotate the main concepts in most cases, even though it annotates incorrectly with some concepts such as 'snow' in Figure 2(b).

In general, SCCM algorithm outperforms significantly NB algorithm and KNN algorithm in annotating video shots.

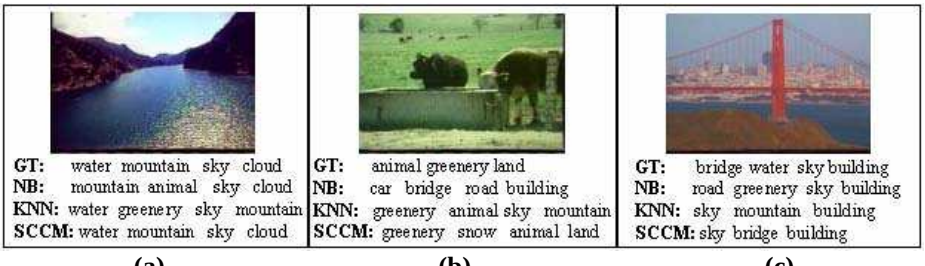


Fig. 2. Automatic annotation of three methods

5.2 Results: Ranked Retrieval of Videos

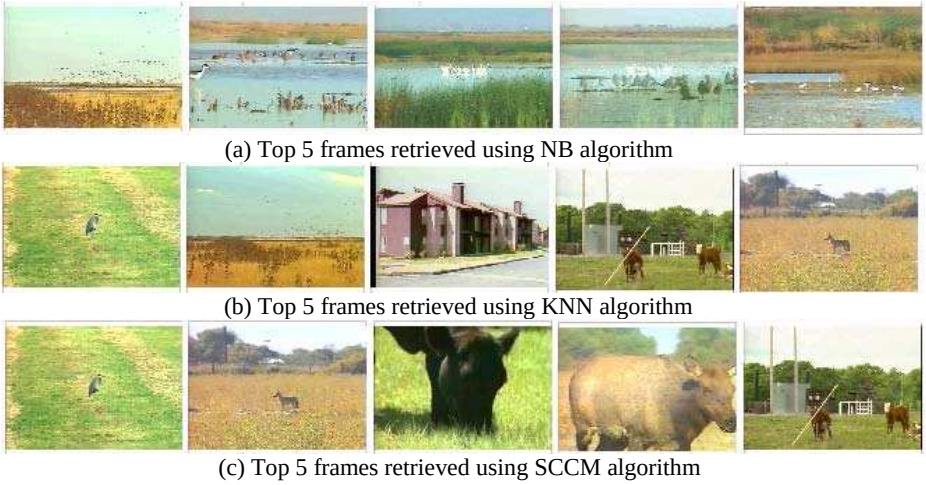
In this section we turn our attention to the problem of ranked retrieval of shots. In the retrieval setting we are given a text query $Q=\{w_1, w_2, \ldots, w_k\}$ and a testing collection of unannotated key frames. For each testing frame f we use equation (4) to get the conditional probability $P(Q|f)$. All frames in the collection are ranked according to the conditional likelihood $P(Q|f)$. In our retrieval experiments, we use four sets of queries, constructed from all 1-, 2-, 3- and 4-word combinations of concepts that occur at least twice in the testing set. A frame is considered relevant to a given query if its manual annotation (ground-truth) contains all of the query words. We use precision and recall averaged over the entire query set as our evaluation metrics.

Table 4 shows the performance of our method on the four query sets, contrasted with performance of the NB algorithm and KNN algorithm on the same data. We observe that our method substantially outperforms the NB algorithm and KNN algorithm on every query set except for the recall in 3-concept query, 0.229 of SCCM compared with 0.313 of KNN. It could be said SCCM has the same performance with KNN algorithm in this case because the sum of precision and recall in both methods are about equal (0.467 of KNN and 0.472 of SCCM).

Table 4. The performance on the retrieval task of three methods

#concepts	NB		KNN		SCCM	
	Precision	Recall	Precision	Recall	Precision	Recall
1-concept	0.277	0.414	0.361	0.471	**0.410**	**0.731**
2-concept	0.168	0.313	0.194	0.332	**0.261**	**0.435**
3-concept	0.117	0.154	0.178	**0.289**	**0.258**	0.214
4-concept	0.111	0.125	0.139	**0.313**	**0.25**	**0.313**

Figure 3 shows top 5 frames retrieved in response to the text query “animal greenery”.



(a) Top 5 frames retrieved using NB algorithm

(b) Top 5 frames retrieved using KNN algorithm

(c) Top 5 frames retrieved using SCCM algorithm

Fig. 3. Example of top 5 frames retrieved in response to the text query “animal greenery”

We do not compare the results of our method with that of other papers because it is unfair to make a direct quantitative comparison with their method using different data set. We do not obtain the standard video data set in TRECVID to measure the algorithm performance now.

6 Conclusion

We have proposed a new approach to automatically annotate a video shot with a varied number of semantic concepts and to retrieve videos based on text queries. There are two main contributions of this work. The first one is to propose a simple but efficient method to automatically extract the semantic candidate set based on visual features. The second one is to obtain the annotation with non-fixed length from SCS by Bayesian Inference. Experiments show that our method is useful in automatically annotating video shots and retrieving videos by key words.

References

[1] S. L. Feng, R. Manmatha and V. Lavrenko. Multiple Bernoulli Relevance Models for Image and Video Annotation. In IEEE Conference on Computer Vision and Pattern Recognition, pages 1002~1009, 2004.

[2] Yan rong : Probabilistic Models for Combining Diverse Knowledge Sources in Multimedia Retrieval. Dissertation of Carnegie Mellon University (2005).

[3] Barnard, K., P. Duygulu, and D. Forsyth, N. de Freitas, D. Blei, and M.I. Jordan : Matching Words and Pictures. Journal of Machine Learning Research (JMLR), Special Issue on Text and Images Vol. 3. (2003) 1107-1135.

[4] Tseng, B.T., C.-Y. Lin, M.R. Naphade, A. Natsev, and J.R. Smith : Normalized Classifier Fusion for Semantic Visual Concept Detection. In Proc. of Int. Conf. on Image Processing (ICIP-2003), Barcelona, Spain (2003) 14-17.

[5] Milind Ramesh Naphade : A Probabilistic Framework For Mapping Audio-visual Features to High-Level Semantics in Terms of Concepts and Context. Dissertation of the University of Illinois at Urbana-Champaign (2001)

[6] Ana Belén Benítez Jiménez : Multimedia Knowledge: Discovery, Classification, Browsing, and Retrieval. Dissertation of Columbia University (2005)

[7] J. Jeon, V. Lavrenko and R. Manmatha. Automatic Image Annotation and Retrieval using Cross-Media Relevance Models In Proceedings of the 26th Intl. ACM SIGIR Conf., pages 119–126, 2003

[8] V. Lavrenko, R. Manmatha and J. Jeon. A Model for Learning the Semantics of Pictures, In the Proceedings of the 16th Conference on Advances in Neural Information Processing Systems NIPS, 2004.

[9] Cheng, J., Greiner, R., Kelly,J. & Bell, D., Liu,w. Learning Belief Networks from Data: An Information Theory Based Approach. Artificial Intelligence. (2002)137(1-2):43-90.

[10] Cecil Huang: Inference in Belief Networks: A Procedural Guide. International Journal of Approximate Reasoning Vol. 11 New York (1994) 1-158.

[11] http://www.research.ibm.com/VideoAnnEx

[12] WANG Fangshi, XU De, WU Weixin. “A Cluster Algorithm of Automatic Key Frame Extraction Based on Adaptive Threshold”. Journal of Computer Research and Development Vol.42(10) (2005)1752-1757

Density-Based Image Vector Quantization Using a Genetic Algorithm

Chin-Chen Chang[1,2] and Chih-Yang Lin[2]

[1] Department of Information Engineering and Computer Science,
Feng Chia University, Taichung, Taiwan, 40724, R.O.C.
ccc@cs.ccu.edu.tw
[2] Department of Computer Science and Information Engineering
National Chung Cheng University, Chiayi, Taiwan, 621, R.O.C.
gary@cs.ccu.edu.tw

Abstract. Vector quantization (VQ) is a commonly used method in the compression of images and signals. The quality of VQ-encoded images heavily depends on the quality of the codebook. Conventional codebook training techniques are all based on the LBG (Linde-Buzo-Gray) method. However, LBG-based methods are noise sensitive and are not able to handle clusters of different shapes, sizes, and densities. In this paper, we propose a density-based clustering method that can identify arbitrary data shapes and exclude noises for codebook training. In order to rapidly approach an optimal solution, an improved version of a genetic algorithm is designed that demonstrates efficient initialization of codewords selection, crossover, and mutation. The experiments show that the proposed method is more robust in generating a common codebook than other LBG-based methods.

Keywords: Density-based clustering, genetic algorithms, vector quantization.

1 Introduction

Digital image compression methods are essential for storage and transmission while still maintaining acceptable fidelity or image quality. Among compression techniques, vector quantization (VQ) [6] is a popular lossy image compression method, primarily because of its reasonable compression rate and efficient decoding process. The main idea of VQ is to partition an input image into nonoverlapping blocks of uniform size, called codevectors, then approximate these codevectors using a limit set of vectors, called the codebook. Each vector c_i with index i in the codebook is called a codeword. In the encoding phase, the closest codeword c_i in the codebook is found for each codevector of the original image to minimize the distortion. Then, index i is used to encode the input vector, and finally the original image is represented by the indices of these closest codewords. Clearly, the quality of the VQ-compressed image is significantly influenced by the quality of the codebook. Although many codebook generation methods have been proposed [1, 4, 8, 9, 11, 12], constructing an optimal codebook is still in general difficult.

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 289 – 298, 2007.
© Springer-Verlag Berlin Heidelberg 2007

The most famous codebook design method is the LBG (Linde-Buzo-Gray) algorithm presented by Linde, Buzo and Gray in 1980 [9]. However, the LGB method makes only local changes to the previous clustering result and therefore is highly sensitive to the initialization of the codebook. To improve the initialization problem, TSVQ-based (tree structure vector quantization) methods [1, 8] and DAM (the diagonal axes method) [2] have been proposed. Although these improved methods have a higher probability of selecting better initial representative codewords, these deterministic algorithms still always converge toward the local optimal solution.

Several non-deterministic algorithms have been developed [4, 11, 12] to efficiently approximate the global optimal solution within a limited computation time. These evolution-based methods are devoted to minimizing the distance between the training vectors and their corresponding codewords. Assume that a codebook C contains m codewords $\{c_0, c_1,\ldots, c_{m-1}\}$ and that the training set contains N input vectors $\{b_0, b_1,\ldots, b_{N-1}\}$. The objective of an evolution-based method for codebook design is usually to minimize the average distortion function DF:

$$DF(C) = \frac{1}{N}\sum_{i=1}^{m}\sum_{j=1}^{N} u_{ij}\cdot\left\|b_j - c_i\right\|^2, \tag{1}$$

where $u_{ij} = 1$ if b_j belongs to the cluster c_i; otherwise, $u_{ij} = 0$; $\|b_j - c_i\|$ denotes the Euclidean distance between b_j and c_i.

However, these methods have two common defects: (1) the poor initialization of the codewords selection requires much time to find an approximate global optimal solution. If the initial codewords contain too many similar trial codevectors or useless codevectors, these methods usually converge to a local optimal solution; (2) these methods commonly consider only the distortion function as the objective function for the trial solutions and use the LBG method as the major improvement procedure. A codebook trained by this technique may not be suitable for images not used as training samples since the LBG method is noise sensitive and cannot handle nonglobular clusters or clusters of different sizes and densities.

In this paper, we propose a density-based method for codebook design that uses a genetic algorithm. Density-based clustering method has the following advantages [7]: (1) it can deal with arbitrary cluster shapes, (2) it can tackle clusters of different sizes and densities, and (3) it can identify noise in clusters. Applying the genetic algorithm helps the proposed method approximate the global optimal solution within finite iterations. To speed convergence toward a global optimal solution, we propose an efficient way to select representative codewords for the initial population, use a deterministic inherited method for crossover, and repair the defective codewords during the mutation operation. These techniques help the proposed method reach approximate global optimum in only a few iterations.

2 Previous Works

LBG (Linde-Buzo-Gray) is the most commonly used method for VQ codebook design. To begin, each training image is partitioned into n dimensional nonoverlapping blocks as input vectors. Then, the LBG algorithm randomly selects m

vectors from the input as the initial codebook. The final codebook can be obtained by repeatedly performing LBG iterations. The following sketch summarizes the LBG algorithm.

The LBG algorithm

Step 1: Randomly select m codewords from the input vectors to form the initial codebook.

Step 2: Map each input vector to the closest codeword from the codebook that yields the minimum distortion.

Step 3: Update each codeword by recalculating the centroid of the input vectors that map to it.

Step 4: If the average distortion between the input vectors and their codewords is less than a predefined threshold or the average distortion seems unchanged in recent iterations, then stop. Otherwise, go to Step 2.

In the LBG method, each iteration causes only a local change in the codebook, so a new codebook will not be significantly different from the previous one. Therefore, the initial codebook is the key factor in the LBG method; that is why the LBG method always converges toward a local minimum average distortion.

To improve the initialization of codewords selection, Hu and Chang proposed an efficient way [8] to generate a better codebook based on TSVQ (tree structure vector quantization). In this k-ary tree-structure design method, the codewords are progressively generated level by level to construct the complete k-ary tree. Initially, the root (i.e., level 0) is the centroid of all the input vectors. With this tree growing procedure, the next level of k^d children can be generated, where d represents the d-th level. Then, the LBG method is applied to the k^d children to generate k^d new centroids. The final codebook can be obtained by repeatedly performing the tree growing procedure with the LBG iteration. The detailed algorithm is stated as follows.

Progressive codebook training

Step 1: Calculate the centroid of the input vectors as the root node.

Step 2: Construct the next level (called the d-th level) of k^d children using the tree growing procedure; that is, for an input n-dimensional vector $(x_0, x_1,\ldots, x_{n-1})$, generate k initial codewords $b_0, b_1,\ldots, b_{k-1}$, where $b_i = (x_{i0}, x_{i1},\ldots, x_{i,n-1})$ and $x_{ij} = x_i+(2\times i-m-1)\times\delta$.

Step 3: Generate new k^d centroids using the LBG method.

Step 4: Go back to Step 2 until k^d is equal to the number of desired codewords m.

In the tree growing procedure, the value of δ is suggested in range [4, 10]. Although Hu and Chang's method can improve the LBG method owing to the better distribution of the codewords, these methods are still stuck in the local optimal problem.

3 Genetic Density-Based Clustering for Codebook Design

Conventional codebook design methods are usually based on the LBG approach. However, LBG is often affected by noise and cannot deal with the clusters of different shapes, sizes, and densities. For example, Fig. 1(a) shows a result using the LBG-based method, where the centroid colored gray is obviously affected by the two noise

points. However, if the two noise points can be excluded, the centroid point in Fig. 1(b) can fit better among the main data points. Fig. 1(c) shows another LBG-based result for the data set with nonglobular shapes and different densities using three centroids. Although the average distortion in Fig. 1(c) may be minimized, the centroids in Fig. 1(d) are more truly representative. In our scheme, a point is regarded as a codevector having the same dimensions as the codeword in the codebook. From some experiments, we observe that a noise codevector usually represents a nonsmooth area. Because the human eye is less sensitive to distortion in nonsmooth areas than in smooth areas, a clustering result that excludes noise codevectors can improve visual quality. Furthermore, a centroid surrounded by more codevectors better represents a common codeword.

In this section, we propose a genetic density-based method to generate a more representative codebook. The representation of each chromosome contains m centroids for m clusters. Each centroid has a fixed predefined radius r. Note that all centroids in a chromosome should be different. The details of the proposed scheme are as follows.

3.1 Population Initialization

The quality of the genes in a chromosome affects the convergence speed of the objective function. The initial input data comes from the nonoverlapping partitioned blocks of the input training images. A gene represents a codeword in a codebook. If the genes are randomly selected from the initial data set, exhaustively selecting the best genes from the large data set can be very time-consuming. To improve efficiency, we apply a density-based data reduction algorithm to select most representative genes [10]. This method is based on a k-NN (k nearest neighbors) method and attempts to

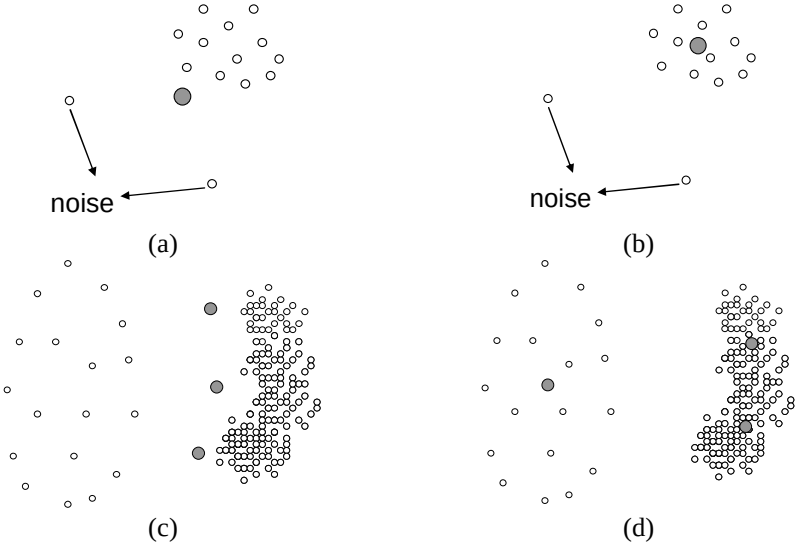


Fig. 1. Comparisons of LBG-based and density-based methods

use a condensed set to represent the entire distribution of the original data set. The point with a higher density has a higher probability of being selected in the condensed set. The condensed set production algorithm is presented as follows.

The Condensed Set Production Method

Input: A set of blocks $BS = \{b_1, b_2, \ldots, b_N\}$ from the training images and the value of k.

Output: A condensed set CS of BS.

Step 1: For each block b_i of BS, calculate the distance d_i of the k-th nearest neighbor of b_i in BS.

Step 2: Select the b_i having the smallest d_i and place b_i in CS.

Step 3: Remove all blocks including b_i from BS that are within the circle with radius $2d_i$ centroided at b_i.

Step 4: Repeat Steps 2 through 3 until BS becomes a null set.

In Step 3, the radius set to $2d_i$ can avoid the selected points over-centralized at the dense area. Further, the value of k used for k-NN controls the condensation ratio and the accuracy of representation. Choosing a proper k is discussed in Section 4. In this method, the circle in the dense area has a smaller radius than in the sparse area, and the dense area has more representative points (or blocks) than the sparse area.

After performing this method, the first $|P| \times m$ points of CS are selected and evenly distributed to m chromosomes, where $|P|$ is the size of the initial population and m is the size of each chromosome. For instance, if there are two chromosomes with size 10 and the points in CS are numbered $\{1, 2, 3, 4, \ldots, |CS|\}$, the two chromosomes are composed of points numbered $\{1, 3, 5, \ldots, 19\}$ and $\{2, 4, 6, \ldots, 20\}$, respectively.

3.2 Genetic Operations

The genetic operations including fitness function design, selection, crossover, and mutation are detailed as follows. For clearly, a circle with a radius r in the proposed method represents a cluster.

Fitness computation: Two factors affect the fitness computation for each chromosome: the coverage rate (*CR*) and average distance rate (*ADR*). The *CR* represents the points covered by the chromosome to the total number of points, and the *ADR* denotes the proportion of the entire average distortion to the square of the radius r. An average distortion of a cluster AD_i is the average distance between each point of the cluster to the cluster's centroid. If the distances of a point to all the centroids are all greater than r, the point is considered as a noise and does not belong to any cluster. Assume that there are n data points $\{b_1, b_2, \ldots, b_n\}$ and S_{c_i} is the set of points covered by the circle with centroid c_i and radius r. The formulas for *CR* and *ADR* are defined as follows:

$$CR = \frac{\left| S_{c_1} \cup S_{c_2} \cup \ldots \cup S_{c_m} \right|}{n}. \tag{2}$$

$$ADR = \frac{1}{r^2}\left(\frac{1}{m}\sum_{i=1}^{m} AD_i\right), \text{ where } AD_i = \frac{1}{\left|S_{c_i}\right|} \times \sum_{b_j \in S_{c_i}} \left\| b_j - c_i \right\|^2. \tag{3}$$

The goal of genetic operations is to maximize *CR* and minimize *ADR*. To keep better diversity among solutions in the early stage, the weight of *CR* should be larger than that of *ADR*. Consequently, the fitness function *F* is defined below, where *t* represents the *t*-th iteration of the genetic iterations.

$$F = (1+\frac{2}{t+1})\times CR - ADR\ . \tag{4}$$

Selection operation: The aim of the selection operation is to pick out better chromosomes from the population for better offspring generation. The most common selection method is the roulette-wheel selection (also called tournament selection) [5]; that is, a chromosome with a higher fitness value has a higher probability of being selected. The selection operation consists of two phases. In the first phase, each centroid in the chromosomes is recomputed based on LBG but excludes the points outside the corresponding circle. In the second phase, chromosomes are selected according to their fitness values.

Crossover operation: This operation joins information from different chromosomes to generate new chromosomes, called offspring. The crossover operation plays an extremely important role in affecting the convergence speed and the quality of offspring. To achieve fast convergence and ensure that offspring have a high probability of inheriting better qualities from their parents, we apply the concepts of the deterministic crossover method proposed by Fränti [4]. This method combines two existing chromosomes to generate a new chromosome. As a result, the size of the new chromosome is between *m* and 2*m* since its parents may contain the same codewords (also called centroids). The size of the new chromosome is then reduced to *m* using the hierarchical PNN (pairwise nearest neighbor) method [3]. The first step in PNN is to search the nearest neighbor that minimizes the merge cost for each codeword. The merge cost d_{ij} is obtained by Eq. (5), where c_i and c_j are the merged codewords, and n_i and n_j represent the numbers of codevectors in the circles centroided at c_i and c_j, respectively. Then, the codeword pair with the least merge cost is merged to obtain a new mean codeword, and the merge process is repeated until the size of the new chromosome is reduced to *m*.

$$d_{ij} = \frac{n_i \times n_j}{n_i + n_j} \cdot \left\| c_i - c_j \right\|^2 . \tag{5}$$

Mutation operation: In the mutation, a codeword with a smaller density (i.e., the number of codevectors contained by the codeword) has a higher probability of mutation. However, mutation can often damage codewords. To improve the mutation operation, we propose a repair procedure for the defective codewords in each chromosome. The repair procedure creates a temporary list *L* of codewords in which each corresponding circle includes at least *p* codevectors. Initially, *L* is constructed from the condensed set *CS* in which the codevectors are selected from top to bottom and the distance between each selected codevector pair is at least *r*. When a codeword is to be mutated, it searches *L* for the first codevector having higher density and the distance between the two vectors is smaller than *r*. If the codevector exists, it replaces the codeword. Otherwise, (1) the codevector of *L* is replaced with the codeword or (2)

the codeword is added to L. Then, the codeword is mutated by a one-point mutation algorithm [5]. In the first case, the codeword should have a higher density than the codevector and the distance between them should also be smaller than r. In the second case, the circle centroided at the codeword should contain at least p codevectors and all the distances between the codeword and each codevector of L should be greater than r. Consequently, the modification of L is dynamic during genetic iterations.

When the genetic iterations are finished, the chromosome with the highest fitness value is selected as the final trained codebook.

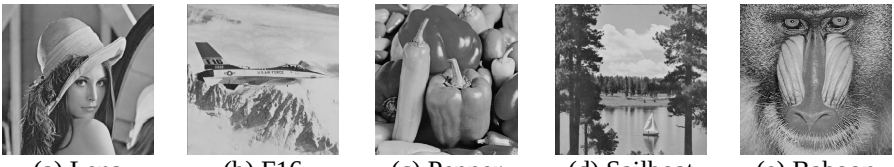
4 Experiments

The experiments conducted for codebook design use five 512×512 standard gray-level images ("Lena," "F16," "Pepper," "Sailboat," and "Baboon") with pixel resolution of 8 bits, as the training images shown in Fig. 2. The image quality of the VQ-encoded image is measured by the peak-signal-to-noise-ratio (PSNR), which is defined as:

$$\text{PSNR} = 10 \times \log_{10} \frac{(255)^2}{\text{MSE}} \text{dB}. \tag{6}$$

The MSE (mean-square error) is defined in Eq (7), where x_{ij} and y_{ij} indicate the pixel values in the position (i, j) of the original image and that of the encoded image, respectively, and w and h are the width and the height of the image.

$$\text{MSE} = \frac{1}{(w \times h)} \sum_{i=1}^{w} \sum_{j=1}^{h} \left(x_{ij} - y_{ij}\right)^2. \tag{7}$$



(a) Lena (b) F16 (c) Pepper (d) Sailboat (e) Baboon

Fig. 2. Five training images

In the experiment, each training image was divided into non-overlapping blocks of 4×4 pixels. The radius r in the proposed method can be estimated from the PSNR equation. For instance, if the desired PSNR of the encoded image is to be about 30, the value of r^2 can be set to about 1040 according to Eq. (8). A larger r^2 achieves a better coverage rate but a higher distortion rate. The value of p used in the mutation operation is set to n/m, where n is the total number of input blocks and m is the codebook size.

$$r^2 = (16 \times 255^2 / 10^{\frac{\text{PSNR}}{10}}). \tag{8}$$

In Section 3.1, the value of k would influence the size of the condensed set. Fig. 3 shows the relationship between k and the number of representative points that are selected from the 81920 input points. It can be observed that k is negatively related to the number of representative points. However, when k is larger than 6, the curve becomes much and much flatter. This confirms the robustness of the data reduction method. In this section, we set k to 7.

The size of the population for genetic operations is set to 15. The crossover selects 15 different pairs to generate 15 children, and the mutation probability is set to 0.01. Table 1 compares image quality (PSNR) using different codebook training methods, where the last two images of the table are not the training images. We perform 15 iterations of genetic operations for both Ying et al.'s method [11] and the proposed method. The value of δ used in Hu and Chang's method is set to 6. From the table we can observe that the proposed method has better image quality and its trained codebook is more representative than other images. Ying et al's method has the worst PSNR because their genetic method is at a low convergence speed.

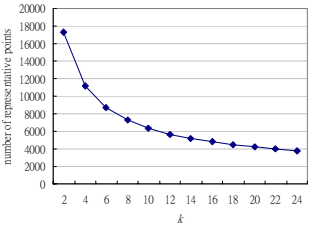
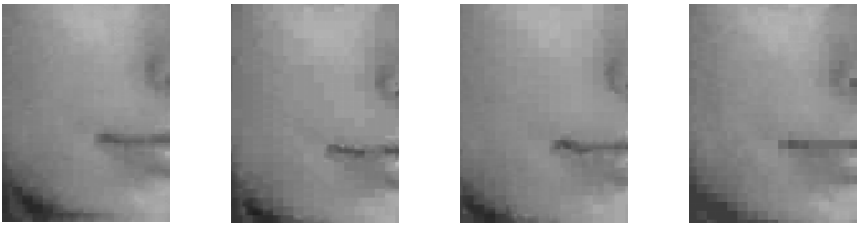


Fig. 3. Variation of k with the number of representative points

The mosaic effect using different methods with a codebook sized 256 is shown in Fig. 4. The subgraph is the face region of Lena. These figures show that the proposed method has the best visual quality (i.e., least mosaic effect) in the smooth area.

Table 1. The comparison of image quality with the codebook sized 256

Images	LBG	TSVQ	Hu and Chang's method	Ying et al.'s method	Proposed method
Lena	29.12	28.91	29.96	27.17	30.76
F16	30.54	30.51	30.73	28.03	30.58
Pepper	29.98	29.76	30.92	27.94	30.98
Sailboat	28.62	28.44	28.91	26.86	28.95
Baboon	24.37	24.31	24.41	23.14	24.46
Tiffany	28.33	27.61	28.70	24.06	28.94
Zelda	34.32	34.15	35.03	29.94	35.72



(a) The proposed method (b) LBG (c) Hu and Chang's method (d) Ying et al.'s method

Fig. 4. Comparison of mosaic effect

Fig. 5 also shows the convergence speed of the proposed method with codebook sized 256 and r^2=1200. The figure confirms that the proposed method has a high convergence speed and most of the improvements are attained within 8 iterations. Furthermore, we also evaluated the equal sized codebook trained by the LBG method with the same r^2 value. An interesting finding is that the *CR* and *ADR* obtained by the LBG method are 0.72 and 0.67, respectively. This reveals that about 30% of the input blocks cannot be covered by the circles with radius r^2 so the LBG based methods are easily affected by these noise-like blocks.

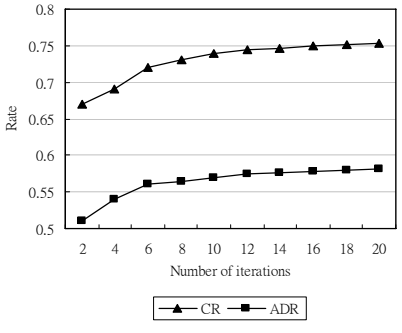


Fig. 5. The convergence speed of the proposed method

5 Conclusions

Conventional LBG-based methods often suffer from the following challenges: (1) the clustering result is susceptible to the initialization of the centroids; (2) the methods often terminate at a local optimum; (3) the methods are sensitive to noise or outlier; (4) they are difficult to handle nonglobular clusters or clusters of different sizes and densities. In this paper, a new codebook training method based on density approach using a genetic algorithm is proposed. The density-based clustering method can avoid the disadvantages of the LBG-based methods and can generate a high quality common codebook. The experimental results show that the proposed method can approach the optimal result within only a few iterations and the density-based

codebook can also alleviate the mosaic effect in images, especially for the VQ-encoded smooth blocks. Although the proposed method may have a lower PSNR than conventional LBG-based methods, it preserves better visual image since the trained codewords are not affected by the noise-like codevectors.

References

1. A. Buzo, A. H. Gray, R. M. Gray, and J. D. Markel, "Speed coding based upon vector quantization," *IEEE Transactions on Signal Processing*, vol. 28, no. 5, pp. 562- 574, 1980.
2. T. S. Chen and C. C. Chang, "Diagonal axes method (DAM): a fast search algorithm for vector quantization," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 7, no. 3, pp. 555-559, 1997.
3. W. H. Equitz, "A new vector quantization clustering algorithm," *IEEE Transactions on Acoustics, Speech, and Signal Processing* vol. 37, no. 10, pp. 1568-1575, 1989.
4. P. Fränti, "Genetic algorithm with deterministic crossover for vector quantization," *Pattern Recognition Letters*, vol. 21, no. 1, pp. 61-68, 2000.
5. M. Gen and R. Cheng, "Genetic algorithms and engineering optimization," *John Wiley & Sons, Inc.*, 2000.
6. R. M. Gray, "Vector quantization," *IEEE ASSP Magazine*, 1984, pp. 4-29.
7. J. Han and M. Kamber, "Data mining: concepts and techniques," *Morgan Kaufmann*, San Francisco, 2001.
8. Y. C. Hu and C. C. Chang, "A progressive codebook training algorithm for vector quantization," *Proceedings of the Fifth Asia-Pacific Conference on Communications and Fourth Optoelectronics and Communications Conference*, Beijing, China, pp. 936-939, 1999.
9. Y. Linde, A. Buzo, and R. M. Gary, "An algorithm for vector quantization design," *IEEE Transactions on Communications*, vol. 28, no. 4, pp. 84-95, 1980.
10. P. Mitra, C. A. Murthy, and S. K. Pal, "Density-based multiscale data condensation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 6, pp. 734 -747, 2002.
11. L. Ying, Z. Hui, and Y. W. Fang, "Image vector quantization coding based on genetic algorithm," *Proceedings of IEEE International Conference on Robotics, Intelligent Systems and Signal Processing*, Changsha, China, pp. 773-777, 2003.
12. Y. Zheng, B. A. Julstrom, and W. Cheng, "Design of vector quantization codebooks using a genetic algorithm," *Proceedings of IEEE International Conference on Evolutionary Computation*, Indianapolis, IN, USA, pp. 525-529, 1997.

Multilayered Contourlet Based Image Compression

Fang Liu and Yanli Liu

School of Computer Science and Engineering, Xidian University, Xi'an, 710071, China

Abstract. After studying contourlet transform and multilayered image representation, we present a Multilayered Contourlet Based Image Compression algorithm (MCBIC) in this paper. It decomposes image into a superposition of coherent layers: piecewise smooth regions layer, directional information layer, e.g., textures and edges. MCBIC uses the best basis to deal with different layers of multilayered representation, and retains the most significant structures of the image. In MCBIC, the first layer of the image is coded in wavelet, which acquires information of piecewise smooth regions; because the contourlet transform is an efficient directional multiresolution image representation, so the second layer of the image is coded in contourlet, which captures directional structure information of the image. Furthermore, each layer is encoded independently with a different transform and the combination of the compressed layers can always be perfect reconstructed. Our experiments demonstrate that MCBIC is efficient in coding images that possess mostly textures and contours. Our experimental results also show that MCBIC is competitive to the contourlet algorithm, SPIHT algorithm and the multilayered image compression approach in terms of the PSNR-rate curves, and is visually superior to these algorithms for the mentioned images.

Keywords: Multilayered representation, Contourlet transform, SPIHT.

1 Introduction

Over the last decade, wavelets have had a growing impact on signal processing and communication such as compression, noise removal, image edge enhancement, and feature extraction. Unfortunately, their good NLA performance for piecewise smooth functions in 1-D is not optimal in 2-D. In essence, wavelets in 2-D are good at isolating the discontinuities at edge points, but will not see the smoothness along the contours. Therefore, more powerful representations are needed for image signals.

M. N. Do and M. Vetterli pioneers a new system of representation, named contourlet, which is a filter bank structure that can deal effectively with piecewise smooth images with smooth contours [1][2]. In the contourlet transform, a Laplacian pyramid [3] is first used to capture the point discontinuities, while directional filter banks (DFB) [4] are used to link point discontinuities into linear structures. The contourlet transform is almost critically sampled, with a small redundancy factor of up to 1.33, so it is much more suitable for image coding applications. Although the

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 299 – 308, 2007.
© Springer-Verlag Berlin Heidelberg 2007

contourlet transform is an efficient directional multiresolution image representation, but when contourlet transform is only used for image compression, grid phenomenon emerges in piecewise smooth regions of images (see Fig.1(a) and Fig.3(b)).

The underlying assumption behind transform coding is that the basis $\{\psi_n\}$ used for compression is well adapted to most images. Unfortunately, this assumption is clearly violated by many observations [5], which results in a consequence that one should be able to reduce the distortion by replacing an orthonormal basis with a richer library of basis functions. By dropping the "orthonormal basis" constraint, it becomes in principle possible to match the local textures with localized cosine functions (for instance), and the piecewise smooth regions with wavelets [6]. François G. Meyer, Amir Z. Averbuch, and Ronald R. Coifman [6] proposed a general framework for image representation and image compression that an image can be decomposed as the sum of two layers: a "cartoon image" and a texture map. The cartoon image provides a description of the piecewise smooth regions, and the texture map deal with the textures. The multilayered approach comes from the fact that different sets of basis functions complement each other: some of the basis functions will give reasonable account of the large trend of the data, while others will catch the local transients, or the oscillatory patterns. So each layer should be represented with different sets of basis functions.

In this paper we used the ideas of multilayered image representation [6]. Our hypothesis is that an image can be decomposed as the sum of two layers: a "cartoon image", a directional information map. The cartoon image provides a description of the piecewise smooth regions. The directional information map permits to fill in the textures in the regions with smooth contours. We propose to represent the cartoon image with wavelets. Our second choose is that the contourlet bases are better suited to capture geometrical structure in pictorial information. Our experimental results show that MCBIC is comparable to the contourlet compression and SPIHT algorithm, especially for a category of images that have a great deal of textures and oscillatory patterns and therefore are not "wavelet-friendly" images.

The paper is organized as follows. The next section mainly explains the multilayered contourlet based image compression. Section 3 illustrates some of the simulation and numerical results achieved by these algorithms. Finally, the main conclusions are outlined in Section 4.

2 Multilayered Contourlet Based Image Compression

The MCBIC is made up of a cascade of compressions applied successively to the image itself and to the residuals that resulted from the previous compression. Wavelet basis is used to compress the input image for obtaining the initial main approximation. This first approximation preserves the general shape of the image, and captures the trend in the intensity function. Then the compressed part is reconstructed, and the residuals error is calculated between the original and reconstructed data. Residuals are composed of textures and contours, namely directional information, and are compressed with contourlet bases which are well adapted to various directions coding.

2.1 The Wavelet Transform

Suppose that V_j is a subspace of Hilbert space $L_2\left(\mathbb{R}^2\right)$, then the corresponding space W_j is the orthogonal complement of V_j in V_{j-1}.

$$V_{j-1} = V_j \oplus W_j \tag{1}$$

Let wavelet function $\varphi(x)$, that generates a basis $\varphi_{i,k}(x)$ of W_j by dilation (index *i*) and translation (index *k*)

$$\varphi_{i,k}(x) = 2^{-i/2}\varphi\left(x/2^i - k\right) \tag{2}$$

Assume $f \in L_2$, the *orthogonal wavelet transform* is

$$Wf(j,k) = \left\langle f, \varphi_{j,k} \right\rangle = \int_{\mathbb{R}} f(t)\overline{\varphi_{j,k}(t)}dt, \tag{3}$$

and the reconstruction formula [7] is

$$f(t) = \sum_{j,k\in\mathbb{Z}} \left\langle f, \varphi_{j,k} \right\rangle \varphi_{j,k}(t) \tag{4}$$

The mechanics of the wavelet transform are, in essence, a two channel filterbank. In the decomposition step, the digital signal is split into two half-size sequences with a conjugate pair of lowpass and highpass filters $\tilde{H}\left(z^{-1}\right)$ and $\tilde{G}\left(z^{-1}\right)$, and then down-sampling the results. The signal is reconstructed by up-sampling, filtering, and summation of the components [8]. In MCBIC, we choice the"9-7" biorthogonal filters [9]. The lowpass filter H and highpass filter G are

$$H(z) = \frac{1}{\sqrt{2}}\sum_{k\in\mathbb{Z}} h_k z^k \qquad G(z) = \frac{1}{\sqrt{2}}\sum_{k\in\mathbb{Z}} g_k z^k \tag{5}$$

These two filters satisfy the QMF condition, so can be exact reconstructed.

2.2 The Contourlet Transform

Contourlet transform is performed by a double filter bank structure, named the pyramidal directional filter bank (PDFB) [10], by combining the Laplacian pyramid with a directional filter bank. It satisfies five key properties of image representations: multiresolution, localization, critical sampling, directionality, anisotropy. A contourlet function is defined in $L^2\left(\mathbb{R}^2\right)$

$$\rho_{j,k}^{(l)}(t)=\sum_{i=0}^{3}\sum_{n} g_k^{(l)}[2n+k_i]\left(\sum_{m\in\mathbb{Z}^2} f_i[m]\phi_{j,n+m}\right)$$
$$=\sum_{m\in\mathbb{Z}^2}\underbrace{\left(\sum_{i=0}^{3}\sum_{n\in\mathbb{Z}^2} g_k^{(l)}[2n+k_i]\, f_i[m-n]\right)}_{c_k^{(l)}[m]}\phi_{j,m}(t) \tag{6}$$

Suppose that the contourlet frame

$$\left\{\phi_{j_0,n}(t),\rho_{j,k,n}^{(l_j)}(t)\right\}_{j\le j_0,0\le k\le 2^l\, j-1,n\in\mathbb{Z}^2} \tag{7}$$

satisfies the anisotropy scaling law

$$l_j=l_{j_0}-\left\lfloor (j-j_0)/2\right\rfloor,\qquad for\ \ j\le j_0 \tag{8}$$

and each contourlet kernel function $\rho_{j,k,n}$ has directional vanishing moments on a set of directions with maximum gap of $W2^{j/2}$ [2]. Then for a function $f\in L_2$, the m-term approximation by contourlet frame obtains

$$\left\| f-\hat{f}_M^{contourlet}\right\|^2\le C(\log M)^3 M^{-2} \tag{9}$$

Because no other approximation scheme can achieve better rate than M^{-2}, then the contourlet transform achieves the optimal approximation rate for piecewise smooth functions with C^2 contours.

2.3 The Ideal of Multilayered Image Compression

The multilayered compression algorithm consists in a cascade of compression applied successively to the image itself and to the residuals that resulted from the previous compression [6]. Let I be an image, I is compressed over the library L_0, used the budget b_0, then we have

$$I=\hat{R}^0+R^1,\quad with\quad \hat{R}^0=\sum_{j\in E_0} q_j^0\psi_j^0 \tag{10}$$

where $\hat{R}^0$ is an approximation of the original image I, R^1 is the approximation error (the residual). In order to discover different features in the image, a different library is used to compress R_1 with a budget of b_1 bit. A best basis, $\{\psi_j^1, j\in E_1\}$, that provides the optimal compression $\hat{R}^1$ of R^1 over the library L_1.

$$R^1 = \hat{R}^1 + R^2, \quad with \quad \hat{R}^1 = \sum_{j \in E_1} q_j^1 \psi_j^1 \tag{11}$$

where $\{q_j^1,\}_{j \in E_1}$ are the quantized coefficients. A second approximation $\hat{I}^1$ of I is reconstructed

$$\hat{I}^1 = \hat{R}^0 + \hat{R}^1 \tag{12}$$

where $\hat{I}^1$ is an image that can be encoded with b_0+b_1 bits. In our work, we choose L_0 to be a wavelet basis, the second library L_1 is contourlet basis.

2.4 The Multilayered Contourlet Based Image Compression Alogrithm

The MCBIC method for a given image consists of the following steps.

1) Decompose the input image with wavelet for the piecewise smooth regions. Set a threshold T1, all the wavelet coefficients to be bigger than T1 are retained and compressed.
2) Reconstruct the compressed part for R^1 and the residual is calculated between the original image and reconstructed data.
3) Decompose the residuals with contourlet for textures and contours. Set a threshold T2, all the contourlet coefficients to be bigger than T2 are retained and compressed.
4) Reconstruct the compressed part of contourlet for R^2. The finally reconstructed image is the sum of $\hat{R}^0$ and $\hat{R}^1$.

3 Experiments and Result Analysis

We compare MCBIC with the contourlet coder, original SPIHT coder and multilayered image representation approach [6] on Barbara image (see Fig.1). And, we also compare MCBIC with the contourlet coder and original SPIHT coder on several images [11] (see Fig.2-3). In our experiments, the wavelet transform uses 3 levels, and the contourlet transform has 32 directions at the finest scale. We used non-separable fan filters of support sizes 23*23 and 45*45 designed in [12]. An arithmetic encoder is used to entropy-code the resulting bit streams of the quantization.

Fig.1 show the coded results of the Barbara image at rate R = 0.125 bpp. The periodic texture on the pants of the lady is very well preserved by the contourlet. The texture on the tablecloth is also well rendered (compare Fig.1(b) and Fig.1(c)). There are some criss-cross patterns on the face of Barbara (compare Fig. 1(c)). Unfortunately, some grid phenomena appear in the smooth floor of Barbara image (see Fig.1(a)). As shown in Fig.1(d), the periodic texture on the pants of the lady and the texture on the tablecloth are very well preserved by MCBIC (compare Fig. 1(b) and Fig. 1(c)). Cheerfully, there are neither some criss-cross patterns on the face of Barbara nor some grid phenomenon in the smooth floor of Barbara image (compare Fig.1(a) and Fig.1(c)). The contourlet provided the optimal library (in preserving directional information) to encode the second layer. Our MCBIC coder outperformed

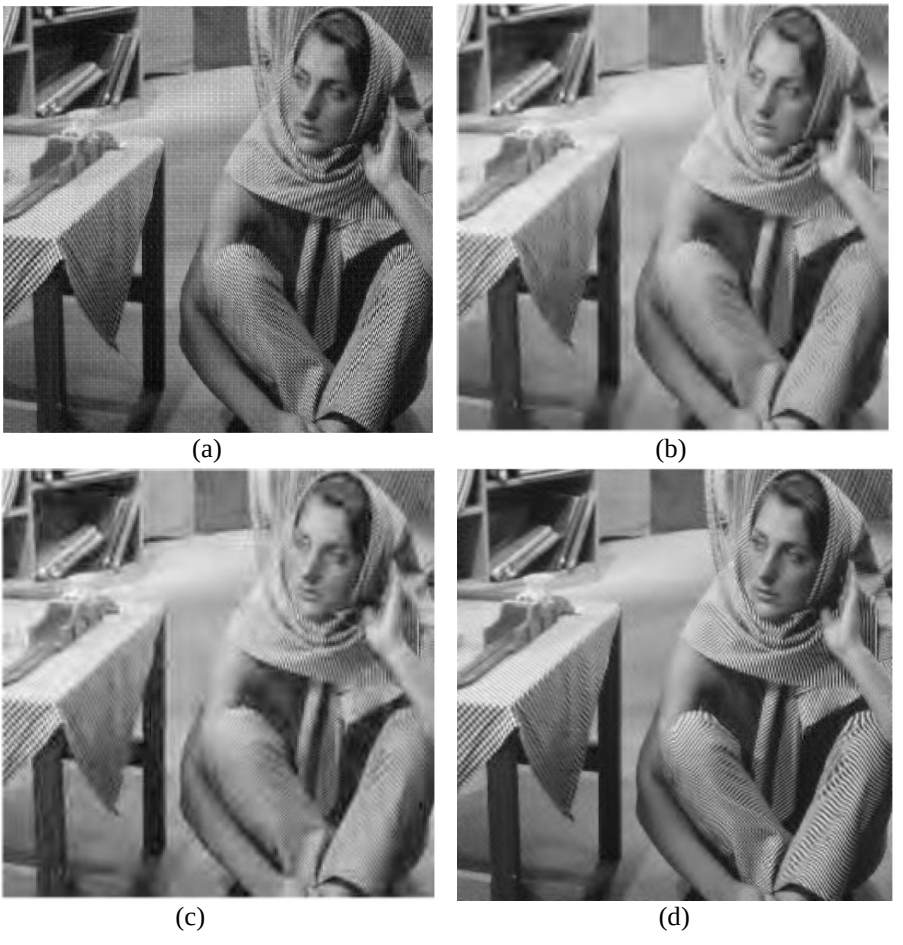


Fig. 1. The barbara image coded at rate 0.125 bpp. (a) result of the contourlet coder (PSNR=28.86). (b) result of the SPIHT coder (PSNR=24.86). (c) result of the Multi-layer coder (PSNR=26.21). (d) result of the MCBIC (PSNR=29.87).

the Multi-layer by 2 to 17.19 dB (see Table1). Our experiments indicated that MCBIC approach is superior in preserving directional information, e.g., textures and edges in the coded images.

Table 1. Barbara: PSNR (IN DECIBELS) FOR VARIOUS BIT RATES

Rate(bpp)	0.0625	0.125	0.25	0.50	0.75
Contourlet	25.87	28.86	32.24	37.61	46.81
SPIHT	23.35	24.86	27.58	31.39	34.25
Multi-layer	24.04	26.21	29.18	32.92	35.30
MCBIC	26.04	29.87	34.54	42.23	52.49



Fig. 2. The houses image coded at rate 0.32 bpp. (a) original image. (b) result of the contourlet coder (PSNR=28.69). (c) result of the SPIHT coder (PSNR=24.17). (d) result of the MCBIC (PSNR=30.17).

Table 2. Houses: PSNR (IN DECIBELS) FOR VARIOUS BIT RATES

Rate(bpp)	0.125	0.20	0.25	0.40	0.50
Contourlet	23.79	26.03	27.09	30.35	32.27
SPIHT	20.98	22.33	23.17	25.06	26.15
MCBIC	24.01	26.74	28.08	32.47	34.72

Fig. 2-3 show the coded results of two images, i.e., the Houses image and the Lighthouse image. As shown in Fig. 2(c), the wavelet coder could not preserve the tiles on the roof of the building (compare Fig. 2(b) and Fig. 2(d)). In fact, our MCBIC coder outperformed SPIHT by 3.03 to 8.57 dB (see Table 2).

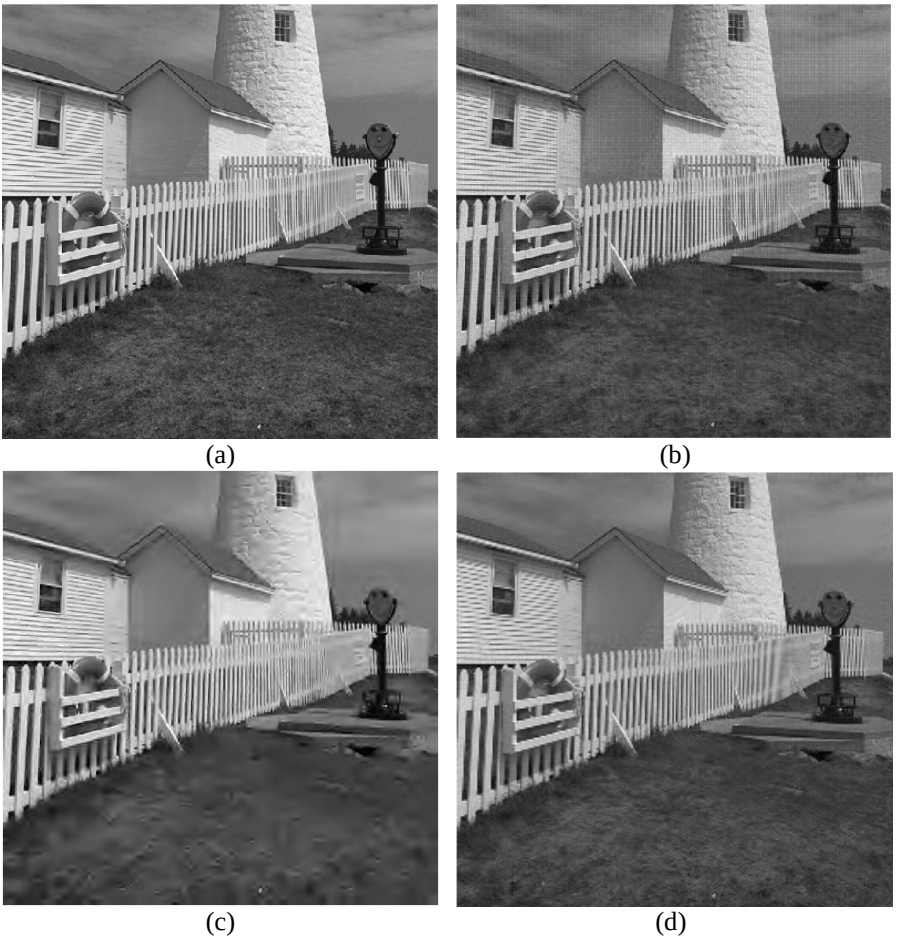


Fig. 3. The lighthouse image coded at rate 0.20 bpp. (a) original image. (b) result of the contourlet coder (PSNR=29.40). (c) result of the SPIHT coder (PSNR=26.58). (d) result of the MCBIC (PSNR=30.84).

Table 3. Lighthouse: PSNR (IN DECIBELS) FOR VARIOUS BIT RATES

Rate(bpp)	0.125	0.20	0.25	0.40	0.50
Contourlet	27.52	29.40	30.44	33.16	35.56
SPIHT	24.98	26.58	27.43	29.29	30.25
MCBIC	28.42	30.84	32.21	35.75	38.35

The contourlet provided again the optimal library to encode the second layer. Our MCBIC coder outperformed SPIHT by 3.44 to 8.10 dB (see Table 3). As shown in Fig.3(c), the wavelet coder could not preserve the tiles on the roof of the two houses, and the lawn of the image (compare Fig. 3(b) and Fig. 3(d)). It could lose some textures on the surface of the tower building (compare Fig. 3(b) and Fig. 3(d)).

Although the contourlet coder could preserve textures, more edges and details information in the coded image, but some grid phenomena emerge in the sky of the Lighthouse image (see Fig. 3(b)). These phenomena indicate that the contourlet coder is not the optimum method to deal with piecewise smooth regions, and they also explain the reason why we combine the ideal of the multilayered image compression with the contourlet.

4 Conclusions

We proposed a multilayered contourlet based image compression algorithm which efficiently coded images that contain a mixture of smooth regions, textured features and edges. Because the contourlet is an efficient directional multiresolution image representation, it can capture directional structure information of most images. But it is not the optimum method to deal with piecewise smooth regions for coding image. So we present a combination of the multilayered image compression and the contourlet. In MCBIC, an image is parsed into a superposition of coherent layers: a piecewise smooth regions layer, a directional information layer. Our simulation results indicated that the MCBIC is visually superior to the contourlet coder, original SPIHT coder and original multilayered image compression approach in preserving details, textures and contours in the coded images.

Acknowledgments. This work is supported by the National Natural Science Foundation of China under Grant No. 60372045 and No. 60133010, the Defence Pre-Research Project of China under Grant No.51406020104DZ0124, the Key Science-Technology Project of Higher Education of China under Grant No. 0202A022 and the National Research Foundation for the Doctoral Program of Higher Education of China No. 20030701013.

References

1. M. N. Do and M. Vetterli, “Contourlets,” in Beyond Wavelets, J.Stoeckler and G. V. Welland, Eds. Academic Press, New York, 2003
2. M. N. Do and M. Vetterli, The Contourlet Transform: An Efficient Directional Multiresolution Image Representation IEEE TRANSACTIONS ON IMAGE PROCESSING,Vol.14, No.12,pp.2091-2106,2005
3. P. J. Burt and E. H. Adelson, “The Laplacian pyramid as a compact image code,” IEEE Trans. Commun., vol. 31, no. 4, pp. 532–540, April 1983
4. R. H. Bamberger and M. J. T. Smith, “A filter bank for the directional decomposition of images: Theory and design,”IEEE Trans. Signal Proc., vol. 40, no. 4, pp. 882–893, April 1992
5. S. Mallat and F. Falzon, “Analysis of low bit rate image transform coding,” IEEE Trans. Signal Processing, vol. 46, pp. 1027–1042, Apr.1998
6. François G. Meyer, Amir Z. Averbuch, and Ronald R. Coifman, “Multilayered Image Representation: Application to Image Compression” IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 11, pp.1072-1080,September 2002
7. S. Mallat. A Wavelet Tour of Signal Processing. Academic Press, San Diego, 1998

8. O. Rioul and M. Vetterli, “Wavelets and signal processing,” IEEE Signal Processing Mag., vol. 8, pp. 11–38, Oct. 1991
9. A. Cohen, I. Daubechies, and J.-C. Feauveau, “Biorthogonal bases of compactly supported wavelets,” Commun. on Pure and Appl. Math., vol. 45, pp. 485–560, 1992
10. M. N. Do and M. Vetterli, “Pyramidal directional filter banks and curvelets,” in Proc.IEEE Int. Conf. on Image Proc., Thessaloniki, Greece, Oct. 2001
11. A. Said, W. A. Pearlman, A new, fast and efficient image codec based on set partitioning in hierarchical trees. IEEE Transactions on Circuits and Systems for Video Technology, 1996, 6(3):243~250
12. S.-M. Phoong, C. W. Kim, P. P. Vaidyanathan, and R. Ansari, “A new class of two-channel biorthogonal filter banks and wavelet bases,” IEEE Trans. Signal Proc., vol. 43, no. 3, pp. 649–665, Mar. 1995

Iterative Image Coding Using Hybrid Wavelet-Based Triangulation

Phichet Trisiripisal, Sang-Mook Lee, and A. Lynn Abbott

Bradley Department of Electrical and Computer Engineering,
Virginia Polytechnic Institute and State University, Blacksburg, VA, USA
{ptrisiri, lsmook, abbott}@vt.edu

Abstract. This paper presents a novel hierarchical subdivision scheme for generating triangular meshes to represent intensity images. The resulting representation is well suited to content-based management of images and videos, including storage and retrieval for multimedia applications. Mesh node locations are selected using multiresolution wavelet-based analysis, and this leads to a piecewise-linear approximation of image intensity values. Interpretation of wavelet coefficients at successively finer levels leads to fast, efficient triangular mesh construction in which the number of approximating elements depends on the image content and on user-specified maximum error. By combining Delaunay and data-dependent triangulation approaches, the result is an efficient multiresolution coding scheme that approximates the image to a desired level of detail, while retaining information related to image content under user-specified triangulation regularity. We demonstrate the utility of this approach for such multimedia applications such as selective compression, segmentation, and eye detection in images of human faces.

Keywords: Image compression and coding, wavelet transform, triangulation, segmentation, region-of-interest, multiresolution analysis, multimedia.

1 Introduction

Content-based image compression and retrieval have received a great deal of emphasis during the past decade, particularly for use in multimedia information systems. This motivation for this work is largely based on the desire to move from image-level to object-level interpretation, where high-level semantic information can be utilized.

This paper presents an image-coding scheme that is well suited for multimedia applications. The approach is based on triangular mesh representation, which is a compact and efficient means of image approximation. The usual approach is to subdivide the two-dimensional (2D) image domain into non-overlapping triangular mesh elements. Image intensity values are stored for vertex locations only, and interpolation is used to estimate image values over each triangular patch. Triangular mesh representations have been used in a wide range of applications, including image compression [1], segmentation [2], range image approximation [3], fingerprint identification [4], and medical applications [5].

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 309 – 321, 2007.
© Springer-Verlag Berlin Heidelberg 2007

Unlike most previous approaches, this paper considers the use of wavelet-based multiresolution methods to select the mesh structure for a given image. The system directly evaluates wavelet coefficients to assess image content within each triangular element at a given scale, and then subdivides and refines triangles based on that evaluation. As the system moves to finer resolution levels, the subdivision and refinement steps are repeated so that the final mesh represents the image function with desired approximation accuracy.

Several researchers have considered the problem of triangular mesh generation for image coding (e.g., [6], [7]). A common approach is first to select node locations in the image, and then to connect the nodes using Delaunay triangulation [8]. This is the method of Yang, et al. [9], for example, which uses an error diffusion algorithm to select mesh node locations. Klein, et al. [10], similarly use Delaunay triangulation to update views of terrain data.

Although Delaunay-based methods are fast, better approximation accuracy is typically possible if data-dependent criteria are employed to select node interconnections. One of the major problems with pure data-dependent triangulation, however, is that very narrow triangles often result [11], and these are difficult to refine through further subdivision steps.

This paper presents a new approach for rapid mesh construction for image approximation. Unlike most previous methods, we employ a "hybrid" technique that seeks an optimum combination of Delaunay and data-dependent mesh criteria. The result is a system that benefits from high approximation accuracy that is possible with narrow mesh elements, but which also retains the ability for further subdivision and refinement at intermediate resolution levels. The method is novel in its direct use of wavelet coefficients for mesh refinement. This work has been inspired by our previous work that involved range data [3], which also motivated others (e.g., [12]).

Section 2 of this paper presents an overview of the wavelet-based multiresolution approach to mesh generation. Section 3 describes our method in more detail. Experimental results are presented in section 4, which also illustrates the efficacy of the approach for segmentation, selective refinement, and eye detection. Finally, concluding remarks are given in section 5.

2 Wavelet-Based Mesh Generation

2.1 Background: Wavelet Decomposition of an Image

Let m=0 represent the resolution level of the original image. Following our derivation in [3], let $\phi_x = \phi(x)$ be a scaling function and $\psi_x = \psi(x)$ be a corresponding wavelet basis of $\mathbf{L}^2(\mathbb{R})$, a set of finite energy functions. Then, the tensor product wavelets given by $\psi^1(x,y) = \psi_x\phi_y$, $\psi^2(x,y) = \phi_x\psi_y$ and $\psi^3(x,y) = \psi_x\psi_y$ and the wavelet family

$$\{\psi^1_{m,i,j}(x,y), \psi^2_{m,i,j}(x,y), \psi^3_{m,i,j}(x,y)\}_{(i,j)\in\Omega^2} \tag{1}$$

constructs a two-dimensional orthonormal basis of detail space at resolution level m. Each $\psi^k_{m,i,j}$ is derived by scaling and shifting the corresponding function ψ^k:

$$\psi^k_{m,i,j}(x,y) = 2^{-m}\psi^k(2^{-m}x - i,\, 2^{-m}y - j) \quad \text{for } 1 \le k \le 3 \ . \tag{2}$$

With scaling function $\phi_{m,i,j}(x,y) = 2^{-m}\phi(2^{-m}x - i)\phi(2^{-m}y - j)$, any two-dimensional image signal $f(x,y) \in \mathbf{L}^2(\mathbb{R}^2)$ can be represented by

$$f(x,y) = \sum_{(i,j)} a_{M,i,j}\phi_{M,i,j} + \sum_{m=1}^{M}\sum_{(i,j)}\sum_{k=1}^{3} d^k_{m,i,j}\psi^k_{m,i,j} \ , \tag{3}$$

where $a_{M,i,j} = < f, \phi_{M,i,j} >$ and $d^k_{m,i,j} = < f, \psi^k_{M,i,j} >$. The symbol $< \cdot, \cdot >$ denotes an inner product, and M is the number of decomposition levels. The equation shows that the original function consists of the coarsest approximation, $\alpha_{M,i,j}$, and detail information, $d^k_{m,i,j}$, at resolution levels $1 \le m \le M$. Hence, to preserve significant details only, a function f can be approximated by

$$\hat{f}(x,y) = \sum_{(i,j)} a_{M,i,j}\phi_{M,i,j} + \sum_{m=1}^{M}\sum_{(i,j)}\sum_{k=1}^{3} \chi_m d^k_{m,i,j}\psi^k_{m,i,j} \ , \tag{4}$$

where

$$\chi(d^k_{m,i,j}) = \begin{cases} d^k_{m,i,j} & |\, d^k_{m,i,j}\,| > \tau_m \\ 0 & otherwise \end{cases} \tag{5}$$

and τ_m is a threshold for detail coefficients at level m.

2.2 Multilevel Hierarchical Mesh Generation

Because the detail coefficients reflect directional variation of data in a given portion of an image, a large magnitude in $d^1_{m,i,j}$, $d^2_{m,i,j}$, or $d^3_{m,i,j}$ tends to be caused by data discontinuities in horizontal, vertical, or diagonal directions, respectively. This can be used in mesh generation and refinement. Consider a two-dimensional data set taken at a grid of sample points V_0 in $\mathbb{R}^2$. A hierarchical mesh is obtained by constructing an initial mesh $\hat{\Gamma}_M$ at the coarsest level, M, with a set of sample points $\hat{V}_M$, and by refining the initial mesh to finer meshes $\hat{\Gamma}_{M-1}$, $\hat{\Gamma}_{M-2}$, …, $\hat{\Gamma}_0$, with sets $\hat{V}_{M-1}$, $\hat{V}_{M-2}$, …, $\hat{V}_0$, respectively. The original dataset is approximated over the entire domain by a piecewise planar function f_m that interpolates all data values at points of $\hat{V}_m$, $0 \le m \le M$. Typically, the number of points in $\hat{V}_m$ increases as m decreases, and generally $\hat{V}_0 \ne V_0$. This implies that $|\hat{\Gamma}_m| < |\hat{\Gamma}_{m-1}|$ for $1 < m \le M$, where $|\cdot|$ represents mesh size, and $|\hat{\Gamma}_0| \ne |\Gamma_0|$. Consequently, the error $\varepsilon_m = \|f - f_m\|_\infty$, defined in $\mathbf{L}^\infty(\mathbb{R}^2)$ norm, which means maximum error, decreases as m does.

In most applications, the main goal of mesh generation is to minimize $|\hat{\Gamma}_0|$ subject to the ε_0 being kept below a given error criterion. However, the minimization of mesh size for a given accuracy is an NP-hard problem and heuristics are needed for practical implementation.

3 Implementation

3.1 High-Level Algorithm

The approach presented here derives from that in [3], which was developed for triangular mesh representation of range data. First, for a given image *f*, a wavelet decomposition is generated. Then from the wavelet detail coefficients, $d^k_{M,i,j}$, $(i,j)\in\mathbb{Z}^2$ at the coarsest level *M*, initial triangulation is performed by selecting from a set of predefined templates. This produces a mesh $\widetilde{\Gamma}_M$ that needs to be refined afterward in an effort to obtain a more accurate and compact representation, that is, $\widetilde{\Gamma}_{M-1}$. Fig. 1a shows the basic templates for flat, vertical, horizontal and diagonal representations while Fig. 1b illustrates variant templates used for solving invalid interconnections between templates. The complete set of 47 predefined templates can be found in [13]. Vertical, horizontal or diagonal templates are assigned when $|d^1_{M,i,j}|$, $|d^2_{M,i,j}|$ or $|d^3_{M,i,j}| = \max_{1\le k\le 3}(|d^k_{M,i,j}|)$ respectively. Otherwise, a flat template is assigned when $\max_{1\le k\le 3}(|d^k_{M,i,j}|) < \tau_M$.

After initial triangulation, the algorithm proceeds to successively finer scales. For each level m 0, the algorithm performs steps of *i*) mesh enhancement, *ii*) wavelet-based region selection, *iii*) mesh refinement, *iv*) mesh reduction, and finally *v*) mesh regularization. These are described in the next sections. The system then performs 2 additional passes to improve the mesh, repeating the steps shown above except that region selection is performed using error criteria rather than wavelet coefficients. Informally, we refer to these as resolution levels –1 and –2. Finally, mesh enhancement is performed one last time to produce the output image.

3.2 Mesh Enhancement and Regularization

An important operation for mesh enhancement is the well-known *edge swap* operation, which tries to improve mesh approximation by analyzing the image content within triangular patches under consideration. This is also known as a data-dependent scheme and is illustrated in Fig. 2. Although this approach provides a good approximation result, it can generate thin triangle patches (known as slivers) that are difficult to process in later subdivision steps. The data-dependent approach, used alone, is therefore not well suited for multiresolution analysis. An alternative is Delaunay triangulation, which tends to avoid thin slivers, but with less approximation accuracy. By the definition of Delaunay triangulation, the circle that circumscribes three vertices of any triangle contains no other vertices. This can be achieved by maximizing the minimum interior angle of each triangular patch.

Our approach combines these two schemes. The decision to perform mesh enhancement and regularization is controlled by the criterion

$$\zeta(t_i,t_j) = \alpha\Re_A + (1-\alpha)\Re_E, \tag{6}$$

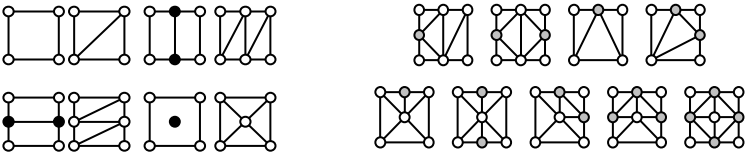
where $\Re_A$ and $\Re_E$ represent alteration ratios of minimum interior angles and approximation errors, respectively. These are defined as follows:

$$\Re_A = \min A(T_i, T_j) / \min A(t_i, t_j) \tag{7}$$

and

$$\Re_E = \{E(t_i) + E(t_j) + 1\} / \{E(T_i) + E(T_j) + 1\}\,, \tag{8}$$

where t_i and t_j represent the triangles under consideration before the edge is swapped, while T_i and T_j represent the new triangles after edge swapping has been performed. Equation (6) represents a trade-off between Delaunay-type (mesh regularization) and data-dependent (mesh enhancement) triangulation, with the mesh regularity factor $0 \ \alpha \ 1$ representing the balance between the two extremes. Since swapping of edges requires good knowledge of neighbor triangles and vertices, we have implemented a hierarchical ring data structure, coupled with a half-edge encoding to support fast access to adjacent vertices and neighbor triangles to speed up this operation [13].



(a) Basic templates. (b) Fundamental variant templates.

Fig. 1. Templates used for constructing initial triangulation

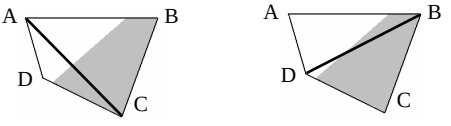


Fig. 2. Mesh enhancement and regularization. Edge *AC* is swapped (replaced) by *DB*, to give a better approximation or to regularize the mesh.

3.3 Wavelet-Based Region Selection for Refinement

During multiresolution analysis, triangles are evaluated for refinement (such as subdivision) using wavelet coefficient energy:

$$e_{m,i,j} = | d^1_{m,i,j} |^2 + | d^2_{m,i,j} |^2 + | d^3_{m,i,j} |^2 > \delta_{w1}\ . \tag{9}$$

If $e_{m,i,j}$ is above the user-specified threshold δ_{w1}, then the triangle under consideration is subdivided. For level –1 and level –2 refinement, the algorithm uses the absolute maximum error, which is expressed as

$$E = \max | f - \hat{f} |\,, \tag{10}$$

where $f(x, y)$ and $\hat{f}(x, y)$ are the original and the reconstructed images, respectively, and the comparison is performed over the entire triangle under consideration. If the

magnitude of E exceeds a given threshold value, the triangle will be tagged as candidate for future subdivision.

3.4 Mesh Refinement

To achieve good approximation during subdivision, we have used wavelet coefficients to determine the cutting direction. First, the cutting direction and centroids of a pair of adjacent triangles are calculated by analyzing wavelet coefficients within the two triangles. This is shown in Fig. 3. The cutting direction can be determined from wavelet coefficients as

$$g^{\perp} = \left[-\frac{\partial f}{\partial y} \quad -\frac{\partial f}{\partial x} \right]^T = -2^m \left[d_m^2 \quad d_m^1 \right]^T . \tag{11}$$

Next, the cutting vector is projected from each centroid onto the shared edge and the cutting point, v_i, is calculated by averaging the two intersection points. Details of this operation can be found in [3], [13].

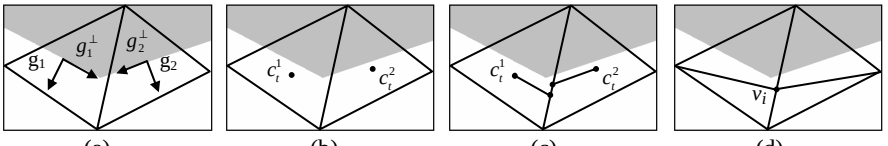


Fig. 3. Directional split. (a) Gradient direction. (b) Centroids. (c) Projection of gradient direction. (d) Midpoint of two intersection points.

3.5 Mesh Reduction

Generally, there are two mesh reduction operations: vertex removal and edge removal. These operations are used to remove redundant triangles at low loss of image quality. For vertex removal, we view the image function as a surface over the image domain, and consider the directions of the normal vectors of each triangle, n_i, that approximate a surface region. The covariance of these normal vectors, $R(v)$, can be defined by

$$R(v) = \frac{1}{k} \sum_{i=1}^{k} n_i n_i^T , \tag{12}$$

where k is the number of triangles that share the same vertex of interest, v. If the following condition is satisfied,

$$\lambda_2 < \lambda_1 < \delta_\lambda \tag{13}$$

indicating that these triangles are relatively flat, then the vertex under consideration can be removed without significantly deteriorating the image quality. This operation is shown in Fig. 4a. The values λ_1 and λ_2 are the eigenvalues of $R(v)$ and δ_λ is a user-specified threshold used to control the flatness tolerance of this operation. Because

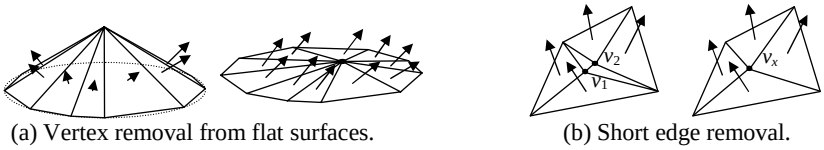
this method is relatively time-consuming, the magnitudes of the wavelet coefficients are considered first to reduce operation time where δ_{w2} is a constant:

$$|d^1_{m,i,j}|^2 + |d^2_{m,i,j}|^2 + |d^3_{m,i,j}|^2 < \delta_{w2}\,. \tag{14}$$

The second type of mesh reduction operation is edge removal. This operation is performed when triangles satisfy the condition

$$\sqrt{(x_{v_1} - x_{v_2})^2 + (y_{v_1} - y_{v_2})^2 + k(I_{v_1} - I_{v_2})^2} < \delta_d\,, \tag{15}$$

where (x_{v_1}, y_{v_1}) and (x_{v_2}, y_{v_2}) represent the geometric locations of the two vertices that share the edge of interest, while $(I_{v_1} - I_{v_2})$ is their intensity difference. Equation (15) can be interpreted as a sphere, indicating that if a neighbor vertex is located within a radius δ_d, of a vertex of interest, the edge between the two vertices can be removed. Note that constant *k* is used to normalize intensity values with geometric quantities. In this research, we have set *k* to 1. An example of this operation is shown in Fig. 4b.



(a) Vertex removal from flat surfaces. (b) Short edge removal.

Fig. 4. Mesh reduction operations

4 Result and Performance Analysis

4.1 Mesh Regularity

First, to determine an appropriate value for the mesh regularity factor α in (6), we empirically evaluated the resulting mesh constructions for several values chosen over the whole range 0 α 1. Four examples are shown in Fig. 5 for the commonly used "pepper" image. All images used in our experiments are 512 × 512 pixels in size. Observe that the data-dependent triangulation in (a) contains many unorganized slivers. These triangles usually have negligible interior area, which makes further refinement impossible. Figs. 5b and 5c show the results of the hybrid schemes with α=0.2 and α=0.6. It can be observed that the average minimal interior angle gets larger as α is increased. At α=1.0, Delaunay triangulation, which fully preserves triangle regularity, is shown in Fig. 5d.

According to the diamond plots in Fig. 6, for the fully data-dependent case (α=0.0), approximately 21% fewer triangles are needed to represent the image as compared to the Delaunay case (α = 1.0), although the PSNR (the accuracy measurement known as peak-signal-to-noise ratio) accuracy values are approximately the same (29.9 and 30.0 dB, respectively). A somewhat surprising result occurs with the choice of α=0.2. In this case, better approximation accuracy (PSNR is 30.6 dB) is achieved with nearly 29% fewer triangles than for the Delaunay case.

Additional results for the well-known "Lena" image are also graphed in Fig. 6. The graphs reveal a general trend toward better approximation accuracy with fewer mesh elements as α approaches 0 (data-dependent scheme), as we might expect.

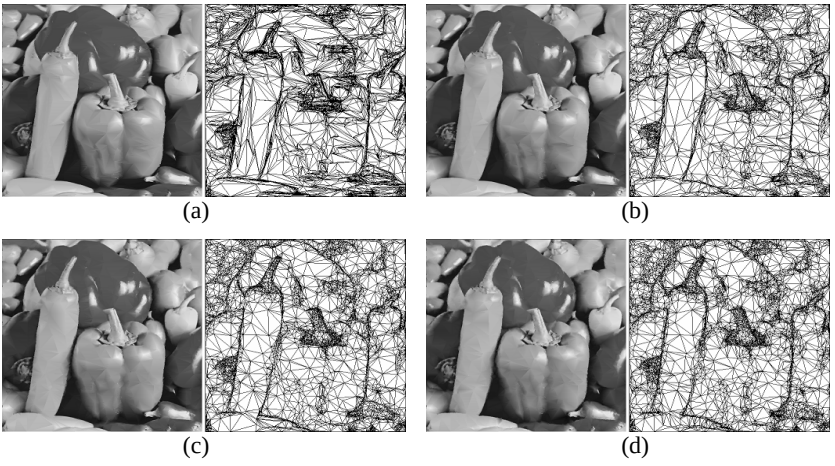


Fig. 5. Reconstructed image comparison for the "pepper" image. Proceeding from top to bottom, mesh regularity factor values are (a) α = 0.0 (fully data-dependent), (b) α = 0.2 (hybrid), (c) α = 0.6 (hybrid), and (d) α = 1.0 (Delaunay triangulation).

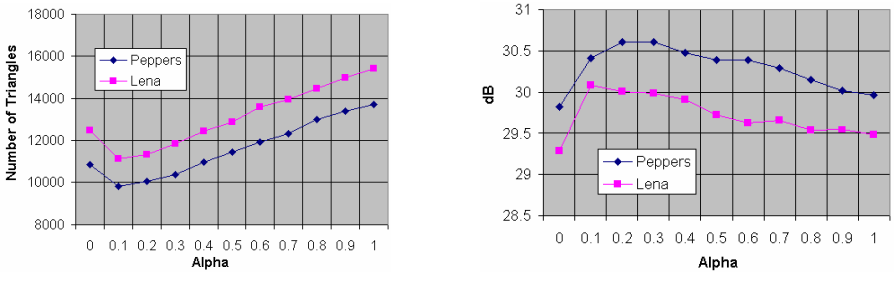


Fig. 6. (a) The number of triangles needed to approximate the "Lena" and "pepper" images at different values of α, the mesh regularity factor. (b) Corresponding PSNR for the images. Empirically, the optimum value for α is around 0.2.

However, there is an optimum value near $\alpha \approx 0.2$ which yields the best results in terms of PSNR accuracy, and it does so with fewer mesh elements than either $\alpha = 0.0$ or $\alpha = 1.0$. We have observed this in our tests with other images as well. This behavior may be explained in part by the fact that very narrow triangles, which result for $\alpha \approx 0$, are not well suited for subdivision during successive refinement steps. The value α in the range 0.1 to 0.3 therefore represents a good trade-off between approximation accuracy and suitability for multiresolution analysis and refinement.

4.2 Level-of-Detail and Mesh Reduction

Next, we demonstrate that the quality of the reconstructed image can be easily controlled by the specification of such parameters as the wavelet energy threshold, δ_{w1} (9), vertex removal threshold, δ_λ (13), and edge removal threshold, δ_d (15). Fig. 7



Fig. 7. Example of level-of-detail control on the "Elaine" image. Beginning from left to right, δ_{w1} is set to 80, 60, 40, and 20, respectively.

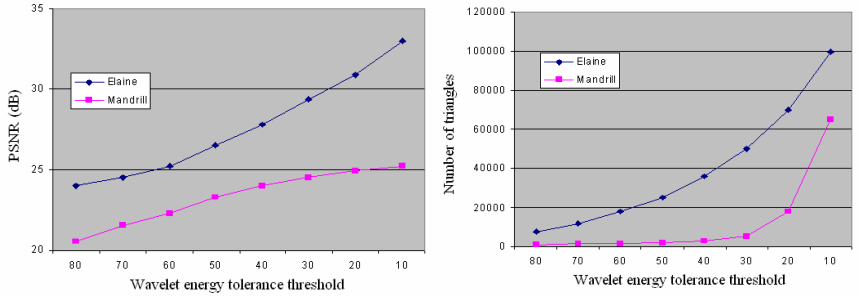


Fig. 8. Level-of-detail analysis. The graphs compare PSNR and the number of triangles used, respectively, for reconstruction at different values of δ_{w1}.

Table 1. Vertex removal performance as a function of vertex removal threshold δ_λ

	Lena			Elaine		
δ_λ	PSNR	$\lvert\Gamma_m\rvert$	$\lvert V_m\rvert$	PSNR	$\lvert\Gamma_m\rvert$	$\lvert V_m\rvert$
0.000	30.66	14862	7454	31.26	19053	9552
0.050	30.26	12881	6461	31.02	16876	8462
0.100	29.62	11183	5610	30.74	14971	7510
0.125	29.07	10393	5215	30.71	13996	7019
0.150	28.70	9477	4754	30.41	12996	6519

shows the reconstructed images for different choices of δ_{w1} on the "Elaine" image. For lower values of this threshold, more mesh elements are chosen as candidate triangles for refinement, thus resulting in better image quality. Fig. 8 shows the comparison of the PSNR of the reconstructed images and the number of triangles used for reconstruction for the "Elaine" and "mandrill" images. In another mesh reduction experiment on "Elaine" and "Lena" images, the effect of the vertex removal threshold, δ_λ , is shown in Table 1. If this parameter was set too low, then very few vertices were removed due to the restricted normal vector deviation, resulting in a higher number of triangles, $|\Gamma_m|$, and vertices, $|V_m|$. From the table, we see that the number of vertices and PSNR of both images decrease approximately linearly with δ_λ.

4.3 Multiresolution and Mesh Generation

Our method also supports progressive refinement of an image during reconstruction. This can be crucial when images are transferred at low data rates. To illustrate this, reconstructed images at different resolution levels are shown in Fig. 9. Because most mesh vertices should lie near intensity edges of an image, wavelet coefficients are efficient for progressive reconstruction. Table 2 shows the numerical results for the operations. The column labels L , $|\Gamma_m|$, and $|V_m|$ represent the level number, the number of triangles, and the number of vertices, respectively. The fourth column $\%V$ represents the percentage of vertices used by the triangulation relative to the number of total pixels in the original image. The bit rate, *bpp*, is an estimate of the number of bits per pixel and is calculated using $(2|\Gamma_m|+12|V_m|)/HW$ as described in [14], [15]. The first row of the table (L=6) represents the result of initial triangulation. The next 6 rows (L=5 to L=0) show the result after each successive refinement step, using the hybrid criteria (α = 0.2). The last 2 rows (levels -1 and -2) represent further refinement using the absolute maximum error alone. The reason for using maximum error for these levels is because wavelet coefficients are not sensitive enough to select further candidates for refinement.



Fig. 9. Reconstructed versions of the "Lena" image at several scales. From left to right, the resolution levels are 6, 4, 2, 0, and -2, respectively.

Table 2. Numerical results for the "Lena" image, for levels 6 to –2

L	$\|\Gamma_m\|$	$\|V_m\|$	**%***V*	*PSNR*	*bpp*
6	294	170	0.065	16.54	0.010
5	294	170	0.065	19.47	0.010
4	726	386	0.147	22.14	0.023
3	1624	835	0.319	24.56	0.050
2	3196	1621	0.618	26.64	0.099
1	5400	2723	1.039	28.25	0.166
0	8412	4229	1.613	29.43	0.258
-1	10800	5423	2.069	30.06	0.331
-2	11106	5576	2.127	30.17	0.340

4.4 Applications

Mesh coding schemes, unlike block coding, facilitate several applications that are of interest for higher-level image analysis. This section illustrates three such applications: selective refinement, image segmentation, and object recognition.

Selective refinement, sometimes called region-of-interest (ROI) refinement, refers to the use of different resolution levels for different portions of an image during reconstruction. A common approach is to provide a mask (either manually or automatically) that indicates a ROI to be reconstructed at a finer scale than the rest of the image. Fig. 10 illustrates selective refinement using two different masks for the same image. During mesh generation, the system only performs refinement at finer scales for triangles overlapped by the mask.

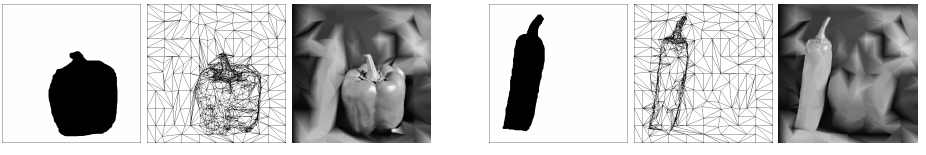
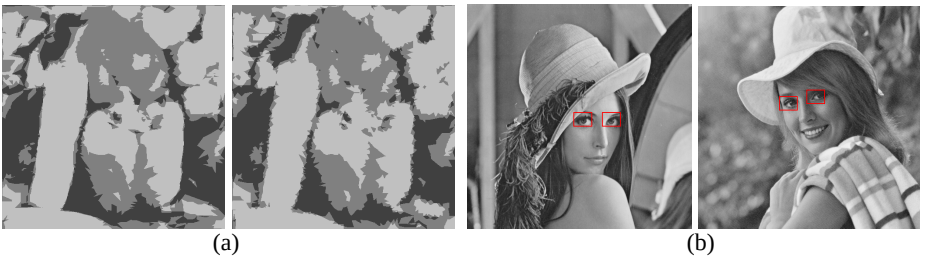


Fig. 10. Illustration of selective refinement. At the left, two different masks have been provided for the "pepper" image. (Both were created manually.) Two respective triangular meshes are shown in the middle. Reconstructed images appear at the right.

Segmentation procedures can also be benefit from this iterative mesh coding. Examples of mesh-based image segmentation are shown in Fig. 11a. The image shown on the right is the result from Delaunay triangulation, and we see that the regions contain unsatisfactory jagged borders. In contrast, our hybrid approach (α=0.2) results in regions with borders that are much smoother, and therefore much more useful for higher-level analysis.

Finally, our triangular mesh coding approach offers advantages for simple object recognition tasks. To demonstrate the feasibility of content retrieval, we illustrate this with the example application of fast detection of human eyes.



(a) (b)

Fig. 11. (a) Segmentation of the "pepper" image. Segmentation result from hybrid triangulation with α = 0.2 (left) and Delaunay triangulation (right). (b) Results of a simple eye detection algorithm, which benefits from triangular mesh coding. Detected eye locations from "Lena" and "Elaine" images are shown in red rectangles.

Because the appearance of the eyes typically contains high-contrast light and dark regions, a mesh created using wavelet information will encode the presence of these high-frequency image components. Fig. 11b shows the results of our (relatively simple) eye-detection algorithm. After mesh construction, triangle sizes and adjacent-triangle contrast were analyzed to determine potential candidates for eye locations. In our 512 × 512 images, 50 to 300 triangles were quickly identified as potential candidates. This represents a significant in data reduction for processing. K-means clustering algorithm was then used to group the triangles. Finally, the algorithm made the assumption that two eyes in an image should be almost identical. Therefore, we verify the detection of eyes by using normalized cross-correlation.

5 Conclusion

This paper has presented a novel wavelet-based method for image coding using a hybrid of Delaunay and data-dependent triangular mesh criteria. Wavelet coefficients are used directly in selecting mesh node locations, in selecting node interconnections, and for mesh refinement. The advantage of using the hybrid approach is to balance the advantages of data-dependent triangulation (better approximation) with Delaunay-type triangulation (more suited to multiresolution subdivision). This method is suitable for multiresolution image coding, content-based image retrieval from databases and next-generation data compression, where high-level (object-level) semantic information is crucial. We have demonstrated this with such multimedia applications as selective compression, segmentation, and eye detection.

References

1. da Silva, L.S., Scharcanski, J.: A Lossless Compression Approach for Mammographic Digital Images Based on the Delaunay Triangulation, Proc. IEEE International Conference on Image Processing, 2005, vol. 2, pp. 758-761.
2. Prasad, L., Skourikhine, A.N.: Vectorized Image Segmentation via Trixel Agglomeration, Pattern Recognition, 2006, vol. 39, pp. 501-514.
3. Lee, S., Abbott, A.L., Schmoldt, D.: Wavelet-Based Hierarchical Surface Approximation from Height Fields, Proc. IEEE Conference on Computer Vision and Pattern Recognition, 2004, vol. 1, pp. 299-305.
4. Liu, N., Yin, Y.L., Zhang, H.W.: A Fingerprint Matching Algorithm Based on Delaunay Triangulation Net, Proc. 5th International Conference on Computer and Information Technology, 2005, pp. 591-595.
5. Flores, J.M., Bloch, I., Schmitt, F.: Base Surface Definition for the Morphometry of the Ear in Phantoms of Human Heads, Proc. 25th Annual International Conference of the Engineering in Medical and Biology Society, Sept. 2003, pp. 541-544.
6. Demaret, L., Dyn, N., Iske, A.: Image Compression by Linear Splines over Adaptive Triangulations, Signal Processing, 2006, vol. 86, pp. 1604-1616.
7. Damaret, L., Iske, A.: Adaptive Image Approximation by Linear Spline over Locally Optimal Delaunay Triangulation, IEEE Signal Processing Letters, 2006, vol. 13, pp. 281-284.

8. Kropatsch, W.G., Bischof, H.: Digital Image Analysis: Selected Techniques and Applications, New York: Springer, 2001.
9. Yang, Y., Wernick, M.N., Brankov, J.G.: A Fast Approach for Accurate Content-adaptive Mesh Generation, IEEE Transactions on Image Processing, August 2003, pp. 866-881.
10. Klein, R., Cohen-Or, D., Huttner, T.: Incremental View-dependent Multiresolution Triangulation of Terrain, Proc. 5th Pacific Conference of Computer Graphics and Applications, Oct. 1997, pp. 127-136.
11. Rila, L., Constantinides, A. G.: Image Coding using Data-dependent Triangulation, Proc. 13th International Conference on Digital Signal Processing, July 1997, pp. 531-534.
12. Battiato, S., Barbera, G., Di Blasi, G., Gallo, G., Messina, G.: Advanced SVG Triangulation/Polygonalization of Digital Images, Proc. SPIE Electronic Imaging - Internet Imaging VI, Jan. 2005, vol. 5670, pp.1-11.
13. Trisiripisal, P.: Image Approximation using Triangulation, M.S. thesis, Virginia Polytechnic Institute and State University, 2003.
14. Rossignac, J.: Edgebreaker: Connectivity Compression for Triangular Meshes, IEEE Transactions on Visualization and Computer Graphics, Jan. 1999, pp. 47-61.
15. Lee, E.S., Ko, H.S.: Vertex Data Compression for Triangular Meshes, Proc. 8th Pacific Conference on Computer Graphics and Applications, Oct. 2000, pp. 225-234.

A Novel Video Coding Framework by Perceptual Representation and Macroblock-Based Matching Pursuit Algorithm*

Jianning Zhang, Lifeng Sun, and Yuzhuo Zhong

Department of Computer Science and Technology,
Tsinghua University,
Beijing 100084, China
zjn01@mails.tsinghua.edu.cn, sunlf@mail.tsinghua.edu.cn,
zyz-dcs@mail.tsinghua.edu.cn

Abstract. This paper presents a novel hybrid video coding framework by perceptual representation and macroblock-based matching pursuit algorithm (PRMBMP), which uses a set of filters to extract the perceptual parts of each video frame. The original video frame is separated into low-frequency image and high-frequency image. The low-frequency image has low sensitivity to human perception and few complex texture details, which can be handled efficiently by traditional H.264 video coding. The high-frequency image is the perceptual representation, which includes more texture details and edges. The proposed macroblock-based matching pursuit algorithm is used to compress the high-frequency image, which speeds up the conventional matching pursuit algorithm efficiently by estimating the local optimization. The experiments show that the proposed framework can achieve 2 to 3 dB improvements compared with the conventional H.264 video coding framework. The proposed framework also has the definition scalability, which can be widely used in bandwidth-variation video applications.

Keywords: Hybrid Video Coding, Visual Perceptual, Video Representation, Filter-based Decompose, Matching Pursuit.

1 Introduction

Video coding is a popular research area over the last few decades. The traditional hybrid video coding framework has been developed such as the completed international video coding standard MPEG1/2, H.263 [1] and H.264 [2]. The motion estimation and compensation are used to search the predictive blocks and calculate the residue blocks for the prediction coding. The DCT is accepted as the transform coding algorithm for each encoding residue block signals. The main problem is that the motion estimation only catches the signals similarity in the spatial domain,

* This work has been supported by the National Natural Science Foundation of China under Grant No. 60503063 and No. 60432030 and supported by the National Basic Research Program of China (973) under Grant No. 2006CB303103.

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 322 – 331, 2007.
© Springer-Verlag Berlin Heidelberg 2007

without considering of the texture changing and deforming. When the video content has the complex texture details, even small texture deforming or light changing can make the poor efficiency in predicting and matching of motion estimation method, which will result in large block residues.

To decrease the effect of the complex texture details dynamics, the object-based video coding framework is proposed and accepted in MPEG-4 video coding standard [3]. The complex video objects extraction and coding without 2D or 3D models are proposed in [4]. But it is only applied in static camera situations. So how to segment and track the complex video objects precisely in dynamic scenes and complex camera changing is still an open issue.

On the other hand, the DCT lacks of considering about the signals features or patterns, which will result in poor transform efficiency while coding the residue blocks with plenty high frequency details. Furthermore, the DCT tends to introduce coding artifacts especially block edges at low bit rates. Wavelet transform has also been used such as [5], which may show ringing artifacts around high contrast edges. To overcome these problems, the matching pursuit decomposing on an over-complete non-orthogonal basis set has been developed in recent years [6][7][8]. Instead of DCT, the matching pursuit is used to decompose the residue signals into the non-orthogonal feature spaces, which can fit the high frequency features and details very well with less numbers of used basis (so called *atom*) than DCT. It shows very high residue coding efficiency for sparse residue signals. But in complex texture details dynamics situations, the residue signals after motion estimation and compensation are not sparse, so the numbers of atoms used to represent the residue signals will increase evidently, which results in the poor coding efficiency. Another problem of the matching pursuit methods is the high computation cost for the computing time and storage space. Some fast matching pursuit algorithms have been proposed by using FFT and picking up more than one atom per iteration [9] or vector norm comparison [10] with the cost of very high memory occupying.

To solve these problems, a novel hybrid video coding framework by perceptual representation and macroblock-based matching pursuit algorithm (PRMBMP) is proposed in this paper. The proposed framework uses a set of filters to decompose the original video frame into low-frequency image and high-frequency image. The low-frequency image has low sensitivity to human perception and few complex texture details, which can be handled efficiently by traditional H.264 video coding. The high-frequency image is the perceptual representation of the original video frame, which includes more texture details and edges. It is very suitable for the matching pursuit algorithm. To speed up the conventional matching pursuit algorithm, the macroblock-based matching pursuit algorithm is proposed, which estimates the local optimization inside each size-predefined macroblock. The proposed framework can not only increase the coding performance, but also have the definition scalability, which can be widely used in bandwidth-variation video applications.

The rest of this paper is organized as follows: Section 2 describes the proposed video coding framework with the filter-based image decomposing. The proposed macroblock-based matching pursuit algorithm for high-frequency image coding is illustrated in Section 3. Section 4 shows the experimental results. The conclusions and discussions are given in Section 5.

2 The Proposed Video Coding Framework

In this section, we propose a novel hybrid video coding framework by perceptual representation and macroblock-based matching pursuit algorithm. First we set $I_1 \ldots I_n$ denote n successive video frames in the original source video sequence where n is the total frame number in the video sequence. The linear decomposition for the K_{th} frame to a set of different frequency images can be shown below simply.

$$I_k = \sum_{f=0}^{M} \alpha_k^f I_k^f + \beta_k e_k , \tag{1}$$

where f denotes frequency band which can be set from 0 to M. Notice that f =0 means the low-frequency band, and f =1…M denotes 1 to F high-frequency bands. I_k^f denotes the component image with frequency band f for the original frame k. α_k^f means the corresponding weight. $\beta_k e_k$ denotes the decomposed noise. So for each video frame, the original image can be decomposed into F component images linearly. In practices, the decomposed noise can be added into the highest frequency band image to avoid the noise modeling and coding with extra coding bits. The equation (1) can be simplified as F frequency bands linear decomposition with equal weight and F=2. This simplification is practical in video coding applications without much performance decrease. And also our proposed framework can be easily extended into multiple-layered video coding with different scalable weight to meet the requirements of scalable video coding. So (1) is simplified as,

$$I_k = I_k^0 + I_k^1 \tag{2}$$

Equation (2) illustrates that the original video frame can be decomposed into two band images: low-frequency image and high-frequency image. The low-frequency image and high-frequency image can be compressed by different video coding strategies.

2.1 The Encoding Framework

The PRMBMP encoder framework is shown in Fig. 1. Each frame I in original video is first decomposed into low-frequency image I^0 and high-frequency image I^1 by the proposed filter-based decomposition method, which will be illustrated below. The low-frequency image sequence is encoded by conventional H.264 video encoder. The high-frequency image consists of high frequency signals including texture details and edges. The hierarchical macroblock-based matching pursuit algorithm is proposed to encode the high frequency image sequence on an over-complete static dictionary. In the implementations, two hierarchical macroblock layers are used: 16x16 and 8x8. The macroblock-based matching pursuit finding process is first applied on 16x16 macroblocks of the high-frequency image and then applied on 8x8 macroblocks of the residue image derived from the above 16x16 macroblock layer matching pursuit. The detail will be shown in Section 3.

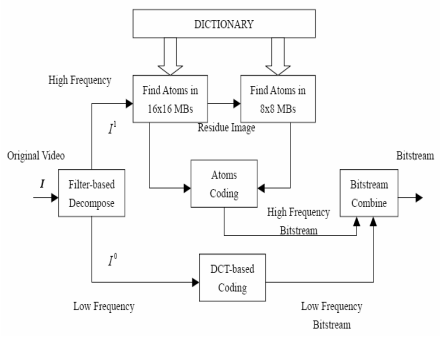


Fig. 1. Encoding Framework

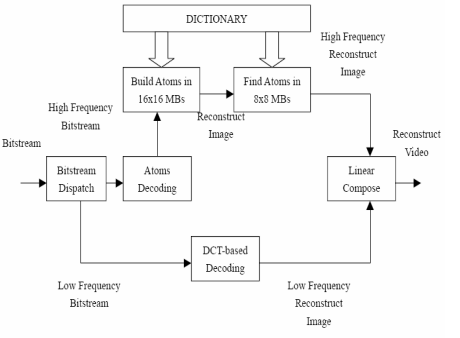


Fig. 2. Decoding Process

2.2 The Decoding Process

The decoding process of the proposed PRMBMP framework is shown in Fig. 2, the encoded bitstream is dispatched into low-frequency and high-frequency bitstream to send to different decoders. The low-frequency decoder is derived from the standard H.264 reference decoder. The high-frequency bitstreams is decoded using the matching pursuit atoms decoder, which decodes each atom in different hierarchical macroblock layers. Then atoms building steps of 16x16 and 8x8 macroblock layers are applied to achieve the final high-frequency reconstruct image. Finally, the high-frequency reconstruct image and low-frequency reconstruct image are linearly combined with the corresponding weights to output the decoded video frame.

2.3 Filter-Based Decomposition

In order to decompose original image into different frequency bands, a set of low-pass filters are designed with different filter size and strength. The Gauss filters are accepted to make the basis of the low-pass filters. One dimension Gauss function can be written as:

$$g(x) = \frac{1}{\sqrt{2\pi\sigma}} e^{-(x-xw/2)^2/\sigma^2}, \tag{3}$$

where x_w is the filter window size in x dimension, so the center of the Gauss will be the center of the filter window. The σ is the strength factor. Then the 2D Gauss filter F(x,y) with the filter size $x_w \times y_h$ is shown as:

$$F(x,y) = g(x) \cdot g(y) = \frac{1}{2\pi\sigma} e^{-((x-xw/2)^2+(y-yw/2)^2)/\sigma^2} \tag{4}$$

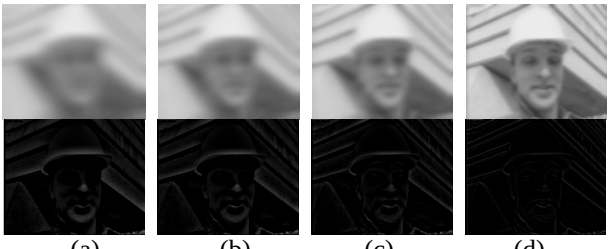
So a set of filters with different sizes and strengths is designed to be applied on the original image signals to decompose the original image into multiple frequency bands images. It is an iterate process, which can be illustrated using the following equations.

$$I^{0'} = I \otimes F^0$$
$$I^{k'} = (I - \sum_{f=0}^{k-1} I^{f'}) \otimes F^k, k = 1,...,M \quad (5)$$

In Equation (5), F^0 … F^M denote M low-pass filters from level 0 to level M derived from equation (4) by different filter sizes and strengths. Practically, we can define F^k to be the 2D Gauss filter with the size of $(2(M-k)+1)\times(2(M-k)+1)$, the strength factor is set to $2(M-k)+1$. In the proposed framework, the filter-based decomposition can result in two frequency images: low frequency image and high frequency image. So after each step k of filtering, the low frequency image and the high frequency image can be calculated as:

$$I^0 = \sum_{f=0}^{k} I^{f'}$$
$$I^1 = I - I^0 \quad (6)$$

Fig. 3 shows an example of filter-based decomposition.



(a) (b) (c) (d)

Fig. 3. Filter-based decomposing. This figure shows the low-frequency image and the high-frequency image derived from each step of filter-based decomposition. (a), (b), (c) and (d) show the results at the step that k=0, 3, 6 and 9 where M = 11. The upper images are the resulted low-frequency images and the lower images are the high-frequency images.

3 Macroblock-Based Matching Pursuit Algorithm

The matching pursuit theory is used for signal decomposition, as shown by [11] and applied to video coding in recent years. The matching pursuit algorithm decomposes a signal using an over-complete dictionary of functions. It can be used to represent residue signals with high frequency and sparse distribution efficiently. The conventional matching pursuit algorithm has high cost in computation and memory, which makes the matching pursuit algorithm hardly to be used in the practical video coding applications especially for large picture size videos. The macroblock-based matching pursuit algorithm can make the tradeoff between coding complexity and coding performance. Using the proposed method, we can speed up the matching pursuit process by estimating the local optimization instead of global optimization and make the algorithm suitable for the high frequency image decomposing and coding.

The over-complete dictionary used to decompose the original residue image signals must be designed according to the signal features and patterns in the residue image. Here, the high-frequency image derived from filter-based image decomposition consists of rich texture details and edges. The characters of texture and edges are arbitrary orientated and Gauss distributed. So the directional Gauss functions with different orientations and different scales are adopted, which are introduced in [12].

The main idea of the macroblock-based matching pursuit algorithm is to estimate the local optimization inside each overlapped redefined macroblock to find a plane of best atoms at one matching pursuit iterating process instead of global optimization for the whole image in conventional matching pursuit. Firstly, in the macroblock-based matching pursuit algorithm, the macroblock size is predefined hierarchically. Set the macroblock size to be $N \times N$ and M macroblocks in the whole high-frequency image, the overlapped search area for each macroblock in the high-frequency image can be extended to $2N \times 2N$ for the reason of protecting the correct atom estimating at the macroblock edges. So the global optimization estimation can be modified to meet the local optimization requirements as the following. For each predefined macroblock m,

$$g_r : \arg\max_{\gamma}\left(p_m = \left\langle f_m, g_\gamma \right\rangle\right),\ f_m \in MB_m, \tag{7}$$

where f_m denotes the signals inside macroblock MB_m . For each matching pursuit iteration, one best atom will be derived inside each overlapped macroblock, and an atom plane with M atoms can be achieved for the whole high-frequency residue image. So after iteration k, the residue image can be calculated as:

$$R_k = R_{k-1} - \sum_{m=1}^{M} p_m g_{\gamma m} \tag{8}$$

The reconstruction of the original image can also be achieved after N iteration:

$$\hat{f} = \sum_{n=1}^{N}\sum_{m=1}^{M} p_m(n) g_{\gamma m}(n) \tag{9}$$

The detail algorithm can be described as:

1) For k iteration process
2) For each macroblock m in the whole residue image
3) For each atom with the index γ in the dictionary D
4) Calculate the inner products inside macroblock m in the searching window NxN.
5) Loop until the end of 3)
6) The best atom $g_{\gamma m}$ is selected for each macroblock m with the maximum inner product by the equation (7).
7) Loop until the end of 2)
8) Save the best atoms for all the macroblocks in the whole residue image and update the residue image using the equation (8)
9) Loop until the matching pursuit completed.

According to [9], the complexity of the conventional matching pursuit algorithm for decomposing a $W \times H$ size image can be calculated as:

$$C = k \cdot N \cdot d \cdot n \log_2 n,\ n = W \times H, \tag{10}$$

where N is the number of iterating times, d is the dictionary size and k depends on the strategy of the atom searching (k<<1). So we can see that for the proposed macroblock-based matching pursuit, if the image is divided into M macroblocks, the matching pursuit complexity will be:

$$C_{MB} = M \cdot k \cdot N \cdot d \cdot \frac{n}{M} \log_2 \frac{n}{M}, \text{ so}$$
$$C_{MB} = C - k \cdot N \cdot d \cdot n \log_2 M \tag{11}$$

So the computation complexity can be reduced according to the number and the size of macroblocks predefined in the high-frequency image. In the same time, there are no needs to cost extra memory for the temporal saving. So the proposed macroblock-based matching pursuit algorithm can reduce the computation complexity with low memory cost.

The predefined macroblock size can be reduced hierarchically to perform multi-pass macroblock-based matching pursuit to refine the signal decomposition accuracy. In the practical framework as shown in Section 2, two macroblock size matching pursuit passes are developed. The original high-frequency image is first decomposed by 16x16 macroblock matching pursuit process to represent large residue signals, and then the result residue image is decomposed by 8x8 macroblock matching pursuit process to catch the smaller residue signals.

After the hierarchical macroblock-based matching pursuit, the found atoms must be encoded to output into the bitstream. Considering the characters of the proposed algorithm, that the atom finding process estimates the local maximization inside each macroblock, the atoms may be redundant or repeated among the macroblocks. An atom clustering method is used to group the same atoms by its dictionary index. The translation parameter b can also be represented by the shorten offset from the center of the macroblock that the atom belongs to. Finally, the conventional position prediction coding is applied before the entropy coding. The conventional position prediction coding and entropy coding can be found in [6] for the details.

4 Experimental Results

The experiments are performed on the basis of the conventional H.264 video coding framework. In the experiments, the proposed PRMBMP video coding framework uses the conventional H.264 video coding scheme to encode the low frequency image sequence derived from filter-based decomposition. And we will compare the PRMBMP video coding framework with the conventional H.264 video coding framework for the coding performance. The testing video sequences are QCIF format sequences including the complex texture and complex motion sequences and the simple texture and low motion sequences. The coding frame rate is 10Hz, and QP is set from 15 to 40 to show the simulation results from high bitrate situations to low bitrate situations. The average PSNR and bitrate are evaluated.

4.1 Coding Performance Comparison

The coding performance comparison will be illustrated by the Rate-Distortion (RD) curves. The RD curves of Foreman and Coast Guard are shown in Fig. 4.

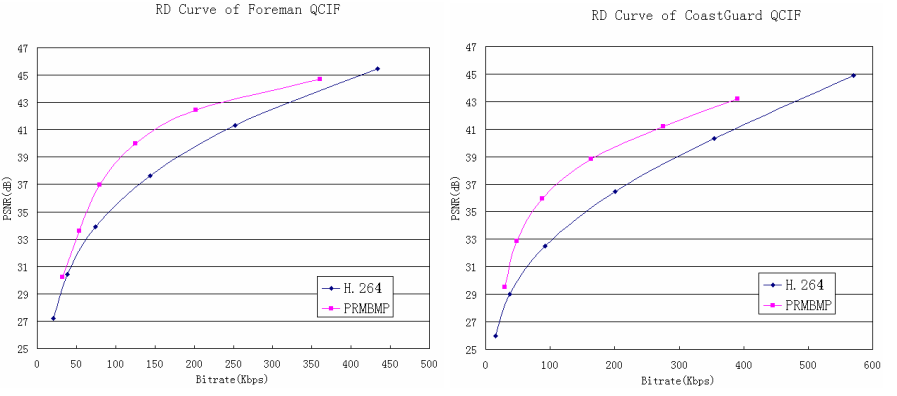


Fig. 4. RD curves of Foreman and GoastGuard QCIF sequences

From the RD curves, we can see that the proposed PRMBMP video coding framework performs better coding efficiency than the conventional H.264 video coding standard. It also can be seen that for the situations with the complex texture details and complex motions, the proposed PRMBMP video coding framework can achieve very high coding efficiency with 2-3 dB increasing of the average PSNR. And for the situations with simple texture and low motions, the PRMBMP framework can achieve 1-2 dB PSNR increasing in the high bitrate conditions compared with the conventional H.264 video coding. For the situation with simple texture and motion, the H.264 video coding can achieve nearly the same coding performance with the PRMBMP video coding with lower bits used, especially in the very low bit rate conditions, in which situation, the atoms coding still remain much bits to represent the high-frequency image signals. Also the proposed PRMBMP framework can increase

Table 1. The Selective Comparison Results

Seq.	QP	H.264		PRMBMP	
		PSNR (dB)	Bitrate (Kbps)	PSNR (dB)	Bitrate (Kbps)
Foreman	25	37.64	143.68	39.99	125.27
	35	30.43	38.25	33.61	53.68
Coast Guard	25	36.44	200.9	38.85	163.58
	35	29.03	37.78	32.86	48.4
Silent Voice	25	38.45	84.89	40.47	83.52
	35	31.21	22.15	34.21	41.1
News	25	39.02	82.21	39.39	71.4
	35	31.43	23.85	33.7	33.2

the coding performance more evidently in the middle and the high bit rate conditions than in the low bit rate conditions compared with the conventional H.264 video coding. The selective comparison results are shown in Table 1.

4.2 Definition Scalability

The definition scalability is a natural scalable feature for the proposed PRMBMP framework, because of the filter-based decomposition and the matching pursuit processing. The low frequency image reconstructed from the decoder can be treated as the base layer video stream with least visual definition to meet the least bandwidth requirement. With the transmission bandwidth increasing, more high frequency atoms are received at the decoder. The more atoms are decoded and reconstructed, the more definition is achieved gradually. This character of the proposed video coding framework can be called definition scalability. Fig. 5 shows one example of the decoding results with different received high frequency atoms numbers. The upper images of each figure show the reconstruct images, which are decoded from the all-received low-frequency bitstream and partial-received high-frequency bitstream. The lower images of each figure denote the atoms reconstruct results from the partial-received high-frequency bitstream. From these figures, we can see that the video frame becomes clearer and higher definition when the decoder receives more bitstream having more atoms. Because the matching pursuit decomposition process guarantees that the most significant effective atoms are found and coded first, which make the scalability more practical. And the video quality becomes acceptable even with partial bitstream receiving.

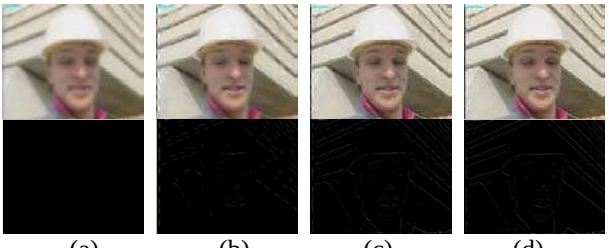


Fig. 5. Definition Scalable Samples for Foreman QCIF, (a) atoms = 0, (b) atoms = 100, (c) atoms = 500, (d) atoms = 1000, the upper images are reconstruct images and the lower images are atoms reconstruct results

5 Conclusions

This paper proposes a novel hybrid video coding framework by perceptual representation and macroblock-based matching pursuit algorithm (PRMBMP). The proposed PRMBMP video coding framework uses a set of filters to extract and represent the perceptual parts of each video frame with high visual sensitivity. The original video frame is separated into low-frequency image and high-frequency image. The low-frequency image has low sensitivity to human perception and few complex texture details. The traditional DCT-based video coding algorithm is applied

on low-frequency image, which can achieve very high coding efficiency. The high-frequency image is the perceptual representation of the original video frame, which includes more texture details and edges. The proposed macroblock-based matching pursuit algorithm is used to compress the high-frequency image, which speeds up the conventional matching pursuit algorithm efficiently by predicting and estimating the local optimization inside each size-predefined macroblock. Hierarchical matching process is developed to refine the signal decomposing gradually. The experiments show that the PRMBMP framework can achieve improvements of average 2 to 3 dB for the situations with complex texture details and dynamics, and average 1 to 2 dB for the simple texture situations compared with the conventional H.264 video coding framework. The proposed video coding framework can not only increase the coding performance evidently but also have the definition scalability, which can be widely used in different bandwidth requirements video coding applications.

Future works will focus on the dictionary design with dynamics adaptive dictionary and more efficient atoms coding for the hierarchical macroblock-based atoms structures. The multi-layers video coding and frequency band scalable video coding using the proposed PRMBMP framework can also be achieved better results than the traditional methods, which are new challenging works.

References

1. ITU-T Recommendation H.263 Version 2, Video Coding for Low Bit Rate Communication. Draft (1998)
2. ITU-T Recommendation H.264/ISO/IEC 11496-10, Advanced video coding, Final Committee Draft, Document JVT-G50, Dec. 2002
3. MPEG Video Group, MPEG-4 video verification model version 18.0. ISO/IEC JTC1/SC29/WG11 N3908 (2001)
4. Hakeem, A., Shafique, K., Shah, M.: An Object-based Video Coding Framework for Video Sequences Obtained From Static Cameras. ACM Conference on Multimeida 2005.
5. Martucci, S.A., Sodagar, I., Chiang, T., Zhang, Y.Q.: A zerotree wavelet coder. IEEE Transactions on Circuits and Systems for Video Technology, vol. 7, no.1, pp. 109-118, February 1997
6. Neff, R., Zakhor, A.: Matching pursuit video coding at very low bit rates. IEEE Transactions on Circuits and Systems for Video Technology, pp. 158-171, February 1997
7. Neff, R., Zakhor, A.: Matching pursuit video coding. I. Dictionary approximation. IEEE Transactions on Circuits and Systems for Video Technology, 12(1): 13-26, January 2002
8. Schmid-Saugeon, P., Zakhor, A.: Dictionary Design for Mathcing Pursuit and Application to Motion Compensated Video Coding. IEEE Transactions on Circuits and Systems for Video Technology, vol. 14, no.6, June 2004, pp 880-886
9. Granai, L., Maggio, E., Peotta, L., Vandergheynst, P.: Hybrid Video Coding based on Bidimensional Matching Pursuit. EURASIP – Journal on Applied Signal Processing, Vol. 2004, No. 17, pp.2705-2714, December 2004
10. Jeon, B., Oh, S.: Fast Matching Pursuit With Vector Norm Comparison. IEEE Transaction on Circuits and Systems for Video Technology, vol. 13, No.4, April 2003
11. Mallat, S., Zhang, Z.: Matching pursuits with time-frequency dictionaries. IEEE Transaction on Signal Processing, 41(12):3397-3415, December 1993
12. Figueras, R., Ventura, I., Vandergheynst, P., Frossard, P.: Low rate and flexible image coding with redundant representations. IEEE Transactions on Image Processing, vol. 15, no. 3, pp. 726-739, March 2006

MetaXa—Context- and Content-Driven Metadata Enhancement for Personal Photo Books

Susanne Boll[1], Philipp Sandhaus[2], Ansgar Scherp[2], and Sabine Thieme[3]

[1] University of Oldenburg, Germany
Susanne.Boll@informatik.uni-oldenburg.de
[2] OFFIS - Institute for Information Technology, Oldenburg, Germany
{sandhaus,scherp}@offis.de
[3] CeWe Color AG & Co. OHG, Germany
thieme@cewecolor.com

Abstract. Making a photo book as a special gift to your beloved can be very time-consuming. One has to carefully select and arrange the pictures nicely over the pages of a previously bought photo book. In these days, photo finisher companies are able to directly print and bind a nice photo book from a selected set of images. But for the users of the software that comes with the album creation, the selection and arrangement of pictures in the album still remains a tedious task. What is missing are easy and good suggestions which pictures to select and how to arrange them into a personal photo book. A higher availability of metadata with the pictures could enable a content-driven and context-driven selection and make album creation better and easier. With MetaXa, we propose a flexible, component-based software architecture that iteratively allows for the multimodal extraction and enhancement of metadata for personal media content. The enhancement process is realized by extraction and enhancement components that each contribute to a well-defined annotation task. Depending on the application domain different components can be configured into a specific instance. With MetaXa, it is hence easy to reuse certain annotation algorithms in different scenarios and to alter a setup by adding or replacing certain enhancement components. MetaXa has been applied to the domain of photo book creation by our project partner CeWe Color and evaluated on a large set of consumer photos.

1 Introduction

Management and organization of one's personal photo collection is a laborious and therefore often never regarded task. Thousands of printed photos rest in the darkness and isolation of shoe boxes. In recent years, digital cameras became wide spread. However, they have not changed or solved the organization problem: Now it is pictures dsc2345.jpg to dsc2399.jpg residing in our digital shoebox, e. g., a folder called birthdayParty05. Currently, we are facing a market in which about 20 bn digital photos are taken per year for example in Europe [6]. At the same

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 332–343, 2007.
© Springer-Verlag Berlin Heidelberg 2007

time, we can observe that many digital photos are never viewed nor used again. It is estimated that from all digital images only about 20% are actually printed [6]. This is not because we forget about these digital souvenirs. In a study [6] that our project partner CeWe Color, the world's leading photo finisher and digital photo service provider, has been carrying out by a market research institute is that most users of digital cameras would like to have their photos printed. Why are not more photos (re)used and printed even though it seems to be the customer's wish? The answer we give here is that the way in which we find and select photos from a large set of photos to print them needs far too much time and effort. The central insufficiency we face here today is the fact that digital photos today are just a poor reflection of the actual event captured. Digital cameras leave us with a pixel-based copy and some context information of what we experienced. Anything else is gone with the camera releaser, at least it is decoupled from the digital copy of the moment. Even though there is research in content-based image analysis for quite some years [25] as well as nice photo management tools [7,1,2], neither an automatic labeling nor a manual annotation of photos has become a success model.

What is needed is a better and more effective automatic annotation of digital photos that better reflects one's personal memory of the events captured. This approach would allow different applications to create value-added services on top of them such as the creation of a personal photo book. For this we propose a context-enhanced, multimodal method to achieve better image-understanding by the development of a novel, component-based software architecture.

Following the related work in Section 2, we present our content-based and context-driven metadata enhancement architecture (MetaXa), for iterative metadata extraction and enhancement of digital photos in Section 3. Based on our previous work on exploiting context for personal media collections [24,3], we elaborate the design of the architecture and its components and present our multimodal metadata enhancement in Section 4. Section 5 describes the exploitation of the derived metadata in a concrete, professional photo book software to suggest a good (pre-)selection and composition of photos into an individual photo book, before we conclude the paper and present an outlook to future work.

2 Related Work

In recent years, personal digital photo collections have received a great share of attention in the multimedia and database research community. In 2003, Rodden and Wood investigated if "advanced multimedia processing (speech recognition and content-based image retrieval) [are] useful in the context of personal photo collections?" [23]. The authors come to the conclusion that time and events are the preferred means of browsing through photo collections rather than advanced multimedia features. They also found out that manual annotation can not be expected form the everyday user. Interestingly, the participants still wanted to have prints. This observation very much complies with the study [6] that our project partner commissioned in 2003.

In the large and established research field of content-based image retrieval, we find approaches that address the domain of personal photo collections. In these approaches, content-based analysis, partly in combination with user relevance feedback, are used to annotate and organize personal photo collections. Prominent early systems are, e. g., MiAlbum [29], AutoAlbum [21], or SmartAlbum [27]. In the context of the DIVA project [19] a learning-based approach for content-based annotation of photos is introduced. In [30] content-analysis was used to automatically annotate photos based on face-recognition of family members. As the retrieved photos need to meet the users' expectation, a recent publication [14] proposes a hypothesis about human perception of image relevance. In the approaches referred to so far, the context of the photos is not included and exploited for the photo management and organization tasks. However, it became clear in content-based image retrieval that "One way to resolve the semantic gap comes from sources outside the image ..." [25].

With the availability of time and location from digital cameras, we find related work that aims to use this contextual information, sometimes in combination with content-based features, for organizing and accessing digital photo collections. In PhotoTOC [22], time and color histograms are used for the organization of photos in the visual user interface. Stating that "time matters" Mulhem et al. [15] define hierarchical temporal events as a clustering and organizational means. Also Graham et al. [8] consider "Time as Essence for Photo Browsing" in a calendar-based browser. Recently, FXPAL presented an elaborated temporal clustering for photo collections [5] based on similarity of time-stamps. In the ATLAS project, location and time are used for the organization of image collections [20]. Naaman et al. also exploit location for the automatic photo organization [17] and combine space and time for photo browsing [16]. In [4,13] the authors discuss the use of content and context for scene classification.

Leaving the content-based field, there is also recent work in which only context information is used to annotate photos. In [18] identity-label suggestions are based on temporal, spatial, and social context. With the availability of EXIF header [11] for photos, this contextual information can be exploited for image understanding. The architecture presented in [12] proposes a context-based keyword creation for mobile video clips.

Considering the related work, we see approaches that either exploit content-based metadata, context-based metadata, or sometime also a multimodal combination of both to manage and organize photo collections. However, a systematic approach that combines content- and context-based metadata extraction and enhancement in the context of personal photo book applications does not yet exist. Consequently, we the propose with the MetaXa[1] architecture an approach that aims at embracing and advancing the state-of-the-art in multimodal metadata enhancement as well as systematically integrating content-based and context-based information. This approach is not only of academic interest but also commercially relevant as our concrete photo book scenario shows.

[1] Metaxa is a Greek liqueur, a blend of brandy and wine. Here, it means an architecture that is a blend of content- and context-based metadata enhancement.

3 Overview of the MetaXa Architecture

The goal of MetaXa is to provide a component-based architecture for context-enhanced multimodal extraction and enhancement of semantic descriptions of personal photo collections. The architecture allows to configure the setup of metadata extraction and enhancement components. By the dynamic creation of an appropriate workflow a concrete instance of the architecture in realized to meet the specific metadata enhancement of a domain. This architecture not only allows to reuse components in different application domains but also to easily extend the architecture by new extraction and enhancement approaches.

3.1 The General MetaXa Architecture

The central elements and features of the MetaXa architecture are illustrated in Figure 1. Input for the architecture are the photos taken by a digital camera. These photos, together with their contextual metadata, e. g., an EXIF header, enter the central *MetaXa Manager*. The image undergoes a sequence of extraction and enhancement steps, realized by separate components, in which metadata is enhanced iteratively. This allows for modularizing the metadata creation process into different steps. Increasing the amount and quality of available metadata, each step contributes to a better semantic description of the photos.

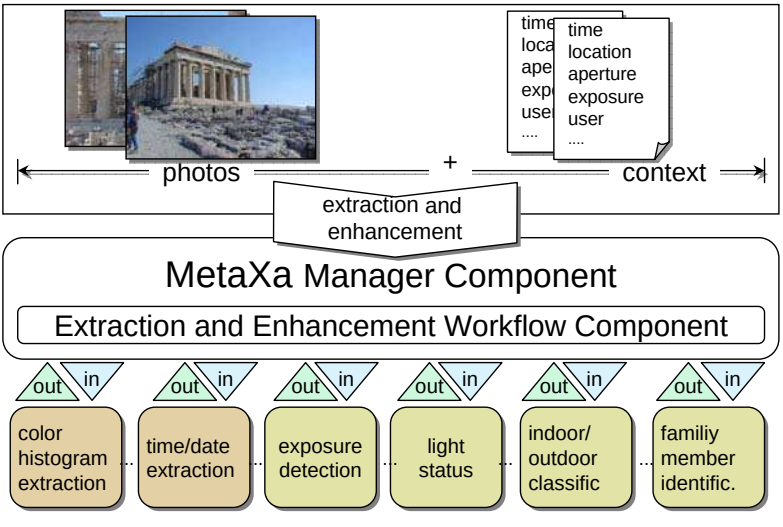


Fig. 1. Metadata extraction and enhancement architecture MetaXa

3.2 Metadata Extraction and Enhancement Workflow

To ensure the correct order of enhancement the different extraction and enhancement steps are driven by a workflow. This workflow configures the sequence of

steps that is carried out for each of the photos. The declarative workflow description identifies the different components and the manager component uses this workflow to drive the extraction and enhancement of the content. This allows to configure the architecture to the actual application needs without having to change the system.

For each extraction and enhancement component, an XML-file specifies which other metadata generated by other components are needed as prerequisites for their metadata extraction and enhancement. A dedicated workflow component is responsible for determining a workflow for this process on basis of the components' XML-specifications. This workflow ensures that all photos pass through the extraction and enhancement components in an reasonable order. This order is calculated by comparing the pre- and post-conditions of each extraction and enhancement component. The workflow component ensures that a specific enhancement component is only called if its pre-conditions are fulfilled. Since extraction components, i. e., a component which bases only on the raw photo data, don not have any pre-conditions, they can always be applied to a photo. The workflow component also detects circular dependencies between the components in order to prevent infinite loops.

3.3 Extraction and Enhancement in MetaXa

In this section, we discuss the general design of the extraction and enhancement components of our system. Input to the architecture are the photos with their metadata. In a first phase, *extraction* components are used to extract relevant metadata and context directly from the photo. Hence, the system starts out with the information that comes with the digital photo itself, together with an optional EXIF header. Each photo is individually analyzed with different content and context feature extraction components that employ state-of-the-art methods. Examples of such components are color histogram extraction, edge detection but also the extraction of time, exposure time or GPS information from the EXIF header. Extraction components, illustrated in gray at the bottom of Figure 2, do not require previously generated metadata and hence form the starting point of the metadata enhancement.

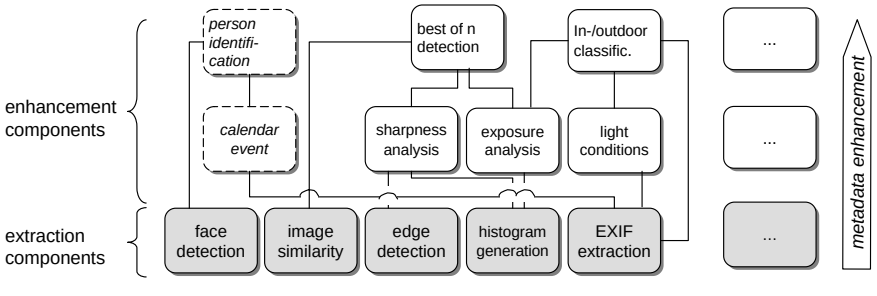


Fig. 2. Different types and dependencies of extraction and enhancement components

On top of the extraction components, *enhancement* components derive new metadata which is stored for further enhancement and use. As illustrated by the edges between the components in Figure 2, enhancement components can use previously extracted or enhanced metadata to create new, potentially higher-level metadata. For this each enhancement component defines the necessary input which it requires to enhance the metadata. All components can still access the raw media content have access to all other photos in the collection, e. g., for similarity detection. Figure 2 illustrates examples of possible extraction and enhancement components and also indicates iterative use and enhancement of the photos' metadata. Each component reveals its function to the MetaXa architecture. Depending on the application needs, our architecture allows to plug-in new components, such as the person identification and calendar event detection currently under development, and to configure the course and extent of extraction and enhancement.

4 Metadata Extraction and Enhancement Components

Having presented an overview of the MetaXa architecture, we now describe the concrete design of the metadata extraction and enhancement components.

4.1 Design of Extraction and Enhancement Components

The different extraction and enhancement tasks are realized by software components. Software components encapsulate their implementation and interact with the environment by means of well-defined interfaces [26]. Thus, a software component comes with a clear specification of what it *requires* and *provides*. As we employ Java to realize the MetaXa architecture, which doesn not natively allow for such a detailed specification, we use an XML-file for every component. These XML-files describe the kind and type of metadata that is necessary (pre-conditions) to apply a specific enhancement component. Since extraction components have no specific pre-conditions, they do not require a pre-condition description. However both extraction and enhancement components contain a section in this XML-file that specifies the provided metadata (post-conditions). The following listing shows an excerpt from such a file for the in-/outdoor enhancement component. This enhancement component requires the light status, time, brightness, and flash usage to determine whether the photo has been captured in-/outdoor.

```
<pre-conditions>
<metadatum description="time" relevance="mandatory">
  <mapping type="time-String" order="1">
    <source>Exif</source>
  </mapping>
  <mapping type="time-String" order="2">
    <source>FileCreationTime</source>
  </mapping></metadatum>
  ...
```

```
10 <post-conditions>
11    <metadatum description="inoutdoor">
12     <mapping>In- or Outdoor</mapping>
13   </metadatum></post-conditions>
```

The example above shows that the component needs several previously generated metadata as input (indicated is only the metadatum time). The metadata entries can be marked as *mandatory* or *optional*. The same kind of metadata can be generated in different ways by different components, e. g., the information when a photo was actually taken. This could either be extracted from the Exif header of a photo or the file modification time. We meet this situation by the introduction of one ore more `mapping` entries in the XML-file. For example, the time metadatum can be achieved from the `source` Exif or file creation time (lines 3-8). A preference for a specific source (e. g., depending on the source's reliability) can be indicated by the parameter `order` (lines 3+6).

4.2 Extraction and Enhancement Components of MetaXa

For MetaXa we developed a set of concrete extraction and enhancement components of which we present representative examples. We developed components that exploit both image content and photo context. For this we employ state-of-the-art technology in content-based and context-based feature extraction and advance it toward context-enhanced multimodal photo metadata annotation.

Extracting content-based features. By content-based extraction we mean the purely pixel-based extraction of features of a photo. Typical components are histogram generation, edge detection, similarity analysis, and face detection. As examples, the similarity analysis and face detection are described in the following.

We developed a simple technique to determine *similarities* between photos. For it, we segment each photo into an 8×8-matrix and calculate the average RGB values. Each photo can then be described by three vectors, one for each RGB-channel. The similarity measure between two photos can then be described as the weighted sum of euclidean distances between the RGB vectors of the two photos. The human eye is not equally sensitive to the the colors red, green and blue. Thus, we weight the three color channels differently according to [9]. It is important to note that the size of the matrix has to be carefully chosen. If the segmentation is too coarse, the differences between two pictures can not be reliably detected as too many details get lost by calculation the averages, which can result in a false high similarity. However, if the segmentation is too fine grained, two pictures that, e. g., only differ in a small horizontal or vertical shift, would likely be considered very different. A good trade-off is a 8×8 matrix.

Face detection in the MetaXa architecture bases on the method presented in [28]. This method uses trained classifiers to rapidly detect faces in pictures. The classifiers were trained with several hundred positive samples and also a few hundred negative examples. The result is a *cascade of boosted classifiers*. The advantage of this algorithm is, that it is very fast and an open source

implementation is available. We use this component to detect the number of faces in the image.

Extracting context-based features. These components utilize purely contextual information such as the EXIF header. *EXIF extraction* is used to extract the EXIF header information which is written to the photos by most consumer cameras. This header varies from camera to camera. However, most cameras at least provide information like timestamp, ISO, aperture, exposure, and if a flash was used. Some cameras also provide the focal length, the orientation and even location information such as a GPS position. Besides this, additionally we developed a component extracting general image information such as width, height, and time from the photo data in case there is no EXIF header available.

Content-based enhancement. Content-based enhancement components solely utilize information from content-based extraction- and other content-based enhancement components. As an example we present a component for sharpness analysis. *Sharpness analysis* is one of the key methods to determine the quality of an image. We developed two simple but fast methods: For both, we assume that most amateur photos contain a region of interest, which is located in the center. Consequently, we segment each photo in a 3×3-matrix and only analyze the center cell. The first method for sharpness detection uses the detection utilizing the Sobel filter [10]. We can use a resulting edge picture to determine an edge histogram. The photo is considered to be sharp if the histogram has high values in the upper bins. The second method utilizes the fact that in image regions, which are considered as sharp, often diverse levels of brightness occur. Thus, we use the brightness histogram and analyze how many bins exceed a certain value. The more of these bins exist the sharper the photo is considered to be. Both components provide this sharpness value to subsequent higher level metadata enhancement components.

Context-based enhancement. Context-based enhancement components solely utilize information from context-based extraction- and other context-based enhancement components. As example we present a component for determining the light conditions of a photo. *Light condition determination* is useful both for search purposes and as information for further analysis of the photos. Light condition can be derived from aperture and exposure time. For our component we employ the method described in [11] to calculate an exposure value from the given values in the EXIF header for aperture (F_n) and exposure time (E_t): $E_v = 2 * \log_2(F_n) - \log_2(E_t)$. The value E_v should be proportional to the brightness in the scene and therefore is a good indicator for the light condition.

Context-enhanced multimodal enhancement. These components utilize both context-based and content-based features for metadata enhancement. Here we present a component for classifying photos as in- or outdoor shots. *In-/Outdoor classification* is provided by a simple yet powerful method that relies on metadata extracted from content and context: the light conditions, daytime, if a flash was fired, and the exposure rating. The first two metadata entries are

generated from context information and the last from the image content. For our in-/outdoor classification we apply a decision tree to the photos. This tree consists of rules that evaluate the before mentioned metadata. If, e. g., the picture is very dark, the flash was fired and it was taken at daytime there is a high probability that it is an indoor shot. A similar method has recently been proposed in [4]. Here Support Vector Machines are used to classify photos as indoor or outdoor. Unlike our approach only context information is used to classify the photos. In contrast, we aim at combining context and content information to achieve a higher precision. In a first step, we evaluated our approach with 437 consumer photos from which 68 are indoor and 369 are outdoor shots. This test set also compromises some ambiguous photos like indoor shots showing the view through a window. The component misclassified 0.5% of the outdoor shots as indoor and 10.1% of the indoor shots as outdoor. 10.5% of all pictures were classified as ambigous. These results are very promising, however, we are aiming at improving the classification accuracy by additionaly taking the EXIF header's object distance information into account.

5 A Concrete Photo Book Application

Maybe one remembers having spent hours to create a nice photo album from a large set of of digital photos that carefully captures the impressions of a vacation or celebration: order prints, select the best pictures, sort and organize the photos along events and experiences, glue the photos into the album and label them – a tedious and time consuming task. In this section, we present the use and exploitation of the extracted and enhanced metadata within the personal photo book creation software of our project partner CeWe Color. An authoring wizard uses the extracted and enhanced metadata for the automatic "best-of" preselection of photos and layout of the photo book pages. The commercial software integrates the MetaXa results and the developed extraction and enhancement components.

The following Figures illustrate phases from photo book creation. Figure 3(a) shows an example of the different types of printed photo books the customer can chose from with CeWe Color's software. For an actual album creation, a user selects the photo collection from which the photo book should be created. Then the actual metadata extraction and enhancement of MetaXa takes place. This metadata is used for both a pre-selection of photos and their composition into a photo book. In a preferences dialog the users can influence this process by indicating which parameters should be taken into account for a "best-of" selection of photos by the software, e. g., sharpness, exposure, and similarity. Figure 3(b) shows a photo book from a trip to Australia. Here, many similar photos especially of the Ayers Rock have been taken. Based on sharpness and similarity analysis, six photos are automatically chosen by the software for the photo book. These photos are shown in the center of Figure 3(b) and are in addition indicated by check marks on the left side. Figure 3(c) shows the results of an automatic selection of backgrounds. This is done, e.g., by taking the color



(a) page of a photo book



(b) preselected photos



(c) background selection

Fig. 3. Screenshots of the photo book application

histograms of the photos on a page into account. Having created a first version of the photobook, the users can still manually add, remove, resize, and rotate any photos and alter the backgrounds.

The goal is to make the photo book creation an intuitive and easy task. The software[2] has been released in June 2006 and has been presented at the Photokina 2006 international trade fair in Cologne. Based on an evaluation of photo book orders, usability studies, and feedback from end users we will work on a refinement of the heuristics for pre-selection of photos and further extraction and enhancement components for MetaXa.

6 Conclusion

With the proposed MetaXa architecture, we presented an approach for the systematic integration of content-based and context-based metadata extraction and enhancement methods for digital photo collections. This architecture easily allows for instantiation of a specific setup of annotation methods to meet the requirements of a specific domain. It also allows to easily reuse components in different domains.

[2] `http://www.cewe-photobook.com/`

We employ and advance the state-of-the-art in semantic understanding of digital media in the domain of personal media. Our enhancement components exploit knowledge from content, context, and domain knowledge to provide for a better semantic understanding of photo collections and lay the grounds for next generation digital photo services.

The results of our MetaXa approach are evaluated by integrating them into the photo book software provided by CeWe Color. For it, we are evaluating and improving our extraction and enhancement components on large test sets provided by our photo finishing partner. We are also developing new extraction and enhancement components such as automatic orientation detection and location clustering.

On the application level, we are currently developing heuristics and probabilistic approaches for automatically suggesting relevant photos for a personal photo book. In addition, we are also working on dynamically determining a content- and context-based layout of the selected images in a photo book. Although the MetaXa architecture presented in this paper is applied in the domain of digital photo book authoring, it is not limited to this kind applications.

References

1. ACDSee Systems Int., Inc. acdsee Pro, 2006. http://www.acdsystems.com.
2. Apple Inc. iPhoto, 2006. http://www.apple.com/de/ilife/iphoto/.
3. Susanne Boll. Image and video retrieval from a user-centered mobile multimedia perspective. In *Proc. of the International Conference on Image and Video Retrieval (CIVR2005)*. Springer, 2005.
4. M. R. Boutell and J. Luo. Beyond pixels: Exploiting camera metadata for photo classification. *Pattern Recognition*, 38(6), 2005.
5. M. Cooper, J. Foote, A. Girgensohn, and L. Wilcox. Temporal event clustering for digital photo collections. *ACM Trans. Multim. Comp. Comm. Appl.*, 1(3), 2005.
6. GfK Group for CeWe Color. Usage behavior digital photography, 2006.
7. Google, Inc. Picasa, 2006. http://picasa.google.com/.
8. A. Graham, H. Garcia-Molina, A. Paepcke, and T. Winograd. Time as essence for photo browsing through personal digital libraries. In *Proc. of the 2nd ACM/IEEE-CS Joint Conf. on Digital Libraries*. ACM Press, 2002.
9. International Radio Consultative Committee (ITU). Encoding parameters of digital television for studios. Technical Report 601, CCIR, 1982.
10. Bernd Jähne. *Digital Image Processing*. Springer, 6th edition, 2006.
11. JEITA. Exif version 2.2. Technical report, April 2002.
12. Lahti, Westermann, Palola, Peltola, and Vildjiounaite. Context-aware mobile capture and sharing of video clips. In *Handbook of Research on Mobile Multimedia*. Idea Publishing, 2006.
13. Jiebo Luo, M. Boutell, and C. Brown. Pictures are not taken in a vacuum. *IEEE Signal Processing Magazine*, 23(2), 2006.
14. J. Martinet, Y. Chiaramella, and P. Mulhem. A model for weighting image objects in home photographs. In *Proc. of the 14th ACM Int. Conf. on Information and knowledge management*. ACM Press, 2005.
15. P. Mulhem and J.-H. Lim. Home photo retrieval: Time matters. In *Proc. of the 2nd Int. Conf. on Image and Video Retrieval*. Springer, 2003.

16. M. Naaman, S. Harada, Q.-Y. Wang, and A. Paepcke. Adventures in space and time: Browsing personal collections of geo-referenced digital photographs. Technical report, Standford University, InfoLab, 2004.
17. M. Naaman, Y. J. Song, A. Paepcke, and H. Garcia-Molina. Automatic organization for digital photographs with geographic coordinates. In *Proc. of the 4th ACM/IEEE-CS Joint Conf. on Digital Libraries*. ACM Press, 2004.
18. M. Naaman, R. B. Yeh, H. Garcia-Molina, and A. Paepcke. Leveraging context to resolve identity in photo albums. In *Proc. of the 5th ACM/IEEE-CS joint Conf. on Digital Libraries*. ACM Press, 2005.
19. W. K. L. P. Mulhem, J. H. Lim and M. Kankanhalli. *Advances in Digital Home Image Albums*, chapter 9. Idea Publishing, 2003.
20. A. Pigeau and M. Gelgon. Building and tracking hierarchical geographical & temporal partitions for image collection management on mobile devices. In *Proc. of the 13th annual ACM Int. Conf. on Multimedia*. ACM Press, 2005.
21. J. C. Platt. Autoalbum: Clustering digital photographs using probabilistic model merging. In *Proc. of the IEEE Workshop on Content-based Access of Image and Video Libraries*. IEEE Computer Society, 2000.
22. J. C. Platt, M. Czerwinski, and B. A. Field. Phototoc: Automatic clustering for browsing personal photographs. Technical report, Microsoft Research, 2002.
23. K. Rodden and K. R. Wood. How do people manage their digital photographs? In *Proc. of the SIGCHI Conf. on Human factors in comp. systems*. ACM, 2003.
24. Ansgar Scherp and Susanne Boll. Context-driven smart authoring of multimedia content with xsmart. In *Proc. of the 13th annual ACM Int. Conf. on Multimedia*, pages 802–803, New York, NY, USA, 2005. ACM Press.
25. A. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. Content-based image retrieval at the end of the early years. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(12), 2000.
26. C. Szyperski, D. Gruntz, and S. Murer. *Component Software: Beyond Object-Oriented Programming*. Addison Wesley, 2nd edition, 2002.
27. T. Tan, J. Chen, P. Mulhem, and M. Kankanhalli. Smartalbum: a multi-modal photo annotation system. In *Proc. of the 10th ACM Int. Conf. on Multimedia*. ACM Press, 2002.
28. P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 2001.
29. L. Wenyin, Y. Sun, and H. Zhang. Mialbum - a system for home photo management using the semi-automatic image annotation approach. In *Proc. of the 8th ACM Intl. Conf. on Multimedia*. ACM Press, 2000.
30. L. Zhang, L. Chen, M. Li, and H. Zhang. Automated annotation of human faces in family albums. In *Proc. of the 11th ACM Int. Conf. on Multimedia*. ACM, 2003.

Context-Sensitive Ranking for Effective Image Retrieval*

Guang-Ho Cha

Department of Computer Engineering, Seoul National University of Technology
Seoul 139-743, South Korea
ghcha@snut.ac.kr

Abstract. Over many years, almost all research work in the content-based image retrieval (CBIR) has used Minkowski metric (or L_p-norm) to measure similarity between images. However, those functions cannot adequately capture the nonlinear relationships in contextual information given by image datasets. In this paper, we present a new similarity measure reflecting the nonlinearity of contextual information. Moreover, we propose a new similarity ranking algorithm based on this similarity measure for effective CBIR. Our algorithm yields superior experimental results on real image database and demonstrates its effectiveness.

1 Introduction and Related Work

The central problems in CBIR are concerned with interpreting the contents of images in a collection and ranking them according to the degree of relevance to the query. Knowing how to extract this information is not the only difficulty; another is knowing how to use it to decide *relevance*. The decision of relevance characterizing user need is a complex problem. Many researchers have proposed the use of *relevance feedback* to improve the retrieval effectiveness [3, 5, 6, 8, 11]. Although relevance feedback is an approach to improve the retrieval effectiveness, its power is still restricted by the similarity measure and the ranking algorithm employed by CBIR system. Another problem with relevance feedback is the *multi-round* feedback that is usually time-consuming and it requires users to have patience.

In this paper, we attempt to address the above problems by capturing the *nonlinear* relationships in contextual information given by image datasets. It has been widely acknowledged in CBIR that a query concept is typically a nonlinear combination of perceptual features (color, shape, texture, etc.) [12].

Recently, Wu et al. [12] proposed a method for formulating a context-based distance function for measuring similarity. It uses the *kernel function* [10] to nonlinearly transform traditional distances into a similarity in a feature space.

DynDex [1] proposed a non-metric distance function, dynamic partial function (DPF), to measure perceptual similarity and proved its closer match with the perceptual similarity than Minkowski metric. However, it is difficult to dynamically select features to be used for distance computation.

* This work was supported by grant No. B1220-0501-0233 from the University Fundamental Research Program of the Ministry of Information & Communication in Republic of Korea.

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 344 – 353, 2007.
© Springer-Verlag Berlin Heidelberg 2007

In this paper, similarly to [12], we first conduct the nonlinear transformation on the original dataset not only to capture nonlinear relationships but also to simulate human perception. However, unlike [12], we do not require human intervention to collect the contextual information, while [12] needs the contextual information in the form of *training data*.

For nonlinear transformation, we adopt a *Gaussian* function because it possesses an excellent nonlinear approximation capability [2, 7]. However, we may also learn the kernel function from the training data as in [12] instead of choosing the Gaussian function in advance. Compared to the Minkowski metric, our approach offers a more accurate modeling of the notion of similarity in a context-sensitive CBIR.

The main contributions of our work are two-fold: (1) we present a new similarity measure that is based on the nonlinear similarity model and that exploits the contextual information in a dataset; (2) we provide two new similarity ranking algorithms based on the developed similarity measure.

2 Nonlinear Similarity Model

2.1 Motivating Examples

Example 1. The user wants to select two images via query-by-example in the handwritten digit image database. Fig. 1(a) is a query image and Figs. 1(b) and 1(c) may be the query result if a human selects two images, and those are the actual result from our similarity search experiment. When we use Euclidean distance measure, on the other hand, Figs. 2(b) and 2(c) are the actual result of the traditional similarity search experiment. This means that there may exist a discrepancy between human perception and the metrics such as Minkowski metric. Therefore, in CBIR, it is necessary to establish the link that bridges the gap between human perception and distance calculation.

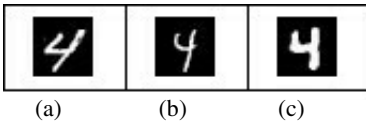


Fig. 1. Human perception based retrieval: (a) is a query image; (b) and (c) are two images retrieved from similarity search

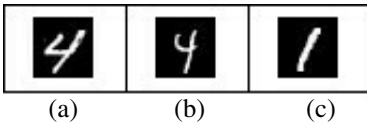
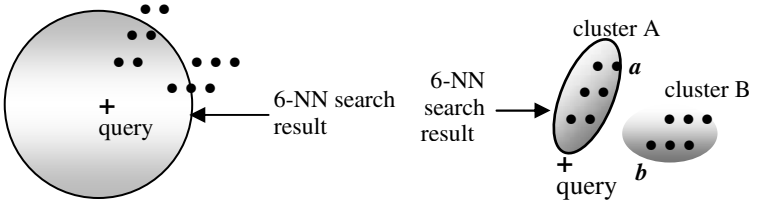


Fig. 2. Euclidean distance based retrieval: (a) is a query image; (b) and (c) are the similarity search result

Example 2. Fig. 3 shows another example to explain our motivation. Assume that we are given a set of points constructed with two clusters. A query point is represented by + and we want six points nearest to the query point +. If we search six nearest neighbors (NNs) to the query by pairwise Euclidean distance, the six NNs resulting from the search are the points within the circle whose center is the query point + (see Fig. 3(a)). However, as described in Example 1, when we consider the distribution of the given dataset, as shown in Fig. 3(b), the six points in cluster A may be more relevant to the query point than the points in cluster B even though some of them have longer Euclidean distance than some points in cluster B.



(a) *k*-NN search by Euclidean distance (b) *k*-NN search considering data distribution

Fig. 3. *k*-NN search results based on two different models

From these motivating examples, we can assume that in image similarity ranking it may be desirable that closer points have more similar ranks than the points far away even though they are behind with respect to Euclidean distance based ranking. If we base similarity ranking on this concept, in Fig. 3(b), the right-most point *a* in cluster A should be ranked to be more relevant to the query point than the point *b* in the left-most in cluster B.

Based on this concept, in CBIR domain, we define the *contextual information* as the information about the distribution and cluster structure of a given dataset.

2.2 Nonlinear Similarity Model

In order to capture the contextual information as well as to simulate human perception for similarity evaluation between images, we first establish a *nonlinear* model. The assumption for the nonlinear approach is that the same lengths of distances do not always give the same degrees of similarity when judged by humans [9]. We adopt a *Gaussian* function as our basic similarity model:

$$G(x_i, x_j) = \exp(-d(x_i, x_j)/\sigma^2) \tag{1}$$

The activity of function G is to perform a Gaussian transformation of the distance $d(x_i, x_j)$, which describes the degree of similarity between x_i and x_j. The scaling parameter σ^2 controls the smoothness of the distance between x_i and x_j and it is specified by a user. The Gaussian function creates a new space called the *feature space* that is a nonlinear transformation of the input space containing the original data. Throughout our work, we conduct the similarity comparison in the induced feature space.

3 Context-Sensitive Similarity Ranking Algorithms

3.1 A Simple Similarity Ranking Algorithm

Our first ranking algorithm performs the similarity comparison in a feature space in which the relationship between data points is more "obvious" as seen in Fig. 4. In order to obtain this effect, we extract the eigenvectors from a similarity matrix made by pair-wise similarity between images.

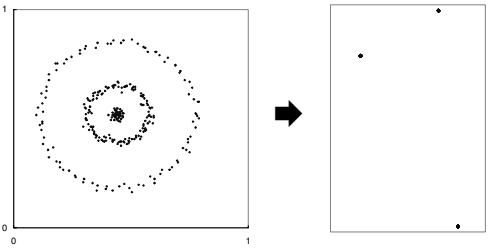


Fig. 4. An actual example of our transformation: it reveals the cluster structure of a dataset

The ranking algorithm is as follows. Given a set of data points $X = \{x_1, \ldots, x_m\} \in R^n$, we transform the data to the points on the eigenvector feature space using the method employed in the spectral clustering [4].

1. Construct a similarity matrix $K \in R^{m \times m}$ defined by
   $K_{ij} = \exp(-d(x_i, x_j)/\sigma^2)$ if $i \neq j$, and $K_{ii} = 0$
2. Construct a diagonal matrix D whose (i, i)-element is the sum of K's i-th row.
3. Form a normalized similarity matrix $K' = D^{-1/2} K D^{-1/2}$.
4. Find $v_1, v_2, \ldots, v_k$, the k largest eigenvectors of K', and form the matrix $V = [v_1\ v_2\ \ldots\ v_k] \in R^{n \times k}$ by stacking the eigenvectors in columns.
5. Form the matrix Z from V by renormalizing each of V's rows to have unit length (i.e., $Z_{ij} = V_{ij} / (\Sigma_j V_{ij}^2)^{1/2}$).
6. Now each row of Z is a point in a feature space R^k.
7. Compute image ranks using the Euclidean distance in this feature space R^k.

In step 1, we construct the matrix K composed of object-object similarities, i.e., K_{ij}, $i \neq j$, gives a similarity value between two points x_i and x_j. K_{ii} is zero to avoid reinforcement of self-similarity value. Note that since the similarity matrix K and the final dataset Z is pre-computed before the search, it is not a burden during the search. Steps 2–3 and 5 provides a suitable normalization necessary for our similarity ranking.

Example 3. To illustrate the effect of this ranking algorithm, let us consider a toy dataset containing 100 data points shown in Fig. 5. The dataset has 3 cluster structures. Every point should be similar to the points in its neighborhood, and furthermore, points in one cluster should be more similar to each other than to points in the

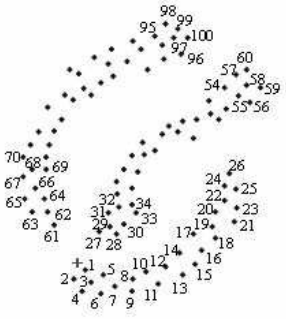


Fig. 5. Ranking on a query point + over 100 data points

other cluster. The query point is + and the number beside each point denotes its rank. As shown in Fig. 5, the ranking algorithm exploits the contextual information of the dataset.

This ranking algorithm works very well in low dimensional spaces, however, it has limited success on real high-dimensional image dataset in our experiments. Therefore, we propose another ranking algorithm for high-dimensional datasets based on similarity distribution and a new similarity measure.

3.2 A Similarity Ranking Algorithm Based on Similarity Distribution

In order to consider the intrinsic structure revealed by the dataset we introduce the concept of *similarity distribution*. Assume a set of points $X = \{x_1, x_2, \ldots, x_m\} \in R^n$ that we would like to rank. Let x_q, $q \notin \{1, 2, \ldots, m\}$, be the query point. We define a vector $s_i = [s_{iq}, s_{i1}, s_{i2}, \ldots, s_{im}]^T$, where s_{ij} is the similarity value between two objects x_i and x_j. The similarity value s_{ij} is computed by Eq. (1), i.e., $s_{ij} = \exp(-d(x_i, x_j) / \sigma^2)$. We consider the vector s_i as the *distribution of similarities* between the point x_i and all other points in a dataset including the query point. The vector $s_q = [s_{qq}, s_{q1}, s_{q2}, \ldots, s_{qm}]^T$ represents the distribution of similarities between the query point x_q and all other points including the query point itself. According to Eq. (1), s_{qq} is defined to be 1.0.

We define the *similarity value* of a point x_i to the query point x_q by the *dot product* of the similarity distribution for x_i and that for x_q in Gaussian feature space. The similarity value s_{iq} of point x_i to the query point x_q is computed by

$$s_{iq} = s_i^T \cdot s_q = s_{iq} s_{qq} + \sum_{j=1}^{m} s_{ij} \cdot s_{qj} = s_{iq} + \sum_{j=1}^{m} s_{ij} \cdot s_{qj} \qquad (2)$$

In the above Eq. (2), s_i and s_q are the similarity distribution vectors for points x_i and x_q, respectively, and the similarity values s_{ij} and s_{qj} are computed by Eq. (1). The similarity measure given by Eq. (2) denotes the actual similarity value between the query point and point x_i plus the linear combination of the similarity values between point x_i and its neighbors, weighted by its neighbors' similarity values to the query point. Therefore, the similarity value of a point affects its neighbors' similarity values, and if two points are close, they are more influenced by each other because their

respective similarity values to query point are weighted by the similarity value between two points. With this similarity metric based on the similarity distribution, the points clustered near the query point are favored in similarity ranking.

Our ranking algorithm is as follows:

[Input] A set of points $X = \{x_1, \ldots, x_q, x_{q+1}, \ldots, x_m\} \in R^n$, where $x_1, \ldots, x_q$ are query points and the rest $x_{q+1}, \ldots, x_m$ are the data points we would like to rank

[Output] The ranked list of data points

1. Construct a similarity matrix $K \in R^{m \times m}$ defined by
$$K_{ij} = \exp(-d(x_i, x_j)/\sigma^2) \text{ if } i \neq j, \text{ and } K_{ii} = 0$$
2. Construct a diagonal matrix D whose (i, i)-element is the sum of K's i-th row.
3. Form a normalized similarity matrix $K' = D^{-1/2} K D^{-1/2}$.
4. Create the initial similarity values s_{ij} between a point x_i and the query point x_j, $1 \leq j \leq q$, $q+1 \leq i \leq m$.
$$s_{ij} = \begin{cases} 1 & \text{for } 1 \leq i \leq q, \text{ i.e., both } x_i \text{ and } x_j \text{ are query points.} \\ \exp(-d(x_i, x_j)/\sigma^2) & \text{for } q+1 \leq i \leq m. \end{cases}$$
5. **for** $i = q+1$ to m **do** // for each data point
6. **for** $j = 1$ to q **do** // for each query point
7. $$s_{ij} = K'_{ij} + \sum_{k=q+1}^{m} K'_{ik} s_{kj}$$
8. **end for**
9. **end for**
10. Compute the similarity score s_i of x_i to q query points by $s_i = \max_{1 \leq j \leq q} \{s_{ij}\}$.
11. Sort the set $S = \{s_{q+1}, s_{q+2}, \ldots, s_m\}$ in non-increasing order and return the top k points as the result.

Example 4. To illustrate the effect of our algorithm, we consider a toy dataset. We use a dataset containing 74 2-dimensional points shown in Fig. 6. The dataset has two cluster structures and two query points are given by + and ×. The number beside each point denotes the rank assigned to that point. As shown in Fig. 6, it is demonstrated that our ranking algorithm exploits the global cluster structure of the dataset. We believe that for many real world applications including image searches this kind of retrieval based on the global cluster structure is superior to the local methods that rank data by pairwise Minkowski distance metric.

4 Experiments

For experimental evaluation of our methods, we use the MNIST database that contains 28 × 28 120,000 handwritten digit images. The MNIST database is the currently used classifier benchmark in the AT&T and Bell Labs, and many methods have been tested with this database. The feature of each image is represented by a 784-dimensional vector. In our experiments, we use only the first 6,000 images from the MNIST database and perform a similarity search to return the k most similar images for the given query images.

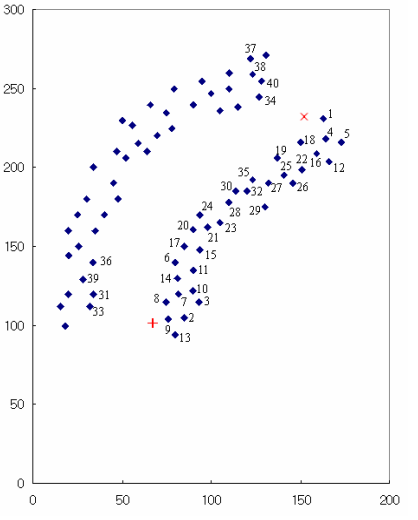


Fig. 6. Ranking on two query points × and +

To obtain an objective measure of performance, we assume that a query concept is an image category, i.e., one of the labels '0', '1', …, '9' given to each digit category.

We evaluate *precision* performance for *k*-NN queries, where *k* is 10 – 100, and precision is computed by the fraction of the returned *k* images that belong to the query image category.

We perform 100 *k*-NN queries and average their performance. The query images are randomly selected from the MNIST database. In order to provide the intuition for our method, we show the *k*-NN search results in Figs. 7 – 10.

Figs. 7 and 8 are the results using single query image. The top-left image is the query image. Note that there are many digits other than '9' in Euclidean distance based ranking in Fig. 8. Figs. 9 and 10 show the results when two images are used as a query. The top-left two images are query images. The first query uses as the query images the two similar images with digits '4'. The second query uses as query images the very different two image for digits '0' and '6'. For multi-point (or disjunctive) queries, we use the aggregate dissimilarity measure of Falcon [11] with the constant $\alpha = -3$. As shown in Figs. 9 and 10, there are many digits other than the query images when we use Falcon's aggregate dissimilarity measure. On the other hand, our method generates the uniform result. This experimental result provides indirect proof of superiority of our method.

Fig. 11 compares the precision performance for *k*-NN queries among our ranking algorithm, Euclidean distance based method, and the SVM_{Active} method [8]. In [8], it is stated that SVM_{Active} outperforms three query refinement methods: (1) query reweighting methods such as MARS [6] (2) the query point movement methods such as MARS [5], MindReader [3], (3) the query expansion methods such as Falcon [11]. Therefore, we compare our method with SVM_{Active}. SVM_{Active} is a relevance feedback

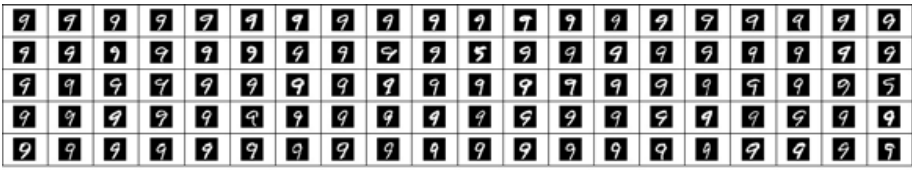


Fig. 7. Top 100 images by our similarity ranking, where the top-left image is the query image

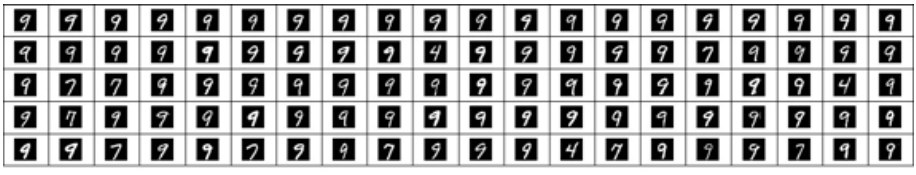


Fig. 8. Top 100 images by Euclidean distance based ranking

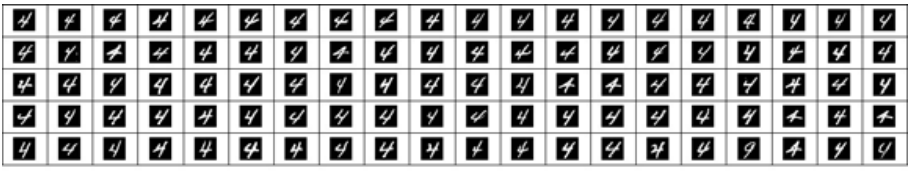


Fig. 9. Top 100 images by our similarity ranking, where the top-left 2 images are the query images

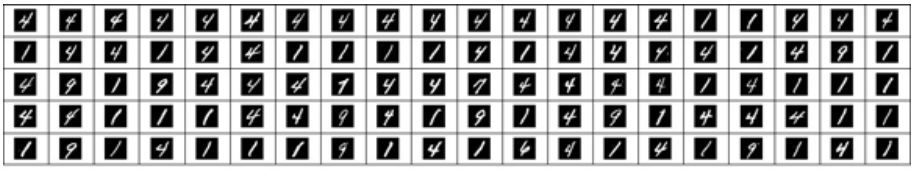
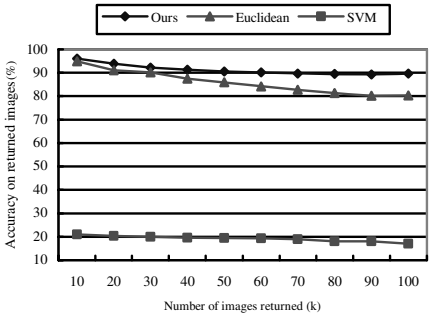


Fig. 10. Top 100 images computed by Falcon's aggregate similarity metric

method based on active learning with support vector machines (SVM) [10]. It retrieves top-*k* images after a few relevance feedback rounds. In each round of relevance feedback, SVM_{Active} determines the images as "relevant" if they have the same label as the query image's. In the experiment of SVM_{Active}, we conduct four relevance feedback rounds and use 100 training images per round. Fig. 11 shows the average top-*k* precision for three different methods. SVM_{Active} shows the worst performance. The poor performance of SVM_{Active} is caused by the size and complexity of the 784-dimensional MNIST database. Our method achieves at least 90% precision on the top-*k* results, whereas the Euclidean distance based method cannot achieve our performance.



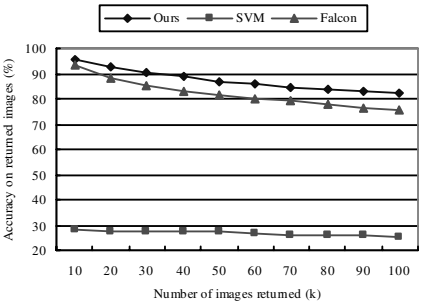


Fig. 11. Single-point queries: average top-*k* precision **Fig. 12.** Multi-point queries: average top-*k* precision

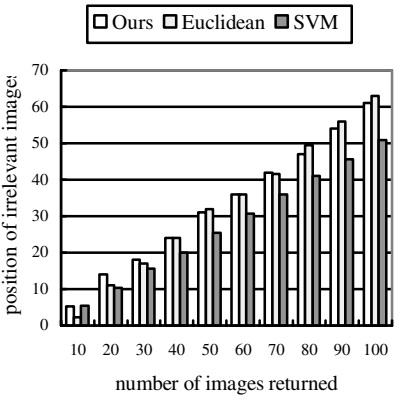


Fig. 13. Positions of irrelevant ones in top-*k* images

Fig. 12 shows the precision result for multi-point *k*-NN searches. Our method achieves over than 80% precision in any cases, whereas SVM_{Active} and Falcon cannot achieve this performance.

Fig. 13 shows the average positions of irrelevant ones in top-*k* images returned. This position indicates where the irrelevant images appear in top-*k* results. It is desirable that the irrelevant images are found in rear positions. In the case of our method, the positions of irrelevant images found are far later compared with other methods when the number of images returned is small, i.e., small *k*. This is a desirable result because users usually do not want to have a large number of images returned.

5 Conclusions

We have presented a new similarity measure and two context-sensitive ranking algorithms based on a nonlinear similarity model for effective image retrieval. This similarity measure and the ranking algorithms consider the intrinsic structure and the distribution revealed by the dataset. Our CBIR scheme has demonstrated its

effectiveness and outperformed the existing image retrieval methods such as SVM_{Active}, Falcon, and Euclidean distance based method. Our scheme takes advantage of the intuition that the same portions of the distances given by Minkowski metric do not always give the same degrees of similarity when judged by humans.

References

1. K.-S. Goh, B. Li, and E. Chang, "DynDex: A Dynamic and Non-metric Space Indexer," *Proc. ACM Multimedia*, 2002, 466-475.
2. S. Haykin, *Neural Networks: A Comprehensive Foundation*, Maxmillan, 1994.
3. Y. Ishikawa, R. Subramanya and C. Faloutsos, "MindReader: Querying databases through multiple examples, *Proc. VLDB Conf*., 1998, 218-227.
4. A.Y. Ng, M.I. Jordan, and Y. Weiss, "On Spectral Clustering: Analysis and Algorithm," *Advances in Neural Information Processing Systems*, 14, 2002.
5. Y. Rui et al., "Relevance feedback: A Power tool for interactive content-based image retrieval," *IEEE Tr. Circuits and Video Technology*, 8(5), 1998, 644-644.
6. Y. Rui, T. Huang, and S. Mehrotra, "Content-based image retrieval with relevance feedback in MARS," *Proc. Int'l Conf. on Image Processing*, 1997.
7. B. Schölkopf et al., "Comparing Support Vector Machines with Gaussian Kernels to Radial Basis Function Classifiers," *IEEE Trans. on Signal Processing*, 45, 2758-2765, 1997.
8. S. Tong and E. Chang, "Support Vector Machine Active Learning for Image Retrieval," *Proc. ACM Multimedia Conf*., 2001, 107-118.
9. R.L.De Valois and K.K.De Valois, *Spatial Vision*, Oxford Science Publications, 1988.
10. V.N. Vapnik, *Statistical Learning Theory*, Wiley, NY, 1998
11. L. Wu, C. Faloutsos, K. Sycara and T.R. Payne, "FALCON: Feedback Adaptive Loop for Content-Based Retrieval, *Proc. of VLDB Conf*., pp. 297-306, 2000.
12. G. Wu, E.Y. Chang, and N Panda, "Formulating Context-dependent Similarity Functions," *Proc. ACM Multimedia*, pp. 725-734, 2005.

Visual Verification of Historical Chinese Calligraphy Works

Xiafen Zhang and Yueting Zhuang

The Institute of Artificial Intelligence, Zhejiang University,
Hangzhou, 310027, P.R.China
{cadal,yzhuang}@cs.zju.edu.cn

Abstract. The problem of historical Chinese calligraphy verification is previously investigated by experienced artists, whereas this paper proposes some objective measures to bear the problem with evidences by analyzing the subtle discrepancies between the images of the suspicious and the genuine. First, features that characterize an individual calligrapher's writing style are extracted and modeled. When a suspicious comes, it is compared with the genuine in the reference database to detect problematic characters and to calculate total accepting probability. The efficiency of the algorithm is demonstrated by a preliminary experiment with 13274 images of calligraphy character.

1 Introduction and Motivation

Historical Chinese calligraphy works are valuable. So the forgeries are introduced into the trade and presented along with the genuine, which makes verification a must in order to distinguish the genuine from the forgeries. This verification problem has long been considered as a problem belonging to the field of the art and the investigation is mostly subjective, whereas we may bring some objective measures from the field of image processing and analyzing to push the limits of the advancement of calligraphy verification. In the field of calligraphy art, artists are successful in recognizing calligraphy characters, identifying calligraphy styles, and classifying the medium on which calligraphy exists. But limited by the human brain, an artist can only remember and be expert on a few masters' works. Artists express their analysis subjectively in a way of what they feel, for example the reason they give for identifying a suspicious as a forgery may be "These characters are fainthearted, and they don't look like the genuine". Such reason is ambiguous and lack of convincible evidences (see [1] and [2]).

This paper propose a way to verify historical Chinese calligraphy like a doctor trying to find out whether a strange patient is really in sick and what's wrong with that patient: The doctor analyzes the checking reports of the patient's each organ. Inspired by this idea, this paper designs different measurements to examine whether each part of a suspicious works conform to those of the genuine. Yueting Zhuang et al. [3] introduced an approach to measure how similar two calligraphy-character shapes are. But for calligraphy verification, the forger is intent to deceive the system by copying the shapes of those genuine works.

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 354–363, 2007.
© Springer-Verlag Berlin Heidelberg 2007

Therefore, instead of finding out how similar two characters are, we need to find out how dissimilar two calligraphy characters are. It is possible because a calligrapher often practice the writing skills and has personal preferences, which is just like "the way in which they do their own business". When forging someone else's, the inconstant writing style will be shown on this or that point and can be detected.

2 Related Works

Many researches on multimedia have been done on the analysis and retrieval of media objects including images, audios and videos. But few researches have been done on artistic images. No published scientific paper has been researched on the problem of Chinese calligraphy verification, mainly because previously it has long been regarded as a problem belongs to the art field. It seems that writer verification may share many same problems and solutions with Chinese calligraphy verification. But actually they are different. The data on which writer verification analyzes is the works written in one's own style, while the data calligraphy verification analyze is intent to hide one's own writing style. The most related works are off-line signature verification, such as [4], [5], [6] and [7]. Both of them contain characters that intend to hide the original writing styles, and have no record of pen trajectory or dynamic pressure. But features used in [4], [5] and [6] have no discriminative power for calligraphy works. Cheng-Lin Liu et al. [7] introduce features of different moments for Chinese writer identification. Parts of the moments do have discriminative power, but the moments alone don't have enough power to verify calligraphy.

3 System Architecture

Key problem for visual verification of calligraphy is to model writer's particularities by learning from the genuine collections. Hence a genuine reference database needs to be built first. When a suspicious is input, it is divided into meaningful parts to compare to those of genuine in the reference database. Fig.1 shows the architecture. Key steps are measurements designing and genuine reference database construction.

The genuine reference database consists of 3 parts: Raw data, Feature data and the map between them. First, page images are segmented into individual characters by employing the approach introduced in [3], and then record the information of individual character's location, page's location, and the writer. This information is organized as maps between the raw image data and its features data. The measurement is designed to measure the different between the suspicious and the genuine is. This paper proposes measurements in two levels: character-based and stroke-based.

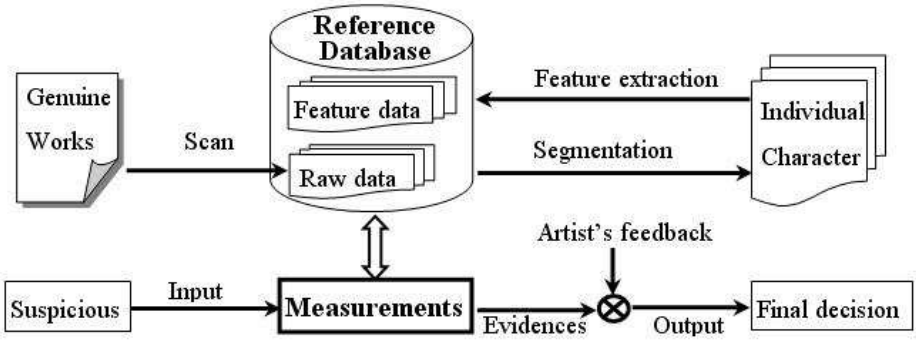


Fig. 1. Architecture of historical Chinese calligraphy works verification

4 Computing Stroke Shape Features

4.1 Jitteriness

An evidence of forgery is jitteriness, which is defined as the situation that when attempting to simulate a genuine curved stroke, one often hesitate and trying to correct the brush trajectory in order to conform to the priori segment. Thus it has more jitteriness when compared with those of genuine. Fig.2 shows an example of the jitteriness of a forged stroke at the turn.



Fig. 2. Example of jitteriness. Left: a forged one with part of the jitteriness zoomed in. Right: a genuine one with the corresponding part zoomed in.

The jitteriness can be measured using fractal dimension measurement, which is first introduced by French mathematician Mandelbrot [8].When a jittered curve is zoomed out with a suitable zoom-ratio, the small jittered part,namely the wrinkly parts, will disappear, which result in shorter curve length. Therefore, a measurement of jitteriness can be written as:

$$Jitterness = \frac{length_{original}}{length_{zoomout}} \tag{1}$$

where $length_{original}$ and $lengh_{zoomout}$ are the length before and after zooming out respectively, $zoomout = 0.5$ is the zoom ratio.

4.2 Thickness Variation

Calligraphy is a special kind of handwriting written by soft brush. The harder the brush is pushed, the thicker the stroke will be. Therefore, to some extent the thickness variation reflects the pen pressure variation. A forger may copy the general shape of a character, but it is difficult to mimic the detail pen pressure variation. The thickness of a skeleton point is defined as the radius of the maximum circle that centered on the skeleton point and covered by the foreground. Let w_i be the thickness of a skeleton point p_i, then the thickness variation tv can be written as:

$$tv = \frac{1}{n}\sum_{i=1}^{n}(w_i - \overline{w})^2, \qquad \overline{w} = \sum_{i=1}^{n} w_i \tag{2}$$

4.3 Curvilinear Style

When writing a curve stroke, a forger constantly analyzes in the mind about brush trajectory and makes corrects in order to conform to the next curve segment. Thus show hesitancy and make subtle differences that are inconsistent with the genuine writing style. Fig.3(b) shows an example written by *changshuo Wu*, who had written many different characters that contain a u shape stroke in different length, different width, different height, or the mouth of the u may in different size. But the curve has only one peak. This feature can not be measured by *Dynamic Programming Matching*(see [9]) or *Hidden MarKov Models*(see [10]). So we propose a measure by finding out how many relative peak-turns in a stroke. A peak-turn is defined as a curve segment that goes in one direction before the peak and then turns to go in an opposite direction.

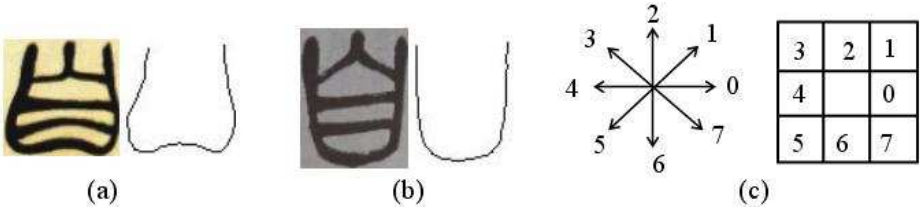


Fig. 3. Example of curvilinear style (a) Character written by an unknown and a u shape stroke skeleton. (b) Character written by Changshuo Wu and the u shape stroke skeleton. (c) Orientation code and the corresponding map for a pixel's 8-neighborhood.

In terms of describing how a curve goes and changes its direction, chain-code is a good representation as shown in Fig.3(c).Let x be a code number, if $x < 4$, then the curve goes in up direction, else if $x > 4$, then the curve goes in down direction. Therefore, it can be seen that there are three peak-turns in the curve of Fig.3(a). But for the stroke in Fig.3 (b), there's only one peak-turn.

Let lb_i and la_i be the length of curve segments that before and after the peak point p_i respectively, let t be the number of total peak-turn, then the measure of relative curvilinear rt can be defined as:

$$rt = 1 + \sum_{i=1}^{t} \frac{t \times li + dif_i}{lengthofcurve}, \qquad l_i = min\{lb_i, la_i\}, \qquad dif_i = |la_i - lb_i| \quad (3)$$

The more times a curve turns, the bigger the value of rt will be.

5 Computing Character Shape Features

Some calligrapher tends to write characters that the height is a little bigger than the width, and the measure can simply be written as:

$$ratio_{h/w} = \frac{height}{width} \quad (4)$$

Such kinds of features are considered to be features of Geometry distribution. They are no easy for human eyes to notice and tend to be overlooked, yet it is easy for computer to detect. For example, the gravity center of a calligraphy character depends on individual calligrapher's preference: It is not in the right center but is somewhere near the center.

5.1 Geometric Mass Distribution

Let M and N be the width and the height of a calligraphy character image $f(x, y)$, then a character's $(p + q)th$ order moment is defined as:

$$m_{pq} = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} x^p y^q f(x, y) \quad (5)$$

Therefore, the gravity center $(\overline{x}, \overline{y})$ of a character can be written as:

$$\overline{x} = \frac{m_{10}}{m_{00}}, \qquad \overline{y} = \frac{m_{01}}{m_{00}} \quad (6)$$

A calligraphy character center moment is defined as:

$$u_{pq} = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} (x - \bar{x})^p (y - \bar{y})^q f(x, y) \quad (7)$$

3rd order center moments have physical meanings and can characterize a calligrapher' particularities. Value of $(x - \bar{x})^3$ indicates the stress variation of a character in horizontal direction. We divide u_{30} into two segments based on the position of the center: the left part (the negative part)$u_{30}{}^-$ and the right part (the positive part)$u_{30}{}^+$. $|u_{30}{}^-| > |u_{30}{}^+|$ indicates that the calligrapher pushed the brush harder in the left part than in the right part. It is the same for the vertical direction. Therefore, stress variation in horizontal s_h and in vertical s_v can be designed as:

$$s_h = \frac{{u_{30}}^+}{{u_{30}}^+ + {u_{30}}^-}, \qquad s_v = \frac{{u_{03}}^+}{{u_{03}}^+ + {u_{03}}^-} \tag{8}$$

Value of $(x-\bar{x})(y-\bar{y})^2$ indicates the extension in the vertical direction. In the same way, it is divided into ${u_{21}}^+$ and ${u_{21}}^-$. ${u_{21}}^- < {u_{21}}^+$ indicates that that the calligrapher slanted more in the bottom part than in the top part. Therefore, the balance slant in horizontal b_h and in vertical b_v can be written as:

$$b_h = \frac{{u_{21}}^+}{{u_{21}}^+ + {u_{21}}^-}, \qquad b_v = \frac{{u_{12}}^+}{{u_{12}}^+ + {u_{12}}^-} \tag{9}$$

6 Visual Verification

Key issue of Calligraphy verification is to extract and model different writing styles by different calligrapher. We organized the above 10 features as a feature vector.

6.1 Writing Style Modeling

Not all the features can characterize an individual calligrapher's particularities, namely the writing style. A feature that characterizes one calligrapher's writing style may not characterize another calligrapher's writing style. So we employ Gaussian distribution model, as shown in Fig.4, to select suitable feature to build individual calligrapher's writing style. The selection follows the two steps:

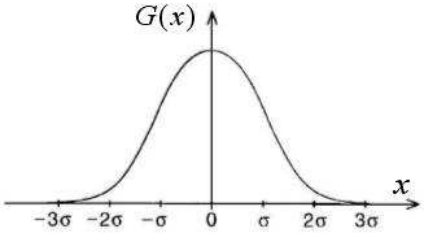


Fig. 4. Gaussian distribution model

First, for each feature f_i, extract and compute the average value u_i and the variance σ_i of all the characters written by different calligraphers. Then compute its average value $\bar{x_i}$ of characters written by an individual calligrapher. A feature is selected as characterizing feature if it satisfies the following formula:

$$|\overline{x_i} - u| > \lambda \times \sigma_i \tag{10}$$

where $\lambda = 1.5$ is a threshold. The corresponding physical meaning is: For feature f_i, if a individual calligrapher's feature value falls in the low probability area of the Gaussian distribution model, then it indicates that this feature is not a common feature for most calligraphers. It represents an individual calligrapher's particularity and so is selected as characterizing feature.

6.2 Problematic Character Detecting

After the individual calligrapher's writing style is modeled, the next step is the comparison of the suspicious and the genuine. Still Gaussian model is employed but with different data. Let $\overline{\sigma_i}$ be the variance of feature f_i for all the characters written by the same writer. A suspicious feature x_i is a forgery feature when:

$$|x_i - \bar{x_i}| > \lambda \times \overline{\sigma_i} \tag{11}$$

where $\lambda = 2.1$ is a threshold. The probability of accepting a suspicious character c as a genuine is defined as:

$$p(c) = \frac{1}{m}\sum_{i=1}^{m} p(x_i), \qquad p(x_i) = 2 \times \int_{t=x_i}^{t=\infty} \frac{1}{\sqrt{2\pi\bar{\sigma_i}}} exp(\frac{(t-\bar{x_i})^2}{2\bar{\sigma_i^2}})dt \tag{12}$$

where m is the total number of characterizing features of an individual calligrapher. If the accepting probability is less than a threshold, then the character is marked as a problematic character.

7 Experiment and Evaluation

In the experiment, we use real calligraphy data of *Changshuo Wu*, a famous calligrapher in Qing dynasty, as a representative test. We obtained 86 *Changshuo Wu*'s genuine calligraphy works and 68 *Changshuo Wu*'s forged instances collected from the market and identified as the forgery by the calligraphy and painting identification center in *Zhejiang University*. The experiment data consists of 3 databases: Database 1 (Db1) is originally built for our early research of calligraphy character retrieval, which contains 12066 genuine calligraphy characters written by different calligraphers in different dynasties. Database 2 (Db2) and Database 3 (Db3) are currently built for the test. Db2 contains 706 *Changshuo Wu*'s genuine calligraphy characters, and Db3 contains 502 *Changshuo Wu*'s forgeries characters.

7.1 Experiment

The input is a suspicious page, and the output is the accepting probability and the detected evidences. First, the input suspicious page image is segmented into individual characters. For each character, features on character level and stroke level are extracted and compared with the genuine's to detect problems and compute the total accepting probability, which can be written as:

$$P = \sum_{i=1}^{n} w_i \times p(c_i) \tag{13}$$

where $w_i = \frac{1}{n}$ is a weight for the character. Fig.5(a) shows a verification example for a suspicious page that claimed to be *changshuo Wu*'s. A user can click each

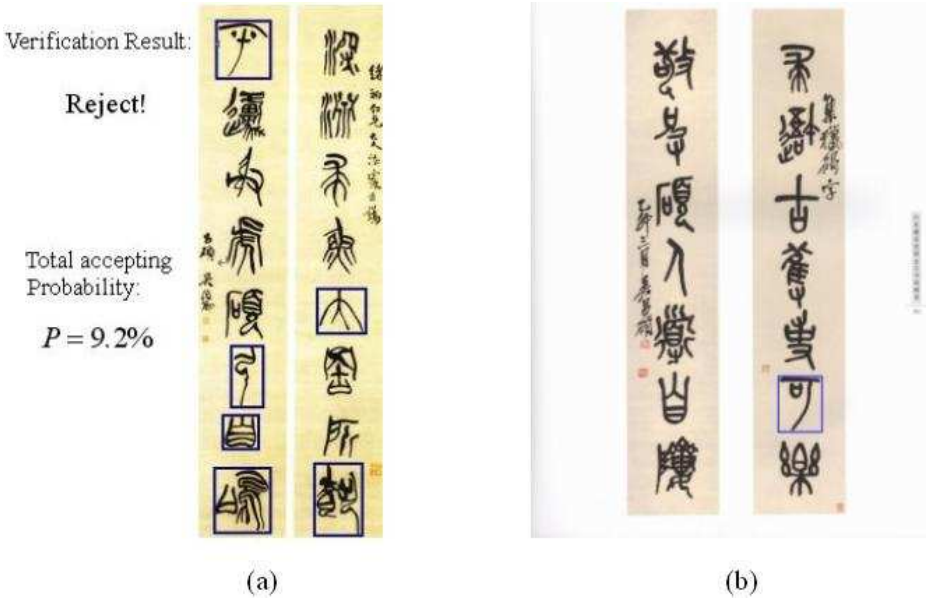


Fig. 5. (a) A verification example with problematic characters marked with blue minimum-bounding box. Reject: $P < 20\%$, Probably reject: $20\% < P < 40\%$, Neutral: $40\% < P < 60\%$, Probably accept: $60\% < P < 75\%$, Accept: $P > 75\%$. (b)Screen-shot of browsing the genuine works, with a blue minimum-bounding box mark out where a specified genuine reference character comes from.

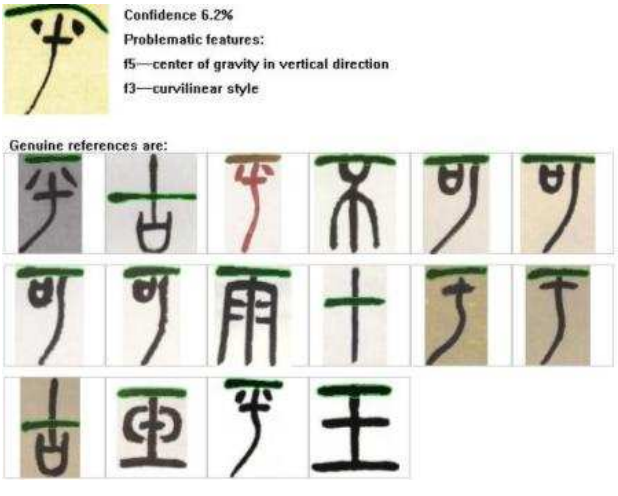


Fig. 6. Screen-shot of evidence show example. The top one is a problematic character and beside it is the problematic features. Contours of the problematic stroke and the corresponding genuine stroke references are marked in green.

character that marked with blue box to see the detail evidences. For example, if the first character in the first column is clicked, then a new page is pop up to present the evidence as shown in Fig.6.

If a user wants to know further about where an alleged genuine character comes from, then page retrieval can be done according to the map between the raw data and the feature data. For example, if a user want to know where the alleged last character in the first reference row in Fig.6 comes from, then with a click a new page will pop up to present its original works as shown in Fig.5(b) .

7.2 Evaluation

False accepts ratio and false rejects ratio are employed to evaluate the performance. If all suspicious are accepted as the genuine, then definitely the false reject ratio is 0% while the false accepts ratio is 100%. And if all suspicious are rejected, then the false reject ratio is 100% while the false accepts ratio is 0%. Both low false accepts ratio and low false rejects ratio are what we're trying to reach. But there is a tradeoff.

In order to find a balance, we draw an error tradeoff curve. For a suspicious, if the total accepting probability $P > threshold$, then it is accepted as a genuine. Else, it is rejected as a forgery. For a fixed *threshold*, we repeat the test 20 times with each time inputting a different suspicious page image, and compute its average false accepts ratio and false rejects ratio. Then we change the *threshold* and repeat the test. Fig.7 gives the comparison of tradeoff curves when using stroke based features and when using both stroke-based and character-based features.



Fig. 7. Error tradeoff curves of when using stroke-based features (SF) and when using features of both stroke based and character based (SCF)

8 Conclusion and Future Works

In form and feature, the forged and the genuine grew like the twins difficult to identify. This paper is a pioneer in giving objective measures to detect subtle

problems and offer detailed evidences for historical calligraphy verification. The measures are straightforward and simple, whereas the approach can be a reference for the analysis and the retrieval of other media objects, which share the same problem of feature selecting and measuring. The experiment is preliminary, yet it gives a clear idea about what can be achieved by objective measure for historical Chinese calligraphy verification.

Our future work includes designing objective measures on the level of page image, finding more features that can characterize calligrapher's particularities, and enlarging the database especially the faked data with the target is to explore low error rate.

Acknowledgements

This work is supported by National Natural Science Foundation of China (No. 60525108, No. 60533090), Science and Technology Project of Zhejiang Province (2005C13032, 2005C11001-05) and China-US Million Book Digital Library Project (www.cadal.zju.edu.cn)

References

1. Bangda Xu, A survey of historical calligraphy and painting verification, published by Chinese culture relics publishing house, 1982.
2. Defu Chen, Chinese calligraphy and painting verification: Fundamental theory, published by Sichuan University publishing house, 1998.
3. Yueting Zhuang, Xiafen Zhang, Jiangqin Wu, Xiqun Lu: Retrieval of Chinese Calligraphic Character Image. *2004 Pacific-Rim Conference on Multimedia, LNCS 3331*, pp. 17-24, 2004.
4. Luan Ling Lee, Lizarraga, M.G., An off-line method for human signature verification, *13th Int'l Conf. on Pattern Recognition*, vol.3, pp. 195 - 198, Aug. 1996.
5. Zhenyu He, Bin Fang, Jianwei Du, Yuan Yan Tang, Xinge You, A novel method for offline handwriting-based writer identification, *8th Int'l Conf. on document analysis and recognition*, vol.1, pp. 242-246, Sept. 2005.
6. Yong Zhu, Tieniu Tan, Yunhong Wang, Biometric personal identification based on handwriting, *15th Int'l Conf. on Pattern Recognition*, vol.2, pp. 797-800, Sept. 2000.
7. Cheng-Lin Liu, Ru-Wei Dai, Ying-Jian Liu, Extracting individual features from moments for Chinese writer identification, *3rd Int'l Conf. On Document Analysis and Recognition*, vol.1, pp.438 - 441, Aug. 1995.
8. Wiley, Fractal Geometry: Mathematical Foundation and Application, New York, 1990.
9. Mario E. Munich, Pietro Perona, Visual Identification by Signature Tracking, IEEE transaction on pattern analysis and machine intelligence, vol. 25, pp.200-217, Feb. 2003.
10. Madabusi, S., Srinivas, V., Bhaskaran, S., Balasubramanian, M., On-line and off-line signature verification using relative slope algorithm, IEEE International Workshop on Measurement Systems for Homeland Security, Contraband Detection and Personal Safety, pp. 11-15, March 2005.

Discovering User Information Goals with Semantic Website Media Modeling

Bibek Bhattarai*, Mike Wong*, and Rahul Singh*

San Francisco State University, San Francisco, CA 94132
{bdb,mikewong}@sfsu.edu, rsingh@cs.sfsu.edu

Abstract. In this work we present an approach to capture the total semantics in multimedia-multimodal web pages. Our research improves upon the state-of-the-art with two key features: (1) capturing the semantics of text and image-based media for static and dynamic web content; and (2) recognizing that information goals are defined by emergent user behavior and not statically declared by web design alone. Given a user session, the proposed method accurately predicts user information goals and presents them as a list of most relevant words and images. Conversely, given a set of information goals, the technique predicts possible user navigation patterns as network flow with a semantically-derived flow distribution. In the latter case, differences between predicted optimal and observed user navigation patterns highlight points of suboptimal website design. We compare this approach to other content-based techniques for modeling web-usage and demonstrate its effectiveness.

Keywords: web usability, multimedia semantics, information foraging.

1 Introduction

The mission of many educational and business websites is to deliver on-demand information. The usability of a website depends on how easily a user finds what they want—the user's *information goals*. Stated another way, a website is highly usable if its *content pages*—pages which are likely to contain information goals—are readily accessible. Improving website usability thus depends on accurately identifying prominent information goals. By enumerating information goals for frequent user sessions, the web designer can refactor the website design, making content pages easily accessible to users. Furthermore, given a set of information goals and the website structure, one can simulate the *flow* (traffic) pattern of users attempting to satisfy the information goals. The user flow can then be visualized and/or compared with the shortest path that satisfies the goals, revealing potential points of sub-optimal website design.

Prior research [11] in *information foraging* theory asserts that users typically navigate towards their information goal by following *link cues*, which are fragments of information within or near hyperlinks and relevant to the user's information goal. A cognitive theory based web-page usability inspection tool based on [11] is

* All authors contributed equally.

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 364 – 375, 2007.
© Springer-Verlag Berlin Heidelberg 2007

presented in [1]. Another work [3] presents two algorithms: one discovers information goals from user sessions and the other predicts probable user flow, given a set of information goals. Both algorithms are based on *spreading activation* theory [2] which stipulates that cognitive processes, such as memory recall, are imprecise and non-isolated, triggering a cascading recollection of semantically related subjects. This theory is modeled using network flow methods [13], where the nodes of a network are primed by an activation weight, and neighboring nodes are recursively activated with iteratively decreasing weights. In the context of web usage, user flow is directly analogous to activation weight. Nodes which have a large number of incoming edges (fan-in) receive activation weight contributions from many sources, and therefore indicate important nodes. In this context, it should be noted that works such as [4] demonstrate that a better understanding of web usage can be obtained by combining web usage mining with web content and structure.

The state-of-the-art is unsatisfactory in two ways. *First*, most approaches rely heavily on textual information as the only significant source of semantics and hyperlinks as the only means of semantic relationships. Because of this limitation both non-textual links and dynamic content are disregarded in analyzing user flow and the contribution of image-based semantics is ignored. *Second*, important pages (content pages) are assumed to be predefined as a consequence of web design. This assumption causes the following two significant problems: (1) the contribution of user context and emergent or exploratory behavior on information goals is missed and (2) pages with a large fan-in unduly influence information goal discovery and user flow prediction. The latter problem which severely hampers current techniques, such as [13], essentially occurs due to the inherent assumption that web site design drives user behavior more than content or user context. It is interesting to note this contradiction from the fundamental assertions of information foraging theory.

This paper presents an approach to capture the total semantic contributions of multimedia web pages and for predicting user flow over text and non-textual links as well as other interface modalities, such as HTML form queries and script-driven interfaces. The proposed method incorporates a semantic representation which allows for a dynamic automated discovery of content pages. This differs from the network flow model for information goal discovery, and thereby avoids static website design dependencies. For user flow prediction, this method improves upon the network flow model by including a semantic cue factor which re-aligns network flow towards content-driven behavior. Visualization of predicted user navigation patterns can then indicate problems in web usability, by showing information goals in the context of the current web structure. The efficacy of the proposed technique in accurately identifying information goals and predicting user navigation habits illustrates that accounting for semantics across multiple media leads to a demonstrably better understanding of user behavior.

The remainder of the paper is organized as follows; Section 2 starts with an outline of the key problems involved in developing an understanding of website semantics. We then describe a model of multimedia websites used in our approach and discuss how the multimedia content of web-pages contributes to the relationships and attributes in this model. The section concludes with a discussion on how these relationships identify information goals for a user session and also how the model predicts user flow for a given set of information goals. Section 3 covers experimental

evaluations and is comprised of three different experiments: (1) a user study which evaluates the effectiveness of the proposed approach in capturing accurate information goals; (2) a goal relevancy comparison to the IUNIS algorithm proposed in [3] and another user study directly comparing the information goals predicted by both the proposed method and IUNIS, and (3) an evaluation of the benefits from multimedia/multimodal analysis over unimedia-unimodal analysis, which also demonstrates a typical use case for the system and the visualization of user flow. These evaluations are performed on the SkyServer [14] website, which is a large (~40TB) multimedia and multimodal site. The SkyServer site is designed to educate the public about astronomy and to serve the data needs of the astronomy community by offering direct access to all the data collected from the Apache Point Observatory in New Mexico.

2 The Proposed Approach

The essential tasks in finding user information goals for a particular user session are as follows: finding meaningful semantic relationships (how information goals are interconnected), representing total semantic information for all media (what to look for), and discovering content pages (where to look).

2.1 Modeling Hyperlinked Multimedia Semantics

We model web pages as containers for media objects, which are classified as *text*, *images*, and *links*. The model associates *semantic annotations*, which are modified term-frequency/inverse-document-frequency (TFIDF) vectors with media objects. Images are described by texture-color vectors, which are described in detail in section 2.4.

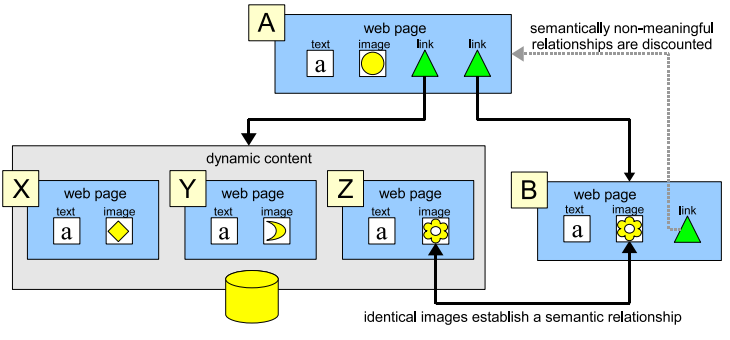


Fig. 1. A sample of the proposed semantic multimedia website model, showing meaningful semantic relationships and interoperability over dynamic and static content

As shown in Fig. 1, the links establish semantic relationships between the web pages. The figure also shows that parametric dynamic content may generate a number of views of the data, which manifest as web pages. For example, as shown in Fig. 1, web page *A* is the parent page for the web page *B* and any instance of the dynamic

content *X*, *Y*, or *Z*. If an image in *B* were to also appear in *Z*, then there would be an image semantics-based relationship between *B* and *Z*. Therefore the information from both pages *B* and *Z* would contribute to the semantics of this image.

2.2 Parent-Child Relationships as Strong Semantic Relationships

The first challenge in finding information goals is to determine how information is organized within a given website. Close relationships between pages in a user session give strong indications of what a user may be looking for. Many web sites have exceedingly interconnected pages; some links are semantically meaningful while others address usability issues. Common usability best practices include: offering a link to the home page on every page, providing a link so users can return to whence they last browsed, or showing a site map so a user can jump to any point in the overall organization of a subsection of the site. These links, in general, don't semantically relate the concepts of two web pages, and therefore need to be discounted when evaluating semantic relationships.

On the other hand, web sites are often organized from very broad and general introductions, down to specific information. For example, a university website may start with general information of the campus, and then link to a page on academic departments, which in turn links to a list of faculty, which finally links to the page about a particular professor, which lists research interests, office hours, location, and contact information. These parent-child relationships are strong semantic relationships and therefore deserve special attention. Our approach identifies the website's parent-child relationships by traversing the links by breadth-first-search. If multiple parent pages link to the same child page, the child is clustered with the most semantically similar parent. This method preserves the semantic relationships between parent and child pages while discriminating links which exist for usability issues alone.

The rationale for finding strong semantic relationships is to capture total media semantics as the user would perceive it. To capture total media semantics given parent-child relationships between pages, the semantics associated with a parent page should include a fraction of each of the child page semantics. In our approach, the fraction of semantic back-annotation is proportional to the semantic similarity of the two pages. If the web page organization is such that there is no true parent-child relationship between two linked topics, then there should be no appreciable semantic similarity between the pages, and no back-annotation would occur.

2.3 Evaluating Multimedia Semantics

Once the strong semantic relationships are identified, the evaluation of multimedia semantics can begin. First we retrieve the web pages from the website. Dynamic content requests which culminate in an *HTTP GET request* are recorded in the usage logs. Therefore the results can be reconstructed and analyzed as if it were static content. Next, we decompose a webpage into its media components. Webpage structure is directly analyzed to separate navigational and ornamental motifs from the most conspicuous or "main" region of the web page. When applicable, the content of the "main" region is used for all semantic evaluations. The two most frequent media for most websites are text and images and therefore we focus on these two media in

our analysis. Textual content is analyzed using a grammarless statistical method, which includes stemming and stop word filtration. This simple method enables the analysis of very large websites, using limited computing resources, in a reasonable amount of time. In the course of this research, several analysis techniques have been empirically evaluated: term-frequency/inverse-document-frequency (TFIDF) [12] and latent semantic analysis (LSA) [8], as well as combinations of TFIDF and LSA. Of these, one variation of TFIDF is satisfactorily found to approximate the semantic content with the least computational expense. This version of TFIDF (which we call DTFIDF) uses a dynamic background document set. This background set is comprised of pages that meet three criteria: they are part of the website, they are semantically similar to the page of interest, and they *link to*, or *are linked from* the page of interest. This helps avoid the unwanted contribution of terms with overloaded semantics by using a smaller, relevant document set as a background set. The semantic annotation of a media object is represented by a DTFIDF-weighted term frequency vector.

To calculate the DTFIDF-weighted term frequency vector for a document (web page), let the document d be represented by a normalized term frequency vector (tf) which has a non-zero value for each term t in d. Let D be the set of all documents in the document collection (web site). Let N be the set of documents in D such that each document e in N has a similarity measure $r_{de} \geq k$ with respects to d and there exists a link between d and e. The value k is an empirically-determined constant threshold. The dynamically-calculated inverse-document frequency (idf) is as follows:

$$idf = \log\left(\frac{|N|}{|\{e|e \in N, t \subset e\}|}\right) \tag{1}$$

Our next goal is to characterize the information represented through images. Image semantics are challenging to ascertain because of the complexities of visual analysis and interpretation. One leading concept, in this context, is the use of image ontology [7], [10] to map semantics to images. The greatest challenge in developing an image ontology lies in finding ways to automatically annotate images. Fortunately, many websites include images for purposes of illustration, and augment the image with some text information. By taking advantage of webpage substructure, we can isolate proximal text which can then be used to annotate images.

There are two challenges associated with extracting image annotations from websites. The first is related to the emergent nature of image semantics; the same image may be used in multiple contexts and have multiple meanings. Second, the image may serve only layout or navigational purposes and not have any relevant semantic contribution to the pages on which it is located. To resolve the first challenge, the semantic annotations of identical images are summed, regardless of where they occur in the website. Images which aren't meaningful tend to be re-used often and for unrelated topic. In these cases, the entropy of the associated semantics of each image is measured. Images whose semantic annotation entropies exceed an empirically-determined threshold are cleared of any semantic annotation; these images are meaningless with regards to identifying user information goals.

2.4 Image Content Analysis and Semantic Associations in Image Clusters

Image content provides semantic value, and similar images may have semantic similarities as well. Therefore it is necessary to analyze the image content. We analyze images by first segmenting the images and then performing color and texture analysis on each segment, which results in a feature vector for each image segment. We use the JSEG [5] color/texture analysis system to identify textures within the image. An example showing the heart of a celestial phenomenon and its coronas is shown in Figure 3. Texture characterization is done with Grey-Level Co-occurrence Matrices (GLCM) [6]. We use eight vectors as the offset parameter for GLCM, and measure four statistical analyses for each co-occurrence matrix: energy, entropy, contrast, and homogeneity. In addition, we generate a low-resolution color histogram for each texture. Relative size, energy, entropy, contrast, homogeneity, and the color histogram are combined to create a feature vector to describe an image segment.

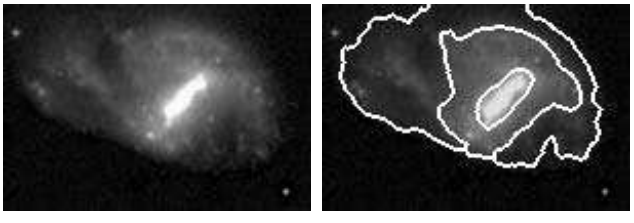


Fig. 2. Texture Segmentation on an image of a galaxy is the first step towards texture-color image feature extraction. Images are clustered by feature similarity. Commonly shared semantics in a cluster contribute more to information goal prediction.

Image similarity is measured through the normalized sum over segment similarity. Images are then clustered by similarity and each cluster is analyzed for frequently co-occurring terms in the semantic annotation of each image. Then, these terms are proportionally weighted. For example, the SkyServer website offers images of galaxies which share a high degree of similarity. These images often share the term *galaxy* in their semantic annotation, which is then given greater importance. Images of galaxy clusters are also similar to one-another, and are annotated with the terms *cluster* and *galaxy*, with *cluster* being more frequent than *galaxy*. Therefore, the term *cluster* is weighted more heavily than *galaxy* for the annotations which contain occurrences of *cluster*.

2.5 Dynamic Content Page Discovery and Information Goal Extraction

The current state-of-the-art takes the web designer's perspective that content pages are static as consequence of web design. We take a fundamentally different approach and assert that content pages are dynamic and relative to the user's information goals and the pages that the user had visited (in the session). Websites are generally organized so that users can drill down from pages with less specific information to pages with more specific information. A page with precise information (*e.g.* a page about a specific professor) is most likely to contain information goals. However, users

are free to traverse this hierarchy in many ways, foiling a designer's efforts to guide them. Measuring the semantic entropy of a given page gives the inverse of the specificity of the page. If a user follows a series of pages with monotonically decreasing entropy, then the web page at the local minima of entropy is likely to provide one or more information goals. These pages with low semantic entropy are identified as content pages. Every page in the user session may contribute towards a user's information goal prediction, but content pages are weighted to have a stronger contribution. Equation 3 shows the formula for the Shannon entropy of a semantic annotation x, being a term-frequency vector consisting of a set of term (t) and normalized frequency-count (c_t) pairs using DTFIDF.

$$H(x) = -\sum_{t \in x} c_t \log(c_t) \tag{2}$$

Information goals are extracted as a subset of the semantic information of web pages visited in a user session using both text and image information. The semantic contribution of each page is considered, with a greater weight placed on the semantic contribution of content pages. The term list is sorted first by page navigation order, and then by weight. Twenty most important terms are then used to form the basis of the predicted information goals. Images that are seen in the user session and have a semantic contribution towards the top 20 terms are also included, completing the predicted multimedia information goals.

2.6 User Flow Prediction and Implications on Usability

User navigation can be modeled as a network flow where the user may either leave the site or visit a different page of the website. Given a set of information goals, our model uses the semantic annotation to generate a probability distribution of user flow. The distribution asserts that links which are likely to lead to a page which satisfies a information goal are more probable than links which do not appear to satisfy a information goal. The criteria for a link which is likely to result in goal satisfaction are: (1) the link is associated with text or image semantics which is relevant to a goal; or (2) the linked page is semantically relevant to a goal.

$$p(l_0) = \frac{\sum_{t \in (S_{l_0} \cap G)} c_t}{2 \sum_{l \in L} \sum_{t \in (S_l \cap G)} c_t} + \frac{r_{GP_{l_0}}}{2 \sum_{l \in L} r_{GP_l}} \tag{3}$$

Users forage for their information goals by following information cues, which are bits of semantic information within (or near) a hyperlink. For a given set of information goals (G) and a given page with a set of links (L), each link (l) associated with semantic information (S_l), we can create a probability distribution which predicts the user flow to the linked page (P_l). The *information cue probability* ($p(l_0)$) for a particular link (l_0) is the probability a linked page will contain or lead to an information goal, is calculated as average of the link cue and the semantic cue. The link cue is the normalized sum of DTFIDF frequency counts of the terms that are

present in both the link semantics and the information goal. The semantic cue is given as the distance of the goal semantics and the linked page.

The user flow is computed by simulating users through an activation function *A(t)* as shown in equation 5. The total percentage of users at a given time in a page depends on total information correlation value for all the links pointing to the page. The dampening factor α represents the probability of the user leaving the website from any given page. The matrix *I* represents the information cue matrix where the rows of the matrix represent a set of links from a web page. Each element of the row is calculated as the information cue probability described in equation 4. *E* simulates users flowing through the links from the entry (or start) page of the usage pattern. The initial activation vector A(1) = E. The final activation vector, A(n), gives the percentage of users in each node of the website after n iterations.

$$A(t) = \alpha IA(t-1) + E \tag{4}$$

For each user session in a usage pattern, the algorithm computes the shortest path which covers all information goals in the predicted order. Our underlying assumption is that the shortest path represents the most optimal (direct) path to the desired information goal. User sessions which fit the predicted user flow are considered to have similar information goals, and are therefore fit for comparison. Comparing these sessions with the optimal shortest path provides an analysis of website design. If the user navigation patterns diverge from the optimal path, then there may be design problems at the point(s) of divergence.

3 Experiments and Results

The evaluation consists of three different experiments: (1) a user study which evaluates the effectiveness of the model in capturing accurate information goals, (2) a goal relevancy comparison to the IUNIS algorithm [3] and another study directly comparing the information goals predicted by both the proposed system and IUNIS, and (3) an evaluation of the benefits from multimedia-multimodal analysis over unimedia-unimodal analysis. Included here is a typical use case for the system and the visualization of user flow. These evaluations are performed on the SkyServer website [14] website, which is a large (~40TB) multimedia and multimodal website. The SkyServer website is designed to educate the public about astronomy and to serve the data needs of the astronomy community by offering direct access to all the data collected from the Apache Point Observatory in New Mexico.

3.1 Evaluation of User Information Goal Discovery

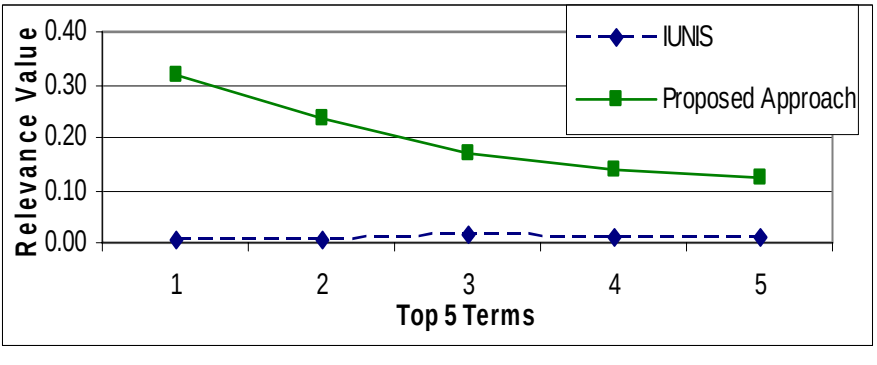
To evaluate the correlation between predicted information goals and users' information goal, a user study was conducted with 8 participants, none of whom were familiar with our research. Ten frequently recurring user sessions were selected from the usage logs and analyzed using the proposed approach, generating ten sets of predicted multimedia information goals. Four sessions were selected from the ten, and each of the web pages for these sessions were printed and bound in chronological order, producing four booklets. Each user was asked to read three of these web

session booklets, one-at-a-time, and for each booklet the user was asked to compare the information they just read to the 10 sets of predicted information goals. Each user was then asked to determine which predicted information goal best describes the information of the given booklet. Overall, the users agreed with the information goals predicted by the proposed approach 15 out of 24 times. The *t-value* for 15 agreements for a binomial distribution where n=24 and p=0.1 is 42.0, which strongly rejects the null hypothesis of no correlation. Therefore, we conclude that it is likely that there is a strong correlation between the predicted goals and the users' information goals. It should be noted that of the 9 incorrect answers, 5 were due to confusion with a very similar set of information goals which shared 60% of the terms, and 25% of the images with the predicted set of information goals.

To cross-evaluate the previous question, users were also asked to rank all 10 sets of goals on a Likert scale from 1 (not relevant) to 5 (highly relevant) with 3 (neutral) as the mid-point. We tested for systemic bias in the study by measuring mean pair-wise Pearson's correlation of answers for each booklet and found a low correlation between answers ($\bar{x}^2 = 0.233, \bar{x} = 0.483, s = 0.280$), indicating no significant systemic bias. Users scored the proposed approach's set of goals high, ($\bar{x} = 4.38, s = 0.824$, out of 5) compared to all other sets of goals ($\bar{x} = 2.56, s = 0.160$), indicating that the proposed approach predicts distinguishably relevant information goals for a given user session.

We next compared goal analysis from IUNIS [3] with the proposed approach over the same ten user sessions from the user study. For each user session, the top five terms of the information goals predicted by IUNIS were compared with the top five terms of the information goals predicted by the proposed approach. As seen in table 1, the top 5 terms score up to 47.6 times more strongly by the proposed approach than by IUNIS; the mean increase in term relevancy is 24.6 times ($\bar{x} = 24.58, s = 18.23$).

Table 1. Comparison of the Relevance Scores for the Top 5 Goal Terms for IUNIS vs. the Proposed Approach Reveals a Mean 24.6 Times Improvement



3.2 Evaluation of Multimedia and Multimodal Analysis

This experiment evaluated the effectiveness of text-and-image analysis versus text-only analysis. The evaluation focused on nine user sessions where graphics provide a

significant contribution to web content. Figure 3 shows the first user session; the other user sessions were identical except for the last page, which correspondingly were *page1.asp* to *page6.asp*, *ngc.asp*, *abell.asp*, and *messier.asp*, respectively. In these sessions, the user visited pages that have thumbnail images of galaxies, spirals, clusters, and so on with the last page displaying specific annotated image examples of these celestial objects.

Base URL: http://skyserver.sdss.org/dr1/en/tools
1: default.asp
2: places/default.asp
3: places/page1.asp

Fig. 3. User Session with Text and Image Media

The text-only model was found to undervalue image semantics leading to the names of the featured galaxies not appearing in the information goal, despite the prominent captions. In contrast, the multimedia model captured image relevance, ranking the galaxy names in the top 20 information goals, while preserving the content and rankings of the top 5 terms as discovered in text-only analysis. Moreover, due to the semantic association of similar images, the top terms showed an improved relevance score, skewing the relevance curve towards the most important terms. The mean relevance improvement was found to be 4.1 times ($\bar{x} = 4.13, s = 1.49$).

The final evaluation is meant to demonstrate the ability of the proposed approach to predict user flow through multiple modalities, specifically hyperlink browsing and script-enabled dynamic content (a simulation of a virtual telescope, allowing the user to view different areas of the night sky). The session shown in Figure 4 considers a typical use case: the user first browses through the static pages and then interacts with the script-enabled dynamic content in the third page of the session. The proposed approach captures the information goals for all pages in the user session, regardless of the interaction mode. As shown in Figure 4, the proposed approach predicts a non-zero probability for user flow towards the dynamic content page, which is a capability not available to the current state-of-the-art approaches.

Base URL: http://skyserver.sdss.org/dr1/en/
1: default.asp
2: /tools/places/default.asp
3: /tools/places/page2.asp
4: /tools/explore/obj.asp?ra=221.546&dec=-0.223

Fig. 4. User Session with dynamic content access

In Figure 5, the lines represent hypertext links and the nodes represent web pages. The figure shows the user flow (orange solid line), the user session path (green dotted line), the shortest path (blue broken line) and the red bar representing the user flow probability computed by the system. Selecting a node shows a thumbnail image of the page and the URL.

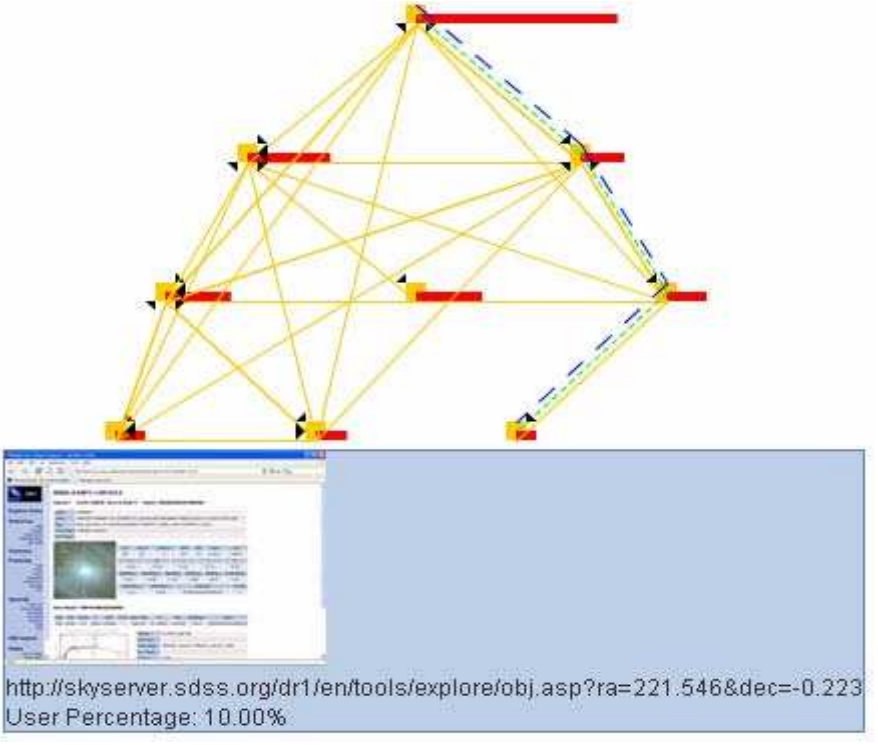


Fig. 5. User flow diagram shows the given user session with dynamic page content access

4 Conclusion

In this paper we presented a new approach for capturing total semantics in multimedia web pages and demonstrated how the method can be used to identify user information goals and predict user flow using a network flow methods. Two key contributions of our research are: (1) an algorithm for dynamic content page identification and assimilation for information goal discovery and (2) a semantically determined (over multimedia content) probability distribution which works over multimodal interfaces for user flow prediction. User study evaluation of this model shows that the model predicts accurate information goals for a given user session with a mean of 24.6 times greater relevance than the current state-of-the-art. This remarkable improvement is due to a fundamentally different approach to localizing semantics and dynamically aligning web page semantics and user navigation. Experimental studies underline the significant improvements brought about by capturing semantics over multimedia content. Finally, we presented a mechanism to predict user flow over rich multimodal interfaces for website dynamic content. In conclusion, this research provided compelling and exciting evidence towards the usefulness of characterizing multimedia-multimodal semantics in context of analysis of web usability.

References

[1] M. H. Blackmon, P. G. Polson, M. Kitajima, C. Lewis, Cognitive Walkthrough for the Web. ACM Proceedings of Conference on Human Factors in Computing Systems (CHI '02), 2002.

[2] J. R. Anderson, P. L. Pirolli, Spread of Activation, Journal of Experimental Psychology: Learning, Memory, and Cognition, Vol. 10, pp. 791-798, 1998

[3] E. Chi, P. L. Pirolli, K. Chen, J. Pitkow, Using Information Scent to Model User Information Needs and Actions on the Web. ACM Proceedings of Conference on Human Factors in Computing Systems (CHI '01)

[4] R. Cooley, The Use of Web Structure and Content to Identify Subjectively Interesting Web Usage Patterns. 2003 ACM Transactions on Internet Technology 3(2), pp 93-116, 2003.

[5] Y. Deng, and B. S. Manjunath, Unsupervised segmentation of color-texture regions in images and video, IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI '01), vol. 23, no. 8, pp. 800-810, 2001

[6] P. Howarth, S. Rüger, Evaluation of Texture Features for Content-based Image Retrieval, Lecture Notes in Computer Science, Volume 3115, pp. 326 – 334, 2004

[7] E. Hyvönen, A. Styrman, S. Saarela, Ontology-Based Image Retrieval. Towards the semantic web and web services, Proceedings of XML Finland 2002 Conference, pp. 15-27, Helsinki, Finland, October 21-22, 2002

[8] T. K. Landauer, P. W. Foltz, and D. Laham, An Introduction to Latent Semantic Analysis, Discourse Processes, Vol. 25, pp. 259-284, 1998.

[9] B. S. Manjunath, and W. Y. Ma, Texture Features for Browsing and Retrieval of Image Data, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 18, no. 8, pp. 837-842, 1996

[10] V. Mezaris, I. Kompatsiaris, M.G. Strintzis, An ontology approach to object-based image retrieval, International Conference on Image Processing vol. 3, no. 2, pp. II- 511-14, 2003

[11] P.L. Pirolli, and S. K. Card, Information foraging. Psychological Review. 106: p. 643-675, 1999

[12] G. Salton, and C. Buckley, Term Weighting Approaches in Automatic Text Retrieval, Technical Report: TR87-881, 1987.

[13] G. Salton, and C. Buckley, On the Use of Spreading Activation Methods in Automatic Information Retrieval, ACM Conference on Research and Development in Information Retrieval, pp. 147-160, 1988

[14] Sloan Digital Sky Survey project's website SkyServer: http://skyserver.sdss.org/dr1/en (English version for Data Release 1)

Online Surveillance Video Archive System

Nurcan Durak[1], Adnan Yazici[1], and Roy George[2]

[1] Computer Engineering Department, METU, 06530, Ankara, Turkey
{nurcan, yazici}@ceng.metu.edu.tr
[2] Dept. of Computer Science, Clark Atlanta University, Atlanta, GA 30314, USA
rkavil@gmail.com

Abstract. Powerful video data models and querying techniques are required to retrieve video segments from the video archives efficiently. Structure of video data model and types of video queries may change depending on an application. In this paper, we present a video data model and wide range of query types considering needs of surveillance video applications. In the video data model, metadata information, automatically extracted moving objects, events and objects' positions are considered. The query set based on our video data model includes semantic queries, spatial queries, regional queries, size-based queries, trajectory queries, and temporal queries. With the developed web-based program, all of the queries can be processed over Internet and their results can be played with streaming technology. With our developed system, the functionality of surveillance video archives is increased.

1 Introduction

With the development of recording technologies, the use of surveillance cameras increases around us. We can see surveillance cameras at shopping centers, airports, banks, embassies, museums, schools etc. The video records of those cameras are captured and archived to retrieve important segments later. Retrieving desired segments instead of watching all videos makes the jobs of the security workers, policemen, detectives, or insurance firms' easier and faster. To access related video segments immediately, video data model and query algorithms should be designed and developed efficiently and effectively. In addition, graphical user interfaces for querying surveillance video archives should be designed by regarding users' needs. We propose a video data model and querying algorithms and have developed a web-based query tool to address the surveillance video archives' requirements.

Most of the video surveillance systems concentrate on the automatic data extraction problem such as moving object extraction, object classification, object identification, object trajectory finding or activity classification [3, 5, 8, 9, 10]. In [9], moving objects are extracted with background subtraction methods and identified with color and texture parameters. With the help of HMM, object's trajectories are trained to classify object behaviors into normal or abnormal events. In [9], object identification and event detection aspects are limited and erroneous. Whereas in [5], IBM smart surveillance engine detects moving objects, tracks multiple objects, classifies objects and events, and then executes event based queries. Similar to this study, in [8] video sequences are retrieved only event based queries. In [5, 8], supported queries are

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 376 – 385, 2007.
© Springer-Verlag Berlin Heidelberg 2007

limited and video data models and query interfaces are not addressed. In another point of view in the surveillance video studies is the cooperation of multiple camera outputs to track moving objects [3, 10]. In these studies, moving object trajectory queries are executed using SQL tags which are not easy to use.

The main contributions of this study are as follows: 1) designing a video data model for surveillance videos; 2) enhancing existing query types; 3) applying query types on surveillance video archives; 4) designing efficient query algorithms; 5) developing easily usable query user interfaces and accessing to the system over Internet.

Our surveillance video modeling and querying system is called SURVIM which models the metadata information, the moving objects, events, and spatial positions of the objects. These entities are extracted automatically by using a software tool [9]. SURVIM presents users a very rich query set including semantic queries, temporal queries, spatial queries, regional queries, size-based queries, and trajectory queries. In the *semantic queries*, objects and events are queried with time interval information. With the *size based queries*, users can retrieve objects according to their sizes. In the *temporal queries*, users can query the temporal relations among events and objects. The other queries are related with the object positions. More specifically, the spatial queries retrieve spatial relations among objects and the regional queries asks for the objects by their positions on the video frames, and the trajectory queries allow users to query the moving objects' paths. We also support conjunctive queries that contain multiple query expressions in the same type.

The rest of this paper is organized as follows: Section 2 describes SURVIM architecture. In Section 3, our video data model is presented and in Section 4, supported queries and their algorithms are described with examples. Conclusion of our work and future research directions are given in Section 5.

2 SURVIM Architecture

SURVIM architecture consists of surveillance cameras, video storage server, data extractor, video database model, query processor, and user interfaces as shown in Figure 1. Raw video files are captured from real time surveillance cameras, after that they are compressed and stored in the video storage server according to their location information. Video name is given by combining of date and time information of the captured video. Captured video file is also sent to the data extractor part, which composes of metadata extractor, moving object extractor, and stationary object extractor subparts. Data extractor gives following attributes of the video file: location, description, date, time, moving objects, stationary objects, events, and spatial positions of the objects. All of the extracted attributes are indexed in a database by using our video data model. In the query processor part, there are number of query algorithms which process query specifications over the model. The query specifications are entered into system via query user interfaces and sent to query processor part. After processing the given query conditions, the query processor sends query results to the user interfaces. If users want to play the retrieved video clips, then clips are streamed into the user's computer.

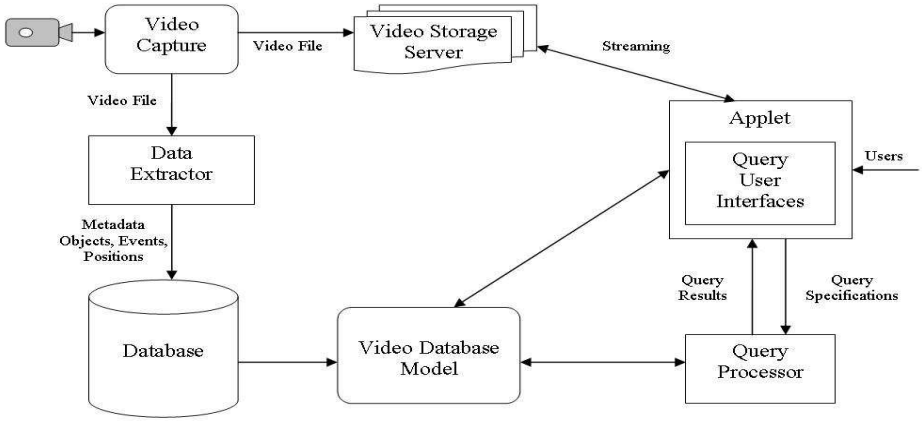


Fig. 1. The Architecture of SURVIM

We have implemented the video data model, the query algorithms, and the user interfaces using Java. For each query type, there are different user interfaces for specifying different characteristics of the query types. Query results in the video clip type are streamed to users with the help of Java Media Framework.

3 Video Data Model

Semantic video data models aim at accessing quickly to video segments having important events or objects [1, 4, 6]. Our video data model captures metadata information, events, objects and their relations as shown in Figure 2. Descriptions of the classes are as follows:

- *Video Metadata:* We keep following video attributes: *VideoId, VideoName, Category, Description, Date, Time, Location, Length*, and *VideoUrl.* Metadata information can be extended according to different kinds of surveillance applications. In our model, each video must have a unique video identity number, which is given from the database automatically. Video metadata class is associated with other classes with 'VideoId' attribute to represent the entity's own video.
- *Time Interval:* Time interval specifies a moving object's entering and exiting time points and an event's starting and ending time points. There can be one or more objects in a certain time interval and all of the objects in the time interval are kept in 'Object List'. If an event occurs in a time interval, then the name of the event is kept in 'Event' field in the data structure. Each time interval must have a unique 'IntervalId' in the own video.
- *Event:* Event is object behavior such as walking, running, packet breaking, etc. An event occurs in a time interval, so each event should have at least one time interval that is represented with 'IntervalId' of the time interval. One event can happen in the different time intervals; therefore, we keep all time intervals' ids in 'Time Interval Linked List'. In an event, there can be one or more objects that are kept in the 'Object Linked List'.

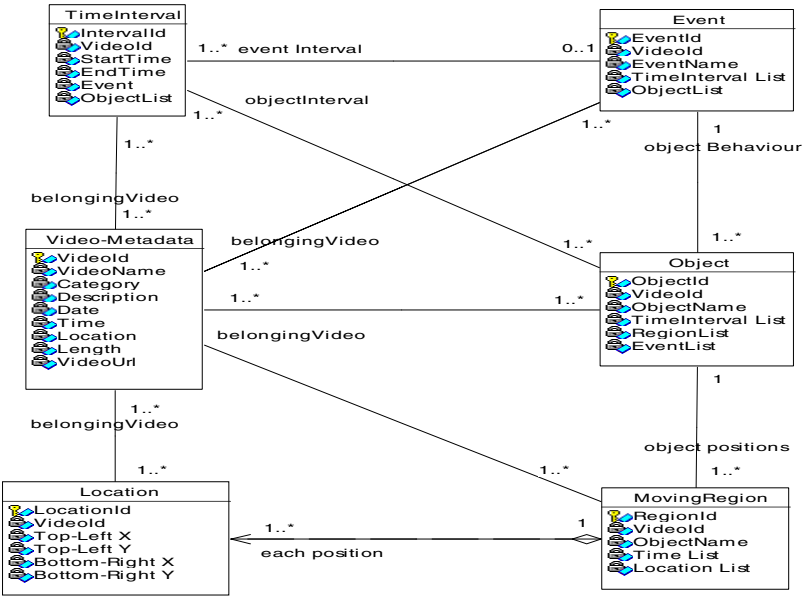


Fig. 2. Video data model in UML

- *Object:* Objects should be associated with time intervals that are kept in 'Time Interval Linked List'. One object can be seen in different events, all events containing the object are kept in 'Event Linked List'. According to our model, an object must have different moving regions in different time intervals and all moving regions of the object are kept in 'Region Linked List'.
- *Moving Region:* Moving region consists of a sequence of spatial locations that are taken in every considerable movements of an object during the object's appearing time interval. We keep spatial locations in the location class and all of the locations in a moving region are kept in 'Location Linked List'. 'Time Linked List' stores the times of the frames at which spatial locations are taken.
- *Location:* The position of an object on a frame is represented with a minimum bounding rectangular that covers the object's top-left point and bottom-right point. These two points are stored in the location class.

4 Querying

SURVIM mainly supports the following query types: semantic query, regional query, spatial query, spatio-temporal query, temporal query, and size-based queries. The visual query interfaces are used to submit queries to the system and to visualize the query results. After a query submission, query conditions are processed using query algorithms over the video data model. In all query types, queries can be processed over a single video or a group of videos specified with metadata information like date, time and place. To specify videos, the video specification interface is used.

In Figure 3, the video specification interface is shown. Users can specify location, date for specifying single video and date interval for specifying a group of video. In this interface, *"parking lot videos between 07/10/2006 and 07/15/2006"* are specified.



Fig. 3. Video Specification User Interface

4.1 Semantic Queries

In semantic queries, semantic entities, which are objects and events, are queried with time intervals. Events, objects and time intervals in the query sentences are entered into system via user interfaces. We support atomic queries which contain one query sentence and conjunctive queries which contain more than one query sentence. In conjunctive queries, each query is processed separately, after that the results of the queries are combined according to specifications. Video semantics are queried in five different ways:

1. A time interval is specified by users, after that objects in the given time interval are queried. An example query is: *"Find all moving objects between 3th and 19th minutes in the 'parking lot' videos on 07/22/2005"*. To evaluate this query, time intervals intersecting with the given interval are found, after that objects in the returned time intervals are listed as a result.
2. An object or an event is specified by users, and then time intervals containing the given semantic entity are queried. An example for such a query is*: "Find all time intervals in which 'pocket breaking' happen in the 'parking lot' videos on 07/22/2005"*. To evaluate this query, specified semantic entity is found, then time interval list of the given semantic entity is listed as a result.
3. An object is specified by users, and then events containing the given object are queried. An example is: *"Find all events having truck object in the traffic videos on 10/07/2005"*. To process this query, the specified object is found and the event list of the specified object is listed as a result.
4. An event is specified by users, and then objects in the given event are queried. An example query is*: "Find all objects of abnormal events in the calculus exam videos on 01/10/2005"*. To process this query, the specified event is found, the object list of the specified event is listed as a result.

5. Objects or events are specified by a user, and then videos containing the given semantic entities are queried. An example query is*: "Find all videos in which ambulance is seen in traffic surveillance videos between 10/15/2005 and 10/30/2005"*. For processing this query, all videos in the specified place and between the specified date and then the specified semantic entity are searched over those videos.

4.2 Size-Based Queries

For some cases, querying the objects based on size is important. Depending on the application, the importance of a moving object can change according to its size. For example, in parking lot videos, the importance of a car and a human is more important than small objects like a cat or a dog. Query specifications are given into the system with query-by-sketch method. We support following size-based query types:

1. Users draw a rectangle on the frame to specify the size and they ask for the objects having a size for smaller or bigger than the drawn size. An example query is: *"Find all objects bigger than size S1 [80x60] from parking lot video on 09/30/2005"*. If the smaller objects are asked, objects' sizes are compared with S1 and smaller objects are put into the result list. Otherwise, the bigger objects are put into the result list.
2. A small rectangle and a big rectangle are drawn by users to represent small and big sizes, and the objects having size between the given two sizes are asked. A sample query: *"Find all objects having size between S1[20x30] and S2[50x60] from entrance video on 09/08/2005"*. In this case, the size of the objects in the specified videos are compared with S1 and S2 and then objects having the size between S1 and S2 are inserted into the result list.

4.3 Temporal Queries

In our system, temporal queries find temporal relations among the events and objects. We support following temporal relations: *before, during, equal, meets, overlaps, starts,* and *finishes* which are calculated according to formulas in [2]. We also support conjunctive temporal queries that contain more than one temporal query sentence. Temporal query types are as follows:

1. A time stamp and a temporal relation are specified by users, then events or objects satisfying the specified temporal relation are queried. An example for such a query is: *"Find all abnormal events 'before 17th minute' in store videos on 06/23/2005*". In the processing phase, if objects are asked, objects in the specified videos are compared with the given time stamp by the given temporal relation. Otherwise, events are compared with the given time stamp.
2. A semantic entity and a temporal relation are specified, and then events or objects satisfying the given temporal relation with the specified entity are queried. An example query is*: "Find all objects overlapping with crashing event in traffic surveillance video on 09/03/2005"*. In this case, a given object or a given event is found in the specified video, and then its time intervals compared with the given entity's time intervals.

4.4 Spatial Queries

In another type of query is the spatial query, in which spatial relations among objects are queried. Spatial relations can be topological such as *inside, contain, cover, overlap, touch* or directional such as *south, north, west, east etc.* in [7]. Köprülü et. al. extend these spatial relations by defining fuzzy membership functions [6]. Fuzzy spatial relations are useful when the spatial relation is not strictly satisfied between two objects. For example, a person can be left of a car with the threshold value of 0.6 means that person can be between left of the car and bottom-left of the car and left relation is satisfied at least 60%. In our study, spatial relations among the stationary objects and the moving objects are queried. For example, a safe-box is a stationary object and there can be a need for a spatial query which highlights the spatial relation between the safe-box and any moving object. Our supported spatial query types along with some examples are as follows:

1. Two objects, a spatial relation and a threshold value are specified by users, afterwards time intervals satisfying the given spatial relation with the given threshold value are queried. An example query is*: "Find all time intervals in which a walker is left of a car with a 0.7 threshold value in the parking lot videos on 07/29/2004"*. To execute this query, common frames that both objects appear together are found. Fuzzy spatial operators introduced in [6] are applied on objects' positions on the common frames.
2. One object, a spatial relation, and a threshold value are specified by users in the query, afterwards all objects satisfying the given spatial relation with the given object in the given threshold range are retrieved. An example is: *"Find all objects overlapping with a safe-box with a 0.9 threshold value in jeweler store videos between 08/20/2005 and 08/30/2005"*. To execute this query, the given object is compared with the other objects according to the specified spatial relation.
3. *Conjunctive spatial queries:* With these queries, multiple spatial query sentences are specified by users, then time intervals satisfying all query conditions altogether are asked for. In the conjunctive case, each spatial query is processed separately then query results of the each query sentence are intersected. An example query is*:* "Find all intervals in which *'a man is left of a car with a 0.6 threshold value'* and *'a car is top of the pedestrian way with a 0.8 threshold value' in parking lot surveillance videos in July, 2005"*.

4.5 Regional Queries

With regional queries, important parts in the camera view are taken under control. Query specifications are entered into system with query-by-sketch method. Threshold value is used in regional queries when two different regions overlap partially. To calculate the matching degree, we use overlapping formula given in (1).

$$\mu = \frac{\text{intersectedArea(Rect1, Rect2)}}{\text{minimumArea(Rect1, Rect2)}} \tag{1}$$

We support following regional query types:

1. An object and a time interval are specified by users, and then frame positions containing the given object in the given interval are queried. A sample query is*: "Find all positions, in which truck is seen between 3rd and 19th minutes of exit way videos on 09/22/2005"*. To evaluate this query, a specified object is found and its moving regions are taken and spatial positions in the given time interval are listed as a result.
2. An object, a position, and a threshold value are specified by a user in the query. Afterwards time intervals satisfying the given conditions are to be retrieved. An example query is: "*Find all intervals where a human is seen in the position given by pixel values [20, 60, 100, 120] with a threshold value of 0.8 in warehouse videos on 03/12/2004"*. To execute this query, the specified object is found and its moving region list is obtained. Every position in the moving region list is compared with the given position by using the fuzzy overlapping formula given in (**1**). If the threshold value is equal or greater than the given threshold value, then time of the position is inserted into the result list.
3. A spatial position and a threshold value are specified by a user, and objects in the given position are queried. A sample query is*: "Find all objects in the position given by pixel values [0, 0, 80, 60] with a threshold value 0.7 in store videos on 07/21/2005"*. To execute this query, positions of the objects in the specified video are compared with the given position according to formula (**1**). Objects overlapping with the given region with equal or greater than the given threshold value are inserted into the result list.

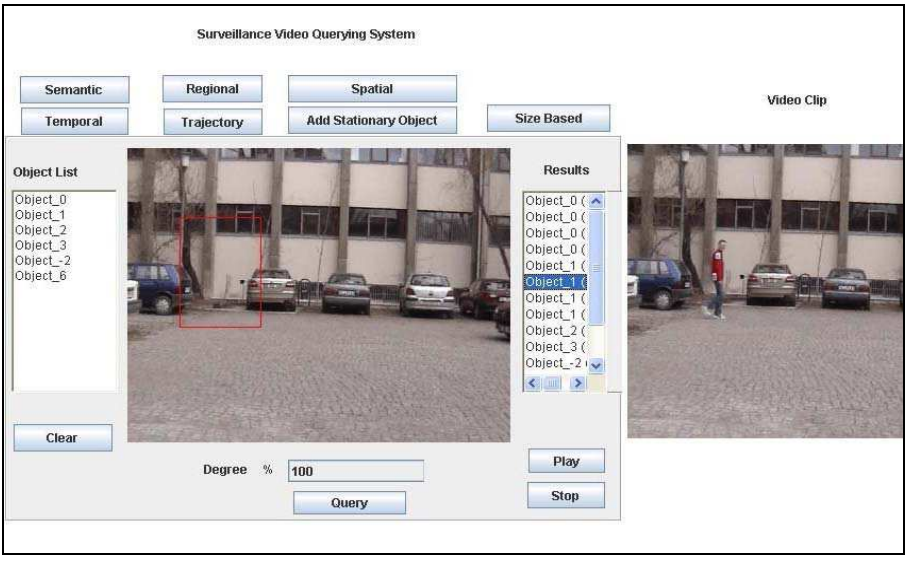


Fig. 4. Querying Web Browser

Figure 4 shows the querying web browser including the regional query panel. In the draw panel, a red rectangle is drawn by the user and the objects passed in the red rectangle are to be retrieved. Results are a group of objects which can be selected for displaying in the video clip panel.

4.6 Trajectory Queries

In the surveillance videos, a moving object follows a path during its appearing time interval. With trajectory queries, moving object paths are queried using fuzzy membership values. In [6], fuzzy membership functions used in a path calculation are defined. Trajectory query types and their examples are as follows:

1. An object is specified by a user, and all paths followed by the specified object are queried. An example query is*: "Find all trajectories of truck objects in the motorway videos on 09/18/2005"*. To execute this query, the specified object is found and its moving regions are taken, after that trajectory is created by connecting center points of all the positions in the moving regions. The created trajectory is inserted into result list. These trajectories can be drawn for users and the interval of the trajectory can be played. In Figure 5, one of the object trajectories is shown with the start rectangle, the end rectangle and the path between them.



Fig. 5. An Object Trajectory

2. A threshold value, starting position and ending position of the desired path are specified in the query by a user. Afterwards all objects that follow the given path in the threshold range are queried. If the object follows the path strictly, then the threshold value of the path is 1 (one). Threshold value is less than 1 (one), when the object follows the specified path with some distortions. An example query is: *"Find all trajectories starting from the position P1 and ending in position P2 with a threshold value 0.5 in parking lot videos on 07/28/2005"*. Users specify the starting position and ending position with query-by-sketch method. To evaluate this query, objects' positions are compared with P1 and P2. If the object is seen on P1 and the following positions are reaching position P2, we create a trajectory and insert this trajectory into the result list.

5 Conclusion and Future Works

In this paper, we present a framework for querying surveillance video archives over Internet. The proposed framework consists of the video data model, data extractor, query processor, and query user interfaces. Our video data model is designed

considering needs of surveillance video archives and semantic features of videos. Functionality of the surveillance video archives is increased with very wide range of query set. We have developed the system in Java Applets in order to easy access and use via internet browsers. Video segments are streamed into the user's computer for providing security of surveillance video archives. Another important point of our study is the successful integration of automatic extractor tool with our video data model. The incoming data into our system depends on the power of the extractor tool. For now, the automatic extractor tool extracts limited number of events and does not classify objects successfully. In future, we plan to integrate our system with a more powerful extractor tool to increase the practice use of the system. In this paper, low level features such as color and velocity are not considered. In future, the model and the query set can be improved by adding low level features into the system.

Acknowledgement

This research is partially supported by NSF Grants DUE-0417079 and HRD-0401679, US Army W913V-06-C-0021and DOD Grant No: DAAD19-01-2-0014. The content of this work does not reflect the position or policy of the sponsors and no official endorsement should be inferred.

References

1. Adalı S., K. S. Candan, Chen S., Erol K., Subrahmanian V.S.:The advanced video info rma tion system:data structures and query processing. Multimedia Systems,V. 4(1996) 172-186
2. Allen J.F.: Maintaining knowledge about temporal intervals. Comm. of ACM, 26 (11), (1983) 832-843
3. Black J., Ellis T., Makris D.: A Hierarchical Database for Visual Surveillance Applications. IEEE ICME (2004) 1571-1574
4. Donderler M.E., SaykoE. l, Ulusoy O., Gudukbay U.: BilVideo: A Video Database Management System. IEEE Multimedia, Vol. 10, No. 1 (2003) 66-70
5. Hampapur, A. Brown, L. Connell, J. Ekin, A. Haas, N. Lu, M. Merkl, H. Pankanti, S. : Smart video surveillance: exploring the concept of multiscale spatiotemporal tracking, Signal Processing Magazine, IEEE Volume: 22, Issue: 2 (2005) 38- 51
6. Köprülü M., Cicekli N.K., Yazici: A. Spatio-temporal querying in video databases. International Journal of Information Sciences. 160(1-4), (2004) 131-152
7. Li J.Z., Özsu M.T., Szafron D.: Modeling of moving objects in a video database. Proceedings of IEEE Int. Conf. on Multimedia Computing and Systems, Canada (1997) 336-343
8. Lyons, D., Brodsky, T., Cohen-Solal, E., Elgammal, A.: Video content analysis for surveillance applications. In: Philips Digital Video Technologies Workshop. (2000)
9. Orten B., Alatan A., Moving Object Identification and Event Recognition in Video Surveillance Systems. Ms. Thesis, Electric and Electronic Department, METU, 2005
10. Rangaswami R., Dimitrijevic Z., Kakligian K., Chang E., Wang Y.F.: The SfinX Video Surveillance System. ICME, Taipei, Taiwan, (2004)

Hierarchical Indexing Structure for 3D Human Motions

Gaurav N. Pradhan, Chuanjun Li, and Balakrishnan Prabhakaran

Department of Computer Science
University of Texas at Dallas, Richardson, TX 75083
{gnp021000, chuanjun, praba}@utdallas.edu

Abstract. Content-based retrieval of 3D human motion capture data has significant impact in different fields such as physical medicine, rehabilitation, and animation. This paper develops an efficient indexing approach for 3D motion capture data, supporting queries involving both sub-body motions (e.g., *Find similar knee motions*) as well as whole-body motions. The proposed indexing structure is based on the hierarchical structure of the human body segments consisting of independent index trees corresponding to each sub-part of the body. Each level of every index tree is associated with the weighted feature vectors of a body segment and supports queries on sub-body motions and also on whole-body motions. Experiments show that up to 97% irrelevant motions can be pruned for any kind of motion query while retrieving all similar motions, and one traversal of the index structure through all index trees takes on an average 15 μsec with the existence of motion variations.

1 Introduction

Several scientific applications, especially those in medical and security field, need to analyze and quantify the complex human body motions. Sophisticated motion capture facilities aid in representing the complex human motion in the 3D space. The 3D human joint data from motion capture facility helps in analysis and comparison of the motions.

Focus of the Paper: Our main objective of this paper is to find similar 3D human motions by constructing the indexing structure which supports queries on sub-body motions in addition to whole-body motions. We focus on content-based retrieval for the sub-body queries such as *Find similar shoulder motions*, *Find similar leg motions* etc., or more regular query on whole body such as *Find similar walking human motion*. Some of the major challenges in indexing large 3D human motion databases are:

- 3D motions are multi-dimensional, multi-attribute and co-related in nature; associated segments of one sub-body (e.g. hand) must be processed always together along every dimension.
- Human motions exhibit huge variations in speed for similar motions as well as in directionality.

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 386–396, 2007.
© Springer-Verlag Berlin Heidelberg 2007

Proposed Approach: In our approach, we represent the positional information of different human body joints in a motion as a feature point. Using these feature points, a composite index structure for 3D human motions comprising five index trees is constructed. Each of the five index trees corresponds to one sub-body part (torso, left hand, right hand, left leg, and right leg). Each level of the index tree is designated to a joint of the corresponding sub-body part depending on the hierarchical structure of the human body joints. The mapped feature points of the joint associated with the level are grouped together. Each level prunes the irrelevant motions for the given query with respect to associated joint. And finally, each index tree gives the relevant motions for the query with respect to corresponding sub-body part. The output of relevant motions is then ranked using a similarity measure. For the whole motion query, the outputs from all index trees are merged and then ranked to get the most relevant motions for the whole-body query.

2 Related Work

In recent years, some approaches have been proposed on motion-retrievals from motion database. [12] constructed qualitative features describing geometric relations between specified body points of a pose and uses these features to induce a time segmentation of motion capture data streams for motion indexing. For each query a user has to select suitable features in order to obtain high-quality retrieval results. In [10], the authors cluster motion poses using piecewise-linear models and construct indexing structures for motion sequences according to the transition trajectories through these linear components. Similarly, posture features of each motion frame are extracted and mapped into a multidimensional vector in [3] for motion indexing. In [9], the authors use a hierarchical motion description for a posture, and use key-frame extraction for retrieving the motions.

Keogh et al. [6] use bounding envelops for similarity search in one attribute time series under uniform scaling. The iDistance [14] is a distance-based index structure, here dataset is partitioned into clusters and transformed into lower dimension using similarity with respect to reference point of cluster. MUSE [13] extends [14] where partitioning of dataset at each level of the index tree is based on the differences between corresponding principal component analysis (PCA).

3 3D Motion Index Structure Design

We need to extract the feature characteristics from the motion matrix; such that joints' motions are represented as entities in the low dimensional feature space (fd-space). When we map the entire matrix for the two walking motions(Figure 1(a, b)), the mapped feature vectors are as shown in Figure 1(e) (Figure 1(f) zooms components corresponding to leg segments). Now, we can also map only the sub-matrix corresponding to leg motion alone, as shown in

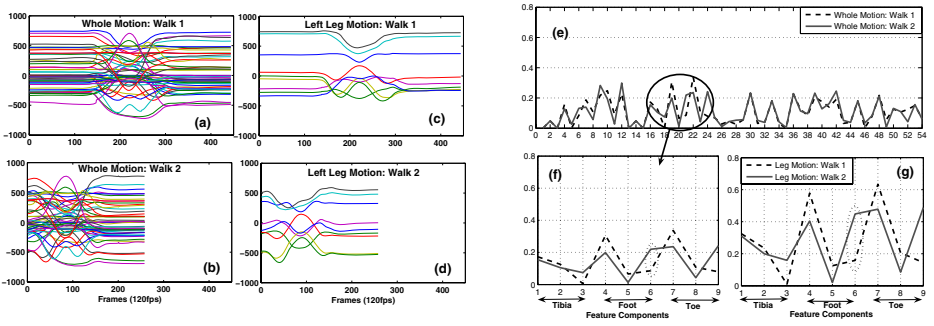
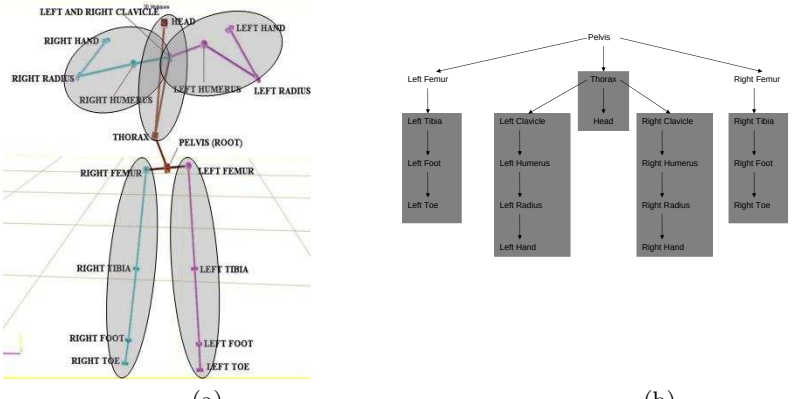


Fig. 1. (a)&(b): X, Y, Z trajectories of all segments for two similar walking motions. (c)&(d): Corresponding trajectories of only leg segments (tibia, foot, toe). (e) Feature components for whole body motions. (f) Feature components associated to only leg segments from(e). (g) Feature components for individual leg segments.

Figure 1(g). The differences between the same feature components are amplified and hence we can determine the similarity or dissimilarity between the two motions in a better way. The way this mapping of matrix/sub-matrix is done influences the query resolution. The details of mapping function are explained in Section 5.1.



(a) (b)

Fig. 2. (a)Segment structure for human body with five major sub-body parts. (b) Hierarchical tree structure of human Body segments.

This motivates us to design the index tree for each sub-body part depending on the hierarchical structure (Figure 2(b)) of the body segments (Figure 2(a)) so that we can resolve both sub-body motion as well as whole body motion queries efficiently. This structure consists of five branches with pelvis segment as the root. This human body structure inspires us to have a composite index structure consisting of five index trees corresponding to each branch.

4 Constructing Sub-body Index Trees

On mapping the joint data in 3D feature space, we can index these mapped points by constructing a corresponding index tree. Most mapping functions in literature [5], [2], [1] provide only similarity measures, i.e. they are not metrics. Due to the non-metric characteristic of the mapping functions available, it is difficult to *strictly* rank similar motions for a given query. Hence, it is better to retrieve the set of motions that lie within a threshold distance from the query motions feature point.

Now, let us consider two correlated joints such as tibia and foot. The movement of foot joint is constrained by or related to the tibia joint. This implies that for retrieving similar leg motions, we first retrieve the group of similar tibia motions, and only retrieved motions are considered for finding similar foot motions, and finally for similar toe motions. This leads us to the index tree structure for leg part of the body as shown in Figure 3.

The number of nodes constructed in Level j are equal to total number of groups present inside the nodes of immediate higher level (i.e. Level $j-1$). Each node has a parent in form of group in immediate high level. In each node, joint feature vectors corresponding to Level j are mapped in 3D-indexed space. But, the patterns present only in parent group are mapped as a 3D-point inside the node.

A node of the index tree has the following structure,

$$\begin{aligned} N &: (G_1, G_2, \cdots, G_e) \\ G_i &: (R, S, C, child - pointer) \end{aligned} \tag{1}$$

A node N consists of e groups of mapped points $G_1, \cdots, G_e$ formed by grouping the feature space. Each entry G_i consists of bounding hyper-rectangular region R, S is a set of n pattern identifiers whose mapped feature points are present in R and $child - pointer$ is a pointer to the node in the next level of an index tree. C is the centroid of the group G_i.

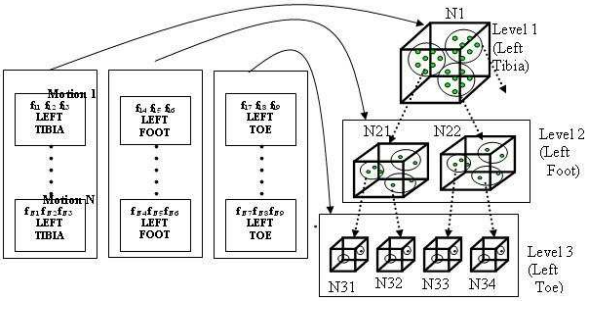


Fig. 3. Construction of hierarchical index tree for leg segments

An example: The construction of the left leg index tree is illustrated in Figure 3. The tibia feature points mapped in indexed space forms the root node

N1 in the Level 1 of left leg index tree. Let us assume that the indexed space is clustered to get three groups G1, G2 and G3, each containing feature points for similar motions. Every group G1, G2, and G3 becomes parent of node N21, N22 and N23 in Level 2 respectively. In node N21, foot feature points of all motions present in G1 are mapped in foot indexed space, which is further partitioned. This process is repeated for all nodes in Level 2. And the construction of the index tree is continued for last level in similar fashion.

4.1 Index Tree Operations

Pattern Insertion: Let us consider the insertion of a new motion in the left arm index tree. For this new motion, let f_x be the feature vector for left arm motion. Let $(f_{x(1-3)})$ feature point mapped in feature space corresponds to clavicle segment, $[f_{x(4-6)}]$ to humerus segment, $[f_{x(7-9)}]$ to radius segment and $[f_{x(10-12)}]$ to hand segment.

Using the threshold δ_c on each component of the feature vector of new inserting pattern, say for instant f_{xc}, we get a range defined as follows,

$$[f_{xc}] \Longrightarrow [f_{xc} - \delta_c, f_{xc} + \delta_c] \tag{2}$$

In first level of arm index tree, we map the new motion of clavicle segment in feature space. So, the new pattern's clavicle segment feature vector becomes a hyper-rectangular region. The dimension of this region is $H_x = [(f_{x1} - \delta_1, f_{x2} - \delta_2, f_{x3} - \delta_3),\ (f_{x1} + \delta_1, f_{x2} + \delta_2, f_{x3} + \delta_3)]$. If this region overlaps with multiple groups inside node, we store new pattern identifier p_x in S structure of each overlapped group. The following routine shows the insertion procedure in $Node$,

The patterns in the next level will be inserted only in child nodes of the overlapped groups. The insertion procedure is same but the thresholds and H_x will change depending on the variations in components of similar feature vectors associated with next level.

Pattern Search: A query search can be very simple: find a group in a node whose boundaries covers the query feature vector or if not, the nearest group to query vector. This group will have copies of all the similar motions from neighboring groups. So there is no need to traverse multiple groups. The query is traversed forward to the corresponding child node pointed by the overlapped or nearest group. When a leaf node is reached, all the motion identifiers included in that leaf node are returned.

Ranking the similar motions: After the index tree has been searched for query, the majority of irrelevant motions should have been pruned. To find out most similar motions to given query we need to rank them in order of similarity. A similarity measure [7] shown in equation(3) can be used to compute the similarity of the query and all returned motions, and the motions with highest similarity has the highest rank or most similar to the query.

$$\Psi(Q, P) = \frac{1}{2} \sum_{i=1}^{k} ((\sigma_i / \sum_{j=1}^{n} \sigma_j + \lambda_i / \sum_{j=1}^{n} \lambda_j) |u_i \cdot v_i|) \tag{3}$$

where σ_i and λ_i are the i^{th} eigenvalues corresponding to the i^{th} eigenvectors u_i and v_i of square matrices of Q and P, respectively, and $1 < k < n$.

4.2 Handling Whole-Body Queries

In some cases, queries on the whole body motions would be more meaningful than queries on sub-body motions. For example, if we want to find motions similar to certain swimming stroke whole body considered together would be more useful.

The five index tree returns the respective similar motions for the sub-body parts. To get similar whole body motion, we need to merge these outputs by taking the intersection of all returned pattern sets. The common patterns in all output sets from index trees will form the answer for the whole body query. The ranking of similarity patterns is again decided by similarity measure using equation (3) with $n = 48$ (all segments).

5 Index Structure Implementation

With the global positions, it becomes difficult to analyze the motions performed at different locations and also in different directions. Thus, we do the local transformation of positional data for each body segment by shifting the global origin to the pelvis segment because it is the root of all body segments. The segments included in the five index trees are highlighted in Figure 2(b).

5.1 Mapping Function for Joint Matrices

An appropriate mapping function is required to map 3D motion joint matrices into 3D feature points in the feature space. In our implementation, we used the linearly optimal dimensionality reduction technique SVD [5] for this purpose. For any $m \times 3$ joint matrix A, the SVD is given as follows,

$$A^{m\times 3} = U^{m\times m} \cdot S^{m\times 3} \cdot V^{3\times 3} \tag{4}$$

S is a diagonal matrix and its diagonal elements are called singular values. And columns of V matrix are called right singular vectors. We add up the three right singular vectors weighted by their associated normalized singular values to construct the features for a joint motion as follows:

$$f_c = \sum_{i=1}^{3} (w_i \cdot v_{ci}) \tag{5}$$

where $w_i = \frac{\rho_i}{\sum_{j=1}^{3} \rho_j}$, $\sum_{i=1}^{3} w_i = 1$, $c = \{1, 2, 3\}$, and $[\rho_1, \rho_2, \rho_3]$ is singular value vector and v_{ci} is the c^{th} component of the i^{th} right singular vector and w_i is the normalized weight for the i^{th} right singular vector. The weighted joint feature vector of length 3 represents the contribution of the corresponding joint to the

motion data in 3D space and also captures the geometric similarity of motion matrices.

The feature vectors for joints are represented as mapped points in feature space. For the similar feature vectors, corresponding mapped points will be close to each other. This creates a need to do grouping of such points which can simplify the similarity search.

5.2 Node Grouping Approach

Several approaches have been suggested in the literature for grouping data [8], [4], [15]. In our work, we formed grouping in all nodes of index tree using the hierarchical, self-organizing clustering approach [11].

Hierarchical, Self-organizing Clustering Approach: In this approach, a node is spliced into two groups if heterogeneity is greater than defined threshold. Heterogeneity(H_t) is a measure to calculate the distribution of the mapped points in a given group. Scattered points in group give high heterogeneity value and closely distributed points give low heterogeneity value. The threshold(T_h) is determined by product of this measure and heterogeneity scaling factor α.

$$H_t = \sum_{j=1}^{D} \frac{\| x_j - C \|_2}{| D |} \qquad T_h = H_t * \alpha$$

where C is the 3D-centroid of the node containing total D patterns and x_j is the mapped 3D coordinates of patterns inside group. We iteratively do the splicing on groups until heterogeneity values of all formed groups are below threshold T_h. These groups become the parent of the nodes in the next level.

Similarly, other index trees are constructed to build whole indexing structure. The (left/right) hand index tree has Level 1 associated with clavicle segment to Level 4 associated with hand segment.

Overcoming Space Partitioning problem: The DGSOT is a space partitioning approach. Due to different variations in performing similar motions, the corresponding mapped points may fall into different groups after clustering causing false dismissals in resulting output of similar motions for the query. Hence, the sizes of the group must be re-adjusted by some "threshold" to include most of the similar motions from the neighboring groups.

To solve the space partitioning problem, we capture the uncertainty of differences in similar motions for a joint in different feature dimension using standard deviation. In a database of M motions, we have E sets of pre-determined similar motions. Let $simDev_c$ be the standard deviation of the differences between similar motions for the c^{th} feature component.

Using $simDev_c$, we get the threshold δ_c to enlarge the group along the c^{th} dimension of the feature space as follows,

$$simTolerance_c = \delta_c = \epsilon * simDev_c \tag{6}$$

$simTolerance_c$ (δ_c) is a final threshold to enlarge the group along c^{th} dimension. ϵ is an input parameter which varies in the range of $0.2 - 1$. The larger ϵ gives high threshold and a group is enlarged to involve more feature points from neighboring groups along all feature dimensions. As a result, the pruning efficiency goes on decreasing and rate of false dismissals falls.

6 Performance Analysis

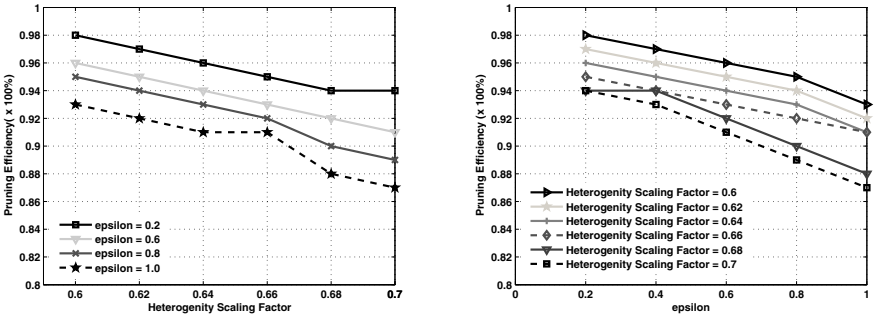
The human motion data was generated by capturing human motions using 16 high resolution Vicon cameras connected to a data station running Vicon iQ software. Our test bed consists of 1000 human motions, performed by 5 different subjects.

Let N_{pr} be the number of irrelevant motions pruned for a given query by the index tree, and N_{ir} be the total number of irrelevant motions in the database. We define the pruning efficiency $\mathcal{P}$ as

$$\mathcal{P} = \frac{N_{pr}}{N_{ir}} \times 100\% \tag{7}$$

6.1 Pruning Efficiency

For each experiment, we issued 1000 queries to calculate the average pruning efficiency for the indexing tree as shown in Figure 4(a) and Figure 4(b).



(a) (b)

Fig. 4. (a) Pruning efficiency for different heterogeneity scaling factors. (b) Pruning efficiency for different input parameters (ϵ).

As we go on increasing α, the heterogeneity threshold goes on increasing, and more pattern-corresponding feature points get accumulated in the same partition, which reduces the pruning power due to inclusion of some irrelevant motions. There is a steady increase in the pruning efficiency as we decrease the heterogeneity scaling factor (Figure 4(a)). The effect on pruning efficiency was

also studied by keeping α constant and varying the input parameter ϵ (Figure 4(b)). As we increase ϵ, the $simTolerance(\delta)$ for all components goes on increasing as a result the pruning efficiency goes on decreasing.

For whole body motion query, average pruning efficiency achieved is 98% where we take "intersection" of the set of relevant motions from five index trees.

6.2 Recall

When a motion query is given to an index tree, ideally it should return all similar motions. However, in practice, resolution of some queries may have some irrelevant motions due to the non-metric nature of the similarity measures used as well as the variations in performing similar motions. Figure 5 shows the average recall for different configurations of the index tree. As input parameter ϵ goes on increasing, the average return of similar patterns for the given query (i.e. hits) goes on increasing.

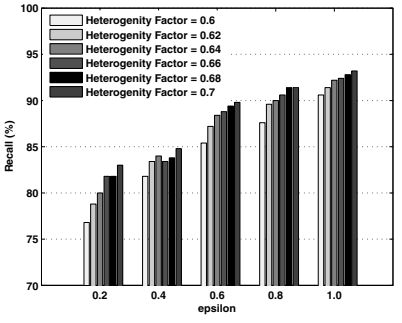


Fig. 5. Recall for different configurations of the index tree

6.3 Comparison with MUSE

The MUSE indexing structure works very well on indexing synthetic hand gesture motions generated using an instrumented device called CyberGlove [13]. Since hand gesture motions have multi-attributes and different variations just like captured 3D human motions, MUSE seems to be the most suitable indexing structure published so far for multi-attribute motion data. For performance comparison, we applied MUSE on our database of 3D human motions. The MUSE structure was constructed using 3 Levels. By querying 1000 3D motions, the pruning efficiency achieved was 5.5%, as compared to 97% of our index tree structure. Also, the average computational time required per query was 0.3 seconds. In our case, for the same set of queries, the average computational time per query is 15 μsec. The lower bound defined by MUSE for pruning irrelevant motions [13] is not tight enough for 3D human motions.

7 Conclusions and Discussions

This paper considered content-based motion querying on a repository of multi-attribute 3D motion capture data. We proposed a composite index structure that maps on to the hierarchical segment structure of the human body. This composite structure comprises five independent index trees corresponding to the five identified body parts: body/thorax, two arms & two legs. In each of these five index trees, a tree level is assigned to one segment feature vector. At each level, similar feature points inside all nodes are grouped together to increase the pruning power of the index tree.

We tested our prototype using a database of approximately 1000 human motions. Our experiments show that up to 96∼97% irrelevant motions can be pruned for any kind of motion query while retrieving all similar motions, and one traversal of the index structure through all index trees takes on an average 15 μsec. Finally, our approach is also applicable to other forms of multidimensional data with hierarchical relations among the attributes.

References

1. R. Agrawal, C. Faloutsos, and A. Swami. Efficient similarity search in sequence databases. In *Proc. 4th Conference on Foundations of Data Organization and Algorithms*, pages 69–84, October 1993.
2. K.-P. Chan and A. W.-C. Fu. Efficient time series matching by wavelets. In *Proc. 15th International Conference on Data Engineering*, pages 126 – 133, March 1999.
3. S.-P. Chao, C.-Y. Chiu, J.-H. Chao, Y.-C. Ruan, and S.-N. Yang. Motion retrieval and synthesis based on posture features indexing. In *Proc. Fifth International Conference Computational Intelligence and Multimedia Applications*, pages 266–271, September 2003.
4. C. Ding and X. He. K-means clustering via principal component analysis. In *ICML '04: Proceedings of the twenty-first international conference on Machine learning*, page 29, New York, NY, USA, 2004. ACM Press.
5. G. H. Golub and C. F. V. Loan. *Matrix Computations*. The Johns Hopkins University Press, Baltimore, Maryland, 1996.
6. E. Keogh, T. Palpanas, V. B. Zordan, D. Gunopulos, and M. Cardle. Indexing large human-motion databases. In *Proc. 30th VLDB Conference*, pages 780–791, Toronto, Canada, 2004.
7. C. Li and B. Prabhakaran. A similarity measure for motion stream segmentation and recognition. In *Proc. MDM/KDD, The sixth International Workshop on Multimedia Data Mining*, Chicago, IL USA, August 2005.
8. B. Liu, Y. Xia, and P. S. Yu. Clustering through decision tree construction. In *CIKM '00: Proceedings of the ninth international conference on Information and knowledge management*, pages 20–29, New York, NY, USA, 2000. ACM Press.
9. F. Liu, Y. Zhuang, F. Wu, and Y. Pan. 3D motion retrieval with motion index tree. *Computer Vision and Image Understanding*, 92:265–284, June 2003.
10. G. Liu, J. Zhang, W. Wang, and L. McMillan. A system for analyzing and indexing human-motion databases. In *Proc. 2005 ACM SIGMOD International conference on Management of data*, 2005.

11. F. Luo, L. Khan, F. Bastani, I.-L. Yen, and J. Zhou. A dynamically growing self-organizing tree (dgsot) for hierarchical clustering gene expression profiles. *Bioinformatics*, 20:2605–2617, May 2004.
12. M. Muller, T. Roder, and M. Clausen. Efficient content-based retrieval of motion capture data. *ACM Transactions on Graphics (TOG)*, 24:677–685, 2005.
13. K. Yang and C. Shahabi. Multilevel distance-based index structure for multivariate time series. In *TIME*, Burlington, Vermont, USA, 2005. IEEE Computer Society.
14. C. Yu, B. C. Ooi, K.-L. Tan, and H. V. Jagadish. Indexing the distance: An efficient method to knn processing. In *Proc. VLDB '01*, pages 421–430, San Francisco, CA, USA, 2001. Morgan Kaufmann Publishers Inc.
15. T. Zhang, R. Ramakrishnan, and M. Livny. Birch: an efficient data clustering method for very large databases. In *SIGMOD '96: Proceedings of the 1996 ACM SIGMOD international conference on Management of data*, pages 103–114, New York, NY, USA, 1996. ACM Press.

Similarity Searching Techniques in Content-Based Audio Retrieval Via Hashing

Yi Yu, Masami Takata, and Kazuki Joe

Graduate School of Humanity and Science, Nara Women's University
Kitauoya nishi-machi, Nara 630-8506, Japan
{yuyi,takata,joe}@ics.nara-wu.ac.jp

Abstract. With this work we study suitable indexing techniques to support efficient content-based music retrieval in large acoustic databases. To obtain the index-based retrieval mechanism applicable to audio content, we pay the most attention to the design of Locality Sensitive Hashing (LSH) and the partial sequence comparison, and propose a fast and efficient audio retrieval framework of query-by-content. On the basis of this indexable framework, four different retrieval schemes, LSH-Dynamic Programming (DP), LSH-Sparse DP (SDP), Exact Euclidian LSH (E^2LSH)-DP, E^2LSH-SDP, are presented and estimated in order to achieve an extensive understanding of retrieval algorithms performance. The experiment results indicate that compared to other three schemes, E^2LSH-SDP exhibits best tradeoff in terms of the response time, retrieval ratio, and computation cost.

1 Introduction

Content-based audio similarity retrieval is not only a very promising research topic, but also one of the main problems in multimedia information processing. Audio sequence data is usually tedious due to the high dimensionality of the features, which makes it inconvenient to utilize the potential content-based information retrieval on the Internet or personal media devices. To access a huge mass of audio information efficiently, it is necessary to explore the audio information, facilitate the management of audio data and serve multimedia applications.

The design of an index-based similarity retrieval system poses the following challenges: a) Characterize a corpus of acoustic objects with a corpus of relevant features. b) Represent audio data sequences as a searchable symbol that can be indexed. c) Locate the desired music segments with a given query in the format of acoustic sequences within the acceptable time. It is observed that the whole procedure is typically a time-consuming operation. In order to expedite the searching procedure many researchers have reported various indexing structures in the study of audio retrieval, for example, hierarchical structure[1], R-trees [2], M-trees[3], KD-trees[4], Locality Sensitive Hashing (LSH)[5]. Retrieving an enormous multimedia database is a challenging task in the content discovery field. One of the basic problems in large-scale audio retrieval is to design appropriate metrics and algorithms to avoid all pairwise comparisons of feature sequences.

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 397–407, 2007.
© Springer-Verlag Berlin Heidelberg 2007

Here we will initiate an interesting direction in the study of effective comparisons of the massive acoustic sequences by taking into account reconstruction of feature sequences that can respond to approximate membership queries. We care about indexing structure on acoustic sequences and reorganization of fragmented sequence. The process of audio content retrieval is carried out according to the following procedure: given a corpus of n musical reference pieces, actually, which can be represented by feature sequences $\mathrm{R} = \{\mathrm{R}_i, 1 \leq i \leq n\}$, (or the frames of all musical reference sequences $R = \{r_{i,j} : r_{i,j} \in R_i,\ 1 \leq i \leq n, 1 \leq j \leq |R_i|\}$, where $r_{i,j}$ is the j^{th} spectral feature of the i^{th} reference piece), are located in a high-dimension Euclidean space (U, d) with a distance metric d. A query sequence is also broken into the format of frames $q_1, q_2, ..., q_Q$. Exact Euclidean LSH (E^2LSH) is used to effectively pick up reference candidates resembling the query sequence. In terms of locality sensitive function $H(\cdot)$, each query frame q_m filters off with a high probability some resemblances, $S_{i,m} = \{r_{i,j} : r_{i,j} \in H(q_m),\ d(q_m, r_{i,j}) \leq \delta\}$, stored in the audio buckets $H(q_m)$. Then the partial audio sequences of the i^{th} reference in the union $\bigcup_m S_{i,m}$ are reorganized and compared with the query by the proposed Sparse Dynamic Programming (SDP). Based on such ideas we present a novel framework to perform audio similarity index and retrieval, and provide scalable content-based searchability.

The rest of the paper is organized as follows: section 2 provides the background of the concepts and notations related to this work, and shows our contribution. Section 3 presents the framework of audio indexing and describes content-based retrieval schemes in detail. Section 4 lists the simulation environment and analyzes the experiment results. Finally Section 5 concludes the paper.

2 Background and Related Work

As a general nonlinear alignment method, DP was originally exploited in the speech recognition to account for tempo variations in speech pronunciation. Many researchers [6][7][8]have studied DP and also applied DP or optimized DP in content-based music information retrieval to match the query input against the reference melodies in the database.

LSH is an index-based data organization structure proposed to improve the scalability for retrieval over a large database, which is an essential spatial access method-constructs some locality sensitive hashing functions to performing indexing in parallel in Euclidean space[9]. And it plays an important part in various applications, e.g., database access and indexing [10], data compression and mining, multimedia information retrieval[5][11]. In particular, the features of the objects are represented as the points-form in the high dimensional space and a distance metric is adopted to judge whether two multimedia objects are similar (such as audio [5], video[11], image[12]). Furthermore, the most interesting thing is that some researchers bring LSH-based data structure into the field of long sequences similarity comparison [5][10].

E^2LSH [13] is to solve Approximate Near Neighbors problem in a high dimensional Euclidean space. It enhances LSH to make it more efficient for the

retrieval with the very high dimensional feature. It performs locality sensitive dimension reduction to get the projection of the feature in different low-dimension sub-spaces. With multiple hash tables in parallel, the retrieval ratio can be guaranteed meanwhile the retrieval speed is accelerated. If two features (q, r) are very similar they will have a small distance $\|q - r\|$ and hash to the same value and fall into the same bucket with a high probability. If they are quite different they will collide with a small probability. A family $H = \{\mathrm{h} : \mathrm{S} \rightarrow \mathrm{U}\}$ is called locality sensitive, if for any features q and r,

$$p(t) = P_{rH}\left[h(q) = h(r) : \|q - r\| = t\right] \tag{1}$$

is a strictly decreasing function of t. That is, the collision probability of features q and r is diminishing as their distance increases. The family H is further called (R, cR, p_1, p_2) $(c > 1,\ p_2 < p_1\ )$ sensitive if for any $q, r \in S$

$$\begin{aligned} &if\ ||q - r|| < R, \quad P_{rH}[h(q) = h(r)] \geq p_1 \\ &if\ ||q - r|| > cR,\ P_{rH}[h(q) = h(r)] \leq p_2 \end{aligned} \tag{2}$$

A good family of hash functions will try to amplify the gap between p_1 and p_2 .

A distribution D over $\Re$ is called p-stable, if there exists p so that for any n real numbers $\bar{v} = (v_1, v_2, ..., v_n)^T$ and i.i.d. random variables $x_1, x_2, ..., x_n, x$ with distribution D, the variable $f_{\bar{v}}(X) = \sum_{i=1}^{n} v_i x_i$ has the same distribution as the variable $(\sum_i |v_i|^p)^{1/p} x$. In E^{2}LSH, a p-stable distribution is adopted in locality sensitive dimension reduction. Each $\bar{v}$ generates a single output. Then $f_V(\cdot)$ with $V = (\bar{v}_1, \bar{v}_2, ..., \bar{v}_m)$ generates an m-dimension vector. And q and r in Eq.(1-2) are replaced with $f_V(q)$ and $f_V(r)$ respectively. We would like to filter some approximate frames in the database that are similar to the query frames. With E^{2}LSH, a single query frame generates a unique hash value in a hash instance and only matches several possible items similar to itself. Accordingly, E^{2}LSH can help to avoid matching a query frame against all frames of a reference melody.

We report retrieval mechanisms of index-based audio sequences, data organizational structure of audio frames as well as recreation and comparison of the fragmented audio sequences. The extensive comparison among the presented four schemes (LSH-DP, LSH-SDP, E^{2}LSH-DP, E^{2}LSH-SDP) shows that the optimal combination-E^{2}LSH-SDP-outperforms the others. We also give the suitable parameters to obtain the best performance. Compared with [13] some significant differences are that the proposed scheme is suitable for time-varying representation of spectrum sequences; the frames selected by the hash tables are reorganized for sequences comparison. Similar to [2][5][14], our retrieval scheme adapts the spectral feature. However, there also are significant distinctions: i) E^{2}LSH is used in our scheme as a filter structure to pick up the reference features similar to the queries. ii) We focus on the sequences reconstruction and efficient comparison. In contrast with the conventional DP, most of the pairwise comparison is avoided in our Sparse DP (SDP) scheme since the match percent usually is very low.

3 Design of Index-Based Audio Content Retrieval

We focus on providing fast and efficient content-based retrieval mechanisms of searching audio sequences data over a large audio sequences database by utilizing a suitable indexing structure. Our retrieval system allows a user to take a fragment of the query song as input, then performs content-based similarity retrieval, and finally returns melodies similar to this query fragment.

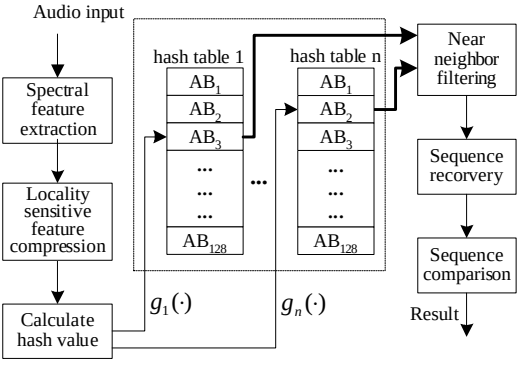


Fig. 1. An index-based audio retrieval framework. It mostly consists of locality sensitive feature compression, parallel hash tables, approximate near neighbor search, sequence recovery and comparison.

3.1 The Basic Framework

This work presents a novel indexable framework for music information retrieval, which involves two main parts: building indexable data structure of acoustic sequences and matching the recovered acoustic sequences. The aim of E^2LSH/LSH is to establish an effective data structure for acoustic-based music information. The aim of SDP/DP is to match the recovered acoustic sequences and obtain the answer closest to the query. Hence, we propose four different retrieval schemes, LSH-DP, LSH-SDP, E^2LSH-DP, and E^2LSH-SDP. Based on this general framework we want to evaluate these four schemes to solve the problem of scalable audio content retrieval and select the best tradeoff according to the response time, retrieval ratio, and comparison cost. The details are discussed in the experiment parts.

We begin by introducing the basic query-by-content framework of acoustic-based music information retrieval. This framework is shown in Fig.1 and its main procedure is summarized as follows: For each frame, its high-dimensional spectral feature, Short Time Fourier Transform (STFT), is calculated. The spectral features of all the reference melodies in form of searchable symbols are stored in the hash tables according to their respective hash values. E^2LSH/LSH provides an effective organization of the audio features, gives a principle to store the fragmented audio sequences, and facilitates an efficient sequences reconstruction. With each frame in the query sequence, the candidate symbols are searched

in the hash table. Then these decomposed frames are recovered and arranged into new spectral sequences and the SDP/DP algorithm is employed to perform an accurate similarity comparison between the query sequence and the partial reference sequences.

3.2 Frame Organization in the Hash Table

Acoustic-based music representations of a large dimension can yield the nice retrieval performance but always with high computational cost. So it is a very important task that the retrieval system can guarantee a quick response time and a low computational burden. we believe that a nice data organization can help to effectively raise the quality of retrieval mechanism.

when we filter audio sequences data with E^{2}LSH/LSH in the preprocessing, an audio sequence is first fragmented to frames. For each frame, STFT is calculated as the spectral feature, and regarded as a searchable symbol in Euclidean space. When LSH is adopted, a feature is directly quantified and its hash value for each hash table is calculated independently. When E^{2}LSH is used, a high-dimensional feature X is first projected to a low-dimension sub-feature $f_V(X)$ with the p-stable random array V. The p-stable array of each hash table is independently taken from the standard normal distribution and the sub-feature is of dimension 8 in our work. Then the sub-feature is quantified and its hash value is calculated according to the details in Section 2. As was stated in[13], multiple hash tables increase the retrieval ratio of E^{2}LSH. Therefore we construct several hash tables, each containing all the frames of the references. The kth hash instance has its own p-stable random array V_k, and an equivalent hash function g_k involving the effect of $f_{V_k}(X)$ and the LSH function $H_k(\cdot)$. In the following, $g_k(\cdot)$ also means an Audio Bucket (AB) storing all the frames in form of searchable symbols with the same hash value. Its meaning is obvious from the context.

The jth spectral feature of the ith reference melody, $r_{i,j}$, is stored in $g_k(r_{i,j})$, a bucket of the kth hash table. Its music number i and the corresponding time offset j are recorded together with the feature, in order to provide facilities for reconstruction of the partial acoustic sequences after the filtering stage.

3.3 Filtering Audio Features with E^{2}LSH/LSH

A group of E^{2}LSH/LSH hash tables are simple and effective data storage structures, which can be accessed randomly and implemented in parallel. They can also make a contribution to perform partial matching audio sequences. In the query stage, a sequence of query frames $q_1, q_2, ..., q_Q$ is used to search and find the reference music closest to the query. With a query frame q_m, the candidate reference frames in the bucket of the kth hash table, $g_k(q_m)$, can be obtained. This AB contains all the frames matching to the same hash value. Though it is probable that the resemble frames lie in the AB, many other non-similar frames also exist due to the limited hash table size. It is necessary to remove these non-similar frames so as to reduce the post computation. Therefore we define a distance function that can quantify the similarity degree, $d(X, Y) = \|X - Y\|_2 / \|X\|_2 \cdot \|Y\|_2$,

the normalized Euclidean distance. Among the indexed frames in the candidate bucket, the ones with a distance to q_m greater than δ are filtered out and discarded by Eq. 3, namely, we would like to retain such frames that lie within the ball centered at q_m with a radius δ.

$$S_{k,i,m} = \{r_{i,j} : r_{i,j} \in g_k(q_m),\ d(f_{V_k}(q_m), f_{V_k}(r_{i,j})) \leq \delta\} \tag{3}$$

The union $S_{k,i} = \underset{m}{\cup} S_{k,i,m}$ gives the candidate features of the ith reference obtained from the kth hash table for the whole query sequence. Since $f_{V_k}(X)$ has a much smaller size than X, the filtering stage of E^{2}LSH can be greatly accelerated in contrast to LSH, as is verified by the simulation.

3.4 Matching of Recreated Audio Sequences

To improve the retrieval ratio, several hash tables are used. The total candidates of the ith reference are obtained from all the hash tables as $S_i = \underset{k}{\cup} S_{k,i}$. These features are reorganized according to the timing relation and form a partial sequence. The obtained frames in S_l are reorganized in the ascending order of their time offset $j_1, j_2, ..., j_R$. Then the query $q_1, q_2, ..., q_Q$ is matched against each reference to find the desired melody with the DP method similar to[7], or the proposed SDP scheme.

In our SDP scheme, the sequence comparison is different from the conventional DP method: we directly utilize the distance calculated in the filtering stage by filling the distance to a DTW table. For each matched pair $< q_m, r_{l,j_n} >$ $(r_{l,j_n} \in S_{k,l,m})$, the reverse of its distance is used as the weight in the matching procedure.

$$w_k(q_m, r_{l,j_n}) = \min\{\delta / d(f_{V_k}(q_m), f_{V_k}(r_{l,j_n})),\ w_{\max}\} \tag{4}$$

The weight is no less than 1. With $w_{\max}$, an occasional perfect match of a single frame has less effect on the sequence comparison. On the other hand, if the pair $< q_m, r_{l,j_n} >$ does not exist, its weight is set to 0.The sequence comparison is to select the path that maximizes the total weight. Despite the absence of most

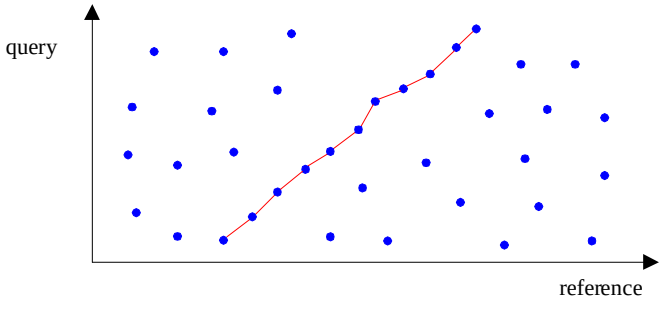


Fig. 2. Sequence comparison with SDP. Since most of the pairs are removed by the E^{2}LSH and the filtering stage, the remaining points are sparse in the DTW table.

matching pairs in Fig. 2, the matching path still contains as many points as possible. The maximum weight is found by the iteration in Eq. 5

$$W_l(m, j_n) = \sum_k w_k(q_m, r_{l,j_n}) + \max \begin{cases} W_l(m-2, j_{n-1}) \\ W_l(m-1, j_{n-1}) \\ W_l(m-1, j_{n-2}) \end{cases} \tag{5}$$

In this equation, the weight of the duplicates from different hash tables are recalculated since the recurrence of a single pair in different hash tables means that the pair is a suitable match with a high probability. For the l^{th} reference melody, its weight W_l is obtained by Eq. 6. Then by Eq. 7 the one with the maximum weight matches the query.

$$W_l = \max_{j_n} W_l(Q, j_n) \tag{6}$$

$$l = \arg \max_l W_l \tag{7}$$

4 Experiments and Results

The experiments have been carried out on the acoustic database with both monophonic and polyphonic melodies. Our music database(166) consists of 44 Chinese folks from 12-girl-band and 122 western instruments melodies. Each piece is segmented into 60-second-long melodic slip. 166 query samples are segmented into 6-8 seconds long. 44 quires corresponding to the Chinese folks are different versions compared with reference melodies to test the robustness of the proposed schemes. The other 122 queries are segmented from the reference melodies.

The melodies are in single-channel wave format, 16bit/sample, and the sampling rate is 22.05KHz. The direct current component is removed and the music is normalized with the maximum value equaling 1. The music is divided into overlapped frames. Each frame of music contains 1024 samples and the adjacent frames have 50% overlapping. Each frame is weighted by a hamming window, and further appended with 1024 zeros to fit the length of FFT. The spectrum from 100Hz to 2000Hz is used as the feature. Accordingly, each feature has the size 177. We use several hash instances, each having 128 entries. In our retrieval system, each melody in the database is accompanied with its feature sequence. When the system boots the hash tables are constructed in the memory. Therefore, utilization of LSH has no extra disk storage requirement, though it does require some memory to hold the hash tables.

4.1 Evaluation Metrics

For the purpose of providing a thorough understanding of our music retrieval mechanism, four different schemes are examined and compared on the basis of the general framework. In the experiment, we mainly consider three metrics that can evaluate every scheme roundly:

Matched percentage. Since LSH/E^{2}LSH is used to avoid pairwise comparison, the first question always touch on how much it can reduce the computation. It needs to be described by a list of parameters, including the following:

- N_{dm}: the number of directly matched pair with LSH/E^{2}LSH.
- N_{tm}: the number of total possible pairwise pair without LSH/E^{2}LSH.
- N_{rm}: the number of remaining frames of matched part in the desired reference melody after the filtering stage in LSH/E^{2}LSH.
- N_{mm}: the number of frames of the matched part in the desired reference melody under the conventional DP.

The ratio N_{dm}/N_{tm} is regarded as Roughly Matched Percentage (RMP) and the ratio Nrm/Nmm is defined as Valid Match Percentage (VMP). RMP reflects how much computation can be reduced while VMP affects the retrieval ratio. With a good design of the hash tables one will expect a low RMP and a high VMP.

Computation time. We will evaluate the filtering time between LSH and E^{2}LSH, the comparison time between DP and SDP. The total time is the sum of the time of calculating hash value for a query, the filtering time and the comparison time.

Retrieval ratio. In our experiment, each of the query pieces is used to retrieve the desired melody from the database. The number of correctly retrieved melodies over that of the total queries is defined as the retrieval ratio.

4.2 Matched Percentage

LSH/E^{2}LSH reduces most of the pairwise comparison. However, the features with the same hash value are not necessarily similar to the query frame. The filtering is done just after indexing the hash tables to keep only the near neighbors of the query. The filtering threshold δ in Eq. 3 plays an important role. It determines how many frames will remain, which in turn affects the computation and the retrieval ratio.

Table 1 shows VMP under different filtering threshold for LSH and E^{2}LSH. Of course a bigger δ leads to a higher VMP. But it also results in heavy computation. By selecting a suitable δ for the filtering stage, most of the unsimilar features are removed while VMP is maintained at a certain level. It is shown later that even a moderate VMP can achieve a high retrieval ratio. Hereafter, by default δ_{LSH} is set to 0.03 and δ_{E^2LSH} is set to 0.0075.

Fig. 3 reveals that the increase of hash tables only results in a little gain in VMP. Therefore, in the following, only three hash tables are used except especially pointed out.

4.3 Computation Time

LSH/ E^{2}LSH hash table construction is usually time-consuming. Fortunately, this is done before the actual query takes place. The hash value of the query is calculated just before retrieval. For a short query, this time is almost negligible.

Table 1. VMP under different filtering threshold (3 hash tables)

δ_{LSH}	0.01	0.02	0.03	0.04	0.05
VMP_{LSH}	0.113	0.255	0.400	0.537	0.669
δ_{E^2LSH}	0.0025	0.005	0.0075	0.01	0.0125
VMP_{E^2LSH}	0.123	0.240	0.363	0.472	0.573

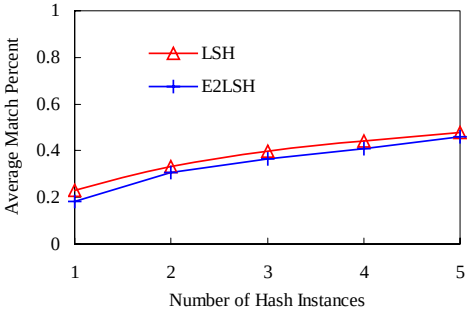
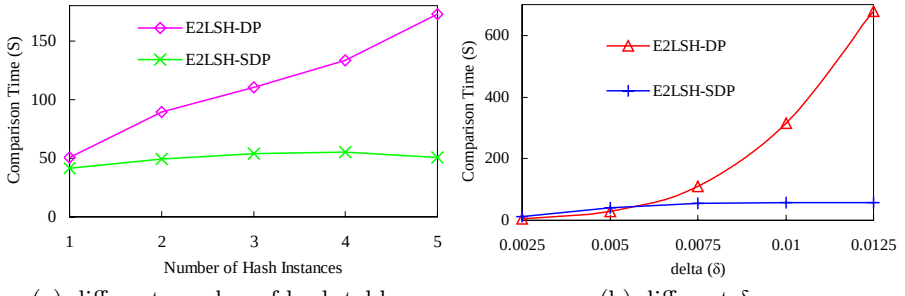


Fig. 3. VMP under different number of hash tables (δ_{LSH}=0.03, δ_{E^2LSH}=0.0075)



(a) different number of hash tables (b) different δ

Fig. 4. Comparison time in DP and SDP under different number of hash tables (δ_{LSH}=0.03, δ_{E^2LSH}=0.0075) or different δ (3 hash tables)

Two comparison methods are used in our schemes. Both the number of hash tables and the filtering threshold affect the post comparison. Fig. 4(a) shows the computation cost with respect to the number of hash instances and Fig. 4(b) shows the similar results under different filtering δ.

DP involves the calculation of pairwise feature distance among the remaining frames. However, its DTW table becomes smaller in case the few reference frames remain. In contrast, in SDP, the feature distance is taken from the filtering stage though the DTW is of full size. When few reference frames are matched, DP and SDP almost have the same retrieval speed. This occurs when there are few hash instances in Fig. 4(a) or the filtering threshold is low in Fig. 4(b). As the number of remaining frames increases, SDP has very obvious superiority over DP since

it avoids the calculation of feature distance and its comparison time approaches a steady value, which guarantees the worst retrieval time. Therefore SDP is preferred when there are more hash instances or the filtering δ is large.

To show the effect of hashing, Table 2 lists the total retrieval time consumed for all the queries under the different schemes. The conventional DP (without hashing) takes 3562.2s. In comparison, E^{2}LSH-SDP reduces the time to 83.4s, accelerating the retrieval speed by 42.7 times.

Table 2. The total retrieval time consumed under different schemes

Scheme	LSH-DP	LSH-SDP	E^{2}LSH-DP	E^{2}LSH-SDP	DP
Time(s)	258.8	213.34	139.5	83.4	3562.2

Table 3. Top-4 retrieval ratio with respect to the number of hash instances (δ_{LSH} =0.03, δ_{E^2LSH} =0.0075)

No. hash table	1	2	3	4	5
LSH-DP	1	0.98	1	1	1
LSH-SDP	1	1	1	1	1
E^{2}LSH-DP	0.98	1	1	1	1
E^{2}LSH-SDP	0.96	1	1	1	1

Table 4. Top-4 retrieval ratio of LSH/E^{2}LSH (3 hash tables)

δ_{LSH}	0.01	0.02	0.03	0.04	0.05
LSH-DP	0.88	1	1	1	1
LSH-SDP	0.94	1	1	1	1
δ_{E^2LSH}	0.0025	0.005	0.0075	0.01	0.0125
E^{2}LSH-DP	0.92	0.98	1	1	1
E^{2}LSH-SDP	0.96	1	1	1	1

4.4 Retrieval Ratio

E^{2}LSH was initially proposed to retrieve from a database by single feature. To acquire a high retrieval ratio, many hash tables are required, which increases the filtering time. In our scheme, E^{2}LSH is used for audio sequence comparison. Even though the retrieval ratio of a single frame is not very high, the following sequence comparison effectively removes the unsimilar sequences. Therefore, in our retrieval system, a few hash tables are sufficient to achieve a high retrieval ratio. Table 3 shows that the retrieval ratio is satisfactory with mere 3 hash tables. Table 4 shows the retrieval ratio under different filtering threshold δ. It is obvious that δ_{LSH} =0.03 andδ_{E^2LSH} =0.0075 are suitable thresholds since lower decreases retrieval ratio while larger increases the computation cost.

5 Conclusions

We have established an audio indexing framework for music information retrieval, proposed and evaluated four different retrieval schemes. Our retrieval system provides a quick response to the query by avoiding pairwise comparisons with every music piece in the entire audio database.We are building a larger database to evaluate the scalability of the proposed schemes in query-by-content audio retrieval.The retrieval speed is accelerated in three folds: LSH reduces the pairwise comparison; E^2LSH further reduces the filtering time by applying locality sensitive dimension reduction; SDP decreases the comparison time by avoiding pairwise distance computation in the sequence comparison stage. From the extensive simulation results it is obvious that E^2LSH-SDP is the optimal choice. We also show that even with only a few hash tables and relatively low filtering threshold, the retrieval with a sequence still has a high successful ratio.

References

1. Bertin, N. and Cheveigne,A.d.,:Scalable Metadata and Quick Retrieval of Audio Signals. ISMIR 2005, pp.238-244, 2005.
2. Karydis, I., Nanopoulos, A., Papadopoulos, A.N., Manolopoulos, Y.:Audio Indexing for Efficient Music Information Retrieval. MMM pp.22-29, 2005
3. Won, J.Y., Lee, J.H., Ku, K., Part, J.and Kim, Y.S.:A Content-Based Music Retrieval System Using Representative Melody Index from Music Databases. CMMR 2004
4. Reiss, J., Aucouturier, J.J., Sandler, M.:Efficient multidimensional searching routines for music information retrieval. ISMIR, 2001
5. Yang,C.: Efficient Acoustic Index for Music Retrieval with Various Degrees of Similarity. ACM Multimedia. (2002) 584–591
6. Tsai, W.H.,Yu, H.M., Wang, H.M.:A Query-by-Example Technique for Retrieving Cover versions of Popular Songs with Similar Melodies. ISMIR2005
7. Jang, J.S.R., Lee, H.R.:Hierarchical Filtering Method for Content-based Music Retrieval via Acoustic Input. ACM Multimedia pp. 401-410 2001
8. Dannenberg, R.B., Hu, N.:Understanding search performance in query-by-humming systems.ISMIR 2004, pp.236-241.
9. Indyk, P.,Motwani, R.:Approximate nearest neighbors: Towards Removing the Curse of Dimensionality. Proc. 30th ACM STOC, 1998
10. Jeremy, b.:Efficient Large-scale sequence comparison by locality seneitive hashing. Bioinformatics Vol.17, No.5, pp.419-428, 2001.
11. Hu, S.:Efficient Video Retrieval by Locality Sensitive Hashing. ICASSP 2005, pp.449-452, 2005
12. Indyk, P., Thaper, N.:Fast color image retrieval via embeddings.Workshop on Statistical and Computational Theories of Vision (at ICCV), 2003.
13. LSH Algorithm and Implementation (E2LSH) http://www.mit.edu/ andoni/LSH/
14. Yu, Y., Watanabe, C., Joe,K.:Towards a fast and Efficient Match Algorithm for Content-Based Music Retrieval on Acoustic Data.ISMIR 2005, pp.696-701 2005.

Fast Answering *k*-Nearest-Neighbor Queries over Large Image Databases Using Dual Distance Transformation

Yi Zhuang and Fei Wu

College of Computer Science, Zhejiang University, Hangzhou, 310027, P.R. China
{zhuangyi,wufei}@cs.zju.edu.cn

Abstract. To support fast *k*-NN queries over large image database, in this paper, we propose a novel *dual distance transformation* method called DDT. In DDT, all images are first grouped into clusters by the *k*-Means clustering algorithm. Then the *start*- and *centroid-distances* of each image are combined to obtain the uniform index key through its dual distance transformation. Finally the keys are indexed by a B^+-tree. Thus, given a query image, its *k*-nearest neighbor query in high-dimensional space is transformed into a search in a single dimensional space with the aid of the DDT index. Extensive performance studies are conducted to evaluate the effectiveness and efficiency of the proposed scheme. Our results demonstrate that this method outperforms the state-of-the-art high dimensional search techniques, such as the X-Tree, VA-file, iDistance and NB-Tree.

1 Introduction

With an explosively increasing number of images on the Internet, content-based image retrieval and indexing have become more important than ever before. However, due to the very known high dimensional curse problem [1], traditional indexing methods such as R-tree [2] and X-tree [4] only work well in low dimensional space. Therefore, the design of efficient indexing techniques for high-dimensional data is still an important and active research area.

In this paper, we propose a high-dimensional image indexing scheme based on the technique of *dual-distance-transformation*, called DDT, to support the progressive *k*-NN search over large image databases. Specifically, in DDT, all images are first grouped into clusters by using the well known *k*-Means clustering algorithm. Then, the dual distance transformation of each image is performed to obtain its index key which is indexed by a B^+-tree. Thus, given a query image V_q and *k*, the *k*-Nearest Neighbor search of V_q in a high-dimensional space is transformed into a search in a single dimensional space with the aid of the DDT index. We have implemented the DDT method and conducted an extensive performance study to evaluate its efficiency. Our results show that the proposed technique has superior to X-tree [4], VA-file [5], NB-Tree [7] and iDistance [8], which are also designed for indexing high-dimensional image databases. The primary contributions of this paper are listed as follows:

1. We propose a *dual-distance-transformation*-based indexing method to facilitate the highly efficient *k*-NN search for large image databases.

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 408 – 417, 2007.
© Springer-Verlag Berlin Heidelberg 2007

2. We propose a uniform index key expression method, called *the dual-distance-transformation-based method*, which nicely combines the dual distance metrics(i.e., *start-distance* and *centroid-distance*) together.
3. We give a cost model for the proposed indexing method to support the query optimization.

The rest of this paper is organized as follows. Related work is surveyed in Section 2. In Section 3, a *dual-distance-transformation*-based high-dimensional indexing scheme is proposed to dramatically improve the query performance of the *k*-NN search. In Section 4, a cost model for DDT is derived to facilitate the query optimization. In Section 5, we performed the extensive experiments to evaluate the efficiency and effectiveness of our DDT index mechanism. Conclusions are given in the final section.

2 Related Work

As we know, content-based image indexing belongs to the high-dimensional indexing problem. The state-of-art techniques can be divided into three main categories [1].

The first category is based on data and space partitioning, hierarchical tree index structure, for example, the R-tree [2] and its variants [3][4], etc. Although these methods generally perform well at low dimensionality, their performance deteriorates rapidly as dimensionality increases due to the “dimensionality curse".

The second category is to represent original feature vectors using smaller, approximate representations, e.g., VA-file [5] and IQ-tree [6], etc. The VA-file [5] accelerates the sequential scan by using data compression. For a search, although VA-file can reduce the number of disk accesses, it incurs higher computational cost to decode the bit-strings. The IQ-tree [6] is also an indexing structure along the lines of the VA-file.

The last category is to use a distance-metric-based method as an alterative direction for high-dimensional indexing, such as NB-tree [7], iDistance [8], etc. NB-tree [7] is a single reference point-based scheme, in which high-dimensional points are mapped to a single-dimension by computing their distance from the origin respectively. Then these distances are indexed using a B^+-tree on which we perform all subsequent operations. The drawback of NB-Tree is that it can not effectively prune the search region and especially when the dimensionality is becoming larger. iDistance [8] is proposed by selecting some reference points in order to further prune the search region so as to improve the query efficiency. However the query efficiency of iDistance relies largely on clustering and partitioning the data and is significantly affected if the choice of partition scheme and reference data points is not appropriate.

3 The Dual Distance Transformation

In this section, we present a novel high-dimensional indexing technique called the *Dual Distance Transformation* (DDT for short) for speed up the query efficiency over large image database.

3.1 Preliminaries

The design of DDT is motivated by the following observations. First, the similarity between images can be derived and ordered based on their distances to a reference image. Second, a distance is essentially a single dimensional value which enables us to reuse existing single dimensional indexing schemes such as B^+-tree. The list of symbols used in the rest of paper is summarized in Table 1.

Table 1. Meaning of Symbols Used

Symbols	Meaning
Ω	a set of images
I_i	the *i*-th image and $I_i \in \Omega$
d	the number of dimensions
n	the number of images in Ω
V_q	a query image user submits
$\Theta(I_q, r)$	the query hypersphere with centre I_q and radius r
$\Theta(O_j, CR_j)$	The *j*-th cluster hypersphere with centre O_j and cluster radius CR_j
$\boldsymbol{d(I_i, I_j)}$	the distance between two images

Definition 1(START DISTANCE). *Given an image I_i, the Start Distance (SD for short) of it is the distance between image I_i and the image $I_o(0,0,...,0)$, formally defined as:*

$$SD(I_i) = d(I_i, I_o) \tag{1}$$

Assuming that n images are grouped into T clusters, the centroid O_j of each cluster C_j is first obtained, where $j \in [1,T]$. We model a cluster as a tightly bounded hyper- sphere described by its *centroid* and *radius.*

Definition 2 (CLUSTER RADIUS). *Given a cluster C_j, the distance between its centroid O_j and the image which has the longest distance to O_j is defined as the cluster radius of C_j, denoted as CR_j.*

Given a cluster C_j, the cluster hypersphere of it is denoted as $\Theta(O_j, CR_j)$, where O_j is the centroid of cluster C_j, and CR_j is the cluster radius.

Definition 3 (CENTROID DISTANCE). *Given an image I_i, its centroid distance is defined as the distance between itself and the cluster centroid O_j, which is denoted as:*

$$CD(I_i) = d(I_i, O_j) \tag{2}$$

where $i \in [1, \|C_j\|]$ and $j \in [1,T]$.

3.2 The Data Structure

Based on the above definitions, we propose the *dual-distance-transformation*-based high-dimensional indexing scheme. Specifically, all images are first grouped into T

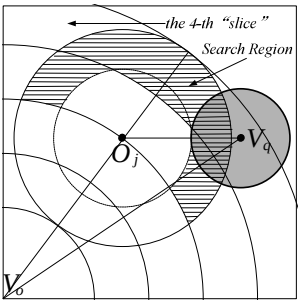


Fig. 1. The corresponding "slices" of the cluster hypersphere $\Theta(O_j, CR_j)$

clusters using a *k*-Means clustering algorithm, then the *start distance* and *centroid distance* of each image are computed, thus I_i can be modeled as a four-tuple:

$$I_i ::= <i, CID, SD, CD> \tag{3}$$

where *i* refers to the *i*-th image and *CID* is the ID of the cluster that I_i belongs to.

However, for each image I_i in a cluster hypersphere, it is hard to combine the *start*- and *centroid-distance* of I_i to get its uniform index key. To address this problem, the *dual-distance-transformation* approach is proposed to obtain a new index key through "slicing" the cluster hypersphere. As shown in Fig.1, assume that $\Theta(I_q, r)$ intersects with $\Theta(O_j, CR_j)$, we first "slice" $\Theta(O_j, CR_j)$ equally into λ "pieces" according to the value of the start-distance, where λ=4. Thus, for an image I_i in the *l*-th "slice" of $\Theta(O_j, CR_j)$, the range of its start-distance is represented: $SD(I_i) \in \left[SD(O_j) - CR_j + \frac{l \times 2CR_j}{\lambda}, SD(O_j) - CR_j + \frac{(l+1) \times 2CR_j}{\lambda} \right]$, therefore the uniform index key of I_i can be defined as:

$$key(I_i) = c \times j + l + CD(I_i) \Big/ MCD \tag{4}$$

where $l = \left\lfloor \frac{\lambda \times SD(I_i) - SD(O_j) + CR_j}{2CR_j} \right\rfloor + 1$ and $MCD = \sqrt{2}$.

Algorithm 1. DDT Index Construction
Input: Ω : the image set, λ : the number of slices in each cluster hypersphere;
Output: *bt:* the index for DDT*;*
1. The images in Ω are grouped into *T* clusters using k-Means clustering algorithm
2. *bt* ← ***newDDTFile***(); */* create index header file */*
3. **for** *j*:=1 to *T* **do**
4. **for** each image I_i in the *j*-th cluster **do**
5. the centroid distance and start distance of each image are computed;
6. the "slice" in the *j*-th cluster that I_i belongs to is identified;
7. *Key*(I_i)= ***TransValue***(I_i, *j*, *l*) ;
8. ***BInsert***(*Key*(I_i),*bt*); */* insert it to B+-tree */*
9. **end for**
10. **end for**
11. **return** *bt*

Fig. 2. The index construction algorithm for DDT

3.3 Building DDT

Fig. 2 shows the detailed steps of constructing a DDT index. Note that the routine ***TransValue***($I_i, CID, SliceID$) is a distance transformation function as given in Eq. (4) and ***BInsert***(*key, bt*) is a standard B^+-tree insert procedure.

3.4 *k*-NN Search Algorithm

For *n* high-dimensional images, *k*-Nearest-neighbor search is the most frequently used search method which retrieves the *k* most similar images to a given query image. In this section, we focus on *k*-NN search for high-dimensional images.

3.4.1 Picking a Value of LB and UB

Assume that a query hypersphere $\Theta(I_q, r)$ intersects with a cluster hypersphere $\Theta(O_j, CR_j)$, we examine which "slices" in $\Theta(O_j, CR_j)$ intersects with $\Theta(I_q, r)$. Let us first introduce two definitions.

Definition 4 (LOW BOUND ID OF "SLICE"). *Given two intersected hyperspheres* $\Theta(I_q, r)$ *and* $\Theta(O_j, CR_j)$, *the low bound ID of "slice" in* $\Theta(O_j, CR_j)$ *is the ID number of the "slice" which is the closest to the origin* V_o, *denoted as* ***LB***(*j*).

Definition 5 (UPPER BOUND ID OF "SLICE"). *Given two intersected hyperspheres* $\Theta(I_q, r)$ *and* $\Theta(O_j, CR_j)$, *the upper bound ID of "slice" in* $\Theta(O_j, CR_j)$ *is the ID number of the "slice" which is the farthest from the origin* V_o, *denoted as* ***UB***(*j*).

For the example shown in Fig. 1, the cluster hypersphere $\Theta(O_j, CR_j)$ is divided into 4 "slices", the "slice" that is closest to the origin V_o is the 3rd "slice", denoted as *LB*(j)=3; similarly, *UB*(j)=4;

Now we focus on the problem of how to get the ***LB*** and ***UB***. Once a query hypersphere intersects with the cluster hypersphere, some continuous "slices" (i.e., from the *LB*(*j*)-th *"slice"* to the *UB*(*j*)-th *"slice"*, where $LB(j) \le UB(j)$) may be affected (i.e., being intersected). Assume that the query hypersphere $\Theta(V_q, r)$ intersects with the cluster hypersphere $\Theta(O_j, CR_j)$, $\Theta(O_j, CR_j)$ is "sliced" into λ pieces. As mentioned before, the slice ID number in a cluster hypersphere is ordered ascendingly in terms of the start-distance value. So we can get the ID number of the low bound "slice"(i.e., *LB*) and upper bound "slice" (i.e., *UB*) in the *j*-th cluster hypersphere as follows:

$$LB(j) = \begin{cases} \left\lceil \dfrac{(SD(I_q) - r - SD(O_j) + CR_j)\,\lambda}{2CR_j} \right\rceil + 1, & if \;\; SD(O_j) - CR_j < SD(I_q) - r < SD(O_j) + CR_j \\ 1 & if \;\; SD(I_q) - r \le SD(O_j) - CR_j \end{cases} \tag{5}$$

$$UB(j) = \begin{cases} \left\lceil \dfrac{(SD(I_q) + r - SD(O_j) + CR_j)\,\lambda}{2CR_j} \right\rceil + 1, & if \;\; SD(I_q) + r < SD(O_j) + CR_j \\ \lambda & if \;\; SD(I_q) + r \ge SD(O_j) + CR_j \end{cases} \tag{6}$$

where $\lceil \bullet \rceil$ is an integral part of $\bullet$.

3.4.2 *k*-NN Algorithm

Fig. 3 illustrates the whole *k*-NN search process which contains three steps: first, when a user submits a query image I_q, the search starts with a small radius, and step by step,

the radius is increased to form a bigger query sphere iteratively (line 3). Once the number of candidate images is larger than *k*, the search stops, then the $(|S|-k-1)$ images which are farthest to the query one are identified (lines 6-7) and removed from *S* (line 8). It is worth mentioning that the symbol $|S|$ has two meanings: (a). the total number of candidate images in *S*; (b). the candidate images in *S* are the images whose distances to the query image I_q are less than or equal to the query radius *r*. In this way, the *k* nearest neighbor images of I_q are returned. Routine ***RSearch*** (I_q, r) is the main range search algorithm which returns the candidate images of range search with centre I_q and radius *r*. ***Search*** (I_p, r) is the implementation of the range search. ***Farthest*** (S, I_q) returns the image which is the farthest from I_q in the candidate image set *S*. ***BRSearch***(*left, right*) is a standard B^+-tree range search function.

```
Algorithm 3. kNN Search
Input: query image I_q , k,  Δr
Output: query results S
1.   r ← 0,  S ← Φ ;                          /*   initialization    */
2.   while ( |S| < k )
3.       r ← r + Δr ;
4.       S ← RSearch( I_q ,r);
5.       if ( |S| > k ) then
6.           for count:=1 to |S| − k − 1 do
7.               I_far ← Farthest(S, I_q );
8.               S ← S − I_far ;
9.           end for
10.      end if
11. end while

RSearch( I_q ,r)
12.   S1 ← Φ , S2 ← Φ ;
13.   for j:=1 to T do                  /*  T is the number of clusters    */
14.       S2 ← Search( I_q ,r, j);
15.       S1 ← S1 ∪ S2 ;
14.       if  Θ(O_j , CR_j ) contains Θ(I_q ,r ) then
17.           end loop;
18.       end if
19.   end for
20.   return  S1 ;                       /*  return candidate images    */

Search( I_q ,r, j)
21.   left ← c × j + LB( j) + (d (I_q , O_j ) − r) / MCD ;
22.   right ← c × j + UB( j) + CR_j / MCD ;
23.   S3 ← BRSearch[left, right];
24.   for each image I_i in the candidate images S3 do
25.       if  d (I_q ,I_i ) > r then
26.           S3 ← S3 − I_i ;          /*  I_i is removed from S3   */
27.   end for
28.   return  S3 ;
```

Fig. 3. *k*-NN search algorithm

4 Cost Model

We now derive a cost model for DDT in which query cost is measured by the number of nodes accessed. Some frequently used symbols are shown in Table 2.

Table 2. Parameters and their meanings

Symbol	Meaning
f	average fanout of a node in B^+ -tree
h	the height of the B^+ -tree
T_S	disk seek time
T_L	disk latency time
T_T	disk transmission time
T_Q	total query time

As mentioned before, we use B^+-tree as a basic index structure for DDT. Both h and n should be met in Eq. (7) below.

$$f \times (f+1)^{h-1} = n \tag{7}$$

By solving Eq. (7), the height of the B^+-tree is as follows:

$$h = \left\lceil \frac{\lg n - \lg f}{\lg(f+1)} \right\rceil + 1 \tag{8}$$

For a range search in the j-th cluster hypersphere, the total number of candidate images accessed can be derived as follows:

$$num(j) = \frac{Vol\left(\left(\overline{\Theta(V_o, SD(V_q)-r)} \cap \Theta(V_o, SD(V_q)+r)\right) \cap \overline{\Theta(O_j, d(V_q, O_j)-r)} \cap \Theta(O_j, CR_j)\right)}{\sum_{i=1}^{T} Vol\left(\Theta(O_i, CR_i)\right)} \times n \tag{9}$$

where $Vol(\bullet)$ refers to the volume of $\bullet$.

Therefore once t cluster hyperspheres are affected, and combining the Eq. (8) with (9) together, the total cost (*height + number of leaf nodes + refinement*) for a range search in DDT denoted as T_Q , is calculated as follows:

$$T_Q = \sum_{j=1}^{t} \left[\left(\left\lceil \frac{lgn - lgf}{lg(f+1)} \right\rceil + 1 + \left\lceil \frac{num(j)}{f} \right\rceil \right) \times (T_S + T_L + T_T) + T_c \times num(j) \right] \tag{10}$$

where T_C is the average CPU cost of the comparison between any two images.

Eq. (10) shows that the range query cost of DDT, which is mainly proportional to the number of images while it is reciprocal to the number of entries in a node. In Eq. (10), T and λ are two tunable parameters which can affect DDT's query performance. Therefore the moderate values of T and λ are critically important to the query optimization.

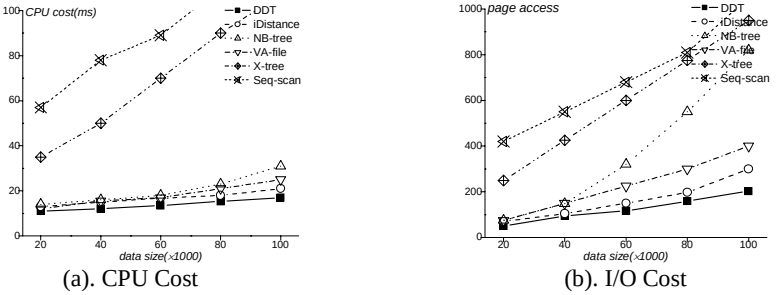
5 Experiments

To demonstrate the practical effectiveness of the new indexing method, we performed an extensive experimental evaluation of the DDT and compared it with the competitors: the X-tree, the VA-file, the iDistance, NB-tree and the linear scan.

We have implemented DDT, NB-Tree, iDistance, VA-file and X-tree in *C* language and used B^+-tree as the single dimensional index structure. All the experiments are run on a Pentium IV CPU at 2.0GHz with 256 Mbytes memory and index page size is fixed to 4096 Bytes. The image data set comprises of color histogram information, containing 32-Dimensional image features extracted from 100,000 images which are downloaded by our web crawler randomly from around 5,000 websites. The values of each dimension are normalized to the range of [0,1]. In our evaluation, we use the number of page accesses and the CPU time as the performance metric.

5.1 Effect of Data Size

In this experiment, 100 various types of 10-NN queries are performed over the image database whose cardinalities ranges from 20,000 to 100,000 with the same dimensionality. Fig. 4a shows the performance of query processing in terms of CPU cost. It is evident that DDT outperforms the other five methods significantly in terms of the CPU cost. It is interesting to note that the performance gap between the tree- based method such as the X-tree and other four techniques, viz., DDT, NB-tree, iDistance and VA-file becomes larger since it is a CPU-intensive operation during query process. In Fig. 4b, the query performance evaluations with respect to the I/O cost are conducted. The experimental result reveals that DDT yields consistent performance and is more efficient than other methods since the increase in data size does not increase the height of B^+-tree substantially, and it can more effectively filter away images that are not in the answer set than others.

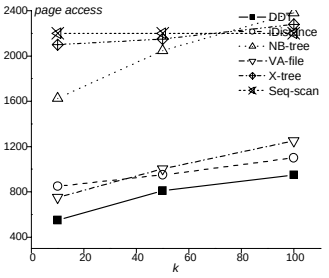


(a). CPU Cost (b). I/O Cost

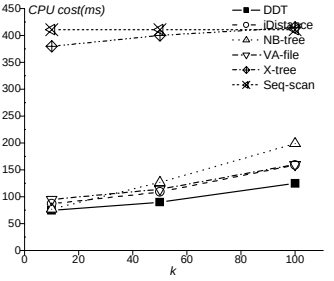
Fig. 4. Effect of Data size

5.2 Effect of *k*

In this experiment, we test the effect of *k* on the *k*-NN search. Fig. 5a demonstrates the experimental results in terms of I/O cost when *k* ranges from 10 to 100. DDT performs the best in terms of page access. The I/O costs of iDistance and VA-file are closely similar and NB-tree exhibits a dramatically increase in terms of page access and it finally exceeds the X-tree when *k* is 45. In Fig. 5b, the CPU cost of DDT is slightly lower than that of NB-tree for all data sets.



(a). *k* vs. I/O cost

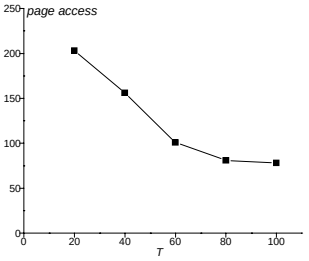


(b). *k* vs. CPU cost

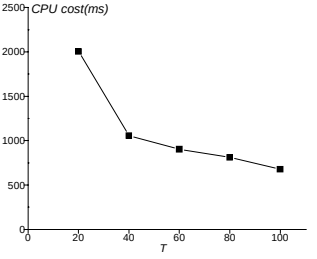
Fig. 5. Effect of *k*

5.3 Effect of *T* on the Efficiency of *k*-NN Search

In this experiment, we study the effect of the number of clusters(*T*) on the efficiency of the *k*-NN search. Figs. 6a and 6b illustrate that with the increase of *T*, the efficiency of the *k*-NN search (including the I/O and CPU cost) first increases gradually since the average search region is reducing as the number of clusters increases. Once *T* exceeds a threshold, the significant overlaps of different cluster hyperspheres lead to the high cost of I/O and CPU in the *k*-NN search. Therefore we should treat *T* as a tuning factor.



(a). *T vs* I/O cost



(b). *T vs* CPU cost

Fig. 6. Effect of *T* on the efficiency of *k*-NN search

5.4 Effect of λ on the Efficiency of Range Search

We use λ to denote the number of "slices" in every cluster hypersphere. In this evaluation, we study the effect of λ on the efficiency of range search with an identical search radius. Fig. 7 illustrates that the number of candidate images by DDT is decreasing gradually as λ increases. That is to say, the search efficiency of DDT could not improve anymore when λ exceeds a threshold, (e.g., λ =40). This is because with the increase of λ, the number of "slices" in a cluster hypersphere increases too. The precision of index key is getting smaller and smaller. The efficiency of DDT dose not improve anymore once λ reaches an optimal value. It is interesting to note that the search efficiency of DDT is better than that of iDistance and NB-tree no matter what value λ is of, and the search efficiency of DDT dose not increase anymore when λ =40. Therefore, λ is also a turning factor to the search optimization. Based on our experiments, we set λ to 40 as an optimal value empirically.

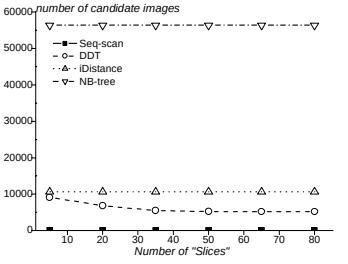


Fig. 7. Effect of λ on the Efficiency of Range Search

6 Conclusion

In this paper, we have presented a *dual-distance-transformation*-based high-dimensional image indexing scheme called *DDT*. We have shown by extensive performance studies that the proposed method is more efficient than the four most competitive techniques including iDistance, NB-Tree, X-tree and VA-file, in addition to sequential scan. Furthermore, being a B^+-tree-based index, DDT can be crafted easily into an existing database backend or implemented as a stored procedure.

Acknowledgments. This work is supported by National Natural Science Foundation of China (No.60533090, No.60525108), 973 Program (No.2002CB312101), Science and Technology Project of Zhejiang Province (2005C13032, 2005C11001-05), and China-US Million Book Digital Library Project (www.cadal.zju.edu.cn).

References

[1] Christian Böhm, Stefan Berchtold, Daniel Keim. Searching in High-dimensional Spaces: Index Structures for Improving the Performance of Multimedia Databases, *ACM Computing Surveys*, 2001. 33 (3).

[2] A. Guttman, R-tree: A dynamic index structure for spatial searching, In *Proc. of the ACM SIGMOD Int. Conf. on Management of Data*, 1984. pp.47-54.

[3] N. Beckmann, H.-P. Kriegel, R. Schneider, B. Seeger, 1990, The R*-tree: An Efficient and Robust Access Method for Points and Rectangles, In *Proc. ACM SIGMOD Int. Conf. on Management of Data*, pp. 322-331.

[4] S. Berchtold, D.A. Keim and H.P. Kriegel. The X-tree: An index structure for high-dimensional data. In *Proc. 22th Int. Conf. on Very Large Data Bases*, 1996. pp. 28-37.

[5] R. Weber, H. Schek, and S. Blott. A quantitative analysis and performance study for similarity-search methods in high-dimensional spaces. In *Proc. 24th Int. Conf. on Very Large Data Bases*, 1998. pp. 194-205.

[6] S. Berchtold, C. Bohm, H.P. Kriegel, J. Sander, and H.V. Jagadish. Independent quantization: An index compression technique for high-dimensional data spaces. In *Proc. 16th Int. Conf. on Data Engineering*, 2000. pp. 577-588.

[7] M J. Fonseca and J A. Jorge. NB-Tree: An Indexing Structure for Content-Based Retrieval in Large Databases. In *Proc. of the 8th Int. Conf. on Database Systems for Advanced Applications*, Kyoto, Japan, 2003. pp. 267-274.

[8] H.V. Jagadish, B.C. Ooi, K.L. Tan, C. Yu, R. Zhang. iDistance: An Adaptive B^+-tree Based Indexing Method for Nearest Neighbor Search., *ACM Transactions on Data Base Systems*, 30, 2, 2005. pp. 364-397.

Subtrajectory-Based Video Indexing and Retrieval

Thi-Lan Le[1,2], Alain Boucher[1,3], and Monique Thonnat[2]

[1] International Research Center MICA
Hanoi University of Technology, Viet Nam
Thi-Lan.LE@mica.edu.vn, alain.boucher@auf.org
[2] ORION, INRIA, 2004 route des Lucioles, B.P. 93, 06902 Sophia Antipolis, France
{Lan.Le_Thi, Monique.Thonnat}@sophia.inria.fr
[3] Equipe MSI, Institut de la Francophonie pour l'Informatique, Hanoi, Viet Nam

Abstract. This paper proposes an approach for retrieving videos based on object trajectories and subtrajectories. First, trajectories are segmented into subtrajectories according to the characteristics of the movement. Efficient trajectory segmentation relies on a symbolic representation and uses selected control points along the trajectory. The selected control points with high curvature capture the trajectory various geometrical and syntactic features. This symbolic representation, beyond the initial numeric representation, does not suffer from scaling, translation or rotation. Then, in order to compare trajectories based on their subtrajectories, several matching strategies are possible, according to the retrieval goal from the user. Moreover, trajectories can be represented at the numeric, symbolic or the semantic level, with the possibility to go easily from one representation to another. This approach for indexing and retrieval has been tested with a database containing 2500 trajectories, with promising results.

1 Introduction

Advances in computer technologies and the advent of the World Wide Web have made an explosion of multimedia data being generated, stored, and transmitted. For managing this amount of information, one needs developing efficient content-based retrieval approaches that enable users to search information directly via its content. Currently, the most common approach is to exploit low-level features (such as colors, textures, shapes and so on). When working with videos, motion is also an important feature. When browsing a video, people are more interested in the actions of a car or an actor than in the background. Moving objects attract most of users' attention. Among the extracted features from object movement, trajectory is more and more used. In order to use the trajectory information in content-based video indexing and retrieval, one must have an efficient representation method allowing not only to index trajectories, but also to respond to the various kinds of queries and retrieval needs. For retrieval aspects, the matching strategies is also of importance.

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 418–427, 2007.
© Springer-Verlag Berlin Heidelberg 2007

Matching to compare between trajectories can be done globally or partially. For global matching, the whole trajectories are compared to each other. However, objects in videos can undergo complex movements, and global matching can prevent from retrieving a partial but important section of the trajectory. In some cases, the user can be interested in only one part of the object trajectory. Therefore, it is useful to segment a trajectory into several subtrajectories and then match the subtrajectories. How to match subtrajectories and how to combine all partial results into a final retrieval result is not as trivial as it seems. In [1], the authors have segmented the object trajectory into subtrajectories with constant acceleration. But this approach do not consider the case where the object changes direction. In [2], after segmenting a trajectory into subtrajectories, the authors computed the PCA coefficients for each subtrajectory. However, with only one matching strategy, it cannot satisfy all user needs.

In the retrieval phase, queries from the user can be done at the numeric, symbolic or semantic level. Many interaction levels allow the user to be more flexible regarding his/her needs. Similarly, distances between a query and the database can be measured according to one level or another. Another important aspect in interactive retrieval is the relevance feedback, which allows an user to refine the query. A good representation scheme and efficient matching strategies must take into account all these aspects, taking care of both indexing and retrieval.

After introducing all these aspects, the main contributions of this paper are the following: Integrate into a representation scheme numeric, symbolic and semantic levels for trajectory-based video indexing and retrieval; Propose a trajectory segmentation algorithm working at the symbolic level, to avoid the problem of sensibility to noise at the numeric level, and invariant to rotation, translation or scaling; Present different trajectory and subtrajectory matching strategies; Go toward semantic trajectory-based video indexing and retrieval.

The rest of the paper is organized as follow. In Section 2, we are proposing a structure for a SubTrajectory-based Video Indexing and Retrieval (STBVIR), which includes control point selection, trajectory representation at the numeric and the symbolic level, trajectory segmentation into subtrajectories and matching strategies. In section 3 we are presenting some aspects linked to semantic. Some experimental results are shown in section 4. Section 5 is concluding this paper with some directions for future work.

2 SubTrajectory-Based Video Indexing and Retrieval

2.1 General Description

We are proposing an architecture of STBVIR (figure 1). In this architecture, object tracking is done by a preprocessing module (not shown here), and object trajectories are taken as input. In the real physical world, a trajectory is represented following 3 dimensions. Without a priori contextual information, trajectories can be represented in 2D. Knowing the application and its context,

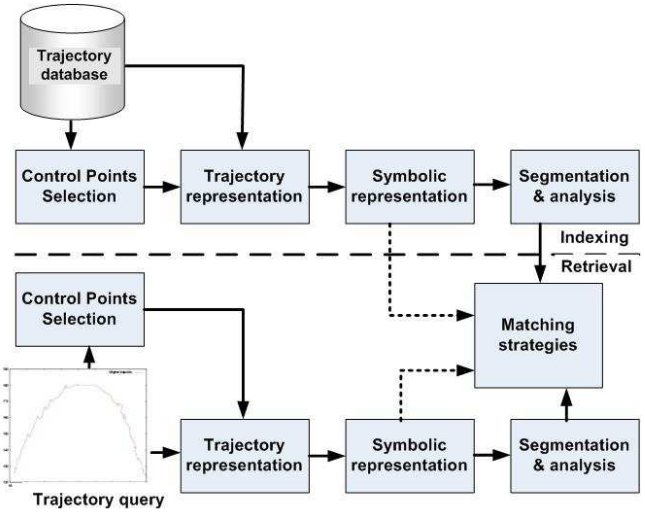


Fig. 1. Architecture for subtrajectory-based video indexing and retrieval

it can be useful to represent a trajectory in 3D or to map the 2D trajectory into the monitored environment [3]. In this paper, we will consider only the general case in 2D, without a priori knowledge on the application.

For indexing, all object trajectories are processed through four modules. The output is a symbolic representation of the global trajectory or its subtrajectories or only some selected control points along the trajectory. For retrieval, given a trajectory query by the user, comparison is made with the trajectories in the database, at the numeric or the symbolic level. Trajectories that are most similar (given a matching strategy) with the trajectory query will be returned to users.

2.2 Control Point Selection

A symbolic representation can take all the individual trajectory points as input. But doing so, computation time can be high, as well for the symbolic representation as for the trajectory matching. Selecting control points with high curvatures along the trajectory before computing its representing can help greatly. Selected control points can capture the trajectory's various geometrical and syntactic features. As one can see in section 4, the results from the two cases, using all points or only some selected control points, are very similar, but the second case takes much less time to compute. Moreover, selected control points and their symbolic representation allow us to propose a segmentation method as described in the next section.

First, we are describing the control point selection method of [4]. Given a sequence, $T = [(x_1, y_1), ..., (x_n, y_n)]$, n being the length of T, T can be represented by $T = [p_1, .., p_n]$. Let $\alpha(p)$ be the angle of a point p in T, determined by

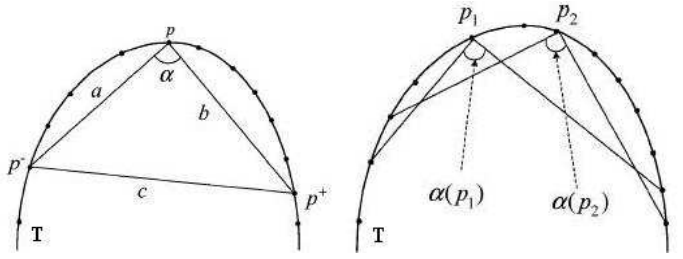


Fig. 2. Selecting control points along a trajectory T. (a) Control points (like p) are shown along the trajectory T. p^- and p^+ are linked to the point p by satisfying the angle constraint given by the Equation 1. (b) p_1 and p_2 are two selected control points too close to each other. The one with the smaller angle α will be chosen as the best control point [4].

two specified points p^+ and p^- which are selected from both sides of p along T (figure 2.a) and satisfy

$$d_{min} \leq |p - p^+| \leq d_{max} \text{ and } d_{min} \leq |p - p^-| \leq d_{max} \tag{1}$$

where d_{min} and d_{max} are two thresholds. d_{min} is a smoothing factor used to reduce the effect of noise from T. With p^+ and p^-, the angle can be computed using

$$\alpha(p) = cos^{-1} \frac{||p - p^+||^2 + ||p - p^-||^2 - ||p^+ - p^-||^2}{2||p - p^+||||p - p^-||} \tag{2}$$

If $\alpha(p)$ is larger than a threshold T_α, set to 150 here, the point p is selected as a control point. In addition to equation 2, it is expected that the two control points are far from each other, to enforce that the distance between any two control points is larger than the threshold defined in (1). If the two candidates are too close to each other, i.e. $||p_1 - p_2|| \leq d_{min}$, the one with the smaller angle α is chosen as the best control point (figure 2.b).

2.3 Numeric Trajectory Representation Module

Working with raw data from the object trajectory is not always suitable because these data are sensible to noise and are affected by rotation, translation and scaling. In order to cope this problem, we have chosen among the existing representation methods the one from [5], which also uses both direction and distance information of the movement.

With a given sequence, $T_A = [(x_{a,1}, y_{a,1}), ..., (x_{a,n}, y_{a,n})]$, n being the length of T_A, a sequence of (movement direction, movement distance ratio) pairs M_A is defined as a sequence of pairs: $M_A = [(\theta_{a,1}, \delta_{a,1}), ..., (\theta_{a,n-1}, \delta_{a,n-1})]$. The movement direction is defined as:

$$\theta_{a,i} = \begin{cases} \arctan\frac{y_{a,(i+1)}-y_{a,(i)}}{x_{a,(i+1)}-x_{a,(i)}} - \pi & \text{if } x_{a,(i+1)} - x_{a,(i)} \prec 0 \text{ and } y_{a,(i+1)} - y_{a,(i)} \leq 0 \\ \arctan\frac{y_{a,(i+1)}-y_{a,(i)}}{x_{a,(i+1)}-x_{a,(i)}} & \text{if } x_{a,(i+1)} - x_{a,(i)} \geq 0 \\ \arctan\frac{y_{a,(i+1)}-y_{a,(i)}}{x_{a,(i+1)}-x_{a,(i)}} + \pi & \text{if } x_{a,(i+1)} - x_{a,(i)} \prec 0 \text{ and } y_{a,(i+1)} - y_{a,(i)} \succ 0 \end{cases} \quad (3)$$

and the movement distance ratio is defined as follows:

$$\delta_{a,i} = \begin{cases} \frac{\sqrt{(y_{a,(i+1)}-y_{a,(i)})^2+(x_{a,(i+1)}-x_{a,(i)})^2}}{TD(T_A)} & \text{if } TD(T_A) \neq 0 \\ 0 & \text{if } TD(T_A) = 0 \end{cases} \quad (4)$$

$$TD(T_A) = \sum_{1 \leq j \leq n-1} \sqrt{(y_{a,(j+1)} - y_{a,j})^2 + (x_{a,(j+1)} - x_{a,j})^2} \quad (5)$$

Raw trajectory data given to this module become a sequence of pairs of movement direction and movement distance ratio. We can use directly this sequence to compare trajectories or we can use it as an intermediate information for the symbolic representation module.

2.4 Symbolic Trajectory Representation Module

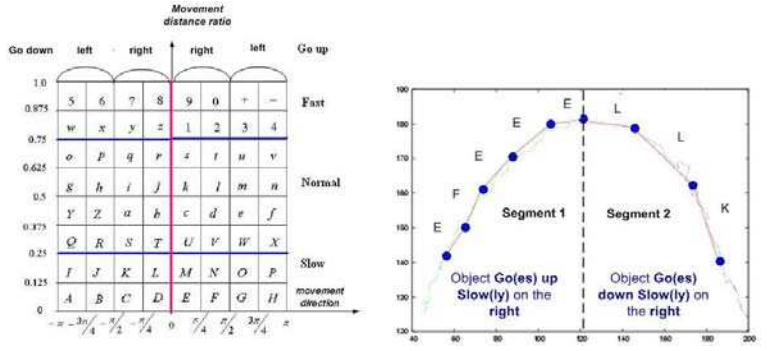
Using the previous numeric representation for trajectories, a proposed symbolic representation from [5] is computed as follows:

Given ϵ_{dir} and ϵ_{dis}, the two dimensional (movement direction, distance ratio) space is divided into $\frac{2\pi}{\epsilon_{dir}} * \frac{1}{\epsilon_{dis}}$ subregions. Each subregion SB_i is represented by two (movement direction, distance ratio) pairs: $(\theta_{bl,i}, \delta_{bl,i})$ and $(\theta_{ur,i}, \delta_{ur,i})$, which are the bottom left and upper right coordinates of SB_i. A distinct symbol A_i is assigned for subregion SB_i of size $\epsilon_{dir} * \epsilon_{dis}$. A pair of movement direction and movement distance ratio $(\theta_{a,i}, \delta_{a,i})$ will be represented by a symbol A_i if $\theta_{bl,i} \leq \theta_{a,i} \prec \theta_{ur,i}$ and $\delta_{bl,i} \leq \delta_{a,i} \prec \delta_{ur,i}$.

2.5 Segmentation

From an original trajectory composed of n points $T = [(x_1, y_1), ..., (x_n, y_n)]$, a new trajectory, shorter than the original one, is obtain after control point selection: $T' = [(x_1, y_1), ..., (x_m, y_m)]$ where $m \leq n$. This trajectory is transformed into a symbolic representation $S = [(A_1, ..., A_m)]$ using the quantization map shown in figure 3.a.

Let A_I be the set of symbols with θ being smaller than 0 and A_{II} be the set of symbols with θ greater than 0. An object going down gets a symbol belonging to A_I (down to the left or to the right), and an object going up gets a symbol belonging to A_{II} (up to the left or to the right). Therefore, by scanning a symbolic representation until a change in direction is detected (i.e. symbol at time t belongs to A_I and symbol at time t+1 belongs to A_{II}, or the opposite), we can create a new subtrajectory including all the points from the last change in direction to this new change, and so on until the end of the trajectory.



(a) (b)

Fig. 3. (a) Quantization map used for symbolic representation with their corresponding semantic expressions. (b) An example of parsing from a symbolic to a semantic representation using this map.

2.6 Subtrajectory-Based Video Indexing and Retrieval

Let $T_Q = [(x_1, y_1), ..., (x_n, y_n)]$ be a query trajectory. Following the segmentation algorithm that we have described in section 2.5, we can segment it in N subtrajectories $T_Q = \{T_{1Q}, ..., T_{NQ}\}$. Let T_D be a trajectory from the indexed databases divided into M subtrajectories $T_D = \{T_{1D}, .., T_{MD}\}$. With all these subtrajectories, we compute their numeric and their symbolic representations. Doing so, we can compare them using either the Edit Distance on Real Sequence (EDR) on the numeric representation or the Edit Distance on Movement Pattern String (EDM) on the symbolic representation [5].

For subtrajectory-based video indexing and retrieval, choosing a good and efficient matching strategy is important to take into account the various user needs. If the user is interested in the whole trajectory, a global matching strategy is a valuable choice, and if he/she just makes more attention in few parts of the trajectory, partial matching is then the privileged choice. Inspiring ourselves from [1], we are giving here some different matching strategies: two global matching strategies (dominant segment (GD) and full trajectory (GF) matching) and three partial trajectory matching strategies (strict partial (SP), relative partial (RP) and loose partial (LP) matching). In the following, $d_{Dist}(T_{iQ}, T_{jD})$ is the distance between a subtrajectory T_{iQ} and a subtrajectory T_{jD}. $Dist$ can be EDR or EDM. Note that L_{iQ} and L_{jD} are the length of T_{iQ} and T_{jD} respectively.

- **Global trajectory matching**
  - ***Dominant segment matching***(GD)**:** Only the dominant subtrajectory is used to match with those in the database. Dominant subtrajectories can be identified as segments with the smallest EDR or EDM distances.

$$d_{Dist}(T_Q, T_D) = min(d_{Dist}(T_{iQ}, T_{jD})) \tag{6}$$

- ***Full trajectory matching***(GF)**:** All subtrajectories in the original trajectory must match those in the database as described above.

$$d_{Dist}(T_Q, T_D) = \sum_{i=1}^{N}\sum_{j=1}^{M} w_{T_{iQ}} w_{T_{jD}} d_{Dist}(T_{iQ}, T_{jD}) \quad (7)$$

$$\text{where } w_{T_{iQ}} = \frac{L_{iQ}}{\sum_{k=1}^{N} L_{kQ}} \text{ and } w_{T_{jD}} = \frac{L_{jD}}{\sum_{k=1}^{M} L_{kD}} \quad (8)$$

– **Partial trajectory matching:** A subset of subtrajectories is selected to match those in the database, and matching is specified in terms of order. Matching between two subtrajectories is computed using:

$$match_{Dist}(T_{iQ}, T_{jD}) = true \text{ if } d_{Dist}(T_{iQ}, T_{jD}) \leq Threshold_{Dist} \quad (9)$$

$$match_{Dist}(T_{iQ}, T_{jD}) = false \text{ otherwise} \quad (10)$$

- ***Strict partial matching***(SP)**:** The matched subtrajectories between the query and the database must be strictly in the same order.
- ***Relative partial matching***(RP)**:** The relative order of the matched subtrajectories between the query and the database must be the same.
- ***Loose partial matching***(LP)**:** No constraint is given on the order of the matched subtrajectories. Just match a subset of subtrajectories between the query and the database.

3 Toward a Semantic Trajectory-Based Video Indexing and Retrieval

The word semantic is more and more used in the information retrieval domain (in many different ways). The given symbolic representation and segmentation method allow us to go toward a more semantic subtrajectory-based video indexing and retrieval. Using the quantization map of figure 3.a, a sequence of symbols representing a trajectory can be translated into a sequence of semantic words, as shown in figure 3.b.

In order to transform a symbolic representation of a trajectory into a semantic representation, some abstraction heuristics must be used. For example, if more than 80% of the symbols of a trajectory belong to the set {'M','N','E','F'} (figure 3.a) it can be said that this trajectory has the 3 characteristics {***Go up***, ***Slow***, ***right***}. In the example of 3.b, the original trajectory is segmented into two subtrajectories. After the symbolic representation phase for the selected control points, the trajectory is represented by 8 symbols 'EFEEELLK'. The first subtrajectory has 5 symbols (EFEEE) while the second one has 3 symbols (LLK). This trajectory can be abstracted saying that first the objet ***Go(es) up Slow***ly on the ***right*** and then it ***Go(es) down Slow***ly still on the ***right***.

Using this scheme, the user can give a query at the numeric, symbolic or semantic level. Comparison between (sub)trajectories can be done so far at the numeric (EDR distance) or at the symbolic (EDM distance) level (figure 4.a).

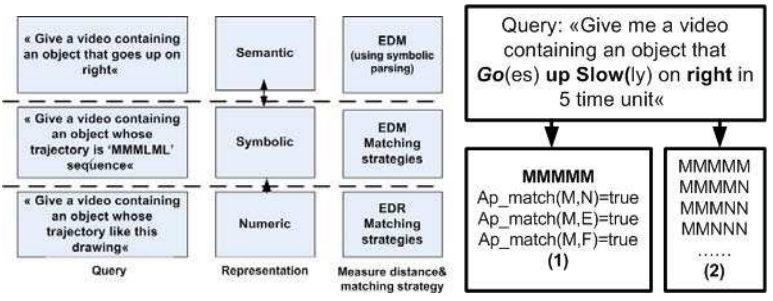


Fig. 4. (a)Three levels for trajectory representation. At the numeric level, the user draws a query, the EDR distance and corresponding matching strategies are used. At the symbolic level, the user gives a query using a sequence of symbols, the EDM distance and the proposed matching strategies are used. At the semantic level, the semantic query is first parsed into a symbolic query. Then, the EDM distance and the symbolic matching strategies are used. (b) A given semantic query is parsed into a symbolic representation following two different parsing methods.

To process a semantic query, one must first parse it into a symbolic representation. But more than one symbol can correspond to a sole semantic word. For this reason, two methods are possible for parsing the semantic query. The first one is to choose a representative symbol from a semantic characteristic of the movement, using an Ap_match comparison like in [5]. A_i Ap_match A_j if and only if $A_i = A_j$ or A_i is neighboor of A_j. The second method is to generate all possible symbolic sequences and then, combine or choose between all the matching results (following some pre-defined strategies). In figure 4.b, given the semantic query "Give me a video containing an object that ***Go(es) up Slow***ly on the ***right*** in 5 time units", then the two corresponding symbolic sequences, according to both parsing methods, are shown.

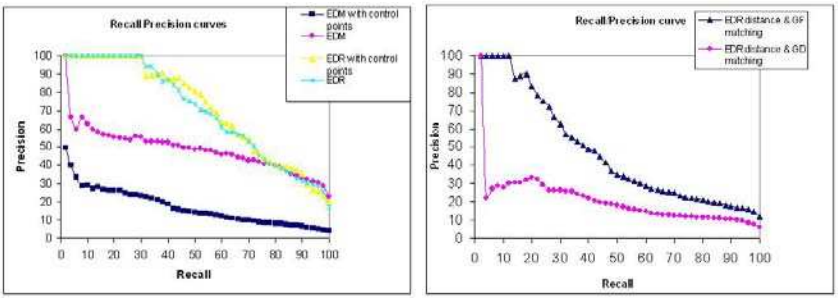
This preliminary discussion about semantic aspect in subtrajectory-based video indexing and retrieval is still on-going work, but we can foreseen some promising results in achieving semantic indexing and retrieval.

4 Experiments and Results

In order to analyze our system performances, we have used the free trajectory database[1] containing 2500 trajectories coming from 50 categories.

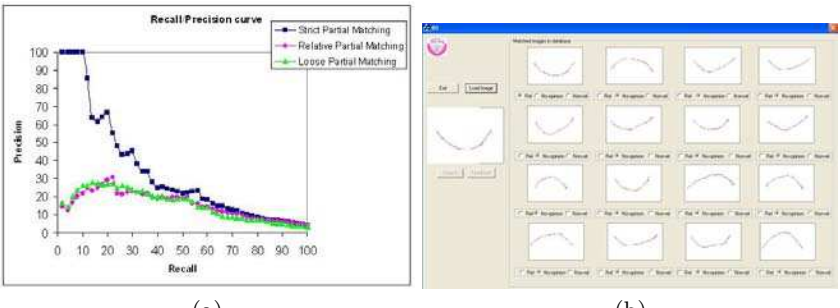
Recall and precision curves are widely used to evaluate performance of information retrieval system. In our tests, we have set $n/20$ and $n/15$ for d_{min} and d_{max} respectively where n being the length of T. We have chosen $\epsilon_{dir} = \pi/4$ and $\epsilon_{dis} = 0.125$ for the symbolic representation. Figure 5.a shows the results of our system with all individual of trajectory or some selected control points using

[1] http://mmplab.eed.yzu.edu.tw/trajectory/trajectory.rar



(a) (b)

Fig. 5. (a) Recall/precision curve for the EDR and EDM distance with all points from the trajectory and only the selected control points. The curve with green stars and the one with yellow triangles present retrieval results using the EDR distance with all points from trajectories and only selected control points respectively. The curve with violet circles and the one with blue rectangles present retrieval results using the EDM distance with all points from trajectories and only selected control points respectively (b) Recall/precision curve for the EDR distance with different matching strategies. The curve with triangles presents retrieval results using the EDR distance with the Full trajectory (GF) matching strategy while the curve with circles presents retrieval results using the EDR distance with the Dominant segment (GD) matching strategy.



(a) (b)

Fig. 6. (a) Results for the SP, RP, LP matching strategies. The curve with green triangles presents the retrieval results using the EDR distance with the SP strategy, the curve with violet circles presents retrieval results using the EDR distance with the RP matching strategy and the curve with blue rectangles presents retrieval results using the EDR distance with the LP matching strategy (b) System interface for retrieval, the trajectory query being on the left and the first sixteen result images on the right shown according to their EDR distance with the trajectory query.

the EDR distance for numeric representation and the EDM distance for symbolic representation. One can realizes that results using selected control points with the EDR distance are comparable with those using all points from the trajectory. The EDR distance in both cases gives better results than the EDM distance.

Figure 5.b shows the results of our system with different global matching strategies, GF and GD with the EDR distance. With this database, the results with the GF matching strategy are better than with the GD matching strategies. However, in the case where the user is interested in only one part of trajectory, the GD matching strategy is an efficient choice. Figure 6.a shows the results for the three partial matching strategies, SP, RP and LP, using a threshold of 60 and the EDR distance.

Query acquisition and result display is an important but difficult task in trajectory-based video indexing and retrieval. We have implemented a retrieval interface, as shown in figure 6.b. The trajectory query drawn on the left and the first sixteen result images on the right are shown sorted with their EDR distance with the trajectory query. It is possible to draw the corresponding trajectories from a numeric or a symbolic representation.

5 Conclusions and Future Work

In this paper, a subtrajectory-based video indexing and retrieval system has been proposed. Our system has some notable characteristics. Firstly, it allows the users to search desirable trajectory or only part of a trajectory (according to many matching strategies). It effects both EMD and EDR distance that count similar subsequences and assign penalties to the gaps in between these subsequences. Thus, unlike Longest Common SubSequence (LCSS)[5], it does consider gaps within sequences. Secondly, it offers a fast searching (because the trajectories or their subtrajectories are compared by matching only their selected control points), and it allows an efficient segmentation method based on a symbolic representation that is invariant to rotation, scaling and translation. Finally, all these advantages allows us to go toward semantic trajectory-based video indexing and retrieval system, although further work is needed to fully achieve it.

References

1. Chenn, W., Chang, S.F.: Motion trajectory matching of video objects. In SPIE 2003.
2. Bashir, F.: Object Motion Trajectory-Based Video Database System Indexing, Retrieval, Classification, and Recognition. PHD Thesis of Electrical and Computer Engineering, College of Engineering, University of Illinois Chicago (2005).
3. Piciarelli, C., Foresti, G.L., Snidara, L.: Trajectory clustering and its applications for video surveillance.Proceedings. IEEE Conference on Advanced Video and Signal Based Surveillance, 2005,40– 45.
4. Hsieh, J. W., Yu, S. L., Chen, Y.S.: Motion-Based Video Retrieval by Trajectory Matching. Proc IEEE Trans. on Circuits and Systems for Video Technology, Vol. 16, No. 3, March 2006.
5. Chen, L., Oŭu, M.T., Oria, V.: Symbolic Representation and Retrieval of Moving Object Trajectories. In MIR'04, NewYork (2004), 15–16

DR Image and Fractal Correlogram: A New Image Feature Representation Based on Fractal Codes and Its Application to Image Retrieval

Takanori Yokoyama and Toshinori Watanabe

Graduate School of Information Systems,
University of Electro-Communications,
1-5-1 Chofugaoka, Chofu-shi, Tokyo 182-8585, Japan
{yokotaka, watanabe}@is.uec.ac.jp

Abstract. In this paper, we propose a novel image feature representation method using fractal codes and its application to image retrieval. A fractal code consists of contractive affine mappings; each mapping represents a similarity relation between two regions in an image. A fractal-coded image is composed of a set of fractal codes. In the encoding process, the region sizes in the fractal codes tend to reflect the local features in the original image. We propose a method to represent this size as a feature image termed as a 'DR image'. We also propose an efficient retrieval method using correlograms of DR images. This retrieval method is an application of DR images and can be used in the compressed domain. The effectiveness of the proposed methods have been illustrated by several experiments using an image database released on the Internet.

Keywords: Fractal-Coded Image, Image Feature Representation, Correlogram, Compressed-Domain Retrieval.

1 Introduction

Generally, an image is represented as a two-dimensional array of pixels wherein the brightness value of each pixel is recorded. In image analysis, images are often transformed into various domains to extract their useful features. For example, there are several transforms such as color transform, distance transform, hough transform, fourier transform, wavelet transform and so on. In this paper, we propose a new feature representation method using fractal codes obtained by fractal transform. Fractal transform is a relatively new compression method proposed by Barnsley [1]. We employ the fractal codes to represent features in the original image, resulting in a feature image that we termed as a 'DR image'. Here, 'DR' is an abbreviation for domain and range. Since DR images can be directly extracted from fractal codes, they can be represented in the compressed domain, thereby eliminating the time-consuming decoding process in image retrieval [2]. We also propose an application of DR images in image retrieval. It is the third in a series of applications of correlograms in image retrieval, after color correlograms [3] and wavelet correlograms [4]. We have formulated a prototype system and evaluated the effectiveness of our method by means of some experiments.

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 428–439, 2007.
© Springer-Verlag Berlin Heidelberg 2007

2 Fractal Codes

Fractals are self-similar and independent of scale. Fractal transform compression is based on the self-similarity feature of an image. Its compression principles were proposed by Barnsley [1]. He used a system of mappings termed as the 'iterated function system (IFS)', and Jacquin [5] improved Barnsley's method to realize a fully automatic encoding process. Since an image is partitioned into regions in the encoding (i.e. affine mapping generation) process, Jacquin's method is generally referred to as a 'partitioned IFS (PIFS)'. The PIFS method forms the foundation of present fractal transform techniques [6]. Therefore, we assume that a fractal code is invariably obtained by PIFS encoding. In this paper, the compression principles and encoding algorithms have not been described in detail. For details, refer [7].

A fractal-coded image consists of a set of contractive mappings. In encoding, an image is partitioned into large regions called 'domains' and smaller regions called 'ranges'. Domains may overlap, while ranges tile the entire image. By determining a contractive mapping w_i for each range R_i from a relevant domain D_i and collating w_i from all the ranges, a fractal-coded image W can be derived as follows.

$$W(\cdot) = \bigcup_{i=1}^{N} w_i(\cdot), \tag{1}$$

$$R_i = w_i(D_i). \tag{2}$$

Here, N is the number of ranges that is equal to the cardinality of $W(\cdot)$. Usually, the *affine* transformation is denoted as w_i.

$$w_i\left(\begin{bmatrix} x \\ y \\ z \end{bmatrix}\right) = \begin{bmatrix} s^i_{00} & s^i_{01} & 0 \\ s^i_{10} & s^i_{11} & 0 \\ 0 & 0 & \alpha_i \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \begin{bmatrix} e_i \\ f_i \\ \beta_i \end{bmatrix} \tag{3}$$

Here, s^i_{00}, s^i_{01}, s^i_{10} and s^i_{11} are the parameters of spatial rotations and flips of D_i, α_i is the contrast scaling parameter and β_i is the luminance offset. An actual fractal code c_i possesses the entire information required to construct w_i (see Fig. 1).

$$c_i = \Big((x_{D_i}, y_{D_i}), (x_{R_i}, y_{R_i}), \mathrm{size}_i, \theta_i, \alpha_i, \beta_i\Big) \tag{4}$$

Here, (x_{R_i}, y_{R_i}) is the upper-left coordinate of R_i and (x_{D_i}, y_{D_i}) is the upper-left coordinate of D_i. The size of R_i is denoted by size_i, and θ_i is the index of the spatial rotation of D_i. We denote the fractal-coded image C of the original image I as a set of c_i.

$$C = \bigcup_{i=0}^{N} c_i \tag{5}$$

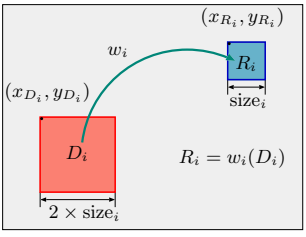


Fig. 1. w_i represents the similarity relation between R_i and D_i

3 DR Image

In this section, we introduce a new feature representation method using fractal codes. Among the information derived from fractal codes, we employ region sizes to create a new feature image. Owing to this transformation, we can apply existing image analysis methods. As the feature image is based on the domain and range data of the fractal codes, we termed it as a 'DR image'.

3.1 Distinct Property of Region Sizes in Fractal Codes

In encoding, a fractal encoder determines similar regions by using adaptive region partitioning. The region sizes depend on the local feature of the regions in the original image. For example, smaller ranges tend to be assigned to high-frequency edges or line regions, and the corresponding domains also exhibit a similar tendency. Larger ranges and domains tend to be assigned to low-frequency flat regions. Combinations of different sizes are assigned to texture regions that contain both high- and low-frequency regions. We assume that these properties of the region sizes in fractal codes can be useful features in image analysis. Furthermore, these features of fractal codes are directly accessible in the compressed domain.

3.2 DR Image and Creation Algorithm

To utilize the useful properties of region sizes in image analysis, we propose a DR image and its creation algorithm. The DR image records both the range and domain sizes on each pixel from the original image, and we can construct it without decoding the fractal-coded image.

We elaborate upon the DR image by means of a concrete example. Fig. 2 shows a pixel value and related regions in the DR image. Each of the fractal codes can employ one of the three different range sizes (4×4, 8×8 and 16×16 pixels) and three different domain sizes (8×8, 16×16 and 32×32 pixels). The brightness value of each pixel in the DR image requires five bits. The most significant two bits denote the range size in which the pixel belongs, and the least significant three bits denote the corresponding domain sizes. As the ranges are usually disjoint, each pixel is included in only one range. If the number of

possible range sizes is three, two bits are sufficient for their representation. In this case, we assign the most significant two bits '00' to the 16×16 range, '01' to the 8×8 range and '10' to the 4×4 range. While domains are allowed to overlap with each other, it is possible that each pixel is included by domains with different sizes. If the number of possible domain sizes is three, three bits are necessary for their representation. In Fig. 2, the 01011-valued pixel indicates that it is included in an 8×8 range and both 16×16 and 32×32 domains. Thus, a DR image can be considered to be a grey-scale representation of the fractal-code contents.

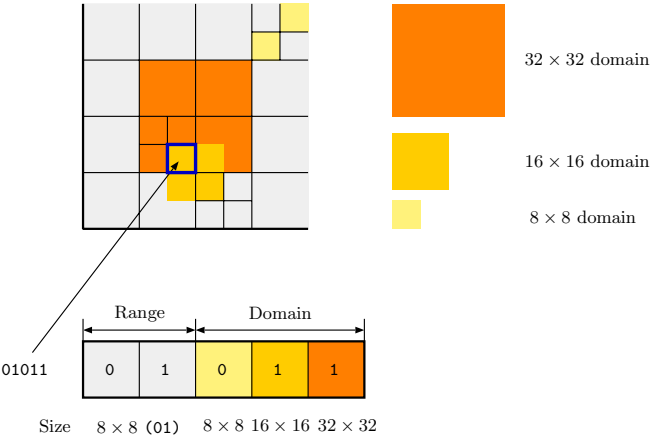


Fig. 2. Definition of a pixel value in DR images. Coloured squares denote domains, and the squares enclosed within lines denote ranges. The pixel in the region indicated by the arrow has an 8×8 range and both 16×16 and 32×32 domains; hence it has a value of 01011.

Fig. 3 shows the DR image creation algorithm. It is evident that the brightness value of each pixel in the DR image obtained using this algorithm represents the region sizes.

4 Retrieval Method for DR Images

In this section, we propose a new image retrieval method using DR images. Since this retrieval method depends on the correlogram, we term this method as 'fractal correlogram'. The correlogram was originally proposed by Jing Huang et al [3] for content-based color image retrieval. First, we investigate the correlogram.

4.1 Correlogram

The correlogram is a type of spatial extension of a histogram. The main advantages of a correlogram are as follows: 1) it includes the spatial correlation of pixels, 2) it describes the global distribution of local spatial correlation of pixels,

CreateDRimage(C) outputs the DR image I_{DR} created from the fractal-coded image C of the original image I.

1. Initialize I_{DR}:
   Instantiate an l-bit gray-scale image with all the pixels having zero value. l denotes the brightness level sufficient to represent the region sizes.
2. Compute each fractal code c_i in C:
   If the pixels in I_{DR} are included in a range R_i or a domain D_i of c_i, set the corresponding bits of the pixels to 1.
3. Return I_{DR}.

Fig. 3. The **CreateDRimage** algorithm

3) it is easy to compute, 4) the feature size is fairly small, and 5) it outperforms the traditional histogram.

Let I be an $n \times n$ gray-scale image. When a pixel p_1 has the brightness value b_1 equal to b_i, the correlogram $\gamma^{(k)}_{b_i,b_j}$ represents the probability of the brightness value b_2 of a pixel p_2 at a distance of k from p_1 assumes the value b_j.

$$\gamma^{(k)}_{b_i,b_j} = \Pr(b_2 = b_j, ||p_1 - p_2|| = k \mid b_1 = b_i) = \frac{\Gamma^{(k)}_{b_i,b_j}(I)}{8kh_{b_i}} \tag{6}$$

For convenience, L_∞-norm is used to measure the distance between the pixels, and the value $8k$ in the denominator is due to this norm. h_{b_i} is the total number of b_i-valued pixels, and $\Gamma^{(k)}_{b_i,b_j}(I)$ is the total number of b_j-valued pixels at a distance k from each b_i-valued pixel.

$$\Gamma^{(k)}_{b_i,b_j}(I) = \#\{(p_1, p_2) \mid p_1 \in I_{b_i}, p_2 \in I_{b_j}, \mid p_1 - p_2 \mid \ = k\} \tag{7}$$

Here, # denotes the cardinality of a set, I_{b_i} denotes the set of b_i-valued pixels. When the number of brightness levels in I is m and the number of distance levels is d, the correlogram dimension becomes m^2d.

The correlogram between pixels of identical brightness ($b_i = b_j$) is referred to as an 'autocorrelogram' and is defined as

$$\alpha^{(k)}_b(I) = \gamma^{(k)}_{b,b}(I). \tag{8}$$

The autocorrelogram is md-dimensional, and its computational cost significantly reduces than a general correlogram without a significant reduction in the retrieval accuracy. Therefore, we apply the autocorrelogram for DR images.

In order to retrieve DR images using correlograms, an appropriate distance measure is required. We employ the following distance d_μ ($\mu = 1$) [8] that is also used in the original paper on correlogram.

$$d_1(I, I') = \sum_k \frac{|\alpha^{(k)}_b(I) - \alpha^{(k)}_b(I')|}{1 + \alpha^{(k)}_b(I) + \alpha^{(k)}_b(I')} \tag{9}$$

Here, the integer 1 in the denominator prevents division by zero.

4.2 Directional Correlogram

L_∞-norm, which is used in calculation of a correlogram is easy to compute; however, it is insensitive to image rotation. Since rotation-sensitive image retrieval is usually required, we extend the original correlogram to obtain a direction-sensitive correlogram. For this purpose, we divide the original L_∞-norm correlogram into four directional sub-correlograms.

Fig. 4 shows the manner of division. Let the white circles denote the b-valued pixels, and the gray circles denote the pixels with brightness values different from that of b. Suppose the central white circle is p_1 (b-valued), and other circles on the thick rectangle denote the k-distant pixels. We divide the k-distant pixels into four parts ($Vertical$, $Horizontal$, $Diagonal_{45}$ and $Diagonal_{135}$) and independently count the b-valued pixels in each division.

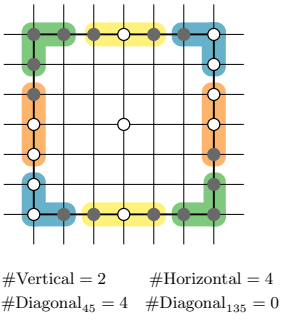


Fig. 4. Determination of the direction-sensitive correlogram by k-distant neighbour division. Four directions were used. White circles on the thick rectangle denote k-distant neighbours with equal brightness. Two of them are in the $Vertical$ part, zero in the $Diagonal_{135}$ part and four in the other two parts.

The pixels on the upper and lower yellow ovals are members of the $Vertical$ neighbour. Let $\lambda_b^{(k,v)}(x,y)$ denote the number of pixels in the $Vertical$ neighbour at a distance k from (x,y).

$$\lambda_b^{(k,v)}(x,y) = \#\{(x \pm i, y \pm k) \in I_b \mid 0 \leq i \leq \frac{k-1}{2}\} \tag{10}$$

Using this definition, the number of matching pixels becomes

$$\Gamma_b^{(k,v)}(I) = \sum_{(x,y)\in I_b} \lambda_b^{(k,v)}(x,y). \tag{11}$$

From the above-mentioned definitions, we can calculate the correlogram for the vertical direction.

$$\alpha_b^{(k,v)} = \frac{\Gamma_b^{(k,v)}(I)}{2kh_b} \tag{12}$$

Here, the value $2k$ in the denominator implies that the value $8k$ due to the L_∞-norm is divided by the four directions. In a similar manner, we can calculate the correlograms for $Horizontal$, $Diagonal_{45}$ and $Diagonal_{135}$ directions.

4.3 Indexing Algorithm

The directional correlogram indexing algorithm is described in this section. A **CreateIndex** function (Fig. 5) outputs the correlogram indexed file for fractal-coded images in the database DB. This index file contains pairs of the filename of the fractal-coded image and its correlograms. By calculating the distance between the correlograms of the query image and each of the elements in the index file and then sorting the filenames according to distance in the ascending order, we can obtain images that are similar to the query image.

CreateIndex(DB) outputs the index file for retrieval created from the DB database that consists of fractal-coded images. The index file consists of pairs of a filename and its correlograms. The distance k is given.

1. Make DR images:
   For each fractal-coded image C in DB, execute **CreateDRimage**(C).
2. Create correlograms:
   For each DR image, calculate the directional correlograms at a distance k, and add them to the index file.
3. Return the index file (discard the DR images if they are unnecessary).

Fig. 5. The **CreateIndex** algorithm

5 Experiments

In this section, we show several experimental results. We use an image database [9] released on the Internet. This database has 1,000 images (324×256 pixels) classified into ten categories, and each category contains 100 images. We used the *Mars* fractal codec software [10] and C (gcc 2.9.53 on Linux 2.4.2) environment for our system implementation. All experiments were performed on an Intel Pentium IV CPU with a speed of 2.80 GHz and a memory of 1 GB.

5.1 DR Image Creation

Table 1 shows some DR images obtained from the fractal-coded images by the proposed creation algorithm. The average time required to create one DR image from 1,000 fractal-coded images is 0.02 s. In DR images, the pixels that have high brightness values correspond to smaller ranges with several different domain sizes. These pixels are observed on edges or texture regions in the original image. The pixels that have low brightness values correspond to larger ranges with fewer

domains. These pixels are observed on flat regions in the original image. We can visually confirm these properties.

5.2 Retrieval Performance

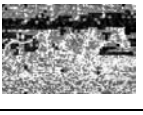
In this section, we evaluate the performance of our retrieval method using DR images and the directional correlogram. We developed a retrieval system with a user-interface generated by a CGI program in C and PHP languages. Users can retrieve images by inputting a query image or randomly selecting an image from some database. Fig. 6 shows the example results given by the system. The retrieved images follow the query image. The corresponding DR images are also outputted.



Fig. 6. Example retrieval results of the prototype system

Comparison of Normal and Directional Correlograms. We compare the performances of a normal correlogram and a directional correlogram. A normal correlogram has 96 dimensions (4 distances × 24 brightness levels). A directional correlogram –each distance having four directions– has 384 dimensions (4 directions × 4 distances × 24 brightness levels). The process time required to create the index file from 1,000 DR images using the normal correlogram is 10.2 s and from the directional correlogram is 26.6 s.

Table 1. Images in the ten categories of [9] and their DR images

Category name	Original image	DR image
Africans		
Beaches		
Buildings		
Buses		
Dinosaurs		
Elephants		
Flowers		
Horses		
Mountains		
Food		

We evaluate the performance of the retrieval system using two indices: the average precision in the first N retrieved images and the average rank of all the correctly retrieved images. Let the query image Q belong to image category A, and rank(I_i) denote the rank of retrieved image I_i for Q. $Y(Q)$ denotes the set of correctly retrieved images for Q: $Y(Q) = \{I_i \mid \text{rank}(I_i) \leq N, I_i \in A\}$. Then, the two indices are defined as

– Average precision

$$P(Q) = \frac{|Y(Q)|}{N}, \tag{13}$$

– Average rank

$$R(Q) = \frac{1}{N_A} \sum_{i=1}^{N_A} \text{rank}(I_i). \tag{14}$$

Here, N_A is the number of images in category A ($N_A = 100$ for all the values of A used here). For a system that randomly retrieves images, the average values of precision and rank are approximately 0.1 and 500, respectively. For ideal retrieval systems, these values are approximately 1.0 and 50, respectively.

Table 2 shows the average precisions ($N = 10$ and $N = 30$), and Table 3 shows the average ranks. From both these results, we can confirm that the performance of the directional correlogram is marginally better than the normal correlogram. The average time required to process one query image using the normal correlogram is 0.11 s, and the directional correlogram is 0.38 s. From these results, we suggest that a directional correlogram can be considered to be a practical option.

Table 2. Average precisions of the normal correlogram and directional correlogram (%)

	$N = 10$		$N = 30$	
	Normal	Directional	Normal	Directional
Africans	45.1	46.7	34.8	36.2
Beaches	32.4	33.5	23.7	25.1
Buildings	48.8	49.9	39.1	40.0
Buses	66.1	73.9	55.6	63.6
Dinosaurs	94.9	95.4	90.8	91.7
Elephants	36.4	37.8	25.6	26.4
Flowers	85.7	84.5	78.9	76.5
Horses	64.5	66.3	50.0	50.9
Mountains	36.2	36.5	28.4	28.9
Food	41.2	43.8	32.5	34.1
Total	55.1	56.8	45.9	47.3

Table 3. Average ranks of the normal correlogram and directional correlogram

	Normal	Directional
Africans	340	338
Beaches	379	373
Buildings	325	325
Buses	181	166
Dinosaurs	106	97
Elephants	362	361
Flowers	139	146
Horses	357	352
Mountains	313	311
Food	294	292
Total	279	276

Comparison of Fractal and Wavelet Correlograms. We compare the proposed method with the wavelet correlogram. The wavelet correlogram is a retrieval method for wavelet coefficients reflecting the local frequency features of the images. The wavelet correlogram used in [11] has 96 dimensions, and the fractal correlogram has 384 dimensions in this comparison.

Table 4 shows the average precisions and average ranks (the results of the wavelet correlogram are quoted from [11]). In the Beaches, Buses and Elephants categories, the performance of the fractal correlogram are notably inferior to the wavelet correlogram. The images in these categories include outdoor scenes that contain foreground objects in front of complicated backgrounds. While the fractal correlogram in the Dinosaurs category outperforms the wavelet correlogram. The images in this category have simple backgrounds. The total performances of the two methods are almost the same; however, the fractal correlogram appears to prefer non-cluttered images as opposed to the cluttered ones preferred by the wavelet correlogram.

Table 4. Average precisions ($N = 100$) and ranks of the fractal correlogram and wavelet correlogram

	Fractal correlogram		Wavelet correlogram [11]	
	Precision (%)	Rank	Precision (%)	Rank
Africans	25.1	338	29.5	288
Beaches	18.4	373	28.9	341
Buildings	28.6	325	29.3	316
Buses	48.5	166	62.7	113
Dinosaurs	71.9	97	26.2	421
Elephants	19.5	361	30.9	241
Flowers	60.0	146	58.6	150
Horses	31.0	352	36.7	267
Mountains	23.4	311	23.0	335
Food	25.7	292	34.7	242
Total	35.2	276	36.1	271

6 Conclusions

In this paper, we have proposed a new concept of DR images and its application to image retrieval using correlograms. A DR image can be created from a fractal-coded image without any decoding. We visually confirmed the versatility of a DR image. Since DR image creation is independent of the decoding process, we can apply this method to any type of fractal-coded images. We have also proposed a correlogram-based retrieval method for DR images and demonstrated the effectiveness of this method. We believe that the DR image can form the foundation over which fractal-code-based compressed-domain image retrieval can be realized.

References

1. Barnsley, M.F.: Fractals Everywhere. Academic Press, San Diego (1993, 1988)
2. Mandal, M.K., Idris, F., Panchanathan, S.: A critical evaluation of image and video indexing techniques in the compressed domain. Image and Vision Computing **17**(7) (1999) 513–529
3. Huang, J., Kumar, S.R., Mitra, M., Zhu, W.J., Zabih, R.: Spatial color indexing and applications. Int. J. Comput. Vision **35**(3) (1999) 245–268
4. Moghaddama, H.A., Khajoieb, T.T., Rouhib, A., Tarzjana, M.S.: Wavelet correlogram: A new approach for image indexing and retrieval. Pattern Recognition **38** (2005) 2506–2518
5. Jacquin, A.E.: Image coding based on a fractal theory of iterated contractive image transformations. IEEE Trans. on Image Processing **1** (1992) 18–30
6. Wohlberg, B., de Jager, G.: A review of the fractal image coding literature. IEEE Trans. on Image Processing **8**(12) (1999) 1716–1729
7. Fisher, Y., ed.: Fractal Image Compression: Theory and Application. Springer-Verlag New York, Inc. (1995)
8. Haussler, D.: Decision theoretic generalizations of the pac model for neural net and other learning applications. Information and Computation **100**(1) (1992) 78–150
9. (`http://wang.ist.psu.edu/~jwang/test1.tar`)
10. (`http://inls.ucsd.edu/~fisher/Fractals/`)
11. Saadatmand-T., M., Moghaddam, H.: Enhanced wavelet correlogram methods for image indexing and retrieval. In: IEEE International Conference on Image Processing. Volume 1. (2005) 541–544

Cross-Modal Interaction and Integration with Relevance Feedback for Medical Image Retrieval

Md. Mahmudur Rahman[1], Varun Sood[1], Bipin C. Desai[1], and Prabir Bhattacharya[2]

[1] Dept. of Computer Science & Software Engineering, Concordia University, Canada
mah_rahm@cs.concordia.ca
[2] Institute for Information Systems Engineering, Concordia University, Canada*

Abstract. This paper presents a cross-modal approach of image retrieval from a medical image collection which integrates visual information based on purely low-level image contents and case related textual information from the annotated XML files. The advantages of both the modalities are exploited by involving the users in the retrieval loop. For content-based search, low-level visual features are extracted in vector form at different image representations. For text-based search, keywords from the annotated files are extracted and indexed by employing the vector space model of information retrieval. Based on the relevance feedback, textual and visual query refinements are performed and user's perceived semantics are propagated from one modality to another. Finally, the most similar images are obtained by a linear combination of similarity matching and re-ordering in a pre-filtered image set. The experiments are performed on a collection of diverse medical images with case-based annotation of each image by experts. It demonstrates the flexibility and the effectiveness of the proposed approach compared to using only a single modality or without any feedback information.

1 Introduction

The digital imaging revolution in the medical domain over the past three decades has changed the way physicians diagnose and treat diseases. A large number of digital images of diverse modalities (e.g. X-ray, CT, MRI, PET, etc.) are generated every day in the hospitals with sophisticated image acquisition devices [1]. Currently, the utilization of the medical images is limited due to the lack of effective search methods and keyword-based searches have been the dominating approach for the medical database management [1,2]. Medical images of diverse modalities can not be effectively indexed or organized with only text-based search techniques. Since, it might be difficult to describe some of the distinct visual features with only keywords, which is very subjective in nature and depends on the expertise of the annotator. This motivates the need for effective way to retrieve

* This work was partially supported by NSERC, IDEAS and Canada Research Chair grants.

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 440–449, 2007.
© Springer-Verlag Berlin Heidelberg 2007

relevant images from such repositories based on their visual content, commonly known as content-based image retrieval (CBIR) [3]. In a typical CBIR system, low-level visual features (e.g. color, texture, shape, edge, etc.) are generated in a vector form and stored to represent the query and target images in the database. While much research effort has been made on the development of the CBIR systems in the medical domain, the performances are still limited and there is no real clinical integration yet.

In general, users would like to pose semantic queries in medical databases using textual descriptions and find images relevant to those semantic queries. For example, one should be able to pose a query like *"Find chest X-ray images with tuberculosis"*. This will be difficult if not impossible with the current CBIR technologies, which may easily find images of chest X-ray, however, it will hardly be able to distinguish between tuberculosis and bronchitis in the lung without any additional information. Complementing an image with words may provide significant semantics. Adding textual information to the system will solve the problem at the first hand. Adding visual information to the text retrieval will also help to distinguish specific visual only features. As an example, for a query like *"Find chest CT images with micro nodules"*, by using merely keywords will only succeed when such an image is sufficiently annotated with the keyword *"nodules"*. However, it would be more appropriate to perform the search based on the image content (e.g., textural properties) in this case, which might describes different types of nodule structures easily. Hence, integration of textual information to a CBIR system or image content information to a text retrieval system might help to close the semantic gap and improve the retrieval performance.

Image retrieval based on multi-modal information sources has been a popular research issue in the last few years [4,5,6]. A simple approach in this direction is to conduct text and content-based retrieval separately and merging the retrieval results of the two modalities [4,5] or combine the textual and visual statistics in a single index vector for content based search [6]. Another approach prompt the user for feedback on the retrieval results and then use this feedback on subsequent retrievals with the goal of increasing retrieval performance. For example, in [4], the well known Rocchio algorithm [7] is utilized in a modified way to incorporate both text and image feature contents. In [5], relevance feedback is utilized to select the appropriate Minkowski metrics and adjust weights to different modalities. Motivated by the above relevance feedback paradigm, this paper presents an interactive framework for medical image retrieval which allows cross-modal (text and image) query by propagating user's perceived semantics from one modality to another. This framework would allow users enough flexibility in searching images and finally provides the retrieval results by integrating the results of the both modalities.

2 Retrieval Approach Based on Image Content

The performance of the CBIR systems depends on the underlying image representation usually in the form of a feature vector [3]. To generate the feature

vectors of complementary nature, low-level color, texture and edge-based features are extracted from different image representations. For global image feature, MPEG-7 based Edge Histogram Descriptor (EHD), $\mathbf{f}^{\mathrm{EHD}}$ and Color Layout Descriptor (CLD), $\mathbf{f}^{\mathrm{CLD}}$ are extracted [8]. To represent the image feature at the semi-global level, a grid based approach is taken into account to divide the images in five overlapping sub-images. These sub-images are obtained by first dividing the entire image space into 16 non overlapping sub-images. From there, four connected sub-images are grouped to generate five different clusters of overlapping sub-regions. The moment-based color and texture features are extracted from each of the sub-region. For moment-based color feature, the first (mean) and second (standard deviation) central moments of each color channel in HSV color space are extracted. Texture features are extracted from the grey level co-occurrence matrix (GLCM) [9]. GLCM is defined as a sample of the joint probability density of the gray levels of two pixels separated by a given displacement. Second order moments, such as energy, maximum probability, entropy, contrast and inverse difference moment are measured based on the GLCM. Color and texture feature vectors are normalized and combined to form a joint feature vector $\mathbf{f}_{\mathrm{r}}^{\mathrm{SG}}$ of 11-dimensions (6 for color and 5 for texture) for each sub-region r and finally obtained a 55-dimensional (5×11) semi-global feature vector $\mathbf{f}^{\mathrm{SG}}$.

Images in the database may vary in sizes for different modalities or within the same modality and may have translations. Resizing them into a thumbnail of a fixed size may overcome the above difficulties. Reduction in size also may reduce some noises due to the artifacts presents in the images, although it may introduce distortion. These kinds of approaches are extensively used in face or finger-print recognition and have proven to be effective. For scaled-based feature vector, each image is converted to a gray-level image and down scaled to 64×64 regardless of the original aspect ratio. Next, the down-scaled image is partitioned further with a 16×16 grid to form small blocks of (4×4) pixels. The average gray value of each block is measured and concatenated to form a 256-dimensional feature vector $\mathbf{f}^{\mathrm{Scaled}}$. By measuring the average gray value of each block, it can cope with global or local image deformations to some extent and adds robustness with respect to translations and intensity changes.

Now, for comparing query image Q_I and target image T_I in the database based on global features, a weighted Euclidean distance measure is used as

$$\mathrm{DIS}_{\mathrm{global}}(Q_I, T_I) = \omega_{\mathrm{CLD}}\mathrm{D}_{\mathrm{CLD}}(Q_I, T_I) + \omega_{\mathrm{EHD}}\mathrm{D}_{\mathrm{EHD}}(Q_I, T_I), \tag{1}$$

where, $\mathrm{D}_{\mathrm{CLD}}(Q_I, T_I) = ||\mathbf{f}_{\mathrm{Q}}^{\mathrm{CLD}} - \mathbf{f}_{\mathrm{T}}^{\mathrm{CLD}}||^2$ and $\mathrm{D}_{\mathrm{EHD}}(Q_I, T_I) = ||\mathbf{f}_{\mathrm{Q}}^{\mathrm{EHD}} - \mathbf{f}_{\mathrm{T}}^{\mathrm{EHD}}||^2$ are the Euclidean distance measures for CLD and EHD feature vector respectively and ω_{CLD} and ω_{EHD} are weights for each feature distance measure subject to $\omega_{\mathrm{CLD}} + \omega_{\mathrm{EHD}} = 1$ and adjusted as $\omega_{\mathrm{CLD}} = 0.4$ and $\omega_{\mathrm{EHD}} = 0.6$ in the experiment. For scaled feature, we also use the Euclidean distance measure as

$$\mathrm{DIS}_{\mathrm{Scaled}}(Q_I, T_I) = ||\mathbf{f}_{\mathrm{Q}}^{\mathrm{Scaled}} - \mathbf{f}_{\mathrm{T}}^{\mathrm{Scaled}}||^2 \tag{2}$$

The semi-global distance measure is described as

$$\mathrm{DIS}_{\text{semi-global}}(Q_I, T_I) = \frac{1}{5}\sum_{r=1}^{5} \mathrm{D}_r(\mathbf{f}_{\mathrm{Q}_r}, \mathbf{f}_{\mathrm{T}_r}) \tag{3}$$

where, $\mathrm{D}_r(\mathbf{f}_{\mathrm{Q}_r}, \mathbf{f}_{\mathrm{T}_r}) = ||\mathbf{f}^{\mathrm{SG}}_{\mathrm{Q}_r} - \mathbf{f}^{\mathrm{SG}}_{\mathrm{T}_r}||^2$ represents the Euclidean distance measure of each region feature vector.

The overall image level similarity is measured by fusing of a weighted combination of individual similarity measures. Once the distances are measured as above, the following function is used to transform them into similarity measures as $\mathrm{S}(Q_I, T_I) = \exp^{-\mathrm{DIS}(Q_I,T_I)/\sigma_{\mathrm{DIS}(Q_I,T_I)}}$, where $\sigma^2_{\mathrm{DIS}(Q_I,T_I)}$ is the distance variance computed for each distance measure separately over a sample image set. After the similarity measures of each representation are determined as $\mathrm{S}_{\text{global}}(Q_I, T_I)$, $\mathrm{S}_{\text{scaled}}(Q_I, T_I)$ and $\mathrm{S}_{\text{semi-global}}(Q_I, T_I)$, they are aggregated or fused into a single similarity matching function as follows:

$$\mathrm{S}_{\text{image}}(Q_I, T_I) = w_{\mathrm{g}}\mathrm{S}_{\text{global}}(Q_I, T_I) + w_{\mathrm{s}}\mathrm{S}_{\text{scaled}}(Q_I, T_I) + w_{\mathrm{sg}}\mathrm{S}_{\text{semi-global}}(Q_I, T_I) \tag{4}$$

Here, w_{g}, w_{s}, and w_{sg} are non-negative weighting factors of different feature level similarities, where $w_{\mathrm{g}} + w_{\mathrm{s}} + w_{\mathrm{sg}} = 1$ and selected as $\omega_{\mathrm{g}} = 0.4$, $\omega_{\mathrm{s}} = 0.2$, and $\omega_{\mathrm{sg}} = 0.4$ for the experiments.

3 Retrieval Approach Based on Textual Content

To incorporate textual information in the retrieval framework, we have decided to use the popular vector space model of information retrieval [10]. Each image in the collection is attached to a manually annotated case or lab report in a XML file. Figure 1, shows an example image (e.g., X-ray of lung with tuberculosis) and its annotation with several XML tags in the right side. The most important information about the case is mainly contained inside the *description* tag. Hence, information from only this tag is extracted for each XML file and preprocessed by removing stop words that are considered to be of no importance for the actual retrieval process. Subsequent to the stopping process, the remaining words are reduced to their stems, which finally form the index terms or keywords.

Vector space model treats documents and queries as vectors in a N-dimensional space, where N is the number of keywords in the collection. So, each annotated file T_D of image T_I in a collection is represented as a vector [10]:

$$\mathbf{f}_{T_D} =< w_{1T_D}, \cdots, w_{NT_D} > \tag{5}$$

where the element w_{iT_D} represents the weights of keyword w_i appearing in document T_D. The element w_{iT_D} can be weighted in a variety of ways. We followed the popular TF-IDF weighting scheme. Both the global and the local weights are considered in this approach [10]. The local weight is denoted as $L_{i,j} = log(f_{i,j}) + 1$, where $f_{i,j}$ is the frequency of occurrence of keyword w_i in



```
<IMAGE>
<GlobalID>934020</GlobalID>
<FileName>LUNG</FileName>
<Title>LUNG</Title>
<ContributeDate>02/06/2004</ContributeDate>
<Annotated>false</Annotated>
<Inappropriate>false</Inappropriate>
<Archived>false</Archived>
<Private>false</Private>
<Description>LUNG: Case# 33633:
PRIMARY TUBERCULOUS. Three year old
with cough and fever. Chest x-ray reveals
right upper lobe consolidation with scattered
air bronchograms. There is hilar fullness
bilaterally and in the right paratracheal region.
No pleural effusion is identified.
case.</Description>
<SourceCollection>PEIR - University of Alabama
at Birmingham Department of Radiology
</SourceCollection>
<Path>Images/00134020.tif</Path>
</IMAGE>
```

Fig. 1. An example image and associated XML file

document T_D. The global weight G_i is denoted as inverse document frequency as $G_i = log(M/M_i)+1$, for $i = (1, \cdots, , N)$, where M_i be the number of documents in which w_i is found and M is the total number of documents in the collection. Finally, the element w_{iT_D} is expressed as the product of local and global weight as $w_{iT_D} = L_{i,j} * G_i$. In vector space model, the direction or angle of the vectors are a more reliable indication of the semantic similarities of the documents. Hence, we adopt the cosine similarity measure between normalized feature vectors of the textual query Q_D and document T_D as a dot product as follows [10]:

$$\mathrm{S}_{\mathrm{text}}(Q_D, T_D) = \sum_{i=1}^{N} w_{iQ_D} * w_{iT_D} \tag{6}$$

The vector space model returns ranked documents in an order. Such an ordering will determine the similarity of a document to the query and will be useful enough when we combine the result from both the text and image based retrieval as discussed in section 6.

4 Textual and Visual Query Refinement by Relevance Feedback

Information retrieval in general is an unsupervised or isolated process as there is no real human-computer interaction, except only when the user submit a query (either with keywords or example images) to the system. However, the performance would be improved if users have some indication of relevant and irrelevant items to use in the ranking, commonly known as relevance feedback [11,12]. It prompts the user for feedback on retrieval results and then use this feedback on subsequent retrievals with the goal of increasing retrieval performance. In a medical image retrieval system, the user at first may want to search images with keywords as it is more convenient and semantically more appropriate. However, a short query with few keywords may not enough to incorporate the user

perceived semantics to the retrieval system. Hence, a query expansion process is required to add additional keywords and re-weight the original query vector. Query expansion is a standard technique for reducing ambiguity in the information retrieval [12]. In this work, the textual query refinement is done based on the well known Rocchio algorithm [7]. The formula for the modified query vector is as follows:

$$\mathbf{f}^m_{Q_D} = \alpha\ \mathbf{f}^o_{Q_D} + \beta \frac{1}{|D_r|} \sum_{\mathbf{f}_{T_D} \in D_r} \mathbf{f}_{T_D} - \gamma \frac{1}{|D_{nr}|} \sum_{\mathbf{f}_{T_D} \in D_{nr}} \mathbf{f}_{T_D} \tag{7}$$

where, $\mathbf{f}^m_{Q_D}$ and $\mathbf{f}^o_{Q_D}$ are the modified and original query vectors, D_r and D_{nr} are the set of relevant and irrelevant document vectors and α, β, and γ are weights. This algorithm generally moves a new query point toward relevant documents and away from irrelevant documents in feature space [7].

Visual features of images also play an important part in a multi-modal system. Therefore, we also need to perform relevance feedback with the image query for better precision. Our idea of image query refinement based on the visual features is the following: user will provide the initial image query vector $\mathbf{f}^x_{\mathrm{Q}(0)}, x \in \{EHD, CLD, SG, Scaled\}$ for each feature to retrieve K most similar images based on the similarity measure function in equation (4). If the user is not satisfied with the result, then he/she will select a set of relevant or positive images compared to the query image. It is assumed that, all the positive feedback images at some particular iteration will belong to the user perceived semantic category and obey the Gaussian distribution to form a cluster in the feature space. We consider the rest of the images as irrelevant and they may belong to different semantic categories. However, we do not consider the negative images in this image-based feedback algorithm. Let, N_{r} be the number of relevant images at iteration k and $\mathbf{f}^x_{\mathrm{T}_j} \in \Re^d$ is the feature vector that represents j-th image for $j \in \{1, \cdots, N_{\mathrm{r}}\}$, then the new query point at iteration k is estimated as $\mathbf{f}^x_{\mathrm{Q}(k)} = \frac{1}{N_{\mathrm{r}}} \sum_{j=1}^{N_{\mathrm{r}}} \mathbf{f}^x_{\mathrm{T}_j}$ as the mean vector of positive images and covariance matrix is estimated as $\mathbf{C}^x_{(k)} = \frac{1}{N_{\mathrm{r}}-1} \sum_{j=1}^{N_{\mathrm{r}}} (\mathbf{f}^x_{\mathrm{T}_j} - \mathbf{f}^x_{\mathrm{Q}(k)})(\mathbf{f}^x_{\mathrm{T}_j} - \mathbf{f}^x_{\mathrm{Q}(k)})^T$. However, singularity issue will arise in covariance matrix estimation if fewer than $d+1$ training samples or positive images are available as will be the case in user feedback images. So, we add regularization to avoid singularity in matrices as follows[13]:

$$\hat{\mathbf{C}}^x{}_{(k)} = \alpha \mathbf{C}^x_{(k)} + (1 - \alpha)\mathbf{I} \tag{8}$$

for some $0 \leq \alpha \leq 1$ and $\mathbf{I}$ is the $d \times d$ identity matrix.

After generating the mean vector and covariance matrix for a feature, we adaptly adjust the Euclidean distance measures with the following Mahalanobis distance measure [14] for the feature x of image Q^x_I and T^x_I as:

$$\mathrm{DIS}_{\mathrm{Maha}}(Q^x_I, T^x_I) = (\mathbf{f}^x_{\mathrm{Q}(k)} - \mathbf{f}^x_{\mathrm{T}})^T \hat{\mathbf{C}}^x{}^{-1}_{(k)} (\mathbf{f}^x_{\mathrm{Q}(k)} - \mathbf{f}^x_{\mathrm{T}}) \tag{9}$$

The Mahalanobis distance differs from the Euclidean distance in that it takes into account the correlations of the data set and is scale-invariant, i.e. not

dependent on the scale of measurements [14]. Basically, at each iteration of the relevance feedback, we generate several mean vectors and covariance matrices for all the individual feature vectors and use it in the distance measure of equation (9). Finally, a ranked based retrieval result is obtained by applying the fusion-based similarity function of equation (4). So, the above image-based relevance feedback approach performs both the query refinement and similarity matching adjustment at the same time.

5 Cross-Modal Interaction and Integration

Various techniques have been proposed to combine or integrate the results from the text and image modalities either simultaneously or sequentially [4,5,6]. This section describes about how to interact with both the modalities in a user's perceived semantical and sequential way. We have considered a pre-filtering and re-ranking approach based on the image search in the filtered image set which is obtained previously by the textual search. This approach might be more appropriate and effective in medical domain as majority of the images are categorized by their disease related names, such as cancer images of the brain or lung. Hence, it would be more appropriate to search the images with the keyword "cancer" and then searching visually similar images of the brain or lung on the top returned images by the textual search. In this method, combining the results of the text and image based retrieval is a matter of re-ranking or re-ordering of the images in a text-based pre-filtered result set. The steps involved in this approach are as follows:

Step 1: For a given query topic or annotation Q_D, perform a textual search and rank the images based on the ranking of the associated annotation files by applying $\mathrm{S}_{\mathrm{text}}$ in equation (6).

Step 2: Obtain user feedbacks about relevant and irrelevant images for the textual query refinement.

Step 3: Calculate the optimal textual query vector $\mathbf{f}^m_{Q_D}$ by applying equation(7) for the text-based search and re-submit it again.

Step 4: Continue the textual feedback process until the user is satisfied or switch to visual only search.

Step 5: Perform visual only search with a initial query image Q_I in a filtered list L obtained from the previous step and rank the images by applying $\mathrm{S}_{\mathrm{image}}$ in equation (4).

Step 6: Obtain user feedbacks about the relevant images for the visual query refinement.

Step 7: Calculate $\mathbf{f}^x_{Q(k)}$ and $\mathbf{C}^x_{(k)}$ for each visual feature,
$x \in \{EHD, CLD, SG, Scaled\}$ for the content-based search in next iteration in L and re-rank the images.

Step 8: Continue the visual feedback iterations until the user is satisfied or the system converges.

Step 9: Finally, combine the image scores or merge the result lists obtained from both the text and image-based search as a linear combination:

$$\mathrm{S}(Q,T) = w_{\text{text}}\mathrm{S}_{\text{text}}(Q_D,T_D) + w_{\text{image}}\mathrm{S}_{\text{image}}(Q_I,T_I) \tag{10}$$

where, $Q = \{Q_D, Q_I\}$, $T = \{T_D, T_I\}$, w_{text} and w_{image} are weighted coefficients subject to $0 \leq w_{\text{text}}, w_{\text{image}} \leq 1$, $w_{\text{text}} + w_{\text{image}} = 1$. The resulting similarity function $S(Q,T)$ serves for the final ranking of the images.

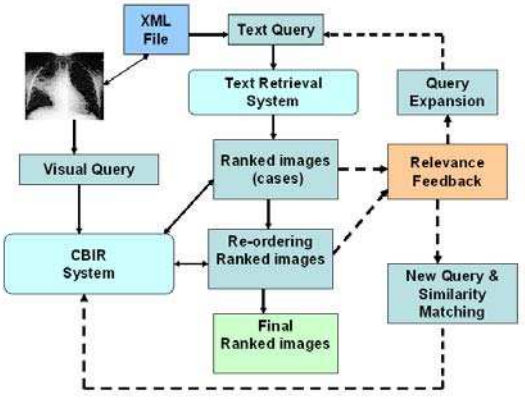


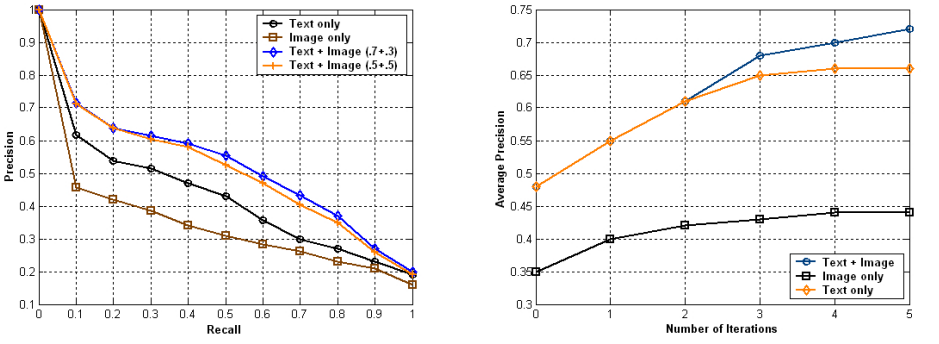
Fig. 2. Process flow diagram of the integration approach

However, the ordering of textual and visual searches might be different for other databases or might depend on user's search criteria. Figure 2 shows the process flow diagram of the above multi-modal interaction and re-ranking approach in general.

6 Experiments and Results

To measure the accuracy of the proposed multi-modal retrieval approach, the experiments are performed in a medical image collection where images are categorized with different diseases, body parts and imaging modalities (such as X-ray images of lung cancer, pathology images of lung cancer, chest X-ray images with tuberculosis, CT images of prostate cancer, etc.). The data set contains around 3000 medical images with 20 different categories and each image has an associated annotation of the case or lab report. The image collection is actually a subset of the large ImageCLEFmed collection [15], where we manually categorize it with the above properties so that both the textual and visual search techniques might be needed for effective retrieval.

For a quantitative evaluation of the retrieval results, the performances are compared based on the precision-recall curves and average precisions on different number of iterations for the RF operations. We have randomly selected five



(a) Precision-recall without RF (b) Average precision with RF

Fig. 3. Accuracy comparison with and without RF

images from each category (e.g., a total of 100 images) as the initial visual query images and their annotations as the initial textual queries. A retrieved image is considered a correct match if it is in the same category as the query image. Figure 3(a) presents the precision-recall curves for different modalities without any RF. For the multi-modal retrieval, texts and images are combined simultaneously with a linear combination of different weights as shown in Figure 3(a). It is clear that the best performance is always achieved when search is based on multi-modal retrieval and when textual modality has more contribution in the similarity matching function. To evaluate the effects of RF, we compared the average precision for the same query set with five iteration rounds. The average precision is based on the top 30 returned images for each query and the feedbacks are performed manually. For image or text only RF evaluations, we utilized the image and text based RF approaches respectively in all the iteration rounds as described in section 4. For the cross-modal RF, we have performed first two iterations for the textual query refinement and next three iterations in a filtered set of $L = 1000$ images for the visual query refinement with a combination of weight as $w_{\text{text}} = 0.7$ and $w_{\text{image}} = 0.3$ as described in section . As shown in Figure 3(b), we obtained better precision by applying visual only feedback in the text-based prefilterd images after two iterations compared to the text only feedback result. There is also a large visible gap between the image only RF and text or cross-modal based RF. This justifies our initial assumption about the requirement of an interactive multi-modal system for effective image retrieval.

7 Conclusions

In this paper, a novel framework for multi-modal interaction and integration is proposed for a diverse medical image collection with associated annotation of the case or lab reports. Unlike in many other approaches, where the search is performed with a single modality and without any user interaction, we proposed to involve the users directly in the retrieval loop and integrate the results obtained

from both the text and imaging modalities. Experiments are performed on a medical image collection with known categories or ground truth, which showed promising results.

References

1. Müller H., Michoux, N., Bandon, D., Geissbuhler, A. : A review of content-based image retrieval applications–clinical benefits and future directions. International Journal of Medical Informatics. **73** (2004) 1–23
2. Tagare, H. D., Jafe, C., Duncan, J. : Medical image databases: A content-based retrieval approach. Journal of the American Medical Informatics Association. **4 (3)** (1997) 184–198
3. Smeulder, A., Worring, M., Santini, S., Gupta, A., Jain, R.: Content-Based Image Retrieval at the End of the Early Years. IEEE Trans. on Pattern Anal. and Machine Intell. **22** (2000) 1349–1380
4. Lu, Y., Zhang, H., Wenyin, L., Hu, C. : Joint semantics and feature based image retrieval using relevance feedback. IEEE transactions on multimedia. **5 (3)** (2003) 339–347
5. Sclaroff, S., Cascia, M. L., Sethi, S., Taycher, L. : Unifying Textual and Visual Cues for Content-Based Image Retrieval on the World Wide Web. Computer Vision and Image Understanding. **75** (1999) 86-98
6. Rong, Z., Grosky, W. I. : Narrowing the semantic gap – improved text-based web document retrieval using visual features. IEEE Transactions on Multimedia. **4 (2)** (2002) 189–200
7. Rocchio, J. J. : Relevance Feedback in Information Retrieval. In: The Smart Retrieval System, Prentice Hall (1979) 313–323
8. Manjunath, B. S., Salembier, P., Sikora, T. (eds.) : Introduction to MPEG-7 – Multimedia Content Description Interface. John Wiley Sons Ltd. (2002) 187–212
9. Haralick, R. M., Shanmugam, Dinstein, I.: Textural features for image classification, IEEE Trans System, Man, Cybernetics. **SMC-3** (1973) 610-621
10. R. Baeza-Yates and B. Ribiero-Neto : Modern Information Retrieval, Addison Wesley, (1999).
11. Rui, Y., Huang, T. S. : Relevance Feedback: A Power Tool for Interactive Content-Based Image Retrieval , IEEE Circuits Syst. Video Technol., **8** (1999)
12. Salton, G., Buckley, C. : Improving retrieval performance by relevance feedback. JASIS. **41 (4)** (1990) 288–297
13. Friedman, J.: Regularized Discriminant Analysis., Journal of American Statistical Association, **84** (2002) 165–175
14. Fukunaga, K.: Introduction to Statistical Pattern Recognition. 2nd edn. Academic Press Professional, Inc. San Diego, CA, USA (1990)
15. Müller, H., Deselaers, T., Lehmann, T., Clough, P., Hersh, W. : Overview of the ImageCLEFmed 2006 medical retrieval and annotation tasks. CLEF working notes. Alicante, Spain, Sep., (2006)

A New Multi-view Learning Algorithm Based on ICA Feature for Image Retrieval

Fan Wang and Qionghai Dai

Department of Automation, Tsinghua University, Beijing 100084, China
wangfan@mails.tsinghua.edu.cn,
qhdai@mail.tsinghua.edu.cn

Abstract. In content-based image retrieval (CBIR), most techniques involve an important issue of how to efficiently bridge the gap between the high-level concepts and low-level visual features. We propose a novel semi-supervised learning method for image retrieval, which makes full use of both ICA feature and general low-level feature. Our approach can be characterized by the following three aspects: (1) The ICA feature, as proved to be representative of human vision, is adopted as a view to describe human perception; (2) A multi-view learning algorithm is introduced to make the most use of different features and dramatically reduce human relevance feedback needed to achieve a satisfactory result; (3) A new semi-supervised learning algorithm is proposed to adapt to the small sample problem and other special constrains of our multi-view learning algorithm. Our experimental results and comparisons are presented to demonstrate the effectiveness of the proposed approach.

1 Introduction

With the rapid increase of the volume of digital image collections, content-based image retrieval (CBIR) has attracted a lot of research interest in recent years [16]. However, most of the features adopted in the previous approaches are pixel based or extracted by cutting the image into blocks or regions, and further extract feature from the blocks. Therefore, these approaches are mostly concerned with low-level features, such as color, texture, shape, etc., which can not fully represent the human perception. Actually, people do not perceive the images on the level of pixels or blocks, they are always interested in high-level concepts instead of the low-level visual features. As a result, the gap between high-level hidden concepts and low-level visual features has become one of the challenging problems of CBIR systems, due to the rich content but subjective semantics of an image, which can not be fully recognized by computer.

Theoretical studies suggest that primary visual cortex (area V1) uses a sparse code to efficiently represent natural scenes, and each neuron appears to carry statistically independent information [20]. Recent researches have shown that, Principal Component Analysis (PCA), and Independent Component Analysis (ICA) of natural static images produce image representation bases resembling the receptive fields of V1 cells [5]. This kind of results, more specifically ICA

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 450–461, 2007.
© Springer-Verlag Berlin Heidelberg 2007

results, also come from the learning procedure named sparse coding. This coincidence of results was already mathematically justified through the identification of the link between the ICA and Sparse Coding formalisms. The results reported in above-mentioned experiments are well fitted to parametric (Gabor or Wavelets) models which were broadly accepted as approximations for V1 receptive fields [14].

To this day, there have been several approaches that adopt features through ICA to improve the retrieval performance. For example, the paper [9] showed that the PCA and ICA features can be used to construct similarity measures for the image retrieval, and through comparison, the conclusion is made that the ICA basis method outperforms the PCA basis method. In [10], an unsupervised classification algorithm was presented based on an ICA mixture model. This method can learn efficient representation of images of natural scenes, and the learned classes of basis functions yield a better approximation of the underlying distributions of the data.

Based on the former research, it is believed that ICA is able to well discover the basis of human vision. We adopt ICA feature in this paper to further approach the human perception. Instead of simply replace the former general visual features with ICA features, a new utilization of ICA features is proposed. While ICA features is some efficient representation of human vision, the low-level features, such as color or texture, carrying abundant statistical information, are the image representation by computer. In other words, ICA features are the representation of images from human's view, while the low-level features can be regarded as the computer's view. Since both of the two views are valuable for the retrieval system, a multi-view learning algorithm is necessary to fully utilize these features.

A well-know tool to bridge the gap between high-level concepts and low-level features in CBIR is relevance feedback, in which the user has the option of labeling a few of images according to whether they are relevant or not. The labeled images are then given to the CBIR system as complementary queries so that more images relevant to the user query could be retrieved from the database. In recent years, much has been written about this approach from the perspective of machine learning [17], [18], [19], [24]. It is natural that the users will be more willing to see satisfied retrieval results only by once or twice feedback instead of many times of labeling. This limits the amount of available labeled data, and here comes the demand of semi-supervised learning algorithm, which reduce the amount of labeled data required for learning.

Multi-view learning algorithms have been studied for several years, and there exist some significant proposals, i.e. *Co-Training* [3], *Co-Testing* [12], *Co-EM* [13], *Co-retrieval* [21]. However, these methods' performance drops dramatically if the labeled data is limited, and they do not take enough consideration of the characteristics of the data and the views.

In this paper, we propose a new image feature based on ICA expansion, and the distance between ICA features are also defined. We novelly integrate semi-supervised learning method into a multi-view learning framework called

Co-EMT, and ICA features are introduced as one of the views to further improve the retrieval performance.

The rest of the paper is organized as follows: Section 2 introduces how to perform ICA expansion and extract ICA features. The multi-view learning algorithm is described in Section3, followed by the proposed semi-supervised algorithm in each single view detailed in Section 4. Section 5 shows the whole scheme of our CBIR system. The experimental results and some discussions of our algorithm are presented in Section 6. Finally this paper is concluded in Section 7.

2 ICA Feature Extraction

ICA is a recently developed statistical technique which often characterizes the data in a natural way. It can be viewed as an extension of standard PCA, where the coefficients of the expansion must be mutually independent (or as independent as possible) instead of being merely uncorrelated. This in turn implies that ICA exploits higher-order statistical structure in data. The goal of ICA is to linearly transform the data such that the transformed variables are as statistically independent from each other as possible [1], [4].

ICA has recently gained attention due to its applications to signal processing problems including speech analysis, image separation and medical signal processing. So far there have been many kinds of algorithms for ICA expansion. However, some may be computationally demanding or have problem of convergence when dealing with data of high dimensionality. In this paper, we choose a fast and computationally simple fixed-point rule of ICA [8] for image feature extraction in consideration of speed. Furthermore, the convergence of this learning rule can be proved theoretically.

Here we apply the method to computing ICA bases of images, the detailed steps are discussed as follows.

Firstly, the n-dimensional data vectors $x(t)$ were obtained by first taking $n^{1/2} \times n^{1/2}$ sample subimages from the available image database. Here t is from 1 to N, which is the total number of data samples for x. In the formulation of the ICA, the data vector is assumed to be mixed by unknown sources, that is

$$x(t) = As(t) = \sum_{i=1}^{m} s_i(t) a_i \tag{1}$$

here the vector $s(t) = [s_1(t), \cdots, s_m(t)]^T$ contains the m independent components $s_i(t)$ for the data vector $x(t)$. $A = [a_1, \cdots, a_m]$ is a $n \times m$ matrix, whose columns are called features or basis vectors. The number of independent components m is often fixed in advance. In any case, $m \leq n$, and often $m = n$.

Data x is preprocessed to have zero-mean and unit variance.

$$x \leftarrow x - E[x] \tag{2}$$

$$x \leftarrow x \Big/ \sqrt{\|x\|^2} \tag{3}$$

The preprocessed vectors were then whitened using standard PCA so that the resulting vectors $v(t)$ had $n-1$ components (one of the components becomes insignificant because of the subtracted mean). The PCA whitening matrix is of the form $V = D^{-1/2}E^T$, where the columns of the matrix E contain the PCA eigenvectors, and the diagonal matrix D has the corresponding eigenvalues as its elements. Standard PCA is used because it can compress the data vectors into an m-dimensional signal subspace and filter out some noise.

W is defined as the $m \times n$ de-mixing matrix, so that the purpose of the ICA learning is to estimate W in

$$\hat{s}(t) = Wv(t) \tag{4}$$

After this, the generalized fixed-point algorithm described in detail in [8] is applied to finding the independent components of the whitened data vectors $v(t)$. In this algorithm, we first initialize the matrix W by the unit matrix I of the same dimension. The update of w_i, denoting the i-th column of W, and the orthonormalization are performed as follows:

$$w_i^*(k+1) = E\left\{vg\left(w_i(k)^T v\right) - g'\left(w_i(k)^T v\right) w_i(k)\right\} \tag{5}$$

$$w_i(k+1) = w_i^*(k+1)/\|w_i^*(k+1)\| \tag{6}$$

here $E\{\}$ denotes the mathematical expectation, $w_i(k)$ is the value of w_i before the k-th update, while $w_i(k+1)$ is the value after it. In practice it is replaced by sample mean computed using a large number of vectors $v(t)$. The function $g(u)$ can be any odd, sufficiently regular nonlinear function, and $g'(u)$ denotes its derivative. In practice, it is often advisable to use $g(u) = \tanh(u)$ [6]. The convergence of this method was proved in [7].

From w_i we can obtain the estimation for the corresponding basis vector a_i of ICA using the formula

$$\hat{a}_i = ED^{1/2}w_i \tag{7}$$

that is, the estimation of the mixing matrix is

$$\hat{A} = (WV)^{-1} = ED^{1/2}W^T \tag{8}$$

For a new image, we can extract ICA feature from it through mapping it to the basis and getting the coefficients. The image is first sampled by taking $n^{1/2} \times n^{1/2}$ subimages from it for K times. Then the prewhitened n-dimensional data vectors $x(i), i = 1, \cdots, K$ are obtained. The ICA feature for this image can be calculated as $S = WX$, where X is composed by the columns $x(i)$.

3 Multi-view Learning Algorithm

Two kinds of image features are utilized in our system: general low-level feature and ICA feature. As mentioned in Section 1, the general low-level feature representation can be regarded as the view of computer when recognizing the image,

while the ICA feature approximates the view of human. That is to say, an image x can be described by these two features in two views.

Previous research proved that if there exist two compatible and uncorrelated views for a problem, the target concept can be learned based on a few labeled and many unlabeled examples. We found the two views mentioned above partially satisfies the condition after some statistical test. This is the similar situation in many real world multi-view learning problems. We use a robust multi-view algorithm called *Co-EMT* [11] which interleaves semi-supervised and active learning, to handle this problem. It has been proved that *Co-EMT* is robust in harsh conditions when the two views are not completely compatible and uncorrelated.

The algorithm *Co-EMT* includes training step and testing step, which adopt *Co-EM* and *Co-Testing*, respectively. *Co-EM* [13] is a multi-view algorithm that uses the hypothesis learned in one view to probabilistically label the examples in the other view. It can be seen as a probabilistic version of *Co-Training* [3].

Let $V1$ denotes the view of general low-level feature, $V2$ the ICA feature. Denote learning algorithms L, which will be talked about later in Section 4. The *Co-EM* can be described as follows:

Firstly, the algorithm trains an initial classifier h_1 in the view $V1$ based solely on the labeled examples by the learning algorithm L. Then it repeatedly performs the following four-step procedure: (1) use h_1 to probabilistically label all unlabeled examples and obtain their labels New_1; (2) in $V2$, learn a new maximum a posterior (MAP) hypothesis h_2 based on the labels New_1 learned in the previous step; (3) use h_2 to probabilistically label all unlabeled examples again, and get New_2; (4) in $V1$, learn a new MAP hypothesis h_1 based on the labels New_2 labeled in the previous step. These steps are repeated for several iterations. At the end, a final hypothesis is created which combines the prediction of the classifiers learned in each view.

Since solely depending on the system's automatic iterations is insufficient for learning, the user's feedback should be added to input new useful information to the system. Here *Co-Testing* [12] is introduced as an active learning algorithm, and *Co-EM* is interleaved with *Co-Testing* to form the *Co-EMT* algorithm. After running *Co-EM* for several iterations on both labeled and unlabeled examples, the two hypotheses in two views have been trained sufficiently. The data points on which the hypotheses on two views disagree the most consist the contention set, which means we are least confident on the label of these samples using the two hypotheses. Labeling these points by the user can provide the system with the most information from the user's perception, thereby enhance the effectiveness of the learning algorithm.

4 Proposed Semi-supervised Learning Algorithm in Each Single View

In the view of general low-level feature, we use Euclidean distance as the distance measure between any two images x_i, x_j:

$$d(x_i, x_j) = \begin{cases} \|x_i - x_j\|_2 \; if \; \|x_i - x_j\|_2 < \varepsilon \\ \infty \qquad\qquad otherwise \end{cases}$$

where ε is a positive threshold to assure the sparsity of the distance matrix. Since the images in positive set R have been labeled relevant, we set the distance between each of them as zero, that is, $d(x_i, x_j) = 0, \forall x_i, x_j \in R$.

In the view of ICA feature, we also need distance measurements between each pair of the features. According to the equations in Section 2, we firstly use the labeled positive examples to train the basis vectors which expand the ICA subspace corresponding to the positive set. For an image x, we sample subimages from it, and map the subimages to the acquired basis vectors to obtain the $m \times K$ coefficient matrix S, which we treat as the feature of image x. Here K is the number of patches sampled from x.

Each column of S is a vector in m-dimensional space, and all the K columns in the feature S of image x form a point set in m-dimensional space, with each of the point in it describes one block of image x. As a result, we can calculate the distance between two images x_i and x_j as distance between the two point sets S_i and S_j.

We use the mean of distance between each of the K points in S_i and S_j as the distance measure. This measure has been widely used in cluster methods, and proved to be robust to noise. The distance between images x_i and x_j in ICA space can be formulated as:

$$d(x_i, x_j) = \frac{1}{K^2} \sum_{l=1}^{K} \sum_{m=1}^{K} (S_l^i, S_m^j) \tag{9}$$

Where S_l^i denotes the l-th column of S_i, S_m^j denotes the m-th column of S_j, and $(\cdot, \cdot)$ denotes inner product of two vectors.

After some easy formulation, we can simplify the distance to

$$d(x_i, x_j) = \frac{1}{K^2} sum(S_i S_j^T) \tag{10}$$

where $sum(\cdot)$ denotes the sum of all the elements of a matrix.

It is easy to see that this distance measurement is quite computationally efficient compared to L2 norm distance between S_i and S_j.

Under the assumption that the images lay on smooth manifolds embedded in image space, and the labeled data is limited, we use a semi-supervised algorithm L to learn the hypothesis in each view.

The original method proposed in [23] is as follows:

Given a set of point $\mathrm{X} = \{x_1, \cdots, x_q, x_{q+1}, \cdots, x_n\}$, $f = [f_1, \cdots, f_n]^T$ denotes a ranking function which assigns to each point x_i a ranking value f_i. The vector $y = [y_1, \cdots, y_n]^T$ is defined in which $y_i = 1$ if x_i has a label and $y_i = 0$ means x_i is unlabeled.

A connected graph with all the images as vertices is constructed and the edges are weighted by the matrix B where $B_{ij} = \exp\left[-d^2(x_i - x_j)/2\sigma^2\right]$ if $i \neq j$ and $B_{ii} = 0$ otherwise. $d(x_i - x_j)$ is the distance between x_i and x_j. B is normalized

by $S = D^{-1/2}BD^{-1/2}$ in which D is a diagonal matrix with its (i,i)-element equal to the sum of the i-th row of B. All points spread their ranking score to their neighbors via the weighted network. The spread is repeated until a global stable state is achieved.

This label propagation process actually minimizes an energy function with a smoothness term and a fitting term. The smoothness term constrains the change of labels between nearby points, and the fitting term forces the classifier not to change too much from the initial label assignment. It has been proved that this iteration algorithm has a closed form of solution $f^* = (I - \alpha S)^{-1} y$ to directly compute the ranking scores of points [22]. From this formula we can discover that the initial value f_0 has no effect on the final result, which is solely determined by y, S and α.

Down to the case of our problem, there are another two issues to take into consideration. Firstly, the scale of our problem is very large, so we prefer using iteration algorithm, instead of direct inverse, which is more time consuming. Our experiment shows that a few iterations are enough to converge and yield high quality ranking results. Secondly, at the beginning of learning in one view, all the examples have been assigned ranking scores by the other view. The examples tending positive have values close to +1, while those tending negative have values near -1. In these scores, some could be changed, but those marked as +1 or -1 by the user in relevance feedback should not be changed since they are absolutely fixed. That means we have prior knowledge about the confidences of the labels proportional to their respective absolute values. Considering that y_i stands for whether the example has a label in the standard semi-supervised algorithm, which can also be regarded as the confidence, we set $y = [y_1, \cdots, y_n]^T$ as the ranking scores obtained from the other view. Since initial f_0 is not crucial in iteration, it can also be set as equal to y at the beginning.

Based on the predefined parameters, iterate $f(t+1) = \alpha S f(t) + (1-\alpha) y$ for several times, Here *alpha* is a parameter in $(0, 1)$, which specifies the relative contributions to the ranking scores from neighbors and the initial ranking scores. At last, each point x_i is ranked according to its final ranking scores f_i^* (largest ranked first).

The result of the propagation f_i^* is normalized separately as the $h_1(x)$ or $h_2(x)$ mentioned above in Section 3, which gives the probability that the sample is positive in separate views. Then we can deduce the disagreement of them by simply calculate their difference.

5 The Combined Scheme for the Proposed CBIR System

The integrated framework will be described in this section. First, the positive image set R^+ is initialized as only the query image and the negative set R^- as empty. The labels of all the images are initialized as zero. The times for relevance feedback is set as N. Other symbols are defined in Section 3. The following steps are performed:

(1) On the positive set R^+, do ICA expansion and the basis vectors are obtained;

(2) Based on general low-level feature, for each image $x_i \in R^+$, we find its k-nearest neighbors $C_i = \{y_1, y_2, \cdots, y_k\}$, then we get the candidate image set $C = C_1 \cup C_2 \cup \cdots \cup C_{|R^+|} \cup R^+ \cup R^-$. T and U are labeled and unlabeled examples in C, respectively, that is, $C = T \cup U$. The labels of images in R^+ are changed to +1, and those in R^- to -1;

(3) Run *Co-EM(L,V1,V2,T,U,k)* in candidate set C for k times to learn h_1 and h_2; L is the algorithm proposed in Section 4 and *Co-EM* can be referred to Section 3. A typical value of 5 for k is enough to let the *Co-EM* algorithm converge to a stable point;

(4) Sort the examples $x \in U$ according to the absolute value of $(h_1(x) - h_2(x))$, those with large values are defined as contention points, that means, the two views are less confident of the labels of these examples. Select several examples with the largest value among contention points and ask user to label them;

(5) The positive examples newly labeled by user are removed from U to R^+, and the negative ones to R^-;

(6) $N = N - 1$. if $N > 0$, return to step (1);

(7) Sort the examples according to $h_1 + h_2$ in descending order, and the final retrieval results are returned as the first several examples with largest value of $h_1 + h_2$, that means, the two views both have high confidence on those examples.

The candidate set is necessary when the whole image database is so large that the computation in the whole set will be time-consuming and needless. Additionally, in each iteration, some new examples are added into positive set, so there is no need to recalculate the basis vectors. When we do the ICA expansion, the de-mixing matrix W can be initialized as the matrix obtained in the previous iteration, and updated only by the subimages sampled from the newly added examples in positive set. This incremental learning advantage benefits from the characters of ICA, and guarantees the speed of our system.

6 Experiments and Discussions

The image database used in our experiments includes 5000 real-world images from Corel gallery. All the images belong to 50 semantic concept categories and 100 images in each category. The following features, which are totally 515 dimensions, are adopted as the general low-level feature: the 256-dimensional color histogram in HSV color space; the 9-dimensional color moments in LUV color space; color coherence in HSV space of 128-dimension; the 10-dimensional coarseness vector; 8-dimensional directionality; and the wavelet texture feature, 104 dimensions.

To investigate the performance of our system , the following three algorithms are implemented and compared:

(a) Our proposed multi-view learning algorithm, one view is ICA feature and the other is general low-level feature;

(b) Our proposed multi-view learning algorithm, the two views are both general low-level feature;

(c) Combine ICA feature and general low-level feature together as a single view, just adopt the semi-supervised algorithm proposed in Section 4.

Each experiment is performed for 500 times, 10 times in each category. To simulate the real query process, the images are randomly selected from each category as the queries. The number of feedback rounds is set as 4 and in each round 3 images are returned as contention points for labeling. Here the system makes the judgement and gives the feedback automatically to simulate the user's action. The retrieval accuracy is defined as the rate of relevant images retrieved in top 20 returns. Whether two images are relevant or not is determined automatically by the ground truth.

The final averaged accuracy of retrieval results are shown in Fig.1, from which we can conclude that, our method (a) outperformed the other two experiments. The first point on each curve represents the accuracy obtained in the first round before any relevance feedback. As the round of feedback increases, the retrieval accuracy is getting higher. One point that has to be mentioned is that, the number of images for labeling and the round of feedback needed in our experiments are so small that it won't make the user feel boring to make labels. Additionally, the time spent in retrieval is about 10s in a PC of P4 2.0GHz CPU and 1G RAM with *Matlab* implementation, which would be accepted by most users.

To make a detailed discussion, we analyze the results in the following two aspects:

ICA Feature vs. General Low-level Feature

In experiments (a) and (b), both of the mechanisms of CBIR are multi-view learning algorithm, but the features adopted are different. In (b), another set of general features replaces ICA feature as the other view. Since the general features are mostly concerning the statistical characteristics of the images, their interaction on each other is not so significant as that between ICA and general feature. This means, the two views in the multi-view learning algorithm should be less correlated to achieve better performance. Our method handles this problem well, because ICA feature is from the view of human vision, while general features is on the view of computer.

Multi-view vs. Single-view

Experiment (a) and (c) are based on exactly the same features, and in (c), the distance between two images is measured as weighted sum of the distance of general feature and that of ICA feature, defined in Section 4. The better retrieval performance of (a) shows that, providing the same features, it is better to divide them into two parts and use the multi-view learning algorithm than to simply combine them together. The reason is that, the two views will interact and mutually provide the information that the other is lack of.

Another remarkable phenomenon should be pointed out is that, when the round of feedback is more than 2, the retrieval accuracy of experiment (a) and

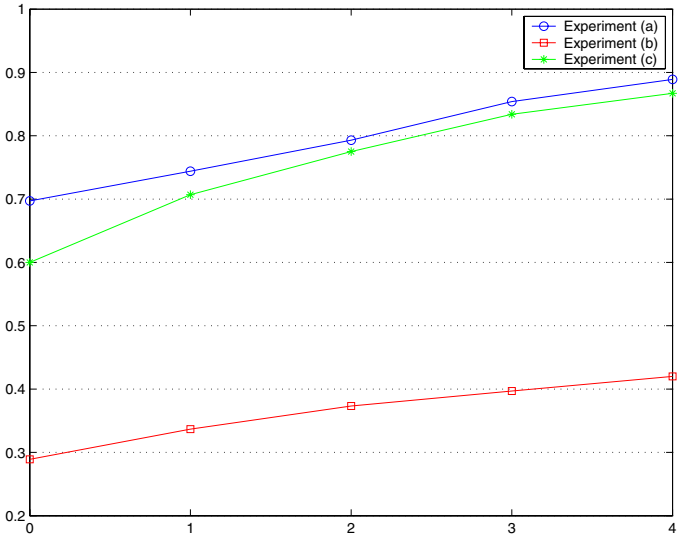


Fig. 1. Retrieval Results Comparison

(c) would be close. The reason is probably that, the features adopted in (a) and (c) are almost the same, so the information that we can ultimately utilize is almost the same. The interaction in multi-view learning only has effects at the first several rounds, and with the increase of rounds, the information provided by the two views has been almost mixed fully and the labels they provide will get close, then they may perform similarly as the system (c) with combined features.

Therefore, we can infer that, even the mechanisms of CBIR system are different, the final retrieval result after sufficient feedback rounds will only be related to the features we adopted and the feedback information provided by user. And this conclusion can be interpreted by the information theory as well. Then the advantage of our proposed system in practical applications is that, we can achieve high retrieval accuracy in the first several feedback rounds, i.e., 2 rounds may be enough, which can significantly improve the efficiency.

7 Conclusions

We have proposed a multi-view learning framework of CBIR, which is further consolidated with the feature extracted by ICA. At first, it is proved in theory that the ICA feature can provide more information than the original general low-level features for it accords with human vision. In the second place, the advantages of ICA feature and general low-level feature are integrated to improve each other in the scheme of the multi-view learning algorithm *Co-EMT*. This dramatically reduce the time of relevant feedback by the users. An the end, the semi-supervised learning algorithm in a single view is designed according to the specialties of the labels and the needs of *Co-EMT*. Owing to the forenamed

characteristics of our proposal, our experimental results demonstrate the outstanding retrieval performance.

Acknowledgements

This work is supported by the Distinguished Young Scholars of NSFC (No.60525111), and by the key project of NSFC (No.60432030).

References

1. A.J. Bell and T.J. Sejnowski. An Information-Maximization Approach to Blind Separation and Blind Deconvolution. *Neural Computation*, 1995, Vol: 7, pp. 1129–1159.
2. A.J. Bell and T.J. Sejnowski. The 'Independent Components' of Natural Scenes are Edge Filters. *Vision Research*, 1997, Vol. 37, No. 23, pp. 3327–3338.
3. A. Blum and T.Mitchell. Combining Labeled and Unlabeled Data with Co-Training. *Proc. of the Conference on Computational Learning Theory*, 1998, pp. 92–100.
4. J.F. Cardoso and B. Laheld. Equivariant Adaptive Source Separation. *IEEE Trans. on Signal Processing*, 1996, Vol. 45, No. 2, pp. 434–444.
5. D. Hubel. Eye, Brain, Vision. *Scientific American Library*, 1995.
6. J. Hurri, A. Hyvarinen, J. Karhunen, and E. Oja. Image Feature Extraction Using Independent Component Analysis. *Proc. IEEE Nordic Signal Processing Symp.*, Espoo, Findland, Sept. 1996, pp. 475–478.
7. A. Hyvarinen. A Family of Fixed-point Algorithm for Independent Component Analysis. *Int. Conf. on Acoustic, Speech and Signal Processing*, 1997, pp.3917-3920.
8. A. Hyvarinen and E. Oja. A Fast Fixed-point Algorithm for Independent Component Analysis. *Neural Computation*, 1997, Vol. 9, No. 7, pp.1483–1492.
9. N. Katsumata and Y. Matsuyama. Similar-Image Retrieval Systems Using ICA and PCA Bases. *Proc. of the International Joint Conference on Neural Networks*, Montreal, Canada, July 31-August 4, 2005, pp. 1229–1334.
10. T.W. Lee, M.S. Lewicki and T. Sejnowski. Unsupervised Classification with Non-Gaussian Mixture Models using ICA. *Adavances in Nueral Information Processing Systems*, 1999.
11. I. Muslea, S. Minton and C.A. Knoblock. Active + Semi-Supervised Learning = Robust Multi-View Learning. In *Proc. of the International Conference on Machine Learning*, 2002, pp. 435–442.
12. I. Muslea, S.Minton, and C.A. Knoblock. Selective Sampling with Redundant View. *Proc. of National Conf. on Artificial Intelligence*. 2000, pp. 621–626.
13. K. Nigam and R. Ghani. Analyzing the Effectiveness and Applicability of Co-Training. *Proc. of Information and Knowledge Mnangement*, 2000, pp. 86–93.
14. B.A. Olshausen and D.J. Field. Sparse Coding with an Overcomplete Basis Set: A Strategy Employed by V1? *Vision Research*, 1997, Vol. 37, No. 23, pp. 3311–3325.
15. A.T. Puga. A Computational Allegory for V1. *Proc. of the 2nd International Symposium on Image and Signal Processing and Analysis*, 2001, pp. 639–644.
16. A. Smeulders, M. Worring, A. Gupta and R. Jain. Content-Based Image Retrieval at the End of the Early Years. *Proc. IEEE Trans. Pattern Analysis and Machine Intelligence*, 2000, Vol. 22, No. 12, pp. 1349–1380.

17. K. Tieu and P. Viola. Boosting Image Retrieval. *Proc. IEEE Conf. on Computer Vision and Pattern Recognitino*, 2000, pp. 228–235.
18. S. Tong and E. Chang. Support Vector Machine Active Learning for Image Retrieval. *Proc. ACM Multimedia*, Ottawa, Canada, 2001, pp. 107–118.
19. N. Vasconcelos and A. Lippman. Learning From User Feedback in Image Retrieval Systems. *Advances in Neural Information Processing Systems*, Denver, Colorado, 1999.
20. W.E. Vinje and J.L. Gallant. Sparse Coding and Decorrelation in Primary Visual Cortex During Natural Vision. *Science*, Feb. 18, 2000, Vol. 287, pp. 1273–1276.
21. R. Yan and A.G. Hauptmann. Co-retrieval: a Boosted Reranking Approach for Video Retrieval. *IEE Proc. Vision Image Signal Processing*, Dec. 2005, Vol. 152, No. 6, pp. 888–895.
22. D. Zhou, O. Bousquet, T. Lal, J. Weston, and B. Schölkopf. Learning with Local and Global Consistency. *Advances in Neural Information Processing Systems*, 2003, vol. 16.
23. D. Zhou, J. Weston, A.Gretton, O.Bousquet, and B. Schölkopf. Ranking on data manifolds. *Advances in Neural Information Processing Systems 16*, Cambridge, MA, 2004. MIT press.
24. X. S. Zhou and T.S. Huang. Comparing Discriminating Transformations and SVM for Learning during Multimedia Retrieval. *Proc. ACM Multimedia*, Ottawa, 2001, pp. 137–146.

A P2P Architecture for Multimedia Content Retrieval

E. Ardizzone, L. Gatani, M. La Cascia, G. Lo Re, and M. Ortolani

Dip. di Ingegneria Informatica, Università di Palermo
Viale delle Scienze, 90128 Palermo, Italy
{ardizzon, gatani, lacascia, lore, ortolani}@unipa.it

Abstract. The retrieval facilities of most Peer-to-Peer (P2P) systems are limited to queries based on unique identifiers or small sets of keywords. This approach can be highly labor-intensive and inconsistent. In this paper we investigate a scenario where a huge amount of multimedia resources are shared in a P2P network, by means of efficient content-based image and video retrieval functionalities. The challenge in such systems is to limit the number of sent messages, maximizing the usefulness of each peer contacted in the query process. We achieve this goal by the adoption of a novel algorithm for routing user queries. The proposed approach exploits compact representations of multimedia resources shared by each peer, in order to dynamically adapt the network topology to peer interests, on the basis of query interactions among users.

1 Introduction

Recent years have witnessed an increasing attention from the research community toward new network paradigms and the focus has gradually shifted from more traditional communication architectures, such as the client/server one, that have historically driven Internet's development, to more decentralized models that carry the promise of improved robustness and scalability, such as the Peer-to-Peer (P2P) paradigm. In a P2P network all participating systems are assumed to run software with equivalent functionality and to operate without requiring central coordination [1]; the research community has shown an intense interest in designing and studying such systems, and file sharing systems such as Napster [2] and Gnutella [3], have gained huge popularity also among end-users. As with other killer applications in the Internet's world, the widespread availability of user-friendly tools has uncovered unforeseeable scenarios; for instance, it is now common for consumers to gather all kinds of diverse digital multimedia contents: consumers capture contents using their digital cameras, digital camcorders and mobile phones and store it on different devices; moreover they are beginning to store videos or images in such amounts that it is becoming increasingly difficult for them to manage, retrieve and ultimately make full use of their own data; in particular, locating and obtaining the desired resource has become a challenging task. Traditionally, user requests in P2P systems begin with the specification of a number of keywords, or of a specific file name pattern, but this

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 462–474, 2007.
© Springer-Verlag Berlin Heidelberg 2007

approach is insufficient when the data collection is huge or distributed, as in the case under consideration. For example, users might use different filenames and keywords to annotate the same file, thus making the data location process error-prone and user dependent; moreover, artificial intelligence technologies cannot provide yet a complete automatic annotation solution that would fill the gap between the semantic meanings and the low-level descriptors.

This paper presents a novel architecture for multimedia content retrieval in P2P networks that exploits an automatic content-based approach. Peers in the network are required to participate both in scattering data storage and in distributing workload of feature extraction and indexing; with respect to current Content Based Image Retrieval (CBIR) systems, enormous image collections can be managed without installing high-end equipment thanks to the exploitation of individual users' contribution; furthermore, we make use of the computational power of peers for image preprocessing and indexing in addition to data storage. As already mentioned, a challenging issue regarding sharing data on P2P systems is related to how content location is determined; this affects both the efficiency of resource usage and the overall system robustness, therefore in order to effectively exploit the potential of CBIR in P2P networks, we propose an adaptive mechanism for query routing that can well balance the storage overhead and the network load. Our approach to CBP2PIR (Content Based Peer-to-Peer Image Retrieval) is driven by the aim to provide scalable and efficient resource discovery in an unstructured P2P environment; an adaptive routing algorithm is implemented in order to avoid flooding of control packets and the network topology takes into account the peers' interests by dynamically adapting through a reinforcement learning process. Each peer maintains a list describing the interests and the resources available to other peers, thus effectively building a profile for all the other participants in the network; before sending out a query, the peer will match it against its profiles in order to find the most suitable route leading to the best-matching peer. Preliminary experiments are encouraging and show that a *small world* network structure can emerge spontaneously from local interactions; this significantly improves both the network traffic cost, and the query efficiency.

The remainder of the paper is organized as follows. Section 2 briefly reviews existing CBIR techniques and describes the video descriptors chosen for representing multimedia resources. Section 3 provides more details on the proposed adaptive routing protocol, while preliminary experimental results are presented in Section 4. Section 5 discusses related work. Finally, our conclusions and final considerations are presented in Section 6.

2 Multimedia Content Representation

Content-based analisys and representation of digital media have been extensively studied in the last decade from researchers working on CBIVR (Content Based Image and Video Retrieval) or, more generally, in the field of digital libraries. The main problem in CBIVR is the gap between the image or video data and

its semantic meaning. Image and video understanding techniques are still very unreliable in general cases, and, moreover, even human provided keywords or textual descriptions usually fail to explicitate all the relevant aspects of the data. Techniques proposed in literature range from semi-automatic to fully automatic. Semi-automatic techniques require a lot of human effort and, consequently, in many cases are not of practical use. On the other hand fully automatic techniques tend to miss the semantic meaning of the data and are mainly related to low-level features such as color histogram, texture, contours, etc. (see [4] for a review).

In our work we focused mainly on fully automatic techniques as we observed that typical users, even if explicitly invited to annotate their data, tend to provide only minimal information that may not be sufficent for acceptable content based retrieval performance.

In the following, we refer to images to indicate either single images, in the case of still image applications, or frames representing a sub-part of a video sequence in the case of video applications. Namely, video representation may be based on the decomposition of the video sequence into shots [5] or into video objects [6]. Shot content representations may be obtained through the description of a few representative frames (r-frames). A limited number of r-frames is selected from each shot, and each r-frame is therefore statically described in terms of its visual content, e.g. through color and texture descriptors. Motion activity may be also taken into account, for example by computing motion features related to short sequences in which r-frames are embedded. The r-frame selection and the computation of visual and motion features may be performed in a number of ways.

Shots are short video units, normally consisting of a few tens or hundreds of subsequent frames, characterized by still or slowly moving camera actions, and normally beginning and ending with abrupt frame content changes or with video editing effects (fade in/out, wipes, etc.).A good representation of a shot in terms of r-frames must strike a balance between adequateness and concision of description. As r-frames must capture the low level semantics of the shot a large number of them are more likely to encode the meaning of the data. On the other hand, it should be profitable to maintain the size of data needed for computation as low as possible. Several attempts have been accomplished to get this goal, also based on heuristics. Early works generally assumed a single r-frame per shot, for example the first frame in the sequence. This choice can be misleading, because two shots of similar content may be considered to be different if representative frames are different. In other cases, the first and the last shot frame have been proposed as representative frames. In general, assuming a fixed number of representative frames per shot is not a good idea, because this can give problems of oversampling for shots with slow dynamics, and undersampling for shots where camera or object motion is noticeable. In our work we adopted a non-linear temporal sampling based on thresholding of the cumulative difference of frame brightness values [7]. Following the lines of [7], we structured the video descriptor in a hierarchical way. At the highest level, this descriptor simply consists of: (*i*) a few keywords, (*ii*) the video duration (in seconds), (*iii*) the number of shots contained in the video, (*iv*) references to the shot descriptor

for each shot belonging to the video. A shot descriptor consists of: (*i*) the shot duration (in seconds), (*ii*) the number of r-frames contained in the shot, (*iii*) a pointer to the r-frame descriptor for each r-frame belonging to the shot. Finally, the r-frame visual descriptor consists of attributes of both static and dynamic kind. Static descriptors are based on texture and color. Motion-based descriptors are based on the optical flow field of the r-frame, and their computation involves considering a few frames before and after the r-frame.

Color is a very powerful feature in finding similar images. Even if textural, geometrical and motion features may be needed to perform effective queries and to eliminate false positive retrieval, it is believed that color indexing will retain its importance due to the fast processing of this kind of queries and to the simpleness in automatically computing color features from raw data. In the last years several color based techniques have been proposed for video annotation (for example, region based dominant color descriptors [8], multiresolution histograms [9], vector quantized color histograms [10]). These techniques in general require color space conversion, quantization and clustering, in order to reduce the descriptor dimension and then improve searching speed. In this work we adopt a simple but effective method [7] based on a 3-dimensional quantized color histogram in the HSV (Hue - Saturation - Value) color space and an Euclidean metric to compare the query image to images contained in the database is proposed. The HSV quantization needed to compute a discrete color histogram is done taking into account that hue is the perceptually more significant feature. Thus a finest quantization has been used for hue, allowing for 18 steps, while only 3 levels are allowed for saturation and value. In such a way we obtain a 162 (18 x 3 x 3) bins HSV histogram, that may be easily represented by a 162 x 1 vector. Texture content of an image is a fundamental feature in classification and recognition problems. Several texture descriptors have been proposed that try to mimic the human similarity concept, but they are normally useful only in classifying homogeneous texture. Generic images usually contain different kinds of texture, so that a global texture descriptor hardly may describe the content of the whole image. The texture features we propose are related to coarseness, directionality and position of texture within the image. All these features are based on edge density measures. Edge density is directly related to coarseness, directionality is addressed by repeating the edge measure for different directions and spatial position is taken into account by a simple partitioning of the r-frame. In particular, we first subdivide the r-frame into four equal regions. For each region we compute the edge maps through directional masks respectively aligned along the directions 0, 45, 90 and 135 degrees. Values of edge map exceeding a fixed threshold are considered edge pixels. The threshold value has been determined experimentally. The ratio between the number of edge pixels and the total number of pixels is the edge density. Since we determine 4 edge density values for each region, we have a 16 x 1 texture-based vector. Optical flow field [11] of the r-frame has been used to compute motion-based descriptors. We use a gradient-based technique and the second-order derivatives to measure optical flow [7]. The basic measurements are integrated using a global smoothness

constraint. This technique allows to obtain a dense and sufficiently precise flow field at a reasonable computational cost.

To code the optical flow in a form adequate for content description we segment the field into four equal regions; for each region we then compute motion based features. The splitting was performed to preserve spatially related information that are not integrated in the computed features. In conclusion, the adopted motion descriptors are a measure of the average motion magnitude in the considered region, and a normalized 18 bins histogram of motion vectors directions.

In summary the visual descriptor of an r-frame, computed automatically by the system, is a 254-dimensional vector $\underline{x} = [\underline{c}\ \underline{t}\ \underline{m}\ \underline{d}]$ where $\underline{c}$ is a 162-dimensional vector representing the global HSV color histogram and $\underline{t} = [\underline{t_{tl}}\ \underline{t_{tr}}\ \underline{t_{bl}}\ \underline{t_{br}}]$ is a 16-dimensional vector representing the edge density computed respectively over the top-left, top-right, bottom-left and bottom-right quadrants of the r-frame. Similarly $m = [m_{tl}\ m_{tr}\ m_{bl}\ m_{br}]$ and $\underline{d} = [\underline{d_{tl}}\ \underline{d_{tr}}\ \underline{d_{bl}}\ \underline{d_{br}}]$ are a 4-dimensional vector and a 72-dimensional vector containing respectively the average motion magnitudes and the 18 bins motion vectors direction histograms computed over the four regions as above.

3 Adaptive Searching Protocol

The key problem addressed in this work is the efficient and scalable localization of multimedia resources, shared in a P2P community. Queries issued by a user are routed to neighbor peers in the overlay network, in order to find resources that satisfy them. At the start the network has a random, unstructured topology (each peer is connected to N_s neighbors, randomly chosen), and queries are forwarded as in the scoped flood model. Then, the system exploits an adaptive approach that selects the neighbors to which a query has to be sent or forwarded. This approach can overcome the limitations of flooding, allowing the peers to form dynamic communities based on commonality of interest. The selection process is carried out with the aim to detect peers that with high probability share resources satisfying the query. The selection is driven by an adaptive learning algorithm by which each peer exploits the results of previous interactions with its neighbors, in order to build and refine a model (profile) of other peers, concisely describing their interests and contents. When a peer enters the network for the first time, a bootstrap protocol returns the address of some existing peers to get started. The new peer can then discover other nodes through these known peers. In particular, our approach is designed in such a way that a peer can discover new peers during the normal handling of queries and responses to its current neighbors. To this aim, each peer maintains a fixed number, N_m, of slots for profiles of known peers. When a peer has to send a query, it dynamically selects the actual set of N_a destinations, among all the the $N_k(t)$ peers known at that time step. This is carried out by means of a ranking procedure that compares the query characteristics with all the information in the stored profiles and sorts all known contacts in order to single out the N_a peers that are the best suited to return good response. The network topology (i.e., the actual set of peers that are

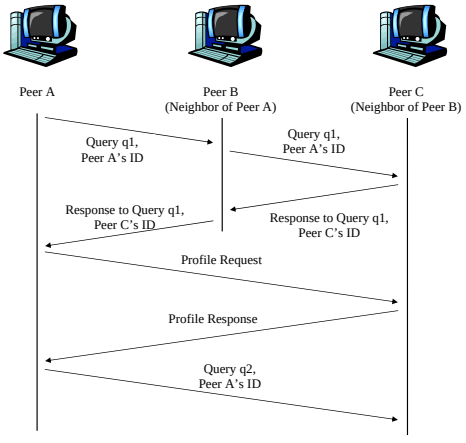


Fig. 1. The process of neighbor discovery

neighbors in the overlay) is then dynamically modified according to the results of the ranking process, and the query is consequently routed according to the predicted match with other peers' resources. A peer that has received a query can forward it to those neighbors whose profiles match the query. To this aim, the peer uses the same selection algorithm applied to locally generated queries (note that the peer automatically excludes both the peer that has forwarded the query, and the peer that has generated the query). To limit congestion and loops in the network, queries contain a Time-To-Live (TTL), which is decreased at each forward, and queries will not be forwarded when TTL reaches 0. When a peer receives the responses for a locally generated query, it can start the actual resource downloading. Moreover, if a peer that has sent a response is not yet included in the list of known peers, a profile request is generated. For this request, the two peers contact each other directly. When the message containing the profile will arrive, the new peer will be inserted among the N_k known peers and its features will be taken into account in order to select actual neighbors for the following queries (see Fig. 1). It is also worth noting that the stored profiles are continuously updated according to the peer interactions during the normal system functioning (i.e., matches between queries and responses). Moreover, a peer can directly request a more up-to-date profile if necessary. Table 2 describes the four basic messages our protocol uses to perform resource searching.

The selection mechanism takes primarily into account the experience that peers acquire during their normal interactions: each new information available is opportunely elaborated and exploited to enrich the system knowledge. Each peer profile maintains a concise representation of the shared resources, by the adoption of different techniques for textual and visual contents. In particular, the system adopts simple taxonomies and Bloom filters [12] to build a binary vector that represents the textual contents. As regards visual resources, after the meaningful features have been extracted from the image database, each peer will work on extracting representative information that may succinctly describe its

whole content. Our implementation makes use of a simple but effective clustering technique through which each peer will roughly partition its data space into separate regions that represent different groups of related images. Specifically, we employ the well-known k-means clustering method [13]. The basic formulation of the algorithm assumes that the number of clusters is known in advance, which may be a too tight constraint for our present scenario, however this requirement may be partially loosened with the use of controlled iterations and of a cluster validity assessment technique [14, 15]. Furthermore, in order to cope with the stream of continuously incoming data, we adopt a variation on the basic k-means algorithm that allows on-line updating of the computed clusters, using the set of cluster representatives as a sort of "signature" for the content of each peer (according to their vectorial representation as reported at the end of Section 2).

Our system supports a basic query language (where a query string is interpreted as a conjunction of keys) for textual information retrieval, while a standard "query-by-example" approach is exploited to search the image database. When asked with a query, the system looks up the information in its profile database in order to obtain a list of candidate peers that might store data matching the query. When a peer receives a query from another peer, it checks its local repository in order to locate the resources that better match with the desired content. In particular, textual resources are searched using a standard keyword-based technique, while visual resources are compared by means of a weighted sum of normalized Euclidean distances, as already presented in [16]. In order to normalize the distances, we estimate a probability distribution for the Euclidean distances of each visual feature (color, texture, motion), comparing each r-frame in a training database with all the others. These distributions are then used to normalize all the distances to the range [0,1]. The similarity between the current query and the general interests of each peer is managed in different ways on the basis of the kind of searched resource. The similarity between textual resources (as well as textual annotations and high-level descriptors associated to multimedia resources) is evaluated exploiting a standard technique for textual retrieval. As regards visual resources, the peer computes the distance to each cluster representative and chooses the closest ones as possible matches. It is worth noting that, while all processing is performed locally, manipulated objects exist in a globally defined vector space; hence all feature vectors, as well as all cluster centroids, are globally comparable; however, clusters are not required to have a global semantic validity as they are only used to compute relative distances. Furthermore, if the resources are opportunely indexed, the system can also exploit the representation of the resources by means of the Bloom filters which are maintained into the peer profiles. This way, it is possible to check, with high probability, if a given resource belongs to the resource set shared by a peer. This approach enhances the topological properties of the emergent overlay network and it is very useful in those applications where resources are uniquely characterized by an identifiers or are semantically annotated.

The base criterion, that exploits the experience of past interactions, gives a good indication about the probability that a contact could directly provide the

Table 1. Selection Criteria

Parameter	Description	Weight
R_n	current estimate of the contact	α
R	old reliability value of the contact (according to past history)	$1-\alpha$
R_a	new reliability value used to rank contacts	-
I	percentage of contact interests with respect to query topics	β
S	percentage of successes provided by the contact	γ
B	result of membership test (produced by Bloom filter)	δ
Q	capability summarization of the contact (bandwidth, CPU, storage, etc.)	ϵ
C	connection characteristic summarization of the contact	ζ

resources searched. In addition to this criterion, a further mechanism is adopted, which is capable of singling out peers that, although not directly owning the desired resources, can provide good references to the resource owners. It is worth noticing that while the first criterion, based on the commonality of interests, tries to increase the overlay network clusterization by the creation of intra-cluster links, the second one typically sets links between different clusters, providing a quick access to peers that are close to several resources.

Furthermore, the selection mechanism considers some additional criteria, in terms of peer capabilities (bandwidth, CPU, storage, etc.) and end-to-end latency, in order to take into account the topological characteristics of the peer community (thus reducing the mismatch between the overlay and the real topology). Regarding the selection algorithm, each contact is associated to a parameter, R, that provides a measure of its reliability. The parameter value is related to the interactions in the peer community and it changes according to the criteria previously described (see also Table 1). Each single criterion gives a partial value. These partial values are then jointly considered by means of a weighted average (see Eq. 1) that produces an estimate of the overall reliability for the current situation.

$$R_n = \beta \cdot I + \gamma \cdot S + \delta \cdot B + \epsilon \cdot Q + \zeta \cdot (1 - C), \tag{1}$$

where

$$\beta + \gamma + \delta + \epsilon + \zeta = 1, \tag{2}$$

$$0 \leq \beta, \gamma, \delta, \epsilon, \zeta \leq 1. \tag{3}$$

This estimate is finally combined with the old R value, generating the new value, R_n for the reliability parameter. In order to smooth the results of the selection process, a kind of temporal memory is employed to balance new information against past experience. The new estimate is then formally computed by the formula:

$$R_a = \alpha \cdot R_n + (1 - \alpha) \cdot R, \tag{4}$$

where

$$0 \leq \alpha \leq 1, \alpha << (1 - \alpha). \tag{5}$$

The R_a value is then exploited to rank all the known peers, according to the estimated reliability. In order to establish a balance between the exploration and

Table 2. Message set

Message type	Usage	Fields
Query	searching for a specific resource	(weighted) query keywords, query ID, source peer ID, generation timestamp, TTL
Query response	responding to a Query message	resource ID, query ID, responder ID, source peer ID, generation timestamp, TTL
Profile request	requesting a peer profile	request ID, source peer ID, target peer ID, generation timestamp
Profile response	responding to a Profile request	profile, request ID, responder ID, source peer ID, generation timestamp,

exploitation of the search space, the algorithm in the early steps can select peers different from the N_a top ones. This random search behavior is characterized by a probability of choosing no optimal contacts:

$$P = \exp(\frac{\delta R_a}{T}), \tag{6}$$

where using an approach similar to that adopted in the "simulated annealing" searching technique [17], δR_a represents the decrease in the reliability value, and the "temperature" T is a control parameter.

4 Experimental Evaluation

The underlying idea of our approach is that an intelligent collaboration among the peers can lead to an emergent clustered topology, in which peers with shared interests and domains tend to form strongly connected communities. The adoption of an adaptive approach, based on a simple, but effective reinforcement learning scheme can better cope with highly dynamic P2P communities. The expected theoretical network topology should have *small world* properties [18] and our experimental evaluation aims to confirm the hypothesis. In such a topology, a flood-based routing mechanism (with limited scope) is well suited, since it allows any two peers to reach each other via a short path, while maximizing the efficiency of communication within clustered peer communities. Furthermore, the approach proposed should take advantage from the adaptive overlay rearrangement, in order to well cope with high node volatility and massive node disconnections.

Since in the studies on deployed P2P networks [19,20,21] the dynamics in peer lifetimes and the complexity of these networks make it difficult to obtain a precise comprehensive snapshot, we decided to use simulation to perform an evaluation of the proposed approach. Simulation of P2P networks can provide a thorough evaluation and analysis of their performances. In order to study the behavior of peer interactions in our system, we designed and implemented a simple simulator (see, also, [22]) that allows to model synthetic peer networks and run queries according to the routing protocol adopted. The goal of the simulator is to analyze the topology properties of emergent peer networks. In the simulations carried out, each peer belongs to one or more groups of interest (let N_g be total number of

groups), according to the resources owned and the query issued; in general, peers have interests that partially overlap each other. As observed in [23], unstructured P2P systems are characterized by high temporal locality of queries (i.e., with high probability a single peer issues similar queries over time). Therefore, in order to better investigate how the adaptive mechanism proposed can support efficient resource searching, we consider that each peer generates queries belonging, with high probability, to one of the group topics (however, a smaller number of queries is generated on a randomly selected topics). It is also worth noting that each resource can be replicated on several peers. For the experimental analysis of emergent topological properties, we consider two network metrics, the clustering coefficient, $C(G)$, and the characteristic path length, $L(G)$, that well characterize the topological properties of dynamic networks. Since in our simulations it is possible that the network is not always strongly connected, we adopt a more practical definition ($L'(G)$) for the characteristic path length, using the harmonic mean of shortest paths that can be computed irrespective of whether the network is connected. We also compute the ratio C/L' that gives a good insight of the overall topological properties: high values are associated with networks that present both a strong clusterization, and a low average separation between nodes. C and L' are computed in the directed graph, based on each peer N_a neighbors, taking a measure at each time step and averaging across simulation runs. Finally, in order to quantify the efficiency of the approach proposed, three further metrics are adopted: the query hit-rate, HR, that represents the percentage of queries successfully replied, the query coverage-rate CR, that represents the average number of nodes reached by a query, and the node message-load ML, that represents the average number of messages that a node has to process during a single time step.

In order to evaluate the algorithm proposed, we performed extensive simulations, considering several scenarios, each of them characterized by the variation of a simulation parameter (namely, TTL, N_a, N, N_g, N_m, and N_s). For each simulation, the aim is to study how network statistics and searching performance change when the parameter value is varied. Furthermore, we studied the impact of dynamic changes in the peer communities, in order to test the robustness of the algorithm against such events. Since the initial random topology can affect the final results, for each simulation, we perform several independent simulations, averaging across all the results. Due to space limitation, we can not present here these experimental results. A detailed performance evaluation of the proposed searching approach can be found in [22], confirming the idea that adaptive routing can properly work and that *small world* network topologies can emerge spontaneously from local interactions between peers, structuring the overlay in such a way that it is possible both to locate information stored at any random node by only a small number of hops (low latency object lookup), and to find quality results quickly and even under heavy demands (high clustering coefficient).

The visual descriptors we adopted, despite its compactness and the availability of simple algorithms to compute, have been proven to encode the visual

content reasonably well. In a previous work [16] extensive experiments to evaluate the retrieval performance based only on visual information have been reported. In particular, to assess the retrieval capabilities of the descriptors we used a normalized version of precision and recall that embodies the position in which relevant items appear in the retrieval list [24]. All the tests were performed using a database containing about 1500 r-frames obtained from about 500 shots. We considered 20 r-frames randomically chosen and evaluated for each one of them the system response to a query by example. Recall and precision measurements require to determine which r-frames are relevant with respect to a posed query, but stating relevance is a very subjective task. To overcome this problem we adopted a subjective criterion: candidate-to-relevance r-frames for each query were determined by four different people (not including the authors) and a r-frame was considered as relevant if at least three people chose it. Once known the correct query result, we are able to evaluate system performances. Experiments showed that visual descriptors are adequate in most cases if the image collections are not too large (less than 10,000 images). For larger image collection, when query results obtained using only visual descriptor tends to become unreliable, the use of textual information greatly improve the results. Preliminary results on our CBP2PIR system using both textual and visual data showed very promising retrieval capability, confirming the feasibility of our searching method for feature vectors derived from multimedia resources.

5 Related Work

Although the lookup of multimedia data in P2P networks represents a new, interesting research field, to the best of our knowledge, only few works exist that address this issue. In [25] the Firework Query Model for CBIR information searching in P2P networks is proposed. The main idea consists of clustering peers with similar resources, using the set of feature vectors as signature value of a peer in order to measure similarity. The Firework Query Model exploits two classes of links (normal random links and privileged attractive links), in order to route queries. A query starts off as a Gnutella-like flooding query. If a peer deems the query too far away from the peers local cluster centroid, it will forward the query via a random link, decreasing the TTL of the query. Otherwise, it will process the query, and forward it via all its attractive links without decreasing the TTL. A similar CBIR scheme for P2P networks, based on compact peer data summaries, is presented in [26]. To obtain the compact representation of a peer's collection, a global clustering of the data is calculated in a distributed manner. After that, each peer publishes how many of its images fall into each cluster. These cluster frequencies are then used by the querying peer to contact only those peers that have the largest number of images present in one cluster given by the query. In [27], the authors investigate a CBIR system with automated relevance feedback (ARF) using non-linear Gaussian-shaped radial basis function and semi-supervised self-organizing tree map clustering technique. The authors apply the CBIR system over P2P networks by grouping the peers into community

neighborhoods according to common interest. In [28] a different overlay setup technique is introduced, in order to cluster peers according to the semantic and feature-based characteristics of their multimedia content. Finally, Wu *et al.* [29] propose a local adaptive routing algorithm that dynamically modify the network topology toward a *small world* structure, using a learning scheme similar to that considered in this paper. However, they design their protocol with the aim of supporting an alternative model for peer-based Web-search, where the scalability limitations of centralized search engines can be overcome via distributed Web crawling and searching.

6 Conclusion

This paper presented an approach to information retrieval in a P2P network that relies on an adaptive technique for routing queries and is specifically targeted to multimedia content search. The main motivation behind our work is that the huge amounts of data, their peculiar nature and, finally, the lack of a centralized index make it particularly difficult to pursue the goal of efficiency in this kind of systems. Our approach employs a decentralized architecture which fully exploits the storage and computation capability of computers in the Internet and broadcasts queries throughout the network using an adaptive routing strategy that dynamically performs local topology adaptations. Modifications in the routing structure are driven by query interactions among neighbors in order to spontaneously create communities of peers that share similar interests; moreover, a *small world* network structure can emerge spontaneously thanks to those local interactions. Network traffic cost, and the query efficiency are thus significantly improved as is confirmed by our preliminary experiments.

References

1. Milojicic, D., Kalogeraki, V., Lukose, R., Nagaraja, K., Pruyne, J., Richard, B., Rollins, S., Xu, Z.: Peer-to-peer computing. Technical Report HPL-2002-57, HP Labs (2002)
2. C-Net News: Napster among fastest-growing Net technologies (2000)
3. Limewire: The Gnutella protocol specification (ver. 0.4). http://www9.limewire.com/developer/gnutella_protocol_0.4.pdf (2001)
4. Datta, R., Li, J., Wang, J.: Content-based image retrieval - approaches and trends of the new age. In: Proc. of ACM MIR, Singapore (2005)
5. Del Bimbo, A.: Visual Information Retrieval. Academic Press (1999)
6. Gunsel, B., Tekalp, A., van Beeck, P.: Content-based access to video objects: temporal segmentation, visual summarization, and feature extraction. Signal Processing **66** (1998) 261–280
7. Ardizzone, E., La Cascia, M.: Automatic video database indexing and retrieval. Multimedia Tools and Applications 4 (1997) 29–56
8. Deng, Y., Manjunath, B., Kenney, C., Moore, M., Shin, H.: An efficient color representation for image retrieval. IEEE Trans. on Image Processing **10**(1) (2001) 140–147

9. Hadjidemetriou, E., Grossberg, M., Najar, S.: Multiresolution histograms and their use for recognition. IEEE Trans. on Pattern Analysis and Machine Intelligence **26**(7) (2004) 831–847
10. Jeong, S., Won, C., Gray, R.: Image retrieval using color histograms generated by Gauss mixture vector quantization. Computer Vision and Image Understanding **9**(1-3) (2004) 44 – 66
11. Horn, B., Schunk, B.: Determining optical flow. Artificial Intelligence **17** (1981) 185–203
12. Bloom, B.H.: Space/time trade-offs in hash coding with allowable errors. Communications of the ACM **13**(7) (1970) 422–426
13. MacQueen, J.: Some methods for classification and analysis of multivariate data. In: Proc. of 5th Berkeley Symposium. (1967) 281–297
14. Halkidi, M., Batistakis, Y., Vazirgiannis, M.: On clustering validation techniques. Journal of Intelligent Information Systems **17**(2-3) (2001) 107–145
15. Pal, N.R., Bezdek, J.C.: On cluster validity for the fuzzy c-means model. IEEE Trans. on Fuzzy Systems **3**(3) (1995) 370–379
16. Ardizzone, E., La Cascia, M.: Automatic video database indexing and retrieval. Multimedia Tools and Applications **4** (1997) 29–56
17. Kirkpatrick, S., Gelatt, C.D., Vecchi, M.: Optimization by simulated annealing. Science **220**(4598) (1983) 671–680
18. Watts, D., Strogatz, S.: Collective dynamics of small-world networks. Nature **393** (1998) 440–442
19. Ripeanu, M., Foster, I., Iamnitchi, A.: Mapping the Gnutella network: Properties of large scale Peer-to-Peer systems and implications for system design. IEEE Journal on Internet Computing, Special Issue on Peer-to-peer Networking (2002)
20. Saroiu, S., Gummadi, K., Gribble, S.: A measurement study of Peer-to-Peer file sharing systems. In: Proc. ACM Multimedia Conferencing and Networking. (2002)
21. Sen, S., Wang, J.: Analyzing Peer-to-Peer traffic across large networks. IEEE/ACM Trans. on Networking **12**(2) (2004) 212–232
22. Gatani, L., Lo Re, G., Noto, L.: Efficient query routing in peer-to-peer networks. In: Proc. IEEE ITRE. (2005) 393–397
23. Markatos, E.P.: Tracing a large-scale peer to peer system: an hour in the life of Gnutella. In: Proc. IEEE CCGrid. (2002)
24. Faloutsos, C., Barber, R., Flickner, M., Hafner, J., Niblack, W., Pektovic, D., Equitz, W.: Efficient and effective querying by image content. Journal of Intelligent Information Systems **3**(3-4) (1994) 231–262
25. King, I., Ng, C.H., Sia, K.C.: Distributed content-based visual information retrieval system on peer-to-peer networks. ACM Trans. on Information Systems, **22**(3) (2004) 477–501
26. Muller, W., Henrich, A.: Fast retrieval of high-dimensional feature vectors in P2P networks using compact peer data summaries. In: Proc. ACM MIR, Berkeley (CA), USA (2003)
27. Lee, I., Guan, L.: Content-based image retrieval with automated relevance feedback over distributed peer-to-peer network. In: Proc. IEEE ISCAS. (2004)
28. Kacimi, M., Yetongnon, K., Ma, Y., Chbeir, R.: HON-P2P: a cluster-based hybrid overlay network for multimedia object management. In: Proc. Intl. Conf. on Parallel and Distributed Systems. (2005) 578–584
29. Wu, L., Akavipat, R., Menczer, F.: 6S: Distributing crawling and searching across web peers. In: Proc. Web Technologies, Applications, and Services. (2005) 159–164

Optimizing the Throughput of Data-Driven Based Streaming in Heterogeneous Overlay Network

Meng Zhang[1], Chunxiao Chen[1], Yongqiang Xiong[2], Qian Zhang[3], and Shiqiang Yang[1]

[1] Dept. of Computer Sci. & Tech., Tsinghua Univ., Beijing 100084, China
{zhangmeng00,chencx05}@mails.tsinghua.edu.cn, yangshq@tsinghua.edu.cn
[2] Microsoft Research Asia, Beijing 100080, China
Yongqiang.Xiong@microsoft.com
[3] Dept. of Computer Sci., Hong Kong Univ. of Sci. and Tech., Hong Kong, China
qianzh@cs.ust.hk

Abstract. Recently, much attention has been paid on data-driven (or swarm-like) based live streaming systems due to its rapid growth in deployment over Internet. In such systems, nodes randomly select their neighbors to form an unstructured overlay mesh (gossip-style overlay construction) and then each node requests desired data blocks from its neighbors (block scheduling). To improve the performance, most of existing works focus on the gossip-style overlay construction issue; however few concentrate on optimizing the block scheduling for improving the throughput of a constructed overlay, especially in heterogeneous environment. In this paper, we propose a scheme to optimize the *throughput* of *data-driven* streaming systems in *heterogeneous* overlay network. We first model the block scheduling problem as a classical min-cost flow problem and thereby derive a global optimal solution. Based on this idea, we then propose DONLE - a fully distributed asynchronous scheduling algorithm. Simulation results verify that DONLE is superior to a number of conventional strategies.

1 Introduction

As the most promising alternative to IP multicast, overlay multicast especially multicast through peer-to-peer (P2P) network has attracted a lot of attention during the past decade. One of the most important applications of overlay multicast is to stream live media content to a huge population of end users through Internet, also known as peer-to-peer streaming.

A lot of measurement studies in P2P overlay networks reveal that the bottleneck bandwidth between the end hosts exhibits extremely heterogeneity. To deal with heterogeneity in streaming multicast applications, numerous solutions has been proposed for both IP multicast [1] and overlay multicast [2,3]. Their basic way is to encode the source video into multiple layers, and each receiver subscribes an appropriate number of layers due to its bandwidth capacity.

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 475–484, 2007.
© Springer-Verlag Berlin Heidelberg 2007

Recently, a new category of overlay streaming multicast protocols called data-driven protocols (or swarm-like protocols) [4,5,6,7] targeting non-interactive streaming multicast applications has been proposed. Unlike conventional tree-based approaches, in data-driven protocol, each node randomly finds some nodes as its neighbors so that an unstructured network is formed. This step is usually called gossip-style overlay construction (or membership management). The next step named block scheduling is also intuitive: the live media content is divided into blocks (or segments, packets) and every node announces what blocks it has to its neighbors. Then each node explicitly requests the blocks of interest from its neighbors according to their announcement. Actually, it is similar to Bit-Torrent protocol [8]. Some systematical studies (such as [9]) show that data-driven approach is better than tree-based approach under many conditions especially in high churn rate of clients. Meanwhile, data-driven based streaming systems are also emerging and rapidly deployed over Internet [4,10,11] in the past two years. Given the significance of data-driven streaming protocol, it is important to study how to improve the throughput of this category of protocols especially under heterogeneous network. Most of existing works in P2P streaming with layered coding [2,3] use stream level scheduling method. However, the scheduling in data-driven protocol is more fine-grained because it needs a block level scheduling. This leads to the challenge: how does a node decide to fetch which block of which layer from which neighbor node under heterogeneous bandwidth constraints.

In our previous work [12], we have studied how to do optimal scheduling in homogeneous environment. In this paper, we propose DONLE, a Data-driven Overlay Network algorithm using LayEred coding to handle the heterogeneity. We first state the basic block scheduling problem, then model the problem as a classical min-cost flow problem and give a global optimal block scheduling solution. After that, we propose a fully distributed algorithm - DONLE, doing local optimal block scheduling at each node. Simulation results show that DONLE is superior to a number of recent proposed scheduling strategies under the same overlay topology. The remainder of this paper is organized as follows. In Section 2 we briefly review the related work. In Section 3, we state the block scheduling problem in detail and formulate the problem. Next, in Section 4, we model this scheduling problem as an equivalent min-cost flow problem and derive the global optimal scheduling algorithm. Section 5 presents the proposed distributed asynchronous algorithm DONLE. The performance of DONLE is evaluated in Section 6. We conclude this paper in Section 7.

2 Related Work

Actually, there are a wealth of research efforts towards improving the overlay multicast throughput. Early researchers in this area mainly focus on how to construct single or multiple application layer tree(s). LION [3] employs a stream-level multi-path based method to improve the throughput of overlay network using network coding. Recently, a new category of overlay multicast

protocols - data-driven (or swarm-like) protocols are proposed [4,5,6,7]. In theses protocols, PALS [7] is an adaptive streaming mechanism from multiple senders to a single receiver using layered coding, which is actually a swarm-like (or data-driven) protocol. PALS mainly focuses on coping with the network dynamics such as bandwidth variations and sender participation. PALS evaluates its performance under the scenario of streaming from multiple senders to a single receiver very detailedly. Yet it does not involve the performance of the data-driven protocol under an overlay mesh and does not aim to improve the throughput of data-driven streaming. Besides, many recent works have been done to improve the gossip-style overlay construction for various purposes [13,14]. However, few works address how to maximize the *throughput* of *data-driven* streaming in a constructed *heterogeneous* overlay mesh.

3 Block Scheduling: Problem Statement and Formulation

In this section, we first intuitively explain what we optimize in data-driven streaming. Then we formulate this problem. Our basic approach is comprehensive. We define a priority for every desired block of each node due to the block importance, such as block layer, and its rarity. Our goal is to maximize the average priority sum of all streaming blocks that are delivered to each node in one request period under heterogeneous bandwidth constraints.

3.1 Block Scheduling Problem

The idea of DON based streaming system is similar to Bit-Torrent protocol [8]. In our protocol, each node will independently find its neighbors in the overlay so that an unstructured random overlay mesh will be formed. The media streaming is encoded with layered coding, and every layer is divided into blocks with the same size, each of which has a unique sequence number. Every node has a *sliding window* which contains all the up-to-date blocks on the node and goes forward continuously at the speed of streaming rate. We call the front part of the sliding window *exchanging window*. The blocks in the exchanging window are the ones before the playback deadline, and only these blocks will be requested if they are not received. The unavailable blocks beyond playback deadline will be no more requested. Every node periodically pushes all its neighbors a bit vector called buffer map in which each bit represents the availability of a block in its sliding window to announce what blocks it holds. Due to the announcement of the neighbors, each node will periodically send requests to its neighbors for the desired blocks in its exchanging window. We call the time between two requests a *request period* (or period for short, typically 1∼6 sec). Each node will decide from which neighbor to ask for which blocks at the beginning of each request period. When a block does not arrive after its request is issued for a while and is still in the exchanging window, it is requested in the following period again. In layered video coding, video is encoded into a base layer and several enhanced layers and a higher layer can only be decoded if all lower layers are available,

Table 1. Notations

Notation	Description
N	Set of all nodes in the overlay except the source node 0
L	Number of encoded layers
$r_l, l = 1, \cdots, L$	The cumulative rate from layer 1 to layer l, blocks per second
$I_i, O_i, i = 0, \cdots, \|N\|$	The inbound and outbound bandwidth capacity of node i
$E_{ik}, i, k = 0, \cdots, \|N\|$	The maximum end-to-end bandwidth from node i to k
$h_{ij} \in \{0, 1\}$	"$h_{ij} = 1$" denotes node i holds block j; "$h_{ij} = 0$", otherwise
NBR_i	Set of neighbors of node i
τ	The request period
π_j^i	The priority of block j for node i
W_T	The exchanging windows size scaled by time
C_i	The current clock time at node i
d_j^i	Play out deadline of block j at node i
D_i	Set of all desired blocks in the exchanging window of node i

namely the block dependency. So in our algorithm, the blocks in lower layer always have higher priority than the ones in the upper layer.

3.2 Problem Formulation

In this section, we will give the formulation of the block scheduling problem (BSP for short). As aforementioned, we try to maximize the average priority sum of all streaming blocks that are delivered to each node in one request period under heterogeneous bandwidth constraints. We consider two types of bandwidth constraints, namely access bottlenecks (inbound and outbound bandwidth capacity) and non-access bottleneck bandwidth (maximum end-to-end available bandwidth). As all the blocks have the same size, we use blocks per second to represent the amount of inbound, outbound, and end-to-end available bandwidth. Table 1 summarizes the notations in the rest of this paper.

Block Priority Definition. We give each block a priority due to its importance for a specified node. Two key factors that have impact on the block importance are considered here: the layer factor and the rarity factor. As in layered coding, the upper layer can be decoded only if the lower layers are available, we should ensure that the blocks of lower layer have higher priority. Besides, many previous works such as [15] demonstrate that requesting the block with rarest holders first brings more diversity to the system and help the block spread more rapidly. The following is the priority value of block j for node i:

$$\pi_j^i = \beta \Pi_R \left(\sum_{k \in NBR_i} h_{kj} \right) + (1 - \beta)\theta \Pi_L(\lambda_j), \text{ where } \beta = (d_j^i - C_i)/W_T \quad (1)$$

We let both function Π_R and Π_L monotonously decreasing. Function Π_L satisfies $\Pi_L(\lambda_j) \gg \Pi_L(\lambda_k)$ when $\lambda_j < \lambda_k$, for any block j and k so as to guarantee the layer dependency requirement. Parameter $0 \leq \beta \leq 1$ represents

the current position block j in the exchanging window. We let θ have relatively large value. Although our block priority definition is a simple linear combination of the two factors, it can guarantee the following key requirements: a) when a lower-layer block is in near the playback deadline ($\beta = 0$), it has much higher priority than any other upper-layer blocks (large value of θ); b) a block with fewer holders has higher priority than the one with more holders in the same layer and the same position in exchanging window.

Formulation. We formulate the block scheduling problem. We define the decision variable x_{kj}^i to denote whether node $i \in N$ should request block $j \in D_i$ from its neighbor $k \in NBR_i$:

$$x_{kj}^i = \begin{cases} 1, \text{node } i \text{ should request packet } j \text{ from neighbor } k \\ 0, \text{otherwise} \end{cases} \tag{2}$$

Our target is to maximize the average priority sum of blocks that each node can receive with heterogenous bandwidth constraints:

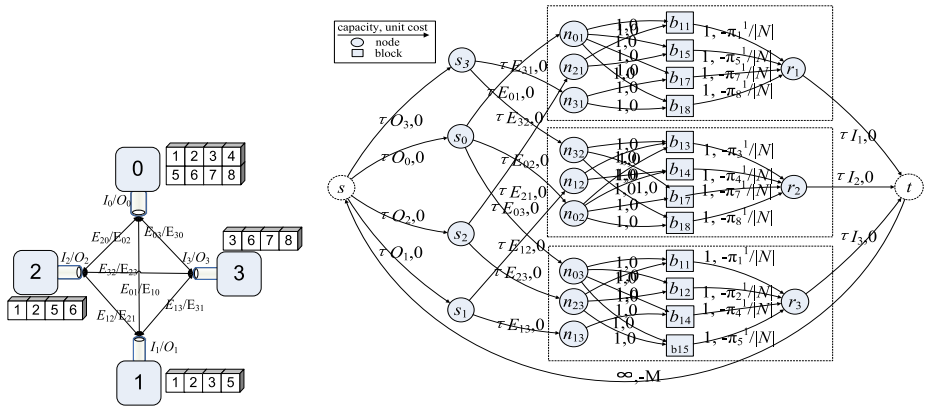
$$\begin{aligned} &\max \frac{1}{|N|} \sum_{i \in N} \sum_{j \in D_i} \sum_{k \in NBR_i} \pi_j^i h_{kj} x_{kj}^i \\ &\text{s.t.} \\ &\text{(a)} \sum_{k \in NBR_i} x_{kj}^i \leq 1, \forall i \in N, j \in D_i \qquad \text{(b)} \sum_{j \in D_i} \sum_{k \in NBR_i} x_{kj}^i \leq \tau I_i, \forall i \in N \\ &\text{(c)} \sum_{i \in NBR_k} \sum_{j \in D_i} x_{kj}^i \leq \tau O_k, \forall k \in N \qquad \text{(d)} \sum_{j \in D_i} x_{kj}^i \leq \tau E_{ki}, \forall i \in N, k \in NBR_i \\ &\text{(e)} \ x_{kj}^i \in \{0,1\}, \forall i \in N, k \in NBR_i, j \in D_i \end{aligned} \tag{3}$$

The formulation is a comprehensive integer linear programming, and we call this optimization problem global block scheduling problem (or global BSP for short). Constraint a) ensures no duplicate blocks are requested. Constraints b) and c) guarantee the blocks numbers that are downloaded from node i and uploaded to node k do not exceed the inbound and outbound bandwidth limitation respectively. Furthermore, constraint d) ensures that the number of blocks transmitted from node k to node i is under the constraint of end-to-end available bandwidth. Finally, constraint e) indicates that it is an integer programming.

4 Modeling and Global Optimal Solution

In this section, we will show that the global BSP (3) can be transformed into an equivalent minimum cost flow problem that can be solved in polynomial time. We call solving such a min-cost flow problem a global optimal solution. The min-cost flow problem is introduced in [16]. By double scaling algorithm [16], the time complexity for min-cost flow problem is bounded with $O(nm(\log\log U)\log(nC))$, where n and m are the number of vertices and arcs while U and C is the largest magnitude of arc capacity and cost respectively.

Fig. 1(a) and Fig. 1(b) show a sample of global BSP with four nodes and its min-cost flow modeling respectively. In Fig. 1(b), the two numbers close to



(a) A global block scheduling problem

(b) Model as a min cost flow problem

Fig. 1. An example of the equivalent MCFP

an arc represent the capacity and per unit flow cost of the arc. Rather than describe the general model formally, we merely describe the model ingredients for these figures. In data-driven streaming, we decompose a node into its three roles: a send, a receiver and a neighbor. We model each sender k as a vertex s_k, each receiver i as a vertex r_i, and each neighbor k of node i as a vertex n_{ik}. Further, we model a desired block j for node i as a vertex b_{ij}. Besides we add two virtual vertices: a source vertex s and a sink vertex t. The decision variables for this problem are whether to request block j from neighbor k of node i which we represent by an arc from vertex n_{ik} to vertex b_{ij} if block j is a desired by node i. These arcs are capacitated by 1 and their per unit flow cost is 0. And we insert arc from vertex n_{ik} to b_{ij} to indicate that neighbor k of node i holds block j. To avoid duplicate blocks, we add arc capacitated by 1 from b_{ij} to r_i and set the per unit flow cost as the priority of block j for node i multiplied a constant $-1/|N|$. To satisfy the outbound bandwidth constraint of node k, we add arc between vertex s and vertex s_k whose capacity is τO_k. And for the maximum end-to-end available bandwidth from neighbor k to node i, we insert arc from vertex s_k to n_{ik} with capacity τE_{ik}. Finally, to incorporate the inbound bandwidth constraint of node i, we introduce arc between r_i and t with capacity τI_i. To guarantee maximum number of blocks are delivered, we insert uncapacitated arc from vertex t to s that has a negative per unit flow cost with large absolute value. Finally we have the conclusion: The min-cost flow problem would yield the optimal solution of the global BSP. We omit the proof here.

5 Heuristic Distributed Algorithm - DONLE

In this section, based on the basic idea of the global optimal solution, we present the heuristic practical algorithm - DONLE which is fully distributed and

asynchronous. In DONLE, each node will decide from which neighbor to fetch which blocks at the beginning of its request period. As the request period is relatively short (such as 2 seconds), our scheduling algorithm should make decision as rapidly as possible. So in our heuristic distributed algorithm, we just do a local optimal block scheduling on each node based on the current knowledge of the block availability among the neighbors. The local optimal block scheduling can also be modeled as a min-cost flow problem. As shown in Fig. 1(b), the sub min-cost flow problem in the each rectangle is just the local optimal block scheduling.

However, one problem to do local scheduling is that each node does not know the optimal flow amount on arcs (s_k, n_{ki}) $(\leq O_k)$. In other words, we should estimate the proper upper-bound of the bandwidth from each neighbor. For simplicity, here we use a purely heuristic way for each node to estimate the maximum rate at which each neighbor can send blocks. Our approach is to use the historical traffic from each neighbor to do this. More formally, let Q_{ki} denote the estimated maximum rate at which neighbor $k \in NBR_i$ can deliver to node i. Of course, Q_{ki} should not exceed O_k. We let $g_{ki}^{(p)}$ denote the total number of blocks received by node i from neighbor k in the p^{th} period. In each request interval, we use the average traffic received by node i in the previous P periods to estimate Q_{ki} in the $(p+1)^{\text{th}}$ period: $Q_{ki} = \gamma \cdot \left(\sum_{\omega=p-P+1}^{p} g_{ki}^{(\omega)}\right)/P\tau$. Parameter $\gamma(>1)$ is a constant called aggressive coefficient. Then we can do a local optimal block scheduling formulated as below and solve it by its equivalent min-cost flow problem in polynomial time. We call it a local BSP. Our distributed algorithm is heuristic and we examine its performance and the gap between DONLE and the global optimal solution by simulation in Section 6.

$$\max \sum_{j \in D_i} \sum_{k \in NBR_i} P_j^i h_{kj} x_{kj}^i \tag{4}$$

s.t.

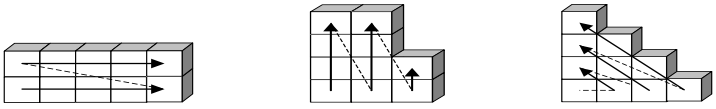
(a) $\sum_{k \in NBR_i} x_{kj}^i \leq 1, \forall j \in D_i$, (b) $\sum_{j \in D_i} \sum_{k \in NBR_i} x_{kj}^i \leq \tau I_i, \forall i \in N$

(c) $\sum_{j \in D_i} x_{kj}^i \leq \tau Q_{ki}, \forall i \in N, k \in NBR_i$, (d) $x_{kj}^i \in \{0,1\}, \forall k \in NBR_i, j \in D_i$

6 Performance Evaluation

In this section, we compare DONLE to other existing block scheduling strategies, and also examine the gap between DONLE and the global optimal solution. Three conventional strategies are compared here:

- Random Strategy: each node will assign each desired block randomly to a neighbor which holds that block. Chainsaw [5] uses this simple strategy. We examine how this method works in layered data-driven streaming.
- Local Rarest First (LRF) Strategy: As Section 3 depicted, a block that has the minimum owners among the neighbors will be requested first. DONet [4]



(a) Conservative (b) Aggressive (c) Tradeoff

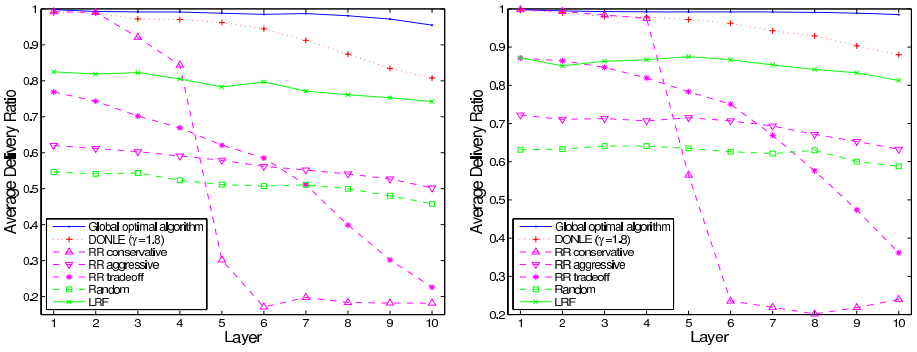
Fig. 2. Three round robin strategies

adopts this strategy. We also introduce this method into layered data-driven streaming and compare it with ours.

– Round Robin (RR) Strategy: All the desired packets will be assigned to one neighbor in a prescribed order in a round-robin way. If the block is only available at one sender, it is assigned to that sender. Otherwise, it is assigned to a sender that has the maximum surplus available bandwidth. In Fig 2, we introduce three conventional block ordering schemes used in the literature. Fig. 2(a) shows the conservative block ordering: it always requests blocks of lower layers first. On the contrary, aggressive block ordering scheme requests blocks of all layers with lowest sequence number (or time stamp) preemptively as illustrated in Fig. 2(b). Fig. 2(c) uses a zigzag ordering (slope=1) which is a tradeoff between the two extreme schemes.

To evaluate the performance, we define ***delivery ratio*** of a layer to represent the number of different blocks that arrive at each node before the playback deadline over the total number of blocks encoded in that layer. Since the total number of blocks in a layer is a constant that relies on the encoding and packetization, the average delivery ratio among all nodes can represent the throughput of the overlay. We compare DONLE and global optimal solution to the following five strategies: random, LRF, RR-conservative, RR-aggressive, RR-tradeoff. To ensure fair comparison, all the approaches have the same physical network and end-host participants in each scenario. Each curve in all the plots is an average over 10 simulation runs. We encode the video into 10 layers, and each layer has a rate of 50Kbps. To evaluate the quality of a specified layer, we average the delivery ratio of that layer over all nodes that can achieve the layer due to their inbound bandwidth. We use 500 nodes in the overlay and set the request period to 2 seconds. We set the node access bandwidth is asymmetric: the inbound bandwidth evenly distributes across 15Kbps to 1Mbps; while the outbound bandwidth of each node is randomly selected between half and one time of its inbound bandwidth. We set the outbound bandwidth of the source node to 2Mbps. Previous study [17] has shown that there is a sweet range of neighbor count or peer degree (roughly between 6 to 14) where the delivered quality to the majority of peers is high. Therefore, in our simulation, each node randomly selects 14 other nodes as its neighbors. We set the exchanging window to 10 seconds so as to avoid large delay and set the sliding window to 1 minute aiming to increase the opportunity of serving more neighbors.

As shown in Fig. 3(a), we compare the global optimal solution and DONLE to five other strategies. In this figure the bottlenecks are configured to be only at



(a) The bandwidth bottleneck is only at last mile (b) The bottleneck is not only at last mile

Fig. 3. Average delivery ratio at each layer

the last mile. We note that the global optimal solution has the best performance, and the delivery ratio in all layers is nearly 1. This demonstrates that the generated topologies have sufficient capacity to support all the nodes to receive all layers that they can achieve. The performance of DONLE is also fairly good. Most of the delivery ratio in lower layers has nearly 1 and most in higher layers is also above 0.9. However, though the RR-conservative method has perfect delivery ratio in layer 1 to 4, the quality has a cliff drop from layer 5. This means all the users can enjoy the video of 4 layers very smoothly, yet few nodes can receive data beyond the 4^{th} layer even if their inbound bandwidth is sufficient to support higher quality. This is because requesting lower layers first leads to bad block diversity among nodes. In contrast, the curve of the RR-aggressive method is flat. We note that most nodes can not watch even the base layer, although more blocks of higher layers are propagated, since this method does not consider the layer dependency. RR-tradeoff methods leverage the previous two methods. Here we use zigzag ordering with slope of 1/10 in RR-tradeoff. We found that the LRF strategy has more deliver ratio than round-robin schemes. Meanwhile, the random strategy has the poorest performance. As shown in Fig. 3(a), our distributed method DONLE outperforms other strategies much with a gain of 10%∼80%. Nevertheless, there is still about 12% gap between the global optimal solution and DONLE. In Fig. 3(b), we investigate the performance of these methods when the bottleneck is not only at last mile. In this figure, we let the maximum end-to-end available bandwidth distribute across 10Kbps 150Kbps. All the other configurations are not changed. The delivery ratio of all methods degrades compared to the results when bottleneck is only at last mile. The performance from the best to the poorest in turn is still DONLE, LRF, round-robin schemes, and random. It is observed that the rarity factor has significant impact on the throughput improvement in data-driven streaming. Therefore LRF strategy has better performance than round-robin and random strategies. Further, DONLE not only considers the rarity factor, but also does a local optimal scheduling that utilize the local bandwidth capacity as sufficient as possible as explained intuitively in Section 3. Hence DONLE outperforms other strategies.

7 Conclusion and Future Work

To improve the *throughput* of *data-driven* streaming in *heterogeneous* network, we propose a global optimal solution and a distributed algorithm - DONLE. Our simulation results show that our proposed algorithm DONLE is superior to a number of conventional strategies. For future work, we will study how to maximize the blocks delivered over a horizon of several periods, taking into account the inter-dependence between the periods. We are also planning to do more experiments on examining the parameter sensitivities in our algorithm.

References

1. McCanne, S., Jacobson, V., Vetterli, M.: Receiver-driven layered multicast. In: ACM SIGCOMM 1996. (1996)
2. Cui, Y., Nahrstedt, K.: Layered peer-to-peer streaming. In: NOSSDAV. (2003)
3. Zhao, J., Yang, F., Zhang, Q., Zhang, Z., Zhang, F.: Lion: Layered overlay multicast with network coding. IEEE Trans. on Multimedia (2007) Accepted to be published.
4. Zhang, X., Liu, J., Li, B., Yum, T.S.P.: Coolstreaming/donet: A data-driven overlay network for efficent media streaming. In: IEEE INFOCOM 2005. (2005)
5. Pai, V., Kumar, K., et al: Chainsaw: Eliminating trees from overlay multicast. In: IEEE INFOCOM 2005, Conell, US (2005)
6. Zhang, M., Zhao, L., Tang, Y., Luo, J., Yang, S.: Large-scale live media streaming over peer-to-peer networks through global internet. In: ACM workshop on Advances in peer-to-peer multimedia streaming (P2PMMS), Singapore (2005) 21–28
7. Agarwal, V., Rejaie, R.: Adaptive multi-source streaming in heterogeneous peer-to-peer networks. In: SPIE/ACM MMCN 2005, San Jose, CA, USA (2005)
8. Cohen, B.: Bittorrent website: http://bitconjuer.com. (2006)
9. Silverston, T., Fourmaux, O.: Source vs data-driven approach for live p2p streaming. In: IEEE International Conference on Networking 2006, Mauritius (2006)
10. GridMedia: http://www.gridmedia.com.cn/. (2006)
11. PPLive: http://www.pplive.com/. (2006)
12. Zhang, M., Xiong, Y., Zhang, Q., Yang, S.: On the optimal scheduling for media streaming in data-driven overlay networks. In: IEEE GLOBECOM. (2006)
13. Venkataraman, V., Francis, P.: On heterogeneous overlay construction and random node selection in unstructured p2p networks. (In: IEEE INFOCOM 2006)
14. Jiang, J., Nahrstedt, K.: Randpeer: Membership management for qos sensitive peer-to-peer applications. (In: IEEE INFOCOM 2006)
15. Bhrarmbe, A.R., Herley, C., Padmanabhan, V.N.: Analyzing and improving a bittorrent network's performance mechanisms. (In: IEEE INFOCOM 2006)
16. Ahuja, R.K., Magnanti, T.L., Orlin, J.B.: Network Flows: Theory, Algorithms, and Applications. (Prentice Hall)
17. Magharei, N., Rejaie, R.: Understanding mesh based peer-to-peer streaming. In: ACM NOSSDAV 2006, Newport, Rhode Island, USA (2006)

LSONet: A Case of Layer-Encoded Video Transmission in Overlay Networks

Hui Guo[1], Kwok-Tung Lo[1], and Jiang Li[2]

[1] Dept. of Electronic and Information Engineering
The Hong Kong Polytechnic University
Hung Hom, Kowloon, Hong Kong
[2] Dept. of Systems and Computer Science, Howard University
2400 Sixth Street, NW, Washington, DC 20059, USA
{enhguo, enktlo}polyu.edu.hk

Abstract. Live media streaming applications are increasing dramatically on the Internet such as IPTV, distance learning, video conference etc. Meanwhile, layered transmission is a promising solution to video streaming over the heterogeneous Internet. This paper presents LSONet, which leverages the advances in both the field of media compression, i.e., layered video coding, and the field of networking, i.e., application-level overlay networking. The purposes are, respectively, to obey the delay requirement, to improve bandwidth efficiency and to adapt to network jitter. The proposed system is completely self-organizing, and it can adapt to network dynamics in a fully distributed fashion. Extensive simulations have been performed. The results show that the system outperforms previous scheme in resource utilization and more robust and resilient for network fluctuation, which demonstrate that the proposed architecture and the associated protocol are well-suited for quality adaptive live streaming applications**.**

Keywords: Overlay network, layered video coding, live media streaming, quality adaptive.

1 Introduction

With the widespread availability of inexpensive broadband Internet connections for home-users, a large number of bandwidth-intensive applications have now become practical. This is the case for multimedia live streaming, such as IPTV [1], distance learning, video conference, news broadcasting and so on. Simultaneously, companying with the deployment of broadband access network for end-users, people found that the bottleneck is now laying on the server side, since the bandwidth required for serving many clients at once is huge and very costly for the broadcasting entity.

For alleviating the streaming server load and make the best use of bandwidth between end-users, the multimedia streaming service, served through application-level overlay (or peer-to-peer) networks is growing rapidly. Peer-to-peer overlay networks shifting the task of content distribution from the server to the users of the

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 485 – 494, 2007.
© Springer-Verlag Berlin Heidelberg 2007

network, which have been proposed in the last few years and provide very encouraging results. However, due to the bit rates are more variable and less predictable than tradition client-server environments, making it difficult to use peer-cooperative based networks to stream video for online viewing. In this situation, how precisely the available channel bandwidth can be estimated, what architecture and its associated protocol are exploited, and the excellent bandwidth adaptability of the source bit stream will play important roles in the end-to-end quality.

In this paper, parallel efforts have been exerted in the media compression field and networking field. We designed a self-organizing peer-assisted streaming architecture and the associated protocol, named as LSONet, which is provided by the scalable coding techniques and inspired by the spirits of peer-to-peer overlay networking. The proposed architecture aims for a better trade-off among bandwidth efficiency, network delay and streaming quality by utilizing the extra available bandwidth that might exist among clients. In the past few years, a number of P2P multicast tree were proposed [2, 3]. As the tree-based approaches are vulnerable with dynamic group variation, we adopt gossip-based mesh-like topology for overlay network construction [4]. Specifically, in tree-based multicast networks, the media contents on the links from the parent to its direct children are almost the same (or at least largely overlapped), whereas in our overlay P2P scheme, multilayered video content are distributed among mesh-like networks and mostly different. Employ data-driven and multi-source transmission scheme, packets can be exchanged among clients efficiently. As a result, the playback quality can be mutually improved and more robust for network fluctuation. Two types of topologies, physical topology and logical P2P topology, are introduced for system evaluation. The physical topology represents a real topology with Internet characteristic mode. The logical topology represents the overlay P2P topology built on top of the physical topology. Simulation and numerical results show that LSONet can achieve improved performance on video delivery quality, bandwidth utilization and service reliability, owing to the peer-assisted multi-path transmission and scalable layer-encoded streaming. Additionally, there is much less control overhead in LSONet comparing with DONet [5] system. The results indicate that the nodes in LSONet can cooperate perfectly that takes advantage of the fine-grained layered coding, and is fully compatible with the best-effort Internet infrastructure.

2 Related Work

Nowadays, a new kind of application is getting success: live streaming applications such as IPTV, distance learning, video conference, news broadcasting etc. Live streaming target a lot of people and consume many resources therefore they need group communication functionalities. Some of optimization prototype systems are proposed such as using a push-pull streaming approach in GridMedia [6], data-driven scheme for DONet in CoolStreaming [5], inter-overlay optimization based scheme in Anysee [7]. Unfortunately, they usually targeting traditional non-scalable video bit stream, do not specially considering quality-scalable video streaming in peer-to-peer environments.

In the coding community, layered coding is often referred to as scalable coding. The scalability includes temporal scalability, spatial scalability, and quality (or SNR) scalability. These scalable coding algorithms have been adopted in advanced compression standards, such as H.263+, MPEG-2, MPEG-4 and H.264. This paper does not specify any particular coding algorithm in the application layer. Nevertheless, a coder with a wide dynamic range, fast responsiveness, and fine granularity in terms of rate control is of particular interest. Examples include the Fine Granularity Scalability (FGS) [8] or Progressively FGS (PFGS) coders [9]. In the layered multicasting field, layered multicast was first proposed in [10], where a stream is separated into multiple layers, and then transmitted through different multicast channels with receiver-driven model. Many follow-up studies work on layer rate allocation mechanisms to maximize the overall streaming quality. Nevertheless, these studies are usually discussed on IP-multicast scenario.

3 Streaming Schemes for Layered Video

The proposed streaming architecture is a mesh-based structure. Unlike other mesh-based ALM structures, this architecture builds a specific overlay for each logical layer (*LL*), i.e., it is a method to construct multiple overlay networks so that different layers of the encoded video can take separated overlay networks for video transmission. It is capable of self-organizing because both the underlying mesh and the delivery path out of it are all dynamically adjustable. Both the dynamic changes in membership, such as client join or leave, and the underlying network conditions will trigger the self-organizing process. Simultaneously, clients in this architecture can mutually improve their quality by exchanging and relaying different logical layers of the streaming data inside the mesh. For example, the connection between client A and client B is used to transmit the first logical layer from A to B and, at the same time, to transmit the second logical layer from B to A. It implies that the manner of how the links in the mesh are utilized is quite different from that in any ALM overlay or tree-base structure. It is exactly the full-duplex connection among clients based on the data-driven request. On the contrary, in any tree-based structure, the connection can only be used to transmit data either from A to B or from B to A that maintained as parent-child relationship.

In LSONet, we stipulate that the clients immediately start to playback once they have received the base layer content, instead of receiving the whole bit stream. Because of the concision and higher streaming priority of base layer content, the transmission of base layer have smaller delay than streaming of traditional non-scalable bit stream. Consequently, the proposed system will not introduce any extra delay (besides the normal relay delay) and achieves shorter start-up latency. In addition, the proposed architecture achieves higher quality of service thanks to the layered video coding, which provides a straightforward means for clients to adjust its transmission policy when handling network dynamics. To guarantee the correct dependencies between bit streams of the physical layers, we impose stronger

dependencies between logical layers. For example, the logical layer 3 is dependent on logical layer 2.

$$LL_i \succ LL_j, \quad \text{for } i > j \geq 1 \tag{1}$$

where LL_i stands for i^{th} logical layer. Equation (1) indicates that a logical layer is decodable only if all its preceding lower logical layers have been accepted correctly. On the other hand, the loss of a higher logical layer will have no influence on lower logical layers. Note that a non-scalable bit stream can be regarded as a special case of a scalable bit stream, i.e., scalable bit stream with only one layer.

Finally, to take into account the diversities in the clients' capabilities such as computation power and network connection, two user configurable parameters, namely in-degree (k_{in}) and out-degree (k_{out}), are introduced. The former limits the maximum number of incoming connections the client can accept (excluding the link to the server) and the latter controls the maximum number of outgoing connections the client is willing to support. In simulations, we found that these two parameters have great influence on the system performance.

Note that LSONet is indeed a multi-sender overlay system, i.e. the receiving node gets different layers of stream from different sender. For the purpose to maximize the delivered quality from multiple senders and adaptive to network bandwidth variation, LSONet leverages data-driven mechanisms and advanced scalable coding techniques. Specifically, the receiver acts as coordinator among multiple senders rely on the layer requested and the message of available data of each sender, we denoted as layer-to-sender mapping mechanism.

4 Protocol Description

In this section, we present the protocol that can achieve all the design goals of the system. Every client in our system maintains a key data structure called a *transmission policy*. A transmission policy includes: (i) receiving which logical layers from which partners (including the server), (ii) relaying which logical layers to which clients and (iii) the available (remaining) inbound and outbound bandwidth. A client's transmission policy is subjective to dynamical change. As the key data structure, most operations of the protocol are about how to create, adjust and optimize the transmission policy. The *transmission policy* structure updated periodically based on the layer available information. In LSONet, each node maintains a Layer Availability Buffer (LAB) to record the specific available layer data it can provide. The LAB messages would be exchanged when a new request received from a receiver, and then the receiver schedules which layer is to be fetched from which partner accordingly based on all LAB messages from its partners. Similar to DONet system, the message delivery of LAB can resort to gossip-based mechanism [4]. However, our proposed system has more stable partnership: A particular layer of stream always relayed from a fixed sender, until the sender has left or failed. In this scheme, data availability message need not be sent out periodically, which have much less overhead for cooperation than DONet.

4.1 Client Join

4.1.1 Initial Join Procedure

When a client wishes to join the session, it contacts the server directly and the session starts immediately through normal unicast. This leads to little start-up delay, which is desirable. Upon joining, the server will allocate a globally unique ID (GUID) to the client. The GUID will not change throughout the life time of the client. The new comer then begins to identify some potential peers. We assume every client can get a list of closely located concurrent session members via a bootstrap mechanism. The bootstrap mechanism may be provided by a central directory or in any out-of-band manner. In practice, a simple but effective method is to ask the server to return some close IP addresses.

The joining client (denoted by X) contacts every client (denoted by C) in the list and collects the following information: (i) underlying network conditions (mainly the available bandwidth and the round trip time (RTT)) measured over the virtual links between X and C; and (ii) current transmission policy of C. In our protocol, the available bandwidth between two members is estimated using the well-established formula [11]:

$$B = \frac{MTU}{t_{RTT}\sqrt{2p/3} + t_{out}\sqrt{3p/8}\; p(1+32p^2)} \tag{2}$$

Where B represents the estimated available bandwidth, MTU is the packet size transmitted over the link, t_{out} is TCP time out, t_{RTT} is the measured round-trip time in seconds, and p is the measured packet loss rate. Having collected enough information, the joining client now selects some clients with relative large available bandwidth as peers and calculates an optimal transmission policy using the algorithm described in the next subsection. Note that the in-degree and out-degree constraints must be obeyed during the peer selection process. Finally, the joining client finishes the whole joining procedure by notifying the server and all peers of its new transmission policy.

4.1.2 Optimal Transmission Policy Decision

Suppose a joining client X chooses a set of N ($N \le k_{in}$) peers as providing peers, $\mathbf{P}=\{P_1, \ldots, P_N\}$, the corresponding available bandwidths are $\{b_1, \ldots, b_N\}$, which is normalized in the unit of logical layers. Let L be the maximum number of logical layers the server feeds into the system, and denote the server as P_0. We first defined a matrix $D=\{d_{ij}\}_{(N+1)\times L}$, this matrix defines the distribution of the multiple sources that X can get contents from. Where $d_{ij}=1$ ($0 \le i \le N$, $1 \le j \le L$) indicates that P_i is receiving logical layer j from the server directly and thus can relay it to X. X is now ready to determine the optimal transmission policy, i.e., which logical layers should be transmitted from which neighbors, so as to maximize its total number of received logical layers while not violating the constraints on link bandwidth and layer dependency policy. The problem can be modeled as a *zero-one integer programming* problem. Let boolean variable $x_{ij} \in \{0,1\}$ represent whether or not X will receive logical layer j from P_i, with $x_{ij}=1$ means X will. Let $J=\{1,2,\ldots,L\}$ and $I=\{0,1,\ldots,N\}$. The problem is then formulated as follows:

$$\text{Maximize} \sum_{i\in I, j\in J} x_{ij} \tag{3}$$

$$\text{s.t.} \sum_{i\in I} x_{ij} \le 1, \forall j \in J \tag{4}$$

$$\sum_{i\in I} (x_{ij} - x_{ij+1}) \ge 0, \forall j \in J \tag{5}$$

$$\sum_{j\in J} x_{ij} \le b_i, \ \forall i \in I \tag{6}$$

$$x_{ij} \le d_{ij}, \ \forall i \in I, j \in J \tag{7}$$

Equation (3) is the objective function, which is to maximize the total number of received logical layers. Equation (4) indicates that the duplicated logical layers from different neighbors should be avoided. Equation (5) expresses the dependencies that must be maintained between logical layers. Equation (6) and (7) are the constraints on bandwidth consumption constrains.

In fact, the complexity of computation is extremely low because both the number of supplying peers and the maximum number of logical layers are relatively small in practice. In LSONet, the receiver can adjust number of delivered layers by joining a different number of multicast sessions. This allows the receiver to regulate overall incoming throughput (and thus overall delivered quality) at the level that does not cause congestion in the network, *i.e.*, the receiver implements some type of congestion control mechanism by regulating incoming throughput.

4.2 Client Leave or Failure

By client leave, we mean that the client notifies all its collaborating peers before it actually leaves the session (explicit leave), while by client failure, we mean that the client did not or failed to notify its collaborating peers when it actually left (implicit leave). In this work, we assume the failure of any client will be detected by its peers, for example, through the periodical heartbeat mechanism. Due to the strong dependency imposed on the logical layers, the protocol must promptly react to any client leave/failure. Specifically, the protocol should suppress the propagation of the bad impact of the client leave/failure. First of all, let us study an example to get a feeling on how a client leave may propagate.

For ease of presentation, we introduce a re-schedule algorithm that is used to handle the loss of one logical layer due to either peer leave/failure or bandwidth fluctuation. For a client with missing a logical layer, if there is another peer (including the server) who can provide the missing layer and under available bandwidth bound, it can simply ask that peer to relay the lost layer. In this case, the client maintains the same quality after the re-scheduling. Otherwise, re-schedule algorithm will try to obtain the lost layer by sacrificing one of the higher (less important) layers recursively. We use the example in Fig.1 to explain the algorithm. In Fig.1-(a), if client F leaves, client A who is forwarding data to F will simply stop forwarding. That is, A is not affected by the leave of F. Unfortunately, client Y has to adjust its transmission policy in time to maintain normal playback since LL_2 is relayed from F. During the re-schedule process, Y first checks with client X. Presently, Y is receiving the layer LL_5 from X. Since X receives only the LL_4 and LL_5 from the server directly, it is not allowed to relay LL_2 to Y. Consequently, Y will resort to Z. Initially Y is receiving the layer LL_4 from Z. Depending on whether Z can relay LL_2 to Y or not, the example is branched into two cases, as shown in Fig.1-(b) and Fig.1-(c).

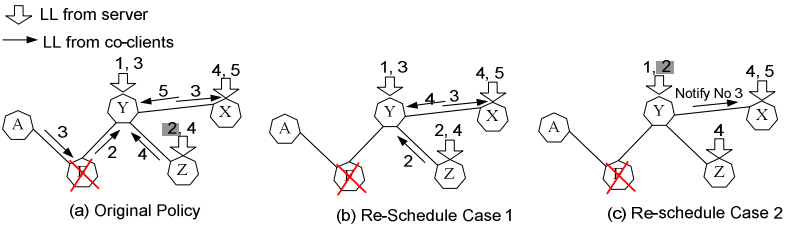


Fig. 1. Examples of adjustment of transmission policy due to client leave/failure

In the first case, Y asks Z to stop sending LL_4 and switch to LL_2 instead. Now, the lost layer at Y shifts from LL_2 to LL_4 now. In the same way, the lost layer is further shifted from LL_4 to LL_5 by requesting X to relay LL_4 instead of LL_5. Clearly, after re-scheduling, Y will receive four logical layers with the new transmission policy. This adjustment ends locally without spreading out to any peer. In the second case, Y can not rescue LL_2 from any peer but the server. However, due to the bandwidth limit on the link from the server to Y, the LL_3 has to be traded for LL_2. In this case, Y will obtain only two logical layers. Also, the missing LL_3 will influence peer X. As a result, Y must notify X of losing LL_3. X will react and run the re-schedule algorithm to adjust its own transmission policy.

5 Performance Evaluation

5.1 Simulation Setup

Two types of topologies, physical topology and logical topology, are generated in our simulation. The physical topology should represent the real topology with Internet characteristics. The logical topology represents the overlay P2P topology built on top of the physical topology. All P2P nodes are in a subset of nodes in the physical topology. The router-level physical network is generated according to the Transit-Stub graph model, using GT-ITM topology generator [12]. In our simulations, we randomly select 500 to 1500 nodes as LSONet nodes, for overly networks construction. The overlay nodes join and leave the network using an exponential on-off distribution. Unless otherwise stated, the default periods of on and off status have mean value of 250 seconds. The scalable source bit stream is composed of 8 logical layers, which has mean value of 256kpbs bandwidth for each layer, and 1Kbyte of each packet size.

5.2 Delivery Quality and Bandwidth Efficiency

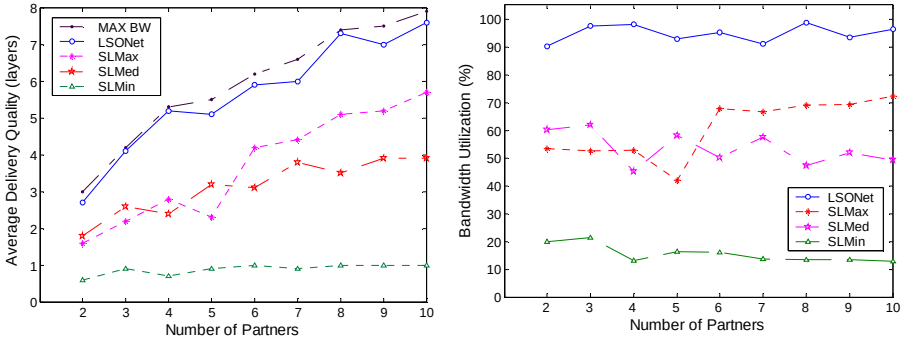
To testify the effectiveness of scalable coding video, we compare the performance of LSONet with the following scenarios:

- Single Layer stream with minimum bandwidth: In this case we employ a 256kpbs CBR source bit stream for media delivery instead of the scalable video stream, denoted as SLMin case.

- Single Layer stream with medium bandwidth: In this case we instead the scalable video stream with a 1Mpbs CBR stream for video delivery, denoted as SLMed case.
- Single Layer stream with maximum bandwidth: In this case we instead the scalable video stream with a 2Mpbs CBR stream for video delivery, denoted as SLMax case.

We compare the average delivery quality of different scenarios where a variable number of partners employed. Fig.2-(a) depicts the average delivered quality by LSONet, SLMin, SLMed and SLMax for different numbers of cooperative partners, ranging from 2 to 10. Note that the average delivery quality normalized by aggregate number of layers in this experiment. We have also shown the maximum deliverable quality as an upper bound for average delivered quality. This figure shows that the average delivery quality by LSONet is higher than the other three scenarios. Lower delivered quality by SLMin, SLMed and SLMax is primarily due to the inability to utilize residual bandwidth from each sender. Meanwhile, we notice that the delivery quality by LSONet is very close to the maximum deliverable quality. The small gap between them represents the residual aggregate bandwidth is insufficient for adding another layer. The result can be concluded that the LSONet has indeed made efficiently use of the available bandwidth and is an effective solution for multi-layer video delivery over pee-to-peer networks.

As mentioned previously, LSONet can adequately use of residual bandwidth of multiple senders. We investigate the utilization of aggregated bandwidth as a function of overlay size, i.e., total number of participating nodes, in the same simulation. Fig.2-(b) depicts the bandwidth utilization by different scenarios for the logical topologies are generated with the number of peers (nodes) ranging from 500 to 1500. The figure shows that the LSONet always keeps higher bandwidth utilization. We also found that the curve line of SLMin and SLMed declines dramatically when the number of participating nodes increases over a threshold. It is the reason that although the gross available bandwidth increases with the augment of overlay size, the streaming throughput remains limited due to the finite bandwidth requirement for non-scalable CBR streams.



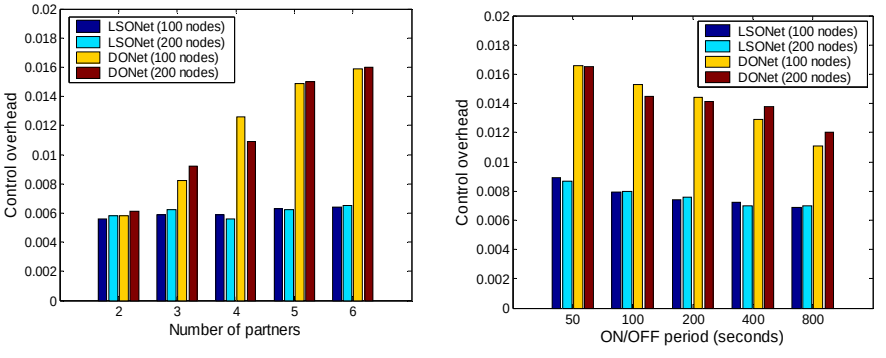
(a) Average Quality (layers) (b) Bandwidth Utilization

Fig. 2. Experimental results for LSONet, SLMax, SLMed and SLMin

5.3 Control Overhead

As mentioned previously, we employ the gossip-based protocol for exchanging data availability of multiple senders. The primary disadvantage of this protocol is larger control overhead due to its property of pure decentralized overlay system. In this experiment, we define the control overhead as the ratio of control traffic volume over video traffic volume at each node. And we present the results of another gossip-based live media streaming system, DONet [5], for comparison. Usually, the number of partners is a key factor to the control overhead.

Fig.3-(a) depicts the normalized control overhead as a function of the average number of partners in a stable environment, i.e., the lifetime of each node equals to the playback duration of streaming, typically as 120 min. The source bit stream is composed of 3 logical layers, which has mean value of 256kpbs bandwidth for each layer. The figure shows that the overhead in DONet system increases with a larger number of partners, while in our proposed system, the control overhead keeps invariability on the whole. The reason is that in DONet, the video stream is partitioned to many segments, each node periodically exchange segment's availability information with partners, and then schedules which segment is to be fetched from which partner accordingly. Unlike DONet, our proposed system has more stable partnership: A particular layer of stream always relayed from a fixed sender, until the sender has left or failed. In this scheme, data availability message need not be sent out until a specific layer request is received from a partner. We also examine the property with dynamic environments. Fig.3-(b) shows the control overhead as a function of ON/OFF period (ΔT). Not surprising, the control overhead increases with a shorter ON/OFF period in both systems. This is because of more dynamic node behaviors. Additionally, the results show that the LSONet can achieve much lower overhead than DONet.



(a) Control overhead as a function of number of partners

(b) Control overhead as a function of ON/OFF period

Fig. 3. Control overhead under stable and dynamic environments

6 Conclusion

In this paper, we propose a layered video live media streaming architecture for overlay networks. The video is encoded into multiple layers and a different overlay structure is maintained for each layer. The design consists of some key technologies, namely multi-source and mesh-based packet exchange among clients, data-driven transmission direction, assign resources based on their locality and delay dynamically, and the protocol is self-organized and operates in a decentralized manner. Maintaining continuous playback is a primary objective for streaming applications. Fortunately, owing to scalable layer-encoded streaming, the client in LSONet maintains continuous playback if only the basic layer can be retrieved from any sender. It also has shorter start-up latency for the sake of the playback immediately start as long as enough base layer data has been fetched. Furthermore, more stable streaming partnership makes LSONet nodes need not send data availability message periodically, which have trivial overhead for gossip-based protocol.

References

1. Alfonsi, B.: I Want My IPTV: Internet Protocol Television Predicted a Winner. IEEE Distributed Systems Online, vol. 6, no. 2, 2005.
2. Chu, Y., Rao, S. G., Zhang, H.: A case for end system multicast. In Proc. of ACM SIGMETRICS, June 2000.
3. Cui, Y., Li, B.C., Nahrstedt, K.: oStream: Asynchronous streaming multicast in application layer overlay networks. IEEE Journal on Selected Areas in Communications, 2004, 22(1):91-106.
4. Ganesh, J., Kermarrec, A.-M., Massoulie, L.: Peer-to-peer membership management for gossip-based protocols. IEEE Transactions on Computers, 52(2), Feb. 2003.
5. Zhang, X. Liu, J., Li, B., Yun, T.-SP: CoolStreaming/DONet: A Data-driven Overlay Network for Live Media Streaming. In Proc. of IEEE INFOCOM'05, Miami, FL, USA, March 2005, 2102 – 2111
6. Zhang, M., Luo, J.G., Zhao, L.: A Peer-to-Peer Network for Live Media Streaming–Using a Push-Pull Approach. In Proc. of the ACM Multimedia'05, Singpore, November 2005, 287~290
7. Liao, X., Jin, H., Liu, Y.: AnySee: Peer-to-Peer Live Streaming. To appear at IEEE INFOCOM 2006, Barcelona, Spain, April 2006.
8. Li, W.: Overview of Fine Granularity Scalability for Internet Video. IEEE Transactions on Circuits and Systems for Video Technology. Vol. 11, pp. 301-317, 2001.
9. Wu, F., Li, S., Zhang, Y-Q.: A Framework for Efficient Progressive Fine Granularity Scalable Video Coding. IEEE Transactions on Circuits and Systems for Video Technology. Vol. 11, pp. 332-344, 2001.
10. McCanne, S., Jacobson, V., Vetterli, M.: Receiver-driven layered multicast. In Proc. of ACM SIGCOMM'96, Stanford, CA, Aug. 1996, 117–130
11. Padhye, J., Firoiu, V., Towsley, D.: Modeling TCP throughput: a simple model and its empirical validation. In Proc. of ACM SIGCOMM 98, Sept. 1998. 303~314
12. Zegura, E. W., Calvert, K., Bhattacharjee, S.: How to Model an Internetwork. In Proc. of IEEE INFOCOM'96, SF, CA, Mar. 1996.

A Strategyproof Protocol in Mesh-Based Overlay Streaming System

Rui Sun[1], Ke Xu[2], Zhao Li[2], and Li Zhang[1]

[1] School of Software, Tsinghua University, Beijing, P.R. China
[2] Department of Computer Science and Technology, Tsinghua University, Beijing, P.R. China
{sunrui99, xuke, lizhao}@csnet1.cs.tsinghua.edu.cn,
zhangli@tsinghua.edu.cn

Abstract. CoolStreaming is the first protocol which introduces the mesh structure into Application Layer Multicast(ALM) in media streaming application, i.e. each agent may have two or more parents instead of only one parent. However, since the agents in Mesh-based ALM(MALM) are strategy and selfish, the effect of cheating behavior may not be ignored. To solve this problem, we apply the VCG mechanism design into MALM network model, and devise a strategyproof mechanism to avoid the agent cheating. As a result, the goal to maximize the system outcome can be achieved. In addition, we design a distributed algorithm to realize our mechanism. The algorithm can dynamically adapts to form a better multicast mesh, though ALM network parameters and constraints change dynamically in reality. The correctness and performance of this distributed algorithm are verified by the following experimental results.

1 Introduction

Application Layer Multicast(ALM)[13,14,17] is one multicast vehicle achieved in application layer. Comparing to IP multicast achieved in network layer, ALM build an overlay network out of unicast tunnels across cooperative participating end-hosts, called overlay agents, and multicast data is relayed among these overlay agents. [5,6] build a tree structure for Tree-based ALM(TALM), and for solving the bandwidth and dynamic problems, CoolStreaming [1] constructs a mesh structure, which is called Mesh-based Application Layer Multicast(MALM), to data delivering.

In MALM, data is delivered among the end hosts instead of the obedient routers, and relay agents are now selfish and strategic end hosts. Therefore, the cooperative behavior among the routers cannot be taken for granted. The selfish and strategic overlay agents may optimize their own utility. As a result, these selfish agents are not always like the routers to optimize the global utility.

[2,9] studies the theory of mechanism design and introduces the VCG mechanism. VCG mechanism is widely used in strategyproof problem to encourage the agents to tell truth.One goal of us is to design a strategyproof mechanism based

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 495–504, 2007.
© Springer-Verlag Berlin Heidelberg 2007

on VCG, which will make each agent tell truth, to build the truthful multicast mesh and optimize the MALM's system outcome. Another contribution of this paper is to design a distributed and trustable algorithm to realize the mechanism according to our theoretical model.

The remainder of this paper is organized as follows. Sec. 2 discusses the related work. Sec. 3 introduces some background information on the mechanism design and the VCG mechanism. Sec. 4 gives the description of network model. Sec. 5 focuses on the distributed algorithm design to realize the strategyproof mechanism. Extensive simulations and analysis are conducted in Section 6. Finally, conclusions of this paper are presented in Section 7.

2 Related Work

Nisan and Ronen first tried to solve network problems through introducing the idea of Algorithm Mechanism Design(AMD) [9,15]. Since then, many computer scientists have joined this field [10,16,18]. In concrete problems of multicast, [11] designed distributed payment algorithms using VCG mechanism to encourage multicast receivers to tell truth in multicast tree. [3] applies mechanism design into link-weighted ALM to solve the problem of receiver cheating. For simplify the process of constructing and maintaining the ALM tree, [4] design a scheme of mechanism design through building a truthful minimum cost multicast tree. Deferent with our work, their multicast are all tree-based, although [4] first builds a mesh-based overlay network.

For solve the bandwidth problem in TALM, some studies introduced a mesh structure into the ALM of streaming. After MALM is introduced into ALM, the study of incentive mechanism in p2p streaming applications also becomes a hotpot, [7,8]. In these incentive mechanisms, one assumption is that the nodes of p2p streaming network are all honest. However, the assumption is not taken for granted in peer-to-peer network. In our work, we design a strategyproof mechanism to encourage each agent to declare real private type to the public system, so that the real maximum outcome can be achieved. At this point, this is the first work in p2p streaming application as far as we known.

3 Background Knowledge

Consider the model of a n-players static game of non-complete information, the n-players is denoted by n-agents $\{a_1, a_2, a_3, ..., a_n\}$. Each agent a_i has a set of possible private information (termed its private types) $T_i = \{t_i^1, t_i^2, t_i^3, ..., t_i^n\}$. a_i has a private type $t_i \in T_i$, which is the real private type of a_i. Let the vector $T = \{t_1, t_2, t_3, ..., t_n\}$ represents the set of all agents' private types. When a_i is needed to declare its private type t_i to the public system, it can declare its real t_i as $s_i \in T_i$. Since agent a_i is a selfish and strategy player, its s_i may not equal to t_i just to gain more benefit. s_i is also called a strategy, then agent a_i's strategy space is denoted as $S_i = \{s_i^1, s_i^2, s_i^3, ..., s_i^n\}$.

In a mechanism problem, when most of agents try to choose the same strategy according to the rule, the strategy is considered as a dominant strategy. In other words, the dominant strategy will maximize the agent's utility u_i, no matter what the other agent do.

We say that a mechanism is an implementation with dominant strategies (or in short just an implementation) if

Definition 1 (A Dominant Strategy). *A dominant strategy equilibrium s^* satisfies the condition $u_i(s_i^*, s_{-i}) \geq u_i(s_i, s_{-i})$ for all agents a_i and all strategies (s_i, s_{-i}).*

In first item, s_{-i} is the strategies space, simply means the set of strategies chosen by all agents except a_i. We define the utility u_i of the agent a_i is the sum of the valuation v_i and the payment p_i, $u_i = v_i + p_i$.

In order to motivate each agent to become honest, or at least does not have any incentive to cheat, in economics, the strategyproof mechanism is introduced to solve the problem. We give the definition of strategyproof mechanism as follow.

Definition 2 (A Strategyproof Mechanism). *A mechanism is strategyproof if for every agent a_i:*

1. *the strategy space S_i is to declare their types, $S_i = T_i$;*
2. *declaring the true type is a dominant strategy, $s_i^* = t_i$.*

The Vickrey-Clarke-Groves (VCG) mechanism has been proved to be strategyproof, defined as follows.

Definition 3 (A VCG Mechanism). *A Vickrey-Clarke-Groves (VCG) mechanism is the family of mechanisms $M(s) = (o(s), p(s))$ such that:*

$$o^*(s) \in argmax \sum_{j=1}^{n} v_j(s_j, o(s)) \tag{1}$$

$$p_i(s) = \sum_{j \neq i} v_j(s_j, o^*(s)) - \sum_{j \neq i} v_j(s_j, o^*_{-i}(s_{-i})) = \sum_{j \neq i} (v_j - v_j^{-i}) \tag{2}$$

In the definition, the VCG mechanism defined the output function Eq. (1) and the payment strategy function Eq. (2). $o^*(s)$ is the desired equilibria outcome, which is obtained by maximizing the sum of all agents' valuations. According to the VCG payment function, we have

$$\begin{aligned} u_i &= v_i + p_i \\ &= v_i + (\sum_{j \neq i} v_j - \sum_{j \neq i} v_j^{-i}) \\ &= \sum_{j} v_j - \sum_{j \neq i} v_j^{-i} \end{aligned} \tag{3}$$

From Eq. (3), we can conclude that the payment p_i of a_i is independent with the valuation v_i of a_i, since the payment p_i is computed by the valuations of all agents except agent a_i. Therefore, in this situation, the agent a_i has no incentive to cheat, because it just obtains the same payment needed to pay no matter what it does cheat or not.

4 Notations and Network Model

Consider an Peer-to-Peer network modeled as a directed graph $G(N, E)$, where N is defined to represent the finite nonempty set of network nodes and E illuminates the set of all edges e of graph G while $E \subseteq N \times N$. Let $n = |N|$ be the number of agents in p2p network.

Let $r_{ij} \geq 0$ represents the throughput of link ij. r_{p_i} is the total throughput of all parent links of node i. Since there are multiple parent nodes to forward data to the same child node in MALM, a key practical issue here is how to divide one buffer into segments and receive different segments from different parents. In order to do their endeavor to disseminate data to downstream nodes and gain the most data transmission performance, each node should allocate all of its outgoing bandwidth into its child nodes according to the amount of the incoming width of child node. Since one goal of data transmission is data from the different parent will finish the transmission on the same time, or the difference in transmitting time is very small, we educe an algorithm of the link throughput. For link (p_j, a_i):

$$r_{P_j,a_i} = \frac{L_{in,a_i}}{\sum_{i=0}^{m} L_{in,C_{\{P_j,i\}}}} \times L_{out,p_j} \tag{4}$$

Additionally, since $r_{P_{a_i}}$ should be less than L_{in,a_i}, when the answer of Eq. (4) exceed the limited incoming throughput of a_i, we need adapt the link throughput of each parent through the equation as Eq. (5).

$$r'_{P_j,a_i} = \frac{r_{P_j,a_i} \times L_{in,a_i}}{r_{P_{a_i}}} \tag{5}$$

5 Strategyproof Mechanism and Implementation

In this section, we shall apply the VCG mechanism to our network model. Firstly, we have to quantify the notion of each node's valuation and utility.

5.1 The Valuation Function

Consider the benefit b_i of the agent to the function of each agent's receiving one fixed-length multicast data message, according to the discussion above, it is reasonable that the benefit b_i of the agent a_i is the function of r_{P_i}, i.e. $b_i = b_i(r_{P_i})$. Similarly, we consider the total cost Tc_i of agent a_i is $\sum_{j \in \{Ch_i\}} (Percent_i^j \times c_i)$, where $Percent_i^j$ is defined to represent the percent of the link throughput r_{a_i,Ch_j} in the total throughput $r_{P_{Ch_j}}$ of agent Ch_j. In our paper, we consider the valuation v_i of each agent is in the form of benefit minus cost. Therefore

$$v_i = b_i - Tc_i = b_i(r_{P_i}) - Tc_i = b_i(\sum_{j \in \{Pa_i\}} r_{P_j,a_i}) - \sum_{j \in \{Ch_i\}} (Percent_i^j \times c_i) \tag{6}$$

5.2 The Payment Function Design

The key point in strategyproof mechanism design is the payment strategy. Considering the computing feasibility, our algorithm is distributed. Therefore, for obtaining the utility of one agent, we need to compute its payment first.

First, we expand the payment function according to our network model. We divide the set of the network agents into four disjoint subsets, which are the set of a_i's parents, the set of a_i's descendants which can find another second-best parents set, the set of a_i's descendants which can not find another second-best parents set, the set of other agents.

The cost of each agent's forwarding one unit message can be regarded as identical. We take the unit forwarding cost as c for each agent. Since all parents of the agent a_i supply one unit data to a_i together, and no parent relay redundant data, the sum of the percent of the data relayed by each parent being in the unit data becomes one, i.e. $\sum_{j\in\{P_i\}} Percent_j^i = 1$. According to the analysis, the VCG payment function can be expanded as follows:

$$\begin{aligned}
p_i &= \sum_{j\in\{P_i\}}(v_j - v_j^{-i}) + \sum_{j\in\{GC_i\cap\exists P_j^{-i}\}}(v_j - v_j^{-i}) + \sum_{j\in\{GC_i\cap\exists P_j^{-i}\}}(v_j - v_j^{-i}) \\
&\quad + \sum_{j\in Allagents\ (G_i\cup P_i)}(v_j - v_j^{-i}) \\
&= \sum_{j\in\{P_i\}}(-Percent_j^i \times c_j) + \sum_{j\in\{GC_i\cap\exists P_j^{-i}\}}(b_j - b_j^{-i}) \\
&\quad + \sum_{j\in\{GC_i\cap\exists P_j^{-i}\}}(b_j - c\times\sum_{k\in\{Ch_i\}} Percent_i^k) + \sum_{j\in All\backslash(G_i\cup P_i)}(b_j - b_j^{-i}) \qquad (7) \\
&= -(\sum_{j\in\{GC_i\cap\exists P_j^{-i}\}}\sum_{k\in\{Ch_i\}} Percent_i^k + 1)\times c + \sum_{j\in All\backslash\{P_i\}} b_j(\sum_{j\in\{Pa_i\}} r_{ji}) \\
&\quad - \sum_{j\in All\backslash\{P_i\cup(GC_i\cap\exists P_j^{-i})\}} b_j^{-i}(\sum_{j\in\{Pa_i\}} r_{ji}) \\
&= Tc + Tb + Tb^{-i}
\end{aligned}$$

5.3 The System Outcome Function Design

Consider the sum of the system valuation as the system outcome, we should maximize the sum. So each agent should select the subset which can maximize the function $\sum_j v_j - \sum_{j\neq i} v_j^{-i}$, i.e. each agent will maximize the sum of system valuation after it joins the multicast. Therefore we consider that the equilibrium s^* of the strategyproof MALM, should satisfies

$$s^* \in max(\sum_j v_j(s_i) - \sum_{j\neq i} v_j^{-i}(s_i)) \qquad (8)$$

For assuring that the system valuation will not be negative, we have a participating constraint. If $max(\sum_j v_j(s_i) - \sum_{j\neq i} v_j^{-i}(s_i)) < 0$, the agent a_i should not join the game.

According to Eq. (3), $max(u_i) = max(\sum_j v_j - \sum_{j \neq i} v_j^{-i})$. Additionally, according to the participating constraint, we can conclude that maximizing the system outcome is the same as maximizing the u_i^+, i.e. $max(u_i^+)$.

5.4 Distributed Algorithm Design

Since computing each link's throughput is a common algorithm, which is used in the algorithm of computing payment, we independently give the algorithm as follow.

Calculate Each Link Throughput

```
program calculateThroughput (limitOut_i, limitInSet_Ch_i)
  var totalLimitIn: sum(limitInSet_Ch_i)
  for each limitIn in limitInSet_Ch_i
    linkT(i,j): limitOut_i * (limitIn / totalLimitIn)
  end for
```

The first term Tc in Eq. (7) can be calculated easily according to the descendant private information. The agent should be consider that it does not find the second best parents set, when its incoming link throughput is zero, i.e. $\sum_{j \in \{Pa_i\}} r_{ji}^{-a_i} = 0$. So the algorithm of calculating Tc is shown as follow.

Calculate Total Cost

```
program calculateTc()
  msgIn: recvMsg()
  payment: msgIn.payment
  for each Ch in GC_i
    linkT_noI: msgIn.linkT_noI
    if linkT_noI == 0 && Ch not calculated
      sum: sum + msgIn.sumOfPer
    end if
  end for
  Tc: -(sum + 1) * c
```

Calculating the term Tb_j and $Tb_j^{-a_i}$ is similar, so we'll obtain these two value through a single algorithm. Essentially, we need to obtain the descendant's r_j and $r_j^{-a_i}$ of the agent a_i and the descendant of the agent a_i's parents set except a_i. This can be achieved by one message with its each parent. When we start to calculate the descendants of each parent, we may find that some descendants may be calculated twice or more, since one agent can be the child of several parents. So the algorithm need avoid this situation through marking if one agent is computed.

Calculate Total Benefit With a_i and Without a_i

```
program calculateTb()
  for each c in Ch
    TbWithI: TbWithI + benefit(TbWithI(c))
    if totalRWithoutI(c) != 0
      TbWithoutI: TbWithoutI + benefit(totalRWithoutI(c))
    end if
  end for
  for each p in Parent
    limitOut: p.limitOut
    calculateThroughput(p, Ch(p))
    for each j in Descendent{p.Children except a_i}
      if a_i in p.Children
        TbWithI: TbWithI+benefitOfIUpdate(totalRWithI(j))
        TbWithoutI: TbWithoutI+benefitOfIUpdate(totalRWithoutI(j))
      else if a_i not in p.Children
        TbWithI: TbWithI+benefitOfIJoin(totalRWithI(j))
        TbWithoutI: TbWithoutI+benefitOfIJoin(totalRWithoutI(j))
      end if
    end for
  end for
```

Additionally, when one node wants to calculate its utility, it may be an existing child or potential child of the agent in the new parent set. For an existing child, the term Tb_j should be calculated by adding the answer of the function $benefitOfIUpdate()$ together and for an potential child, Tb_j should be calculated by adding the answer of the function $benefitOfIJoin()$ together. $benefitOfIJoin()$ means the benefits of the descendants of P_i after a_i becoming the child of the agent p_i. The algorithm is shown as follow.

6 Implements and Experimental Evaluation

In our simulation of single-source MALM session, all topologies are generated by GT-ITM [19]. The agent number n is chosen from 100, 200, 500, to 1000; and the network density d is assigned 20%, 60%, 80% and 100%, respectively. The number of each agent's parents is changed from 1, 2, 4, to 10. The throughput limit of each agent is randomly generated in uniform distribution. The incoming throughput limit is in 10-50Kbps and the outgoing throughput limit is in 30-100Kbps. In our experiment, we define the benefit and cost function to be $b(r_{P_i}) = 5 \times r_{P_i}$ and $Tc_i = 10 \times (\sum_{j \in Ch_i}^{P} Percent_i^j)$.

Fig. (1) and Fig. (2) evaluate the correctness of our distributed algorithms and protocol implementations. We track the system total valuation over time, when each agent separately has 1, 2, 4 and 10 parents in the system of $n = 500$,

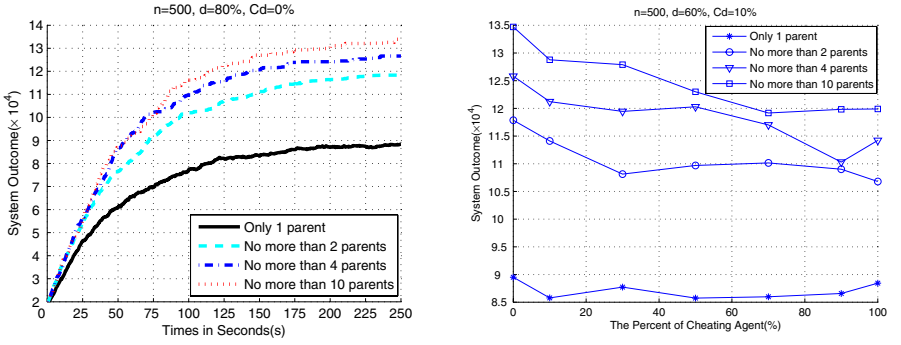


Fig. 1. The changes of system outcome and total throughput

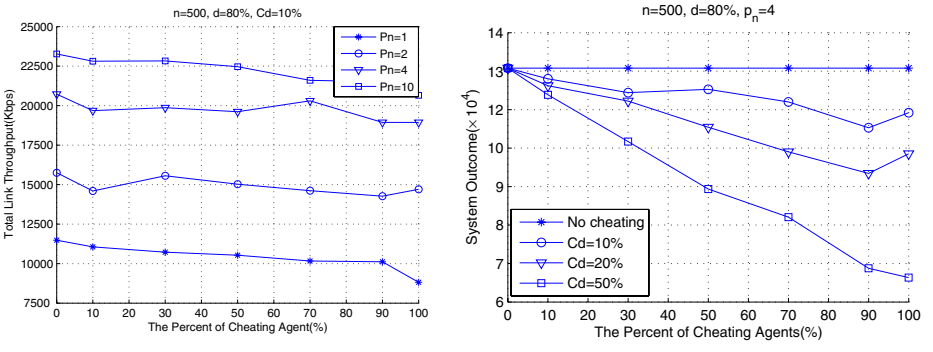


Fig. 2. The changes of system outcome and total throughput

$d = 80\%, Cd = 0\%$ (When the parent number of each agent is 1, the multicast system actually become TALM).

The graphes in the Fig. (1) and (2) show that when in the environment with deferent parent node number, cheating degree, how the system outcome and the total link throughput change. We can observe that all of the maximum values are in the position which the cheating agent percentage is 0%, i.e. no cheating.

We compare our scheme with a random scheme, and compare the situation of no cheating with cheating, separately showing in Fig. (3) and Fig. (4). Obviously, we can observe that in Fig. (3), system outcome of VCG scheme is 4.5×10^4, total throughput is $10802Kbps$, system outcome of random scheme is 2.6×10^4, total throughput is $6752.1Kbps$. System outcome is a 73% improvement and total throughput is a 60% improvement. In Fig. (4), when there exists cheating behavior, system outcome, is 4.5×10^4 if cheating and 2.1×10^4 if no cheating-approximately a 53% loss, total throughput is $10802Kbps$ if cheating and $6004.7Kbps$ if no cheating- approximately a 44% loss.

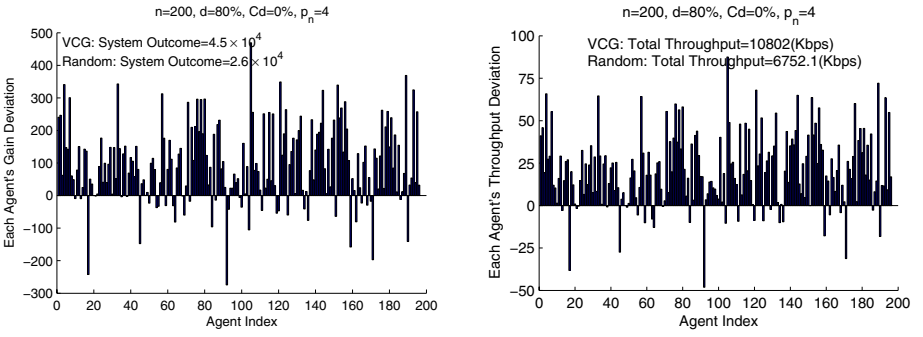


Fig. 3. Compare with random scheme

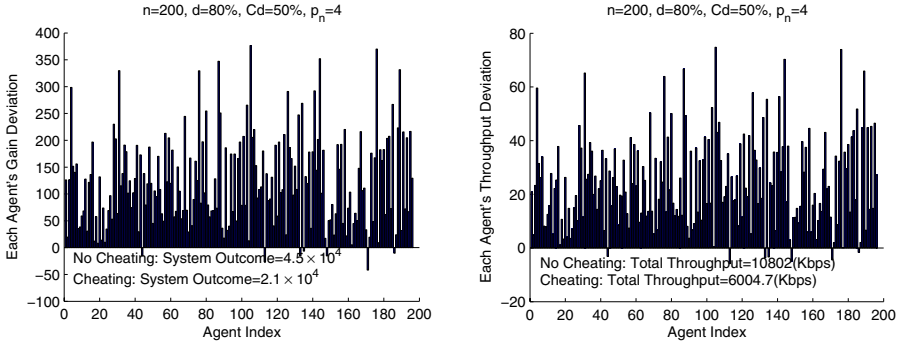


Fig. 4. Compare with cheating

7 Summary and Future Work

The three contributions are mainly finished by this paper. Firstly, for solving the problem of agent cheating behavior in MALM session, we apply the principle of algorithm mechanism design to the MALM network model. Secondly, we design a practical algorithm to realize our principle. Each agent in our algorithm will have no incentive to cheat, so that the real maximum outcome will be achieved. Thirdly, we conduct extensive simulation and analysis to study the correctness of our algorithm, the improvement in the system outcome and total throughput of our algorithm comparing to the random scheme, the effect of cheating behavior.

However, in our work we don't consider the situation of collusion when agents cheat. Therefore, to avoid group cheating and design a group-strategyproof algorithm are our future works. In addition, our future work expect that the parent number of each agent will be varied according to the demand of optimizing the system outcome.

Acknowledgment

This work was supported by grants NSFC-60473082 and NSFC-60303006.

References

1. Xinyan Zhang, Jiangchuan Liu, Bo Li, and Tak-Shing Peter Yum, *"CoolStreaming/DONet: A Data-Driven Overlay Network for Efficient Live Media Streaming"*, Proceedings of INFOCOM 2005.
2. Professor Botond Koszegi, *"Mechanism Design"*, The lecture notes of Economics Theory Course in berkeley, Spring semester 2006.
3. Dan Li, Yong Cui, Jiangchuan Liu, Ke Xu, Jianping Wu.Defending, *"Receiver Cheating in Link-Weighted Application Layer Multicast."* under review
4. Wei Zhou, Ke Xu, Jiangchuan Liu,Chi-Hung Chi, *"Truthful Application-Layer Multicast in Mesh-based Selfish Overlays."* IPCCC WMSN workshop 2006.
5. L. Guo, S. Chen, S. Ren, X. Chen, and S. Jiang, *"PROP: a scalable and reliable P2P assisted proxy streaming system"*, in Proc. ICDCS'04, Tokyo, Japan, Mar. 2004.
6. V. N. Padmanabhan, H. J. Wang, P. A. Chou, and K. Sripanidkulchai, *"Distributing streaming media content using cooperative networking"*, in Proc. NOSSDAV02, USA, May 2002.
7. Yang-hua Chu, John Chuang, Hui Zhang, *"A Case for Taxation in Peer-to-Peer Streaming Broadcast"*, ACM SIGCOMM'04 Workshop on Practice and Theory of Incentives in Networked Systems (PINS), August 2004.
8. A. Habib and J. Chuang, *"Incentive Mechanism for Peer-to-Peer Media Streaming"*, 12th IEEE International Workshop on Quality of Service (IWQoS'04), June 2004.
9. N. Nisan, A. Ronen, *"Algorithmic Mechanism Design"*, Games and Economic Behavior, vol. 35, pp. 166-196, 2001.
10. J. Feigenbaum and S. Shenker, *"Distributed Algorithmic Mechanism Design: Recent Results and Future Directions"*, in Proc. of ACM Dial-M, Atlanta, Georgia, September 2002.
11. Selwyn Yuen, Baochun Li, *"Strategyproof Mechanisms for Dynamic Multicast Tree Formation in Overlay Networks"*, Proceedings of INFOCOM 2005.
12. L. Mathy, N. Blundell, *"Impact of Simple Cheating in Application-Level Multicast"*, IEEE INFOCOM 2004, Hong Kong, China, Mar 2004.
13. Y. D. Chawathe, Scattercast: *"an architecture for Internet broadcast distribution as an infrastructure service"*, PhD thesis, Stanford University, September 2000.
14. P. Francis, Yoid: *"Your Own Internet Distribution"*, http://www.isi.edu/div7/yoid/, March 2001.
15. Noam Nisan, *"Algorithms for selfish agents"*, Lecture Notes in Computer Science, vol. 1563, pp. 1-15, 1999.
16. J. Feigenbaum, C. Papadimitriou, R. Samiy, S. Shenker, *"A BGP-based Mechanism for Lowest-Cost Routing"*, in roceedings of the 2002 ACM Symposium on Principles of Distributed Computing., 2002, pp. 173-182.
17. Y. Chu, S.G. Rao, H. Zhang, *"A case for end system multicast"*, Proc. ACM SIGMETRICS June (2000) 1-12.
18. T. Groves, *"Incentives in Teams"*, Econometrica, Vol. 41, No.4, pp. 617-631(July. 1973).
19. E. Zegura, K. Calvert, S. Bhattacharjee, *"How to Model an Internetwork"*, IEEE INFOCOM 1996, San Francisco, CA, USA, Mar 1996.

Utility-Based Summarization of Home Videos

Ba Tu Truong and Svetha Venkatesh

Department of Computing, Curtin University of Technology,
Perth, Western Australia

Abstract. The aim of this work is to devise an effective method for static summarization of home video sequences. Based on the premise that the user watching a summary is interested in people related (how many, who, emotional state) or activity related aspects, we formulate a novel approach to video summarization that works to specifically expose relevant video frames that make the content spotting tasks possible. Unlike existing approaches, which work on low-level features which often produce the summary not appealing to the viewer due to the semantic gap between low-level features and high-level concepts, our approach is driven by various utility functions (identity count, identity recognition, emotion recognition, activity recognition, sense of space) that use the results of face detection, face clustering, shot clustering and within-cluster frame alignment. The summarization problem is then treated as the problem of extracting the set of keyframes that have the maximum combined utility.

1 Introduction

A video sequence normally contains a large number of frames. In order to ensure that humans do not perceive any discontinuity in the video stream, a frame rate of at least 25fps is required, that is, 7500 images for one hour of video content. This sheer volume of video data is a barrier to many practical applications and therefore there is a strong demand for a mechanism that allows the user to gain certain perspectives of a video document without watching/addressing the video in its entirety. This mechanism is termed *video abstracting*.

There are two types of video abstracts: (a) keyframe or static summarization and (b)video skim or moving-image abstract. The *focus* of our work is static summarization. Many different techniques are proposed in the literature for extracting keyframes, ranging from simple ones such as the uniform sampling of the video sequence or using the first frame of every shot as the keyframe, to more complex methods requiring mathematical modelling [1,2,3]. We refer the reader to the comprehensive survey of the field in [4] for a review of previous work. The review also describes fundamental aspects of current approaches in keyframe extraction as depicted in Figure 1. These aspects are: the size of the keyframe set, the base unit, the representation scope, the underlying computational mechanisms. The way these aspects are addressed differentiates one summarization technique from another.

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 505–516, 2007.
© Springer-Verlag Berlin Heidelberg 2007

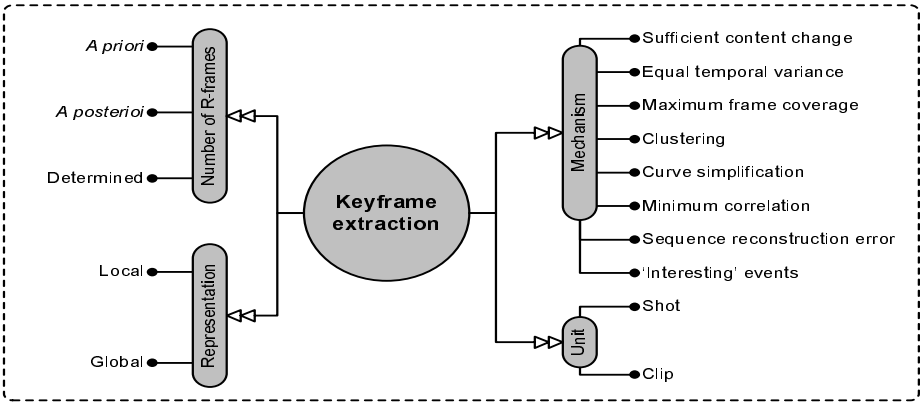


Fig. 1. Features of keyframe extraction methods

The main *problem* with existing static summarization approaches is that they work at the pixel and color levels. They aim mainly at producing a set of keyframes that best represent the visual space of the video sequence. While this is acceptable for shot-level summaries, it fails to address the user's needs in the content-browsing task, especially when the video contains multiple shots. This is because the user looks at the summary from the semantic-content perspective, not the visual perspective, and therefore the summary optimal in visual aspects is not necessarily the one that the user wants. In addition, low-level visual models such as the color histogram do not necessarily reflect the visual space as perceived by humans. For most kinds of videos, especially home videos, the main interest of the user is to gain a knowledge of characters in the video and their activities.

The *purpose* of this work is to devise a new framework for extracting the optimal summary for a video sequence from a user viewpoint. To this end, we define a set of five perspectives important to the user in viewing a summary: How many people (identity count), who they are (identity recognition), their emotional state (emotion recognition), what they are doing (activity recognition), and the sense of space where the video is captured (sense of space). Automatic detection of these semantics is extremely difficult. Instead, we acknowledge that the summary is to be viewed by the user who can detect these semantics effectively, if presented with the right data. Therefore, a video summary should focus on visually "exposing features" of these semantics. We argue that the best summary is essentially the one that best aids the user in these content-spotting tasks. Our idea is best explained with some examples. For the user to know about characters in the video, frames with faces detected should be used, and one face for one person is generally enough. If the emotional state of the character is to be identified, bigger faces should be used. In addition, shots from various angles should improve the ability of the viewer to understand the activity unfolding in the video. From these observations, we formulate utility functions for the above five perspectives. These functions use the results of face detection,

face clustering, shot clustering and within-cluster frame alignment. The video summarization problem is then formulated as the problem of extracting the set of keyframes that have the maximum combined utility.

With respect to Figure 1, our technique addresses both cases where the size of keyframe set is specified as the constraint (*a priori*) or as the outcome (*posteriori*) at the same time. Our technique is clip-based as we aim to produce a concise summary of the entire video sequence, not individual shots. Each selected keyframe has a global representation scope, accounting for the entire video sequence rather than representing a local segment enclosing the keyframe. Our technique is completely novel, differing significantly from those listed in Figure 1. The closest resemblance would be the Maximum Coverage approaches, as they also address the optimality as a set, and each keyframe has a global representation scope. However, they are limited to low-level visual features.

2 Pre-processing

Prior to applying the summarization algorithm, the following processing is carried on video sequences:

Shot segmentation. Shot boundaries are detected by a simple method of applying an adaptive-threshold on the discontinuity curve [5]. For home video sequences in our data set, this simple method is highly reliable, giving only a couple of false detection and missed boundaries.

Shot-based keyframe extraction. We use a simple, efficient method for extracting representative frames of a shot described in [4]. In this method, the first frame is always a keyframe, and the current frame is selected as keyframe, if its visual appearance significantly differs from that of the last keyframe. We also force the last frame of the shot as the keyframe. While this set of shot-based keyframes can be considered a summary of the video sequence, it is very low level and contains too many frames. Our aim is to extract a small-sized subset of these keyframes at a higher-level of abstraction.

Session/scene boundaries identification. A scene or a session in home videos is delimited by temporal and/or spatial discontinuities. It is defined as a collection of consecutive shots, captured at the same place and at the same time. Here, we assume session boundaries are available, as they can be marked by the user when filming using the built-in feature of the camera or by power on/off operations. Alternatively, given the time information associated with each shot, we can easily define an effective classifier to automatically locate session boundaries. Since they can be considered as self-contained story/semantic units, especially so for home videos, we identify session/scene as the appropriate level where summaries can be generated independently of each other. Then, for the rest of paper, a video sequence means a scene/session.

Face Detection. Faces are detected by the CMU Neural Network based technique [6]. Unlike many systems, which are limited to detecting upright, frontal faces, this system detects faces at any degree of rotation in the image plane. This is desirable in our work, since unlike news videos and features in the home

video context, many shots are captured without upright framing and/or contain non-upright faces. Each face returned by the face detector is represented by four variables (x, y, s, σ), which denote the center (i.e. the nose), size and angle of the face to the y-axis. In addition, the CMU face detector returns relatively consistent bounding boxes for detected faces, essential for the accurate modelling of face-based utilities.

Shot Clustering. In our recent work [7], robust shot clusters can be extracted via the use of SIFT features. Each cluster generally corresponds to one view of the action, possibly with different shot distances or camera focal lengths, and they often lie in the same side of a 180-degree axis. From summarization perspective, shots from different clusters should be used since they represent different viewpoints of the event unfolding.

Face Clustering. Using the same technique as shot clustering, in [7] a set of face clusters associated with different individuals can also be extracted robustly. For summarization purpose, we assume that one individual in the video is associated with one and only one cluster.

3 Formulation and Algorithm

In this section, we describe our formulation of the video summarization as a optimization problem and outline some measures for utility functions based on the result of face clustering, shot clustering and frame alignment.

3.1 Problem Formulation

There are two different options for determining the number of keyframes in an automatic keyframe extraction process, and they strongly shape the underlying formulation of the optimal keyframe set. The size of the keyframe set can be fixed as a known *priori*, left as an unknown *posteriori*.

A Priori. The number of keyframes is decided beforehand and given as a constraint to the extraction algorithm. It can be assigned as a specific number or a ratio over the length of the video that may vary according to the user knowledge of the video content. Also called 'rate constraint keyframe extraction', this approach is suitable and often required in mobile device systems where available resources are limited. For these systems, the number of keyframes are distributed differently, depending on the transmission bandwidth, storage capacity or display size of the receiving terminal. A special yet common case is when one keyframe is selected per shot, which is often the first frame, the middle frame or the frame closest to the average content of the shot (also see Section 2.2). The controllability in this manner has a disadvantage in that it does not ensure all important segments in a video contain at least one keyframe. The keyframe extraction problem with *a priori* size, N, can be formulated as the optimization problem of finding the frame set $\mathcal{R} = \{f_{r_1}, f_{r_2}, ..., f_{r_\mathsf{K}}\}$ that is least different from the video sequence with respect to a certain summarization perspective:

$$\mathcal{R} = \arg\min_{\mathcal{R}} \{\mathcal{D}(\mathcal{R}, \mathbf{V}, \rho) \mid 1 \leq r_i \leq \mathsf{N}\}, \tag{1}$$

where N is the number of frames in video $\mathbf{V}$, ρ is the summarization perspective that the user is interested in, and $\mathcal{D}$ is a dissimilarity measure. $\mathbf{V}$ denotes the video sequence. This model is intuitive. For example, if the primary interest of the user is knowing who is appearing in the video sequence then the best summary needs to contain shots of different people in the video and the difference $\mathcal{D}(.)$ is equal to Zero with respect to the user interest.

In the utility-based approach, the difference between the original video sequence and the summary set can be presented as the ratio of utility values. That is:

$$\mathcal{D}(\mathcal{R}, \mathbf{V}, \rho) = 1 - \frac{\mathsf{U}(\mathcal{R}, \rho)}{\mathsf{U}(\mathbf{V}, \rho)},$$

where $\mathsf{U}(\mathcal{R}, \rho)$ denotes the utility of the keyframe set $\mathcal{R}$ with respect to the perspective ρ.

The utility function $\mathsf{U}(\mathcal{R}, \rho)$ needs to satisfy the following characteristics.

- Lies within the (0,1) range.
- $\mathsf{U}(\mathcal{R} \cup \{f^*\}, \rho) \geq \mathsf{U}(\mathcal{R}, \rho), \forall f^* \in \mathbf{V}$. This means that the utility never decreases when more frames are added to the summary set, which is intuitively desirable. This leads to the property that the set of all candidate frames will have the maximum utility, and that $\mathcal{D}(\mathcal{R}, \mathbf{V}, \rho)$ always lie within the (0,1) range.

Since $\mathsf{U}(\mathbf{V}, \rho)$ is constant with respect to variable $\mathcal{R}$, Equation 1 can be reformulated as:

$$\mathcal{R} = \arg\max_{\mathcal{R}} \mathsf{U}(\mathcal{R}, \rho). \tag{2}$$

Most current keyframe extraction techniques have ρ as 'visual coverage', which aims to cover as much visual content with as few frames as possible. However, as demonstrated in this work ρ can also be a combination of various semantic concepts.

A Posteriori. In this approach, one does not know the number of extracted keyframes till the process finishes. For the low-level visual-based approach, the number of keyframes is often determined by the level of visual change itself. The formulation of the keyframe extraction problem with no specified size requires a dissimilarity tolerance ε, also called the *fidelity* level. First, the number of keyframes is determined as:

$$\mathsf{K} = \min_{r_i}\{\mathsf{K} | \min_{\mathcal{R}}\{\mathcal{D}(\mathcal{R}, \mathbf{V}, \rho)\} < \varepsilon, 1 \leq r_i \leq \mathsf{N}\}, \tag{3}$$

which is transtated to:

$$\mathsf{K} = \min\{\mathsf{K} | \max_{\mathcal{R}}\{\frac{\mathsf{U}(\mathcal{R}, \rho)}{\mathsf{U}(\mathbf{V}, \rho)}\} > 1 - \varepsilon\}. \tag{4}$$

Once K is determined, the best summary is determined as in Equation 2. In other words, the problem of extracting the best summary with a fidelity constraint is equivalent to finding the set of keyframes satisfying following constraints:

– the overall utility of the set is close enough to the total utility.
– the size of set is minimum.
– the overall utility of the set is maximum.

For example, if the primary interest of the viewer is to know who is in the video and the fidelity value of 0.3 is set and the labelled video contains 10 people in 9 shots, with one shot containing two people then the optimal summary will contain 6 keyframes of 6 shots which together shows 7 people.

Existing techniques often offer only one option for the size of the keyframe set. However, if the algorithm produces the number of keyframes progressively as demonstrated in our work (see Algorithm 1), two options can be addressed at the same time: the algorithm can stop when the number of keyframes reaches *a priori* value or when certain criteria are satisfied (i.e., *a posteriori*).

For static sequence-based summaries cannot capture the dynamic progression and audio characteristics of the video sequence in all cases, it can be assumed that, for computational efficiency, the set of all shot-based keyframes represent the content of the video in its fullness and hence have the maximum possible utility. Hence $\mathbf{V} = \{f_{i_1}, f_{i_2}, ..., f_{i_\mathsf{N}}\}$, where $\{f_{i_1}, f_{i_2}, ..., f_{i_\mathsf{N}}\}$ is the set of all keyframes extracted in Section 2, and our aim is to extract a subset of these keyframes to represent the content of the video sequence.

3.2 Utility Functions

The main problem is to find appropriate functions for modelling individual utilities corresponding to different aspects of the summarization perspective ρ.

Identity Count U_{ic}. This utility indicates the ability of the summary in providing the user with an estimate of the number of dominant characters in the video. Dominant characters are defined as those whose faces are detected more than once. Given the detected face clusters, which we assume to be correct, this function should increase when the summary contains keyframes with faces appearing in different face clusters. On the other hand, the utility should not increase if a frame with a small face is added and the current summary already possesses a frame from the same face cluster. The appropriate function is therefore:

$$\mathsf{U}_{\mathsf{ic}}(\mathcal{R}) = \mathsf{U}_{\mathsf{ic}}(\mathcal{R}^{\mathsf{F}}) = \frac{1}{\mathsf{L}_{\mathsf{F}}} \sum_{i=1:\mathsf{L}_{\mathsf{F}}} \mathsf{U}_{\mathsf{ic}}(\mathcal{R}_i^{\mathsf{F}}) = \frac{1}{\mathsf{L}_{\mathsf{F}}} \sum_{i=1:\mathsf{L}_{\mathsf{F}}} \max_{\mathsf{F}_j \in \mathcal{R}_i^{\mathsf{F}}} \mathsf{U}_{\mathsf{ic}}(\mathsf{F}_j),$$

where L_{F} is the total number of face clusters formed on all candidate keyframes in the video sequence. $\mathcal{R}^{\mathsf{F}}$ is the set of faces detected in $\mathcal{R}$ and $\mathcal{R}_i^{\mathsf{F}}$ is the subset of $\mathcal{R}^{\mathsf{F}}$ which belong to the i-th face cluster formed on all faces detected in $\mathbf{V}$.

Let us consider how to formulate $\mathsf{U}_{\mathsf{ic}}(\mathsf{F}_j)$. The simplest way is to set $\mathsf{U}_{\mathsf{ic}}(\mathsf{F}_j) = 1$. However, we observe that if the face is too small, it is often a false positive and the viewer cannot spot it easily anyway. Therefore, we use a logistic (sigmoid) function instead, which is of the form:

$$\mathsf{U}_{\mathsf{ic}}(\mathsf{F}_j) = \frac{1}{1 + \exp(-a(x-b))}, \tag{5}$$

where x is the size of face F_j, parameter a controls how fast $\mathsf{U}_{\mathsf{ic}}(.)$ accelerates, and b corresponds to the size of the face where the utility is 1/2. These parameters need to be empirically defined. The sigmoid function lies within the (0,1) range.

Identity Recognition U_{ir}. We model the identity recognition utility similar to the Identity Count.

$$\mathsf{U}_{\mathsf{ir}}(\mathcal{R}) = \mathsf{U}_{\mathsf{ir}}(\mathcal{R}^{\mathsf{F}}) = \frac{1}{\mathsf{L}'_{\mathsf{F}}} \sum_{i=1:\mathsf{L}_{\mathsf{F}}} \mathsf{U}_{\mathsf{ir}}(\mathcal{R}_i^{\mathsf{F}}) = \frac{1}{\mathsf{L}'_{\mathsf{F}}} \sum_{i=1:\mathsf{L}_{\mathsf{F}}} \max_{\mathsf{F}_j \in \mathcal{R}_i^{\mathsf{F}}} \mathsf{U}_{\mathsf{ir}}(\mathsf{F}_j),$$

where L'_{F} is the number of face clusters that have at least one face from keyframes in $\mathcal{R}$. Normalizing by L'_{F} instead of L_{F} means that this utility is only influenced by faces available in the summary set.

In addition, the face attributes (size, location within frame) should influence the ability of viewers to identify different people. Generally, it is easier to spot a person when the face is large and in the centre of the frame. In the current implementation, we ignore the position of face within the frame. We also use the sigmoid function with a higher b value to model the utility function $\mathsf{U}_{\mathsf{ir}}(\mathsf{F}_j)$.

Emotion/Expression Recognition U_{er}. This captures the ability to provide a general perception of the inner emotion of characters appearing in the footage. The only source for emotion perception in static keyframes is facial expression. Currently, we model it the same way as Identity Recognition. The only difference lies in the modelling of individual utilities from the face information. Generally, we require a face of a reasonably large size to be able to recognize the person emotion, and as the face size increases, the ability to uncover what is in a character mind (especially from the eye) increases.

$$\mathsf{U}_{\mathsf{er}}(\mathcal{R}) = \mathsf{U}_{\mathsf{er}}(\mathcal{R}^{\mathsf{F}}) = \frac{1}{\mathsf{L}'_{\mathsf{F}}} \sum_{i=1:\mathsf{L}_{\mathsf{F}}} \mathsf{U}_{\mathsf{er}}(\mathcal{R}_i^{\mathsf{F}}) = \frac{1}{\mathsf{L}'_{\mathsf{F}}} \sum_{i=1:\mathsf{L}_{\mathsf{F}}} \max_{\mathsf{F}_j \in \mathcal{R}_i^{\mathsf{F}}} \mathsf{U}_{\mathsf{er}}(\mathsf{F}_j)$$

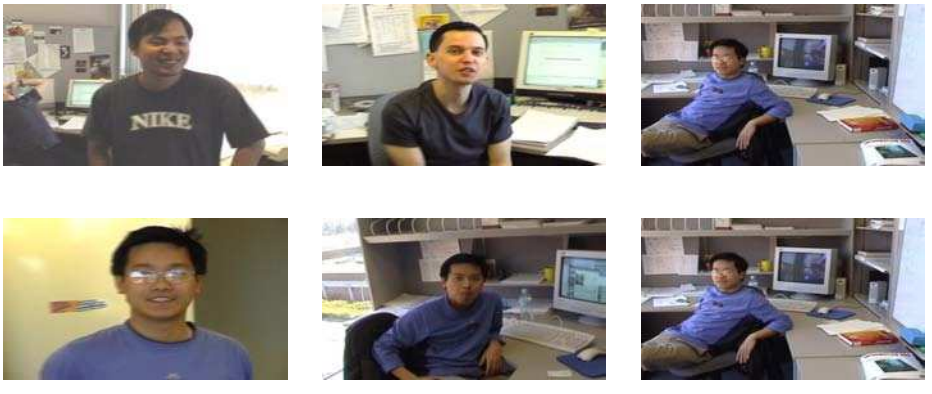


Fig. 2. Keyframes organized in decreasing order of $\mathsf{U}_{\mathsf{ic}}, \mathsf{U}_{\mathsf{ir}}$ and U_{er}

Fig. 2 shows some examples of potential summary frames in the decreasing order of utility value. The second row contains faces from one single face cluster,

i.e, one character. It maps well to the decreasing supports for the viewer to identify the characters as well as their emotional states. With respect to these three utilities alone, the first two frames of each row should be included in the summary set.

Sense of 3D Space $\mathsf{U_{ss}}$. The user is often interested in knowing the place where the video is captured. Obviously, using non-overlapping shots filmed from various angles (i.e., from different shot clusters) improves the space perception. We use a model similar to previous utilities with face clusters being replaced by shot clusters.

$$\mathsf{U_{ss}}(\mathcal{R}) = \frac{1}{\mathsf{L_S}} \sum_{i=1:\mathsf{L_S}} \mathsf{U_{ss}}(\mathcal{R}_i^{\mathsf{S}}) = \frac{1}{\mathsf{L_S}} \sum_{i=1:\mathsf{L_S}} \max_{f_j \in \mathcal{R}_i^{\mathsf{S}}} \mathsf{U_{ss}}(f_j, \mathbf{V}_i),$$

where $\mathsf{L_S}$ is the number of shot clusters detected for the entire set of frames in $\mathbf{V}$ and $\mathbf{V}_i$ represents all frames in the i-th shot cluster. $\mathcal{R}_i^{\mathsf{S}}$ denotes the subset of $\mathcal{R}$ that is also in $\mathbf{V}_i$. The utility function $\mathsf{U_{ss}}(f_j, \mathbf{V}_i)$ means that the $\mathsf{U_{ss}}$ utility for a frame f_j will be computed from the information about all frames and faces contained in its shot cluster, and its value is independent of information outside the cluster.

The important question now is how to compute the utility function $\mathsf{U_{ss}}(f_j, \mathbf{V}_i)$, which indicate how much space is contained in the keyframe. The information about faces and frames in i-th shot cluster is used to estimate the shot size (close-up, medium, etc). First, all frames in $\mathbf{V}_i$ are aligned according to the SIFT feature matches (from the shot clustering algorithm) and the scaling parameters indicate the shot size respectively to the median frame. If faces are detected in $\mathbf{V}_i$, they are used to estimate the shot-size of the median frame. Otherwise, the median frame is assumed to be of medium shot. Finally, another sigmoid function defined over the shot size is used to estimate the utility.

Activity Recognition $\mathsf{U_{ss}}$. The summary should provide the viewer with suitable video frames so that he can spot the activity unfolding in the video. Generally, long distance shots provide more information, since they express the relationship between objects/characters in the scene. Since one shot may not provide enough information for activity recognition and the ability to recognize activity tends to increase if we increase the number of views, we do not normalize the individual utility by the number of shot clusters.

$$\mathsf{U_{ar}}(\mathcal{R}) = \min\{1, \sum_{i=1:\mathsf{L_S}} \mathsf{U_{ar}}(\mathcal{R}_i^{\mathsf{S}})\} = \min\{1, \sum_{i=1:\mathsf{L_S}} \max_{f_j \in \mathcal{R}_i^{\mathsf{S}}} \mathsf{U_{ar}}(f_j, \mathbf{V}_i)\}$$

The min function ensures that the utility will not exceed 1. Currently, $\mathsf{U_{ar}}(.)$ is modelled in the same way as $\mathsf{U_{ss}}$.

Fig. 3, in contrast to Fig. 2, shows potential summary frames in the decreasing order of $\mathsf{U_{ss}}$ and $\mathsf{U_{ar}}$. Each row is associated with one shot cluster. This ordering corresponds well to the decreasing support for viewers to identify the place, its relationship to the character and potentially the activity. With respect to these two utilities alone, the first two frames should always be selected for the summary.

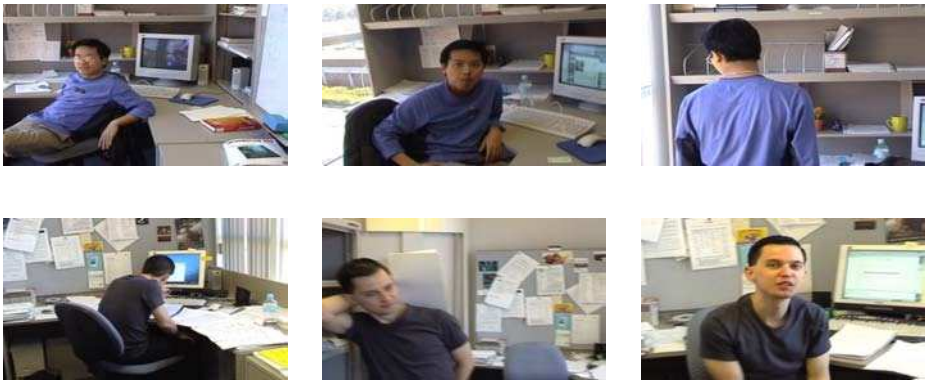


Fig. 3. Keyframes organized in decreasing order of U_{ss} and U_{ar}

3.3 Combining Different Utilities

The next problem is to combine individual utilities that reflect different summarization perspectives into one single utility value. Here the user can specify different weighting values w_ρ indicating which perspective should gain the preference over the other. The most common model is the linear combination. That is:

$$U(\mathcal{R}, \rho) = \frac{1}{W}(w_{ic}U_{ic}(\mathcal{R}) + w_{ir}U_{ir}(\mathcal{R}) + w_{er}U_{er}(\mathcal{R}) + w_{ss}U_{ss}(\mathcal{R}) + w_{ar}U_{ar}(\mathcal{R})),$$

where $W = w_{ic} + w_{ir} + w_{er} + w_{ss} + w_{ar}$.

Alternatively, we can treat the Identity Count utility U_{ic} as the weight for other face-based utilities, and the model becomes:

$$U(\mathcal{R}, \rho) = \frac{1}{W}(U_{ic}(\mathcal{R})(w_{ir}U_{ir}(\mathcal{R}) + w_{er}U_{er}(\mathcal{R})) + w_{ss}U_{ss}(\mathcal{R}) + w_{ar}U_{ar}(\mathcal{R})),$$

where $W = w_{ir} + w_{er} + w_{ss} + w_{ar}$.

This overall utility function satisfy constraints set out in Section 3.1.

3.4 Search Algorithm

From a set of N candidate frames, to find the summary of size K with the maximum utility, we need to examine $\binom{K}{N}$ sets of candidate frames. This is only computational feasible while K and N are both small. If both are large, an approximate method needs to be used. In our current implementation a simple greedy search is employed, as shown in Algorithm 1.

At each iteration, this algorithm basically adds a new frame to the summary set, which maximizes the overall utility of the new set. It stops when the size of the summary or the fidelity level is reached.

Algorithm 1. GreedySearch

1. Initialize $\mathbf{V} = \{f_{i_1}, f_{i_2}, ..., f_{i_N}\}$, $k = 0$ and $\mathcal{R} = \emptyset$.
2. Compute total utility $\mathsf{U}^* = \mathsf{U}(\mathbf{V}, \rho)$.
3. Let $f_i = \arg\max\{\mathsf{U}(\mathcal{R} \cup \{f_i\}, \rho), f_i \in \mathbf{V} - \mathcal{R}\}$.
4. Set $\mathcal{R} = \mathcal{R} \cup \{f_i\}$ and $k = k + 1$.
5. If $k < \mathsf{K}$ and $\frac{\mathsf{U}(\mathcal{R},\rho)}{\mathsf{U}^*} \leq 1 - \varepsilon$ goto Step 2.

3.5 Filtering of Face and Shot Clusters

The examination of face and shot clusters produced by the algorithm in [7] reveals that clusters with single items are better ignored from the summary set, since it either contains false positives in face detection or erratic camera movements. It sometimes contains random shots, which are generally redundant for the summarization purpose.



Fig. 4. Summary Examples with Different Parameter Sets: Office Sequence

4 Experimental Results

Figure 4 shows some preliminary results of applying our technique on video sequence captured at a work place using different parameters sets. There are three people and three dominant "locations" in the video. The size of summary is fixed as 3. In (a), the summary is generated by randomly sampling the candidate keyframe set, which are clearly not as good as other summaries. The summary in (b) focuses on characters (we set $w_{ic} = w_{ir} = w_{er} = 1$, and $w_{ss} = w_{ar} = 0$), and it displays all 3 characters in the video but only 1 dominant location is captured. The activity and sense of space is best presented in (d), where all character-based weights are set to 0 and therefore not all characters are included in the summary. In (c), all weights are set to 1, which is neutral in representing characters, sense of space and activity. Figure 4(e) shows that we need at least 4 keyframes to be within 70% of maximum utility (i.e., $\varepsilon = 0.3$). These four keyframes include all characters and views of the scene.

Figure 5 shows the summaries generated for a video sequence captured at home with different parameter sets. The fixed-size summary is set as 4 while



Fig. 5. Summary Examples with Different Parameter Sets: Family Sequence

the fidelity value is specified as 0.2. There are three characters in the sequence; however there are four face clusters in the clustering results. It shows similar patterns to the previous example. The repetition of a character in second summary is due to the errors in clustering results. Figure 5e shows that in order to be with 80% of total utility, 7 keyframes are required.

5 Conclusions

We have described a new framework for video summarization. Preliminary results have demonstrated the correctness of this technique. However, further improvements are still required to fully realize its merits. We plan to carry out extensive user studies using a large collection of home videos to effectively evaluate the quality of generated summaries. In addition, utility functions can be extended and improved by incorporating more variables such as the shot duration (shots with longer duration should be given priority over shorter shots), shot dominance (shots in larger clusters should have priority over shots in smaller clusters).

References

1. Chang, H.S., Sull, S., Lee, S.U.: Efficient video indexing scheme for content-based retrieval. IEEE Transactions on Circuits and Systems for Video Technology **9** (1999) 1269–1279
2. Lee, H.C., Kim, S.D.: Iterative key frame selection in the rate-constraint environment. Signal Processing: Image Communication (2003) 1–15
3. Porter, S.V., Mirmehdi, M., Thomas, B.T.: A shortest path representation for video summarisation. In: 12th International Conference on Image Analysis and Processing. (2003) 460–465
4. Truong, B.T., Venkatesh, S.: Video abstraction: A systematic review and classification. Accepted for ACM Transactions on Multimedia Computing, Communications and Applications (ACMTOMCCAP) (2006)
5. Truong, B.T., Venkatesh, S.: Finding the optimal segmentation of video sequences. In: ICME05.(2005)
6. Rowley, H., Baluja, S., Kanade, T.: Rotation invariant neural network-based face detection. In: CVPR'98. (1998)
7. Truong, B.T., Venkatesh, S.: Linking identities and view points in home videos using robust feature matching. In: International Multimedia Modeling Conference (MMM07), Singapore(2007)

Performance Analysis of Multiple Classifier Fusion for Semantic Video Content Indexing and Retrieval

Rachid Benmokhtar and Benoit Huet

Département Communications Multimédias
Institut Eurécom
2229, route des crêtes
06904 Sophia-Antipolis - France
(Rachid.Benmokhtar, Benoit.Huet)@eurecom.fr

Abstract. In this paper we compare a number of classifier fusion approaches within a complete and efficient framework for video shot indexing and retrieval[1]. The aim of the fusion stage of our sytem is to detect the semantic content of video shots based on classifiers output obtained from low level features. An overview of current research in classifier fusion is provided along with a comparative study of four combination methods. A novel training technique called Weighted Ten Folding based on Ten Folding principle is proposed for combining classifier. The experimental results conducted in the framework of the TrecVid'05 features extraction task report the efficiency of different combination methods and show the improvement provided by our proposed scheme.

1 Introduction

Multimedia digital documents are readily available, either through the Internet, private archives or digital video broadcast. Tools are required to efficiently index this huge amount of information and to allow effective retrieval operations. Unfortunately, most existing systems rely on the automatic description of the visual content through color, texture and shape features whereas users are more interested in the semantic multimedia content. In practice an important gap remains between the visual descriptors and the semantic content. New tools for automatic semantic video content indexing are highly awaited and an important effort is now conducted by the research community to automatically bridge the existing gap [1,2].

The retrieval of complex semantic concepts requires the analysis of many features per modalities. The task consisting of combining all these different parameters is far from trivial. The fusion mechanism can take place at different levels of the classification process. Generally, it is either applied on signatures (feature fusion) or on classifier outputs (classifier fusion). Unfortunately, complex signatures obtained from fusion of features are difficult to analyze and it results in classifiers that are not well trained despite of the recent advances in machine learning. Therefore, the fusion of classifier outputs remains an important step of the classification task.

[1] This work is funded by France Télécom R&D under CRE 46134752.

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 517–526, 2007.
© Springer-Verlag Berlin Heidelberg 2007

This paper starts with an overview of our semantic video content indexing and retrieval system. It is followed by a brief description of state of the art combination methods and classifiers, including Gaussian Mixture Model, Neural Network and Decision Template. In an effort to evaluate their classification and fusion ability, the previously mentioned approaches have been implemented within our system along with a number of training schemes. Among the training scheme evaluated here, we propose an alternative to the Ten Folding approach; the Weighted Ten Folding. This study reports the efficiency of different combination methods and shows the improvement provided by our proposed scheme on the TrecVid'05 dataset. Finally, we conclude with a summary of the most important results provided by this study.

2 System Architecture

This section describes the workflow of the semantic feature extraction process that aims to detect the presence of semantic classes in video shots, such as building, car, U.S. flag, water, map, etc ...

First, key-frames of video shots, provided by TrecVid'05, are segmented into homogeneous regions thanks to the algorithm described in [3]. Secondly, color and texture are extracted for each region obtained from the segmentation. Thirdly, the obtained vectors over the complete database are clustered to find the N most representative elements. The clustering algorithm used in our experiments is the well-known k-means. Representative elements are then used as visual keywords to describe video shot content. To do so, computed features on a single video shot are matched to their closest visual keyword with respect to the Euclidean distance.

Then, the occurrence vector of the visual keywords in the shot is build and this vector is called the Image Vector Space Model (IVSM) signature of the shot. Image latent semantic analysis (ILSA) is applied on these features to obtain an efficient and compact representation of video shot content. Finally, support vector machines (SVM) are used to obtain the first level classification which output will then be used by the fusion mechanism [4]. The overall chain is presented in figure 1.

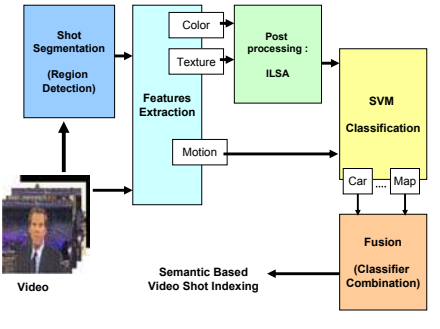


Fig. 1. General framework of the application

2.1 Visual Features Extraction

For the study presented in this paper we distinguish two types of visual modalities: HSV Histogram and Gabor filters features.Two visual features are selected for this purpose: Hue-Saturation-Value color histograms and energies of Gabor's filters [5]. In order to capture the local information in a way that reflects the human perception of the content, visual features are extracted on regions of segmented key-frames [6]. For the sake of computation complexity and storage requirements, region features are quantized and key-frames are represented by a count vector of quantization vectors.

2.2 ILSA

In [7], Latent Semantic Analysis was efficiently adapted from text document indexing to image content. The singular value decomposition of the occurrence matrix of visual keywords in some training shots provides a new representation of video shot content where latent relationships can be emphasized.

2.3 Classification

Classification consists in assigning classes to video shots given some description of its content. The visual content is extremely rich in semantic classes, but limited data is available to build classification models. Classification is therefore conducted on individual features in order to have enough training data with respect to input vector sizes. Allwein and al [8] showed that it was possible to transform a multi-classes classification problem into several binary classification problems. They propose a *one-against-all method*, which consists in building a system of binary classification by class. In our work, this method is adopted using the SVM classification.

Support Vector Machines are one of the most popular machine learning techniques, since they have shown very good generalization performance on many pattern classification problems. They have the property to allow a non linear separation of classes with very good generalization capacities. The main idea is similar to the concept of a neuron: separate classes with a hyperplane. However, samples are indirectly mapped into a high dimensional space thanks to a kernel function that respects the Mercer's condition [9]. This leads the classification in a new space where samples are assumed to be linearly separable. The selected kernel denoted $\mathcal{K}(.)$ is a radial basis function for which normalization parameter σ is chosen depending on the performance obtained on a validation set. The radial basis kernel is chosen for its good classification results comparing to Polynomial and Sigmoidal kernels [4].

3 Classifier Fusion

Combining classifier is an active research field [10,11]. There are generally two types of classifier combination: classifier selection and classifier fusion [10]. The classifier

selection considers that each classifier is an expert in some local area of the feature space. The final decision is taken only by one classifier, as in [12], or more than one "local expert", as in [13]. Classifier *fusion* [14] assumes that all classifiers are trained over the whole feature space, and are considered as competitive as well as complementary. Duin and Tax [11] have distinguished the combination methods of different classifiers and the combination methods of weak classifiers.

The objective of the following section is to present an overview of classifier fusion methods and attempt to identify new trends that can be used in this area of research.

3.1 Non Trainable Combiners

Here, we detail the combiners that are ready to operate as soon as the classifiers are trained, i.e., they do not require any further training. The only methods to be applied to combine these results without learning are based on the principle of vote. They are commonly used in the context of handwritten text recognition [15]. All vote based methods can be derived from the majority rule E with threshold expressed by:

$$E = \begin{cases} C_i \text{ if } \max\left(\sum_i^K e_i\right) \geq \alpha K \\ \text{Rejection else} \end{cases} \tag{1}$$

where C_i is the i^{th} class, K is the number of classifiers to be combined and $e_i \in [0, 1]$ is the classifier output.

For $\alpha = 1$, the final class is assigned to the class label most represented among the classifier outputs else the final decision is rejected, this method is called **Majority Voting**. For $\alpha = 0.5$, it means that the final class is decided if more half of the classifiers proposed it, we are in **Absolute Majority**. For $\alpha = 0$, it is a **Simple Majority**, where the final decision is the class of the most proposed among K classifiers. In **Weighted Majority Voting**, the answer of every classifiers is weighted by a coefficient indicating their importance in the combination [16].

Soft label type classifiers combine measures which represent the confidence degree on the membership. In that case, the decision rule is given by the **Linear Methods** which consist in a linear combination of classifier outputs [17]:

$$E = \sum_{k=1}^{K} \beta_k m_i^k \tag{2}$$

where β_k is the coefficient which determines the attributed importance to k^{th} classifier in the combination and m_i^k is the answer for the class i.

3.2 Trainable Combiners

Contrary to the vote methods, many methods use a learning step to combine results. The training set can be used to adapt the combining classifiers to the classification problem. Now, we present four of the most effective methods of combination.

Neural Network (NN): Multilayer perceptron (MLP) networks trained by back propagation are among the most popular and versatile forms of neural network classifiers. In the work presented here, a multilayer perceptron networks with a single hidden layer and sigmoid activation function [18] is employed. The number of neurons contained in the hidden layer is calculated by heuristic.

Gaussian Mixture Models (GMM): The question with Gaussian Mixture Models is how to estimate the model parameter M. For a mixture of N components and a D dimensional random variable. In literature there exists two principal approaches for estimating the parameters: *Maximum Likelihood Estimation* and *Bayesian Estimation*. While there are strong theoretical and methodological arguments supporting Bayesian estimation, in this study the maximum likelihood estimation is selected for practical reasons. For each class, we trained a GMM with N components, using Expectation-Maximization (EM) algorithm.The number of components N corresponds to the model that best matches the training data. During the test, the class corresponding to the GMM that best fit the test data (according to the maximum likelihood criterion) is selected.

Decision Template (DT): The concepts of decision templates as a trainable aggregation rule was introduced by [10]. Decision Template DT_k for each class $k \in \Omega$ (where Ω is the number of classes) can be calculated by the average of the local classifier outputs $P_m^n(x)$.

$$DT_k(m,n) = \frac{\sum_{x \in T_k} P_m^n(x)}{Card(T_k)} \tag{3}$$

where T_k is a validation set different from the classifier training set. Decision Template is a matrix of size $[S, K]$ with S classifiers and K classes. To make the information fusion by arranging of K Decision Profiles (DP), it remains to determine which Decision Template is the most similar to the profile of the individual classification. Finally, the decision is taken by the maximum of the similarity difference.

Genetic Algorithm (GA): Genetic algorithms have been widely applied in many fields involving optimization problems. It is built on the principles of evolution via natural selection: an initial population (chromosomes encoding possible solutions) is created and by iterative application of genetic operators (selection, crossover, mutation) an optimal solution is reached, according to the defined fitness function [7].

3.3 Alternative Training Approaches

In the case of large sets of simple classifiers, the training is performed modified versions of the original dataset. Three heavily studied training alternatives are Adaboost (also known as boosting), Bagging (Bootstrapping), Random Subspaces and Ten Folding. In addition to the known methods, we propose an alternative to Ten Folding, which we call Weighted Ten Folding and is detailed at the end of this section.

Adaboost: The intuitive idea behind Adaboost is to train a series of classifiers and to iteratively focus on the hard training examples. The algorithm relies on continuously changing the weights of its training examples so that those that are frequently misclassified get higher and higher weights: this way, new classifiers that are added to the set are more likely to classify those hard examples correctly. In the end, Adaboost predicts one of the classes based on the sign of a linear combination of the weak classifiers trained at each step. The algorithm generates the coefficients that need to be used in this linear combination. The iteration number can be increased if we have time and with the overfitting risk [19].

Bagging: Bagging builds upon bootstrapping and adds the idea of aggregating concepts [20]. Bootstrapping is based on random sampling with replacement. Consequently, a classifier constructed on such a training set may have a better performance. Aggregating actually means combining classifiers. Often a combined classifier gives better results than individual base classifiers in the set, combining the advantages of the individual classifiers in the final classifier.

Ten Folding (TF): In front of the limitation (number of samples) of TrecVid'05 test set, *N-Fold Cross Validation* can be used to solve this problem. The principle of Ten Folding is to divide the data in $N = 10$ sets, where $N - 1$ sets are used for training data and the remaining to test data. Then, the next single set is chosen for test data and the remaining sets as training data, this selection process is repeated until all possible combination have been computed as shown in figure 2. The final decision is given by averaging the output of each model.

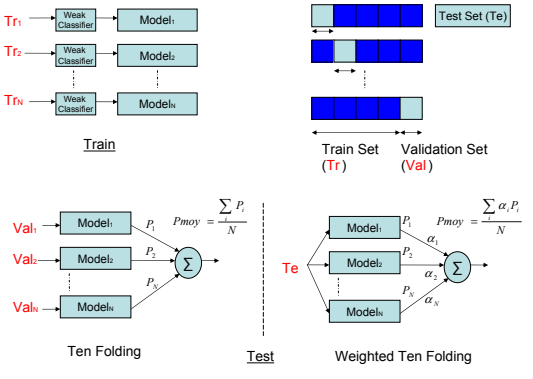


Fig. 2. The standard Ten Folding and Weighted Ten Folding combination classifier

Weighted Ten Folding (WTF): With TrecVid'05 test set limitation in mind, the well-known Bagging instability [20] (i.e. a small change in the training data produces a big change in the behavior of classifier) and the overfitting risk for Adaboost (i.e. when the iteration number is big [19]), we propose a new training method based on Ten Folding

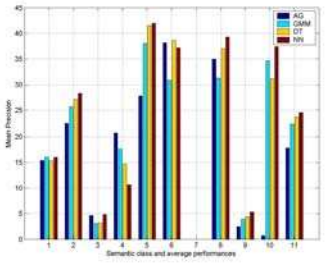


Fig. 3. Comparison of Genetic Algorithm, Decision Template method, GMM fusion method and Neural Network fusion method

that we call *Weighted Ten Folding*. We use the Ten Folding principle to train and obtain N models weighted by a coefficient indicating the importance in the combination. The weight α_i of each model is computed using the single set to obtain the training error ϵ_i. In this way, we obtain models with weak weight if the training error ϵ_i is high and models with high weight when ϵ_i is low.

$$\begin{cases} \epsilon_i = \sum_{j=1}^{N} (y(x_j) - f(x_j))^2 \\ \alpha_i = \frac{1}{2} \log(\frac{1-\epsilon_i}{\epsilon_i}) \end{cases} \tag{4}$$

The final decision combines measures which represent the confidence degree of each model. The weighted average decision in WTF improves the precision of Ten Folding by giving more importance for models with weak training error, contrary to the Ten Folding who takes the output average of each model with the same weight.

4 Experiments

Experiments are conducted on the TrecVid'05 databases [2]. It represents a total of over 85 hours of broadcast news videos from US, Chinese, and Arabic sources. About 60 hours are used to train the feature extraction system and the remaining for the evaluation purpose. The training set is divided into two subsets in order to train classifiers and subsequently the fusion parameters. The evaluation is realized in the context of TrecVid'05 and we use the common evaluation measure from the information retrieval community: the Average Precision.

The feature extraction task consists in retrieving shots expressing one of the following semantic concepts: *1:Building, 2:Car, 3:Explosion or Fire, 4:US flag, 5:Map, 6:Mountain, 7:Prisoner, 8:Sports, 9:People walking/running, 10:Waterscape*.

Figure 3 shows Mean Precision results for the trainable combiners. Of the four fusion scheme compared in this work, the Genetic Algorithm performs worst. This is clearly visible on the semantic concept (5, 10 and 11: Mean Average Precision), where the GA approach suffered from overfitting. The Decision Template and the Gaussian Mixture Model provide only marginally weaker performance than the Neural Network which performed best.

In the next experiment, Adaboost and Bagging principles are employed to increase the performances of GMM and Neural Network methods, considering them as weak classifier. As seen in figure 4, on average for all semantic concept the *Weighted Ten Folding* approach outperforms in turn boosting, bagging and Ten Folding technique in spite of the lack of datum. Significant improvement have been noticed for the following semantic concepts (4, 5, 6, 8 and 11:Mean Average Precision). This can be explained by the weight computation, which is computed on a validation set independently to training set. This allows to have more representative weights in the test for the whole classifier. So, we have best level-handedness of whole classifier contrary to boosting, where the weights computation is made by the training set.

Figure 5 consists in group of plots that represent the evolution of precision and recall values for 3 semantic concepts (Building, Car, Sports), using GMM and NN methods. We observe that the NN-based system has higher precision values for the "Car" and "Sports" concepts. These concepts present a rich motion information compared with "Building" which have no motion. Similar poor results are obtained using "Map" and "Mountain" concepts. Therefore, the choice and the selection of features is very important and must be made by taking into account the behavior semantic concepts. In the same way, use audio features for "Building, Map, US flag and Mountain" concepts will give no positive improvement, but it will be more beneficial for "Explosion" and "Sports" concept for example. A careful selection of the features is therefore necessary to improve our system such that it becomes more selective and less tolerant to changes. This question of features selection will be the object of our future works.

The table 1 presents the TrecVid'05 results submissions for [21], [22], [23] and our system. For this comparison task, we compute the Mean Average Precision (MAP) on the first 1000 retrieved shots as a measure of retrieval effectiveness. Our system presents very promising results, using SVMs classification and Weighted Ten Folding for NN Fusion. Models are trained per raw features and per concept. Looking at those results in some details, shows that the proposed system outperforms the top three systems for 6 of the 10 semantic concepts featured in TrecVid'05. Overall, the mean average precision is the best but only by a small (3%) improvement. We can explain this results by the system scheme classification, when we built a system of binary classification by class for each feature, it protects the correlation between the features. After, we fuse here response using neural network.

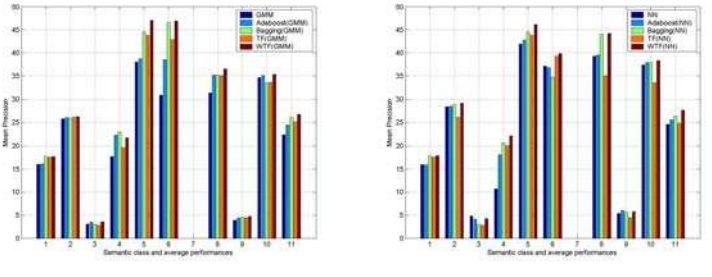


Fig. 4. Comparison of performance using Adaboost, Bagging, Ten Folding and Weighted Ten Folding for GMM and NN

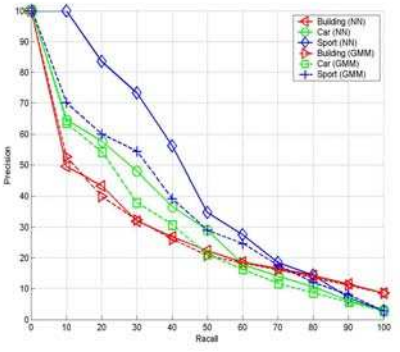


Fig. 5. Mean precision vs recall curves for three different objects (building, car, sports) using NN and GMM methods

Table 1. Mean Average Precision scores for TrecVid'05

Concepts	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	MAP
System A	39%	23.7%	2.8%	7.1%	15.1 %	18.4%	0%	31.2%	15.4 %	23.9%	17.66 %
System B	45%	27.9%	10.7%	24.6%	37.4%	37.8 %	2%	44.6%	27.5%	41.1%	29.86%
System C	47.6%	36.%	9.7%	18.7%	52.4%	45.4%	3%	40.1%	31.9%	47.6%	33.29%
Our System	**45.61%**	**48.49%**	**5.23%**	**38.49%**	**58.19%**	**50.43%**	**0%**	**38.08%**	**17.67%**	**58.89%**	**36.10%**

5 Conclusion

In this paper, we have presented an automatic semantic video content indexing and retrieval system where four different methods for combining classifiers are investigated in details. The Neural network based fusion approach managed all the features most effectively and appears therefore to be particularly well suited for the task of classifier fusion. Our newly proposed training scheme for combining weak classifiers, Weighted Ten Folding, achieved the best retrieval results. Adaboost and Bagging as they were originally proposed did not show a significant improvement, despite their special base model requirements for dynamic loss and prohibitive time complexity. It is due to the TrecVid'05 test set limitation and overfitting risk as the number of iteration increases. The later is solved by our proposed WTF which explains the performance improvement.

References

1. M. Naphade, T. Kristjansson, B. Frey, and T. Huang, "Probabilistic multimedia objects (multijets): a novel approach to video indexing and retrieval," *IEEE Trans. Image Process.*, vol. 3, pp. 536–540, 1998.
2. TRECVID, "Digital video retrieval at NIST," *http://www-nlpir.nist.gov/projects/trecvid/*.

3. P. Felzenszwalb and D. Huttenlocher, "Efficiently computing a good segmentation," *Proceedings of IEEE CVPR*, pp. 98–104, 1998.
4. F. Souvannavong, "Indexation et recherche de plans video par contenu semantique," Ph.D. dissertation, Phd thesis of Eurecom Institute, France, 2005.
5. W. Ma and H. Zhang, "Benchmarking of image features for content-based image retrieval," *Thirtysecond Asilomar Conference on Signals, System and Computers*, pp. 253–257, 1998.
6. C. Carson, M. Thomas, and S. Belongie, "Blobworld: A system for region-based image indexing and retrieval," *Third international conference on visual information systems*, 1999.
7. D. Souvannavong, B. Merialdo, and B. Huet, "Multi modal classifier fusion for video shot content retrieval," *Proceedings of WIAMIS*, 2005.
8. E. Allwein, R. Schapire, and Y. Singer, "Reducing multiclass to binary : A unifying approach for margin classifiers." *Journal of Machine Learning Research*, vol. 1, pp. 113–141, 2000.
9. N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines*. Cambridge University Press, 2000, ch. Kernel-Induced Feature Spaces.
10. L. Kuncheva, J.C.Bezdek, and R. Duin, "Decision templates for multiple classifier fusion : an experiemental comparaison," *Pattern Recognition*, vol. 34, pp. 299–314, 2001.
11. R. Duin and D. Tax, "Experiements with classifier combining rules," *Proc. First Int. Workshop MCS 2000*, vol. 1857, pp. 16–29, 2000.
12. L. Rastrigin and R. Erenstein, "Method of collective recognition," *Energoizdat*, 1982.
13. R. Jacobs, M. Jordan, S. Nowlan, and G. Hinton, "Adaptive mixtures of local experts," *Neural Computation*, vol. 3, pp. 1409–1431, 1991.
14. L. Xu, A. Krzyzak, and C. Suen, "Methods of combining multiple classifiers and their application to hardwriting recognition," *IEEE Trans. Sys. Man. Cyb.*, vol. 22, pp. 418–435, 1992.
15. K. Chou, L. Tu, and I. Shyu, "Perfmrmances analysis of a multiple classifiers system for recognition of totally unconstrained handwritten numerals," *4th International Workshop on Frontiers of Handwritten Recognition*, pp. 480–487, 1994.
16. B. Achermann and H. Bunke, "Combination of classifiers on the decision level for face recognition," *technical repport of Bern University*, 1996.
17. T. Ho, "A theory of multiple classifier systems and its application to visual and word recognition," Ph.D. dissertation, Phd thesis of New-York University, 1992.
18. G. Cybenko, "Approximations by superposition of a sigmoidal function," *Mathematics of Control, Signal and Systems*, vol. 2, pp. 303–314, 1989.
19. Y. Freud and R. Schapire, "Experiments with a new boosting algorithms," *Machine Learning: Proceedings of the 13th International Conference*, 1996.
20. M. Skurichina and R. Duin, "Bagging for linear classifiers," *Pattern Recognition*, vol. 31, no. 7, pp. 909–930, 1998.
21. M. Cooper, J. Adcock, R. Chen, and H. Zhou, "Fxpal at trecvid 2005," in *Proceedings of Trecvid*, 2005.
22. S.-F. Chang, W. Hsu, L. Kennedy, L. Xie, A. Yanagawa, E. Zavesky, and D. Zhang, "Video seach and high level feature extraction," *Proceedings of Trecvid*, 2005.
23. A. Amir, J. Argillander, M. Campbell, A. Haubold, G. Iyengar, S. Ebadollahi, F. Kang, M. Naphade, A. Natsev, J. Smith, J. Tesic, and T. Volkmer, "Ibm research trecvid 2005 video retrieval system," in *Proceedings of Trecvid*, 2005.

Video Semantic Concept Detection Using Multi-modality Subspace Correlation Propagation

Yanan Liu and Fei Wu

liu.yanan@yahoo.com, wufei@cs.zju.edu.cn

Abstract. Interaction and integration of multi-modality media types such as visual, audio and textual data in video are the essence of video content analysis. Although any uni-modality type partially expresses limited semantics less or more, video semantics are fully manifested only by interaction and integration of any unimodal. A great deal of research has been focused on utilizing multi-modality features for better understanding of video semantics. In this paper, we propose a new approach to detect semantic concept in video using SimFusion and Locality Preserving Projections (LPP) from temporal-sequenced associated cooccuring multimodal media data in video. SimFusion is an effective algorithm to reinforce or propagate the similarity relations between multi-modalities. LPP is an optimal combination of linear and nonlinear dimensionality reduction method. Our experiments show that by employing the two key techniques, we can improve the performance of video semantic concept detection.

Keywords: multi-modality semantic concept detection, SimFusion, LPP, temporal-sequenced associated cooccurrence.

1 Introduction and Related Work

Research in content-based multimedia retrieval is motivated by a growing amount of digital multimedia content in which video data has a big part. Video data comprises plentiful semantics, such as people, scene, object, event and story, etc. Much research effort has been made to negotiate the "semantic gap" between low-level features and high-level concepts. In general, three modalities exist in video, namely the image, audio, and text modalities. How to utilize multi-modality features of video data effectively to better understand the multimedia content remains a great challenge.

A multimodal analysis method for semantic understanding of video includes a fusion step to combine the results of several single media analysis. The two main strategies of fusion are early fusion and late fusion[1]. And most existing methods for video concept detection are based on these two schemes.

As described in [1], early fusion is a scheme that integrates unimodal features before learning concepts, whereas late fusion is a scheme that first reduces unimodal features to obtain separately learned concept scores, then these scores are integrated to learn concepts.

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 527 – 534, 2007.
© Springer-Verlag Berlin Heidelberg 2007

When taking early fusion scheme, unimodal features first extracted. After analysis of the various unimodal streams, the extracted features are combined into a single representation, where simply uses concatenation of unimodal feature vectors to obtain a fused multimedia representation. Early fusion yields a truly multimedia feature representation, but it is still a great difficulty to combine features into a common representation properly and effectively.

In contrast to early fusion, approaches for late fusion learn semantic concepts directly from unimodal features, then combine learned unimodal scores into a multimodal representation. Though late fusion focuses on the individual strength of modalities, the expensiveness in terms of the learning effort of separate supervised learning stage for every modal and an additional learning stage for combination is a big disadvantage.

However, the multimodal media types such as image, audio and text in video are in essence of temporal-sequenced associated occurrence. For instance, during a period of time, although the multi-modality data of continuous video frames, transcripts and audio signal may not occur at once, i.e. asynchronously, they convey the uniform semantics. That is, the multi-modality features extracted from video data present a temporal-sequenced associated cooccurence (TSAC) characteristic, which neither traditional early fusion nor late fusion strategy takes into account.

Several major aspects claim attention when considering TSAC characteristic of video. First, how to propagate similarity correlations between distinct modalities. That is, for some semantics, suppose that a video object presents more similar in one modality, then we need find a way to "re-inforce" the similarities in other modalities based on the given "stronger" similarity. And it is worth notice that the relationships in uni-modality and between multi-modalities are complementary. And the intra-modality similarity can reinforce the inter-modality relationship. Thereby how to effectively propagate corresponding correlations between multi-modalitis is a noticable problem. Secondly, "the curse of dimensionality" has been a well-known problem caused by high dimensionality, which video features inevitably face especially when multi-modalities fuse together. So it is important to find a better dimensionality reduction method. Furthermore, statistical learning methods will be a powerful tool for constructing models.

[2] presents a unified similarity-calculation algorithm SimFusion. This approach uses a Unified Relationship Matrix (*URM*) to represent a set of heterogeneous data objects and their interrelationships. By iteratively computing over the *URM*, SimFusion can effectively integrate relationships from heterogeneous sources when measuring the similarity of two data objects. A Unified Similarity Matrix (*USM*) is defined in this process to represent the similarity values of any data object pairs from same or different data spaces. Thus through SimFusion, we can achieve better results of multi-modality subspace correlation propagation.

As we know, the curse of the dimensionality [6] refers to the fact that in the absence of simplifying assumptions, the sample size needed to estimate a function of several variables to a given degree of accuracy (i.e. to get a reasonably low-variance estimate) grows exponentially with the number of variables.

The problem of dimensionality reduction is introduced as a way to overcome the "curse of the dimensionality" when dealing with vector data in high-dimensional spaces and as a modeling tool for such data [7]. It is defined as the search for a low-dimensional manifold that embeds the high-dimensional data.

Now several techniques for dimensionality reduction have been proposed, usually divide into two parts – linear and nonlinear methods. Linear methods reduce dimension through the use of linear combinations of variable, and nonlinear methods do so with nonlinear functions of variable. The linear combinations can be considered as linear projection, and guaranteed to discover the true structure of data lying on or near a linear subspace of the high-dimensional input space. Principle component analysis (PCA)[8] and projection pursuit[9] are typical methods of this type. Although linear methods are simple to implement, explainable, efficient computable and more extensible, many data sets contain essential nonlinear structure that are invisible to PCA and other linear ways, e.g. the classical "Swiss roll" data set, which intrinsically distribute in a nonlinear manifold. As the research for manifold learning, several traditional non-linear methods have been proposed, such as locally linear embedding (LLE)[10], Isomap[11], and Laplacian eigenmap[12]. All of these algorithms are able to discover the intrinsic nonlinear structure, but they are not able to extend to out-of-sample data directly. That is, they are defined only on the training dataset and it is difficult to evaluate the map for new sample. But then, locality preserving projections (LPP) is a combination of linear and nonlinear aspects.

LPP builds a graph incorporating neighborhood information of the data set. Then using the notion of the Laplacian of the graph, a transformation matrix that maps the data points to a subspace is computed. This linear transformation optimally preserves local neighborhood information in a certain sense. The representation map generated by the algorithm may be viewed as a linear discrete approximation to a continuous map that naturally arises from the geometry of the manifold. In deed, LPP may be simply applied to any new data point to locate it in the reduced representation space.

In this paper, we propose a new approach for semantic concept detection in video. Obviously, multi-modality fusion is adopted instead of uni-modality method. For text features, we use Latent Semantic Analysis (LSA) [3] to discover the intrinsic structure of document space. Considering the important temporal-sequenced associated cooccurence characteristic of video, we use SimFusion to propagate correlation from one modality to another, and for much more precise correlations between different modalities. Locality Preserving Projection (LPP) [4] is a noval linear dimensionality reduction algorithm that also shares many of the data representation properties of nonlinear techniques. That is to say, LPP may be simply applied to any new data point to map it in the manifold subapce rather than only defined on the training data set. So we adopt LPP to reduce the high dimension of fused multi-modalities. And at last Support Vector Machine (SVM) is used to detect video semantics.

The organization of this paper is as follows: In Section 2, the proposed method for semantic concept detection in video is presented. Section 3 reports our experiments with TRECVID 2005 news video data. Finally, Section 4 summarizes the results with conclusions.

2 Video Semantic Detection Through Multi-modality Correlation Propagation

In our approach, a single shot is taken as a basic unit of video semantic detection. We perceive of semantic concept detection in video as a pattern recognition problem. Given pattern *x*, part of shot *i*, the aim is to obtain a measure, which indicates whether semantic concept *w* is present in shot *i*.

2.1 Low-Level Feature Extraction

Low-level features are extracted for each shot. Low-level means the features directly extracted from the source – videos, which distinguish from the high-level semantic concept of video. And the motivation of this paper is to use the labeled training video to classify unkown video into different semantic classes. As video carries multi-modality information including visual, audio, and textual data, the low-level features also compose of three parts.

Image features. A shot is the basic unit; therefore, one key frame within each shot is obtained as a representative image for that shot. Image features are then based on the features extracted from the representative image. There are three different types of image features: color histograms, textures and edges.

Audio features. For each shot, we extract the according audio signal as a "audio clip", and divide the audio clip into overlapped "short-time audio frame". Then a frame feature vector is formed based on the audio features extracted from each audio frame. Because of the variable lengths of shots, we calculate the statistic (mean or variance) of audio frame feature vectors for each shot.

Text features. The source text is the ASR transcript. The dimension of text features is much larger than the other modality features, and text contains abundance of semantic information, therefore we use Latent Semantic Analysis (LSA) to reduce the text dimension and discover the semantic structure. This pre-processing step also reduces the dimension of text features effectively first.

2.2 Multi-modal Subspace Correlation Propagation

As mentioned before, shot is the basic processing unit, so our final result we want is the semantic relationships among shots. However, a shot composes of image, audio and text the three multiple modalities; it is difficult to calculate the similarities among shots directly. Also, the temporal-sequenced associated cooccurence characteristic of video reminds of utilizing the multi-modality relationship propagation to gain much more precise and stable similarities among different shots.

The similarity in same modality is easy to calculated, such as the Euclidean distance between image and image, but the correlation between different modalities is hard to obtain, i.e. the relationship of image and text. Thus SimFusion is an effective way to

combine relationships from multiple modalities and achieve multi-modal subspace correlation propagation.

Suppose we have N shots in the training data set X in R^n.

The Unified Relationship Matrix (*URM*) *Lurm* is defined as below:

$$L_{urm} = \begin{vmatrix} \lambda_{11}L_{image} & \lambda_{12}L_{i-a} & \lambda_{13}L_{i-t} & \lambda_{14}L_{i-s} \\ \lambda_{21}L_{a-i} & \lambda_{22}L_{audio} & \lambda_{23}L_{a-t} & \lambda_{24}L_{a-s} \\ \lambda_{31}L_{t-i} & \lambda_{32}L_{t-a} & \lambda_{33}L_{text} & \lambda_{34}L_{t-s} \\ \lambda_{41}L_{s-i} & \lambda_{42}L_{s-a} & \lambda_{43}L_{s-t} & \lambda_{44}L_{shot} \end{vmatrix} . \quad (1)$$

Here L_{image}, L_{audio}, L_{text} and L_{shot} are the intra-modality similarity matrix of image, audio and text spaces respectively. And $L_{i\text{-}a}$, $L_{i\text{-}t}$, $L_{i\text{-}s}$ represent the correlations between image and audio, image and text, image and shot, respectively. The same are the other submatrices. Each submatrix L is $N \times N$. The set of parameters λs are defined to adjust the relative importance of different inter- and intra-modality relationships, and $\sum_{\forall j} \lambda_{ij} = 1,\ \forall i, j, \lambda_{ij} > 0$.

L_{image} and L_{audio} can be calculated based on Euclidean distance, while L_{text} is from Cosine similarity.

Also, the Unified Similarity Matrix (*USM*) is defined as follows:

$$S_{usm} = \begin{vmatrix} 1 & s_{12} & \cdots & s_{1T} \\ s_{21} & 1 & \cdots & s_{2T} \\ \vdots & \vdots & \ddots & \vdots \\ s_{1T} & s_{2T} & \cdots & 1 \end{vmatrix} . \quad (2)$$

where each element $s_{a,b}$ represents the similarity value between data objects a and b (in this case, between image, audio, text and shot) in the unified space. T is the total number of objects in the unified space, i.e. $T = 4 * N$. It is worth mentioning that the order of data objects presented in Susm and Lurm are similar. Having *URM* and *USM* defined, the similarity reinforcement assumption can be represented as :

$$S_{usm}^{new} = L_{urm} S_{usm}^{original} L_{urm}^{T} . \quad (3)$$

Equation (3) is the basic similarity reinforcement calculation in the SimiFusion algorithm. And it can be continued in an iterative manner until the calculation converges or a satisfactory result is obtained, as shown in Equation (4):

$$S_{usm}^{n} = L_{urm} S_{usm}^{n-1} L_{urm}^{T} = L_{urm}^{n} S_{usm}^{0} (L_{urm}^{T})^{n} . \quad (4)$$

In practice, the initial *USM* is often set to be an identity matrix.

The final iterative result S_{usm} can be separated into 4*4 submatrices as L_{urm}. And the last submutrix $W_{N\times N}$ represents the similarity between shots, which is ultimately what we want in this step and will be one input of the next dimension reduction process.

2.3 Dimension Reduction

As mentioned in section 1, LPP is an efficient mean that combines linear and non-linear features of manifold learning.

Given the training data set $X = \{x_1, x_2, \ldots x_N\}$ in R^n as section 2.2, the calculation of LPP will find a transformation matrix A that maps these N points to a set of points $y_1, y_2, \ldots, y_N$ in R^l ($l << n$), such that y_i "represents" x_i, where $y_i = A^T x_i$.

The main procedure of LPP is formally stated below:

1. Choose the weight of the adjacency graph that constructed with the training data set. Here we use $W_{N\times N}$ computed above from SimFusion.

2. Compute the eigenvectores and eigenvalues for the generalized eigenvector problem:

$$XLX^T a = \lambda XDX^T a \,. \tag{5}$$

where D is a diagonal matrix whose entries are column sums of W, $D_{ii} = \sum_j W_{ji}$. $L = D - W$ is the Laplacian matrix. The i^{th} column of matrix X is $\mathbf{x}_i$.

Let the column vectors $\mathbf{a}_0, \ldots, \mathbf{a}_{l-1}$ be the solutions of equation (5), ordered according to their eigenvalues, $\lambda_0 < \cdots < \lambda_{l-1}$. Thus, the embedding is as follows:

$$\mathbf{x}_i \rightarrow \mathbf{y}_i = A^T \mathbf{x}_i,\ A = (\mathbf{a}_0, \mathbf{a}_1 \ldots, \mathbf{a}_{l-1}) \,. \tag{6}$$

where $\mathbf{y}_i$ is a l-dimensional vector, and A is a $n \times l$ matrix.

2.4 Model Training

Among the large variety of supervised machine learning approaches available, the Support Vector Machine (SVM) framework [16] has proven to be a solid choice. The SVM is able to learn from few examples, handle unbalanced data, and handle unknown or erroneous detected data. An SVM tries to find an optimal separating hyperplane between two classes by maximizing the margin between those two different classes. Finding this optimal hyperplane is viewed as the solution of a quadratic programming problem.

In our approach, we use SVM to construct the classification model. The input takes the features that are processed through above steps in stead of the original data.

3 Experiments

Our experiments are designed to evaluate the effectiveness of using the above way to detect semantic concepts in video. We use the TRECVID 2005 video track, which composes of 168 hours digital video (MPEG-1) from LBC(Arabic), CCTV4, NTDTV(Chinese), and CNN, NBC, MSNBC(English).

We chose the semantic concepts “explosion/fire”, “sports”, “Military”, “Government Leader” and “Airplane” to test our method. The ground-truth of the presence of each concept was assumed to be binary (either present or absent in a video shot).

3.1 SimFusion vs. Late and Early Fusion

First we compare the results of utilizing SimFusion, traditional Early and Late Fusion.

Table 1. Average Precision (%) of Video Concept Detection using different fusion methods

Concept	SimFusion	Early Fusion	Late Fusion
Explosion/Fire	60.8	46.7	53.4
Sports	55.1	38.5	48.3
Military	47.5	30.9	41.2
Government Leader	39.3	27.4	34.7
Airplane	23.6	15.2	19.8

Table 1 shows that using SimFusion gets more accurate results than simply using early fusion and late fusion. Thus the correlation propagation between multi-modalitis by SimuFusion can impact the detection results, and make better performance.

3.2 LPP vs. PCA and ISOMAP

In this section, we compare the performance of different dimensionality reduction techniques PCA, ISOMAP and LPP, which represent linear, nonlinear methods and combination of the both, respectively.

Table 2. Average Precision (%) of Video Concept Detection using different dimension reduction methods

Concept	LPP	PCA	ISOMAP
Explosion/Fire	60.8	52.8	54.9
Sports	55.1	43.6	47.3
Military	47.5	38.2	39.1
Government Leader	39.3	29.7	31.5
Airplane	23.6	18.5	20.4

In table 2, we can see that LPP performs better than PCA and ISOMAP. Since LPP preserves the intrinsic nonlinear structure of the dataset in low-dimensional manifold. And LPP is essentially a linear method, it is more efficient than ISOMAP.

4 Conclusions

Semantic understanding of video is a hard but important research topic. The ideas for semantic concept detection in video are still in process. In this paper, the new

approach we present to detect semantic concepts from video shots is based on SimFusion and LPP. And this method focuses on the temporal-sequenced associated cooccurence characteristic of video. The experiments show that our method achieves improved performance than traditional methods. In the future, we plan to design specific processes for different concepts to obtain better results.

Acknowledgments. This work is supported by National Natural Science Foundation of China (No.60603096, No.60525108), Science and Technology Project of Zhejiang Province (2005C13032，2005C11001-05), and China-US Million Book Digital Library Project (www.cadal.zju.edu.cn).

References

1. Cees G.M. Snoek, Marcel Worring, Arnold W.M.Smeulders : Early versus Late Fusion in Semantic Video Analysis. Proceedings of the 13th annual ACM International Conference on Multimedia (2005) 399-402
2. Wensi Xi, Edward A.Fox, etc: SimFusion:Measuring Similarity using Unified Relationship Matrix. The 28th Annual International ACM SIGIR Conference (SIGIR'2005)
3. Susan T.Dumais, George W.Furnas, Thomas K.Landauer : Using Latent Semantic Analysis to Improve Access to Textual Information. Proceedings of the SIGCHI conference on Human factors in computing systems (1988) 281-285
4. Xiaofei He, Partha Niyogi: Locality Preserving Projections. Advances in Neural Information Processing Systems (NIPS 2003)
5. Yi Wu, Ching-Yung Lin, Edward Y.Chang, John R.Smith: Multimodal Information Fusion for Video Concept Detection. International Conference on Image Processing (2004) 2391-2394
6. R.Bellman : Adaptive Control Processes: A Guided Tour. Princeton University Press(1961)
7. Miguel Á. Carreira-Perpiñán.: A Review of Dimension Reduction Techniques. Technical report CS-96-09, Dept. of Computer Science, University of Sheffield, UK
8. I.T.Jolliffe: Principal Component Analysis. Springer, New York, 2nd edition (2002)
9. Guy Philip Nason : Design and choice of projection indices. PhD Thesis, University of Bath
10. Sam T. Roweis, Lawrence K. Saul : Nonlinear Dimensionality Reduction by Locally Linear Embedding, Science Vol.290 (2000) 2323-2326
11. Joshua B. Tenenbaum, Vin de Silva, John C. Langford : A Global Geometric Framework for Nonlinear Dimensionality Reduction, Science Vol.290 (2000) 2319-2323
12. Mikhail Belkin, Partha Niyogi : Laplacian Eigenmaps for dimensionality reduction and data representation. Neural Computation, Vol 15, Issue 6 (2003) 1373-1396
13. M.Belkin and P.Niyogi : Laplacian Eigenmaps and spectral techniques for embedding and clustering. Advances in Neural Information Processing Systems 14, MIT Press, Cambridge (2002) 585-591
14. A.Hauptmann, M.Y.Chen, M.Christel, C.Huang, etc: Confounded Expectations: Informedia at TRECVID 2004.
15. C.G.M. Snoek, M.Worring, et al: The MediaMill TRECVID 2004 Semantic Video Search Engine. In Proc. TRECVID Workshop,Gaithesburg, USA (2004)
16. Chih-Chung Chang and Chih-Jen Lin, LIBSVM : a library for support vector machines, 2001. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm

Enhancing Comprehension of Events in Video Through Explanation-on-Demand Hypervideo

Nimit Pattanasri, Adam Jatowt, and Katsumi Tanaka

Department of Social Informatics, Kyoto University
{nimit,adam,tanaka}@dl.kuis.kyoto-u.ac.jp

Abstract. The objective of this paper is to help users navigate for additional information for more comprehension of events in video through an explanation-on-demand hypervideo system. Given an XML database consisting of MPEG-7 video annotations and ontologies specifying relationships among events in annotations, the system is responsible for dynamically identifying hyperlinks inside a video that users possibly follow to clarify the points they do not understand. Three types of hyperlinks that are helpful in enhancing user comprehension of events are proposed: context, precondition and causal, which are based on an analysis of Fellbaum's verb entailments [4].

Keywords: hypervideo, comprehension, verb ontology, Fellbaum's verb entailments, MPEG-7, OWL, XML database, reasoning, XDD.

1 Introduction

Hypervideo (e.g. [1,14,13]) is a kind of interactive video that requires a multimedia modeling technique for visualizing its content and providing access to the segments through *hyperlinks*. Its objective is to facilitate information searching in video. When users need more details, they can follow link opportunities, appearing somewhere in the screen during some particular moment. This is made possible only after annotators place beforehand the links from source video segments to destination segments.

This paper mainly focuses on two problems in hypervideo. From a system viewpoint, we aim at generating hyperlinks semi-automatically from available metadata in video, thus relieving the burden of manually annotating links. From a user viewpoint, we attempt to solve the problem when users need more explanations to specific events in video they do not understand. It is assumed for user behavior that the purpose of reinvestigating the video is to gain more understanding of specific events of interest. We argue, in this paper, that the system should help users seek for events (i.e. through hyperlinks) that are not only *similar* or *related* but also *clarify* what users might doubt or expect.

The main problem of automatic video hyperlink generation lies in the incapability of both underlying semantics of *similarity* [11] and *relatedness* [10] to capture semantics of *comprehension*. For example, knowing that a video segment, S_a, is similar or related to another, S_b, (by comparing their metadata) it tells

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 535–544, 2007.
© Springer-Verlag Berlin Heidelberg 2007

nothing about whether watching S_a will increase understanding of S_b or vice versa. Rather, approaches employing the two notions can sometimes exacerbate the situation by offering excessive information to users.

The objective of this paper is to enhance user comprehension of events through a hypervideo system. Given an XML database consisting of MPEG-7 video annotations and ontologies specifying relationships among events in annotations, the system is responsible for dynamically identifying hyperlinks inside a video that users possibly follow to clarify the points they do not understand. Informally, the problem of generating hyperlinks, in this context, can be formulated as follows: given a particular event, e_1, of user interest, if another event, e_2, can enhance comprehension of e_1, a hyperlink is established from e_1 to e_2.

Previously, [9] proposes a context-preserving video segment retrieval system capable of retrieving video segments that are not only similar but also related to user queries. The authors did not, however, introduce the notion of user comprehension, its problem in video browsing, and a solution for enhancing comprehension, which is our attempt in this paper.

The contribution of this paper can be summarized as follows. Our work is, to the best of our knowledge, the first to introduce the notion of *comprehension* of events based on Fellbaum's verb entailments [4]. Analysis of verb entailments yields three types of *explanatory events*: context, precondition and causal, which are useful for increasing comprehension of particular events. The ability to discriminate explanatory events from ones that are only *similar* or *related* can reduce information overload to users when they need more explanations.

The rest of the paper is organized as follows. Section 2 briefly reviews the basic of MPEG-7 for annotating video content. Section 3 proposes a methodology for enhancing user comprehension of events. Section 4 presents a rule-based system for implementing an explanation-on-demand hypervideo application. Section 5 shows some implications of this research. Section 6 discusses related work. Section 7 summarizes conclusions and outlines future work.

2 MPEG-7 Video Annotation

A large number of digital videos demand efficient indexing techniques enabling accessibility to those videos. A conventional approach for indexing videos is to assign keywords to them as their metadata. In this way, one can retrieve videos of interest from a system by formulating only a few keywords like searching on the Web. However, for a several-hour-long video such as movies, it is often the case that only parts (called segments) can exactly satisfy user information needs. Thus, there is a need for video indexing at the fine granularity level of segments [15]. Since the current state-of-the-art techniques for automatic video content analysis are restricted to the detection of basic events such as people walking and physical violence [8], there is still a need for human intervention in order to improve quality of metadata [5].

To provide video indexes consistently, a common scheme for interoperability of metadata among communities is required. MPEG-7 [7] is the standard

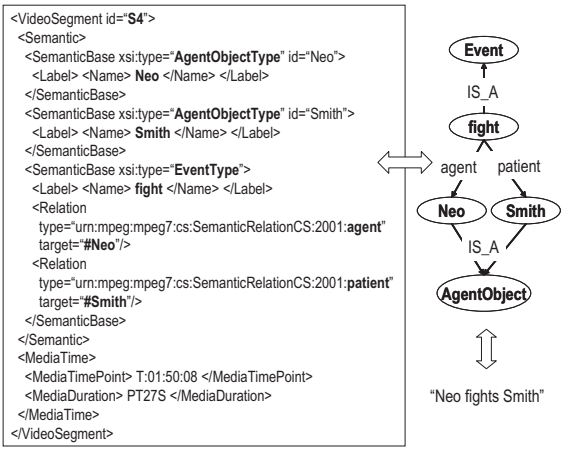


Fig. 1. An example of event-based metadata using MPEG-7 Event DS

XML-based framework for annotating multimedia metadata through descriptors (D) and description schemes (DS). VideoSegment DS is a tool for annotating video attributes such as a source video location, time point and duration in a referenced video. Event DS is selected to describe events in video segments, which are restricted, in this paper, to the form: *subject-verb-object*. Figure 1 shows a metadata example of MPEG-7 Event DS for describing events in a video segment. For the sake of simplicity, we assume that thematic roles of an event, namely *agent* and *patient*, are represented by *subject* and *object*, respectively. Note also that events are represented by verbs.

3 Enhancing User Comprehension of Events

This section reveals limitations of previous approaches, and proposes a solution to the problem of user comprehension in video.

3.1 Verb Ontology

A first step toward increasing user comprehension of events, in this paper, is to acquire knowledge that can semantically *relate* events together. We argue that only IS-A relationships among events are not expressive enough to handle the problem of user comprehension. For example, given an event of interest "A fights B", we can say "A shoots B" is also of our interest; *shoot* is a kind of *fight*. However, taking this kind of similarity between the concepts into account [11], it is difficult to justify that, for example, "A dies" is also relevant to the event of interest since the semantic of *fight* and *die* is not so similar (i.e., *die* is not a kind of *fight* and vice versa). Furthermore, compare two events "A fights B" and "A runs away from B". The semantics of *fight* and that of *run away* are clearly

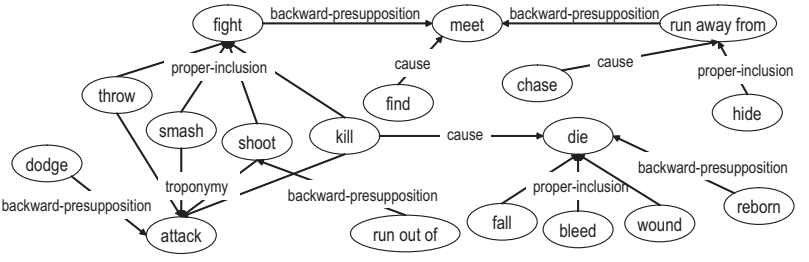


Fig. 2. An example of verb ontology

not similar. However, they seem to be related in some way although it is hard to specify a reason. This implies the incapability of IS-A relationships to capture our intuition for identifying related events.

Since verbs can be used to represent events as illustrated in Sect. 2, it is worth considering existing knowledge about relationships among verbs. Specifically, we exploit Fellbaum's verb entailments [4] to model a *verb ontology*, representing relationships among events.[1] According to Fellbaum's classification, there are four types of entailment relations. Let v_1 and v_2 be verbs.

Definition 1 (Troponymy). *v_1 is a kind of v_2. (ex. limp-walk, lisp-talk)*

Definition 2 (Proper-inclusion). *v_1 can happen in the duration of v_2. (ex. snore-sleep, buy-pay)*

Definition 3 (Backward-presupposition). *v_2 must happen before v_1. (ex. forget-know, succeed-try)*

Definition 4 (Cause). *v_1 causes v_2 to occur. (ex. show-see, give-have)*

Definition 5 (Verb ontology). *Verb ontology is a directed-graph whose nodes are verbs or phrasal verbs and directed edges are one of the four entailment relations: troponymy, proper-inclusion, backward-presupposition and cause.*

Figure 2 shows an example of verb ontology.[2] Considering this ontology and assuming both A and B are persons, if we know that A shoots B, we would imply that A is fighting B; *shoot* is included in the duration of *fight*. A smashes B meaning that A attacks B; *smash* is a kind of *attack*.[3] Given a situation that A kills B, it can be implied that B will die; *kill* causes *die*. Before A runs away from B, it is presupposed that A met B sometimes in the past; *meet* happens before *run away*. This kind of knowledge can be interpreted from the verb ontology, where events can be related through the entailment relations. Let e_i, e_j be events and v_i, v_j be verbs, representing e_i and e_j, respectively.

[1] [12] also suggest the use of verb entailments as the first step to relate events; however, they did not propose a verb ontology and its use to enhance user comprehension.

[2] There are three main sources for verb ontology construction: human labor, WordNet and Web mining (e.g. see [3]).

[3] For troponymy, the duration of both verbs must begin and end at the same time.

Definition 6 (Similar event). *e_i is a similar event of e_j iff there exists in a verb ontology the "troponymy" relation from v_i to v_j or vice versa.*

Definition 7 (Related event). *e_i is a related event of e_j iff there exists in a verb ontology one of the four entailment relations from v_i to v_j or vice versa.*

Note also that constraints of verb attributes (i.e. subjects and objects) cannot be enforced in the representation of the ontology. For example, "A kills B" is not related to "C dies" unless B and C are the same person.

3.2 Classification of Events for Enhancing Comprehension

When users do not understand an event in video they usually search for related events for more explanations. However, not all related events can contribute to comprehension of the event which is of user interest. Rather, some of the events are considered useless, thus causing information overload.

Specifically, the notion of *relatedness* is not appropriate to solve the problem of comprehension because its nature is symmetric. For example, given that an event, e_1, is related to another, e_2, it can be implied that e_2 is also related to e_1. This is, however, not always the case for the notion of *explanation*, which is the basis for enhancing comprehension. To overcome this difficulty, we propose, based on semantics of verb entailments, three types of explanatory events that are useful for enhancing comprehension of events. For the definitions that follow, let e_1, e_2 be events, and v_1, v_2 be verbs, representing e_1 and e_2, respectively.

Definition 8 (Context event). *e_1 is a context event of e_2 iff there exists in a verb ontology an entailment relation, "proper-inclusion", directed from v_1 to v_2.*

Definition 9 (Precondition event). *e_1 is a precondition event of e_2 iff there exists in a verb ontology an entailment relation, "backward-presupposition", directed from v_1 to v_2.*

Definition 10 (Causal event). *e_1 is a causal event of e_2 iff there exists in a verb ontology an entailment relation, "cause", directed from v_2 to v_1.*

Proposition. Context, precondition and causal events increase user comprehension of a current event of interest.
Rationale. Knowing its surrounding context, preconditions and causes help users better understand the current event.

For example, the *meet* event can be considered as a precondition of the *fight* event since *meet* must happen before *fight* (backward-presupposition). In other words, *meet* enhances comprehension of *fight*. The opposite is not always the case; the *fight* event does not increase comprehension of the *meet* event.

Definition 11 (Explanatory event). *e_1 is a explanatory event of e_2 iff e_1 is a context, precondition, or causal event of e_2.*

4 Explanation-on-Demand Hypervideo

This section demonstrates a hypervideo system that exploits explanatory events to enhance user comprehension. Figure 3 shows a screen capture of a prototype for explanation-on-demand hypervideo. A user comprehension model of video browsing is assumed; users browse a video from the beginning to the point they do not understand, and begin to reinvestigate related events in the past. Given a current event being watched as a query, an explanation-on-demand hypervideo system is responsible for dynamically establishing hyperlinks from the current event to its explanatory events where users can follow for clarification. In addition, we assume that event-based metadata for video segments (described in Sect 2) are made available, for example, by annotators. Figure 4 shows an example of metadata, which is also assumed to be precise. Annotators can also consult a verb ontology for describing events in video segments.

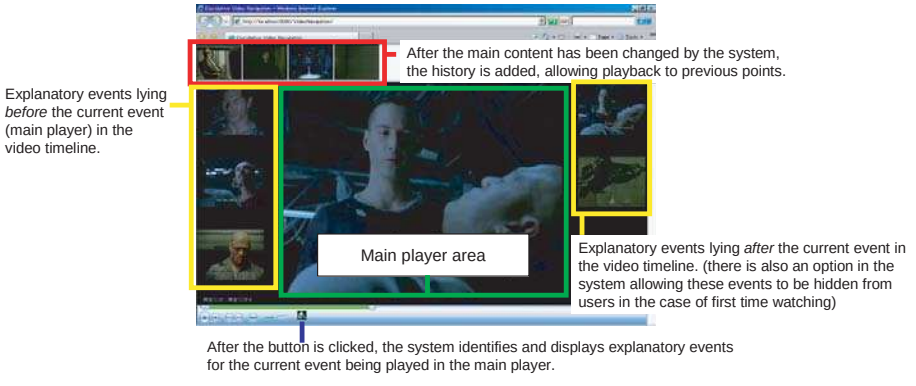


Fig. 3. Screen capture of a prototype for explanation-on-demand hypervideo

4.1 Establishing Hyperlinks Through a Query Model

Although MPEG-7 provides a framework for describing multimedia through descriptors and description schemes, it cannot explicitly express relationships among descriptors, description schemes, constraints, and rules. *XML Declarative Description (XDD)* [16] is a language for modeling XML databases capable of expressing explicit and implicit information through *XML expressions* (variable-embedded XML elements), and relationships, constraints, and rules through *XML clauses*. XML clause is of the form: $H \leftarrow B_1,...,B_n$. where $n \geq 0$, and H and B_i are XML expressions or constraints. When n=0, a clause is called *unit clause*, otherwise *non-unit clause*. The symbol $\leftarrow$ is often omitted in the case of unit clauses; thus, XML elements or documents such as MPEG-7 descriptions and OWL ontologies can immediately become ground XML unit clauses. Given a database, XDB, consisting of XML unit and non-unit clauses and a query, Q, formulated in terms of a non-unit clause, the answers, Q', can be derived by transforming Q repetitively using XML clauses in XDB.

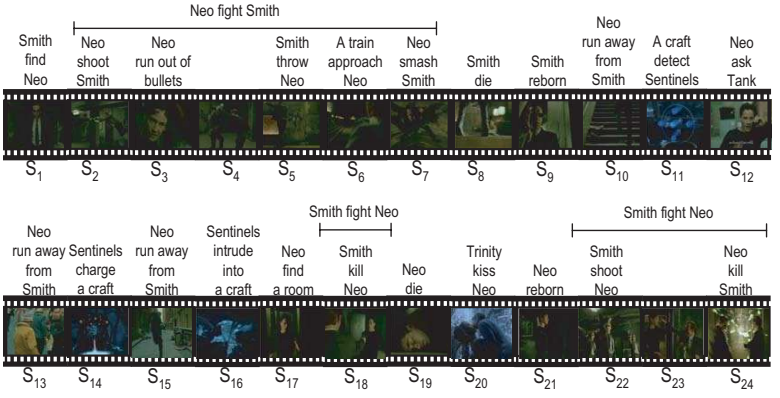


Fig. 4. A metadata example from *The Matrix*

An XML database consists of MPEG-7 documents (XML unit clauses) describing events occurred in video segments, an OWL document (XML unit clause) representing a verb ontology, and rules (XML non-unit clauses) identifying explanatory events. For the sake of simplicity and due to space limitation, we simply describe XDD rules through Horn rules.

```
R1: enhanceComprehension(s1=(e1,tb1,te1),s2=(e2,tb2,te2),type)
                              ← segment(s2=(e2,tb2,te2)),explanatory(e2,e1,type).

R2:              explanatory(e1,e2,"context") ← context(e1,e2).
R3:         explanatory(e1,e2,"precondition") ← precondition(e1,e2).
R4:               explanatory(e1,e2,"causal") ← causal(e1,e2).

R5:      context(e1=(sub1,v1,obj1),e2=(sub2,v2,obj2)) ← entail(v2,"proper-inclusion",v1).
R6: precondition(e1=(sub1,v1,obj1),e2=(sub2,v2,obj2)) ← entail(v2,"backward-presupposition",v1).
R7:       causal(e1=(sub1,v1,obj1),e2=(sub2,v2,obj2)) ← entail(v1,"cause",v2).

where s=(e,tb,te) is video segment metadata where e=(sub,v,obj)
represents an event of the form: subject-verb-object,
tb is the beginning time point, te is the ending time point,
and the semantics of predicates is as follows.

enhanceComprehension(s1,s2,t) ... s2 enhances comprehension of s1
             segment(s) ... there exists a video segment, s, in the database
   explanatory(e1,e2,t) ... e1 is an explanatory event of e2 (of t type)
        context(e1,e2) ... e1 is an context event of e2
    precondition(e1,e2) ... e1 is an precondition event of e2
         causal(e1,e2) ... e1 is a causal event of e2
      entail(v1,rel,v2) ... there exists in the verb ontology an
                            entailment relation rel directed from v1 to v2
```

Note that not only verbs but also their attributes (i.e. subjects and objects) play an important role in the identification of explanatory events. For example, "A kills B" is not an explanatory event of "C dies" unless B and C are the same person. Such constraints can be added in the rules and are omitted for the sake of simplicity.

A query is a non-unit clause which can be formulated, for example, as follows. The te2<tb1 constraint is specified to guarantee that the segments that

users have not yet watched will not be shown (assuming that it is the first time watching).

```
Q: answer(s2,type)
   ← enhanceComprehension(s1=(e1=(Smith,kill,Neo),tb1,te1),s2=(e2,tb2,te2),type),[te2 < tb1].
where inside a bracket is a constraint.
```

Suppose the XML database consists of MPEG-7 metadata illustrated in Fig. 4, and S_{18}, "Smith kills Neo", is a current event of interest. By R1, R2, R5 rules, it can be derived that S_2-S_7 and S_{18} are the explanatory events of S_{18}[4]; explanation-on-demand hyperlinks from S_{18} to S_2-S_7 and S_{18} are established. Note also that S_1, "Smith finds Neo", can also enhance user comprehension of S_{18} although the R1 rule cannot identify it (since there is no annotation about "Smith meets Neo" in the database). The next section proposes a query relaxation algorithm to solve this problem.

4.2 Query Relaxation

The art of making films involves arrangement and composition of video segments, so-called *film grammars* [2]. It is often the case that movie directors drop some events from the films and let viewers infer implicit events from available hints. For example, a movie director may decide to exclude S_{18}, "Smith kills Neo", in the film (see Fig. 4) and suppose a user is currently watching S_{19}, "Neo dies", where he needs more details. The rules defined above will try to identify a segment about *someone killing Neo* as an explanatory event. However, there is no such segment in the database (supposing S_{18} is removed). Common sense tells us that the event of interest is related in some way to the fight event. By considering the verb ontology, *fight* is an explanatory event of *kill*, and *kill* is an explanatory event of *die*; that is, *fight indirectly* increases comprehension of *die*. In this respect, we need an algorithm to reformulate a query when there is no *direct* explanatory event (i.e. *kill*, in this case), identified by the R1 rule.

```
R8: enhanceComprehension(s1=(e1,tb1,te1),s3=(e3,tb3,te3),type)
        ← enhanceComprehension(s2=(e2,tb2,te2),s3=(e3,tb3,te3),type),
          isExplanatoryOf(v2,v1), isExplanatoryOf(v3,v2), ¬segment(s2=(e2,tb2,te2)).

R9:  isExplanatoryOf(v1,v2) ← entail(v2,"proper-inclusion",v1).
R10: isExplanatoryOf(v1,v2) ← entail(v2,"backward-presupposition",v1).
R11: isExplanatoryOf(v1,v2) ← entail(v1,"cause",v2).

where ¬segment(s) ... there is no "s" in the database
```

Query relaxation is performed by adding more rules, R8-R11. The R8 rule first checks in the verb ontology whether there exists, in the database, e_2 as a direct explanatory event for e_1 (an event of user interest). If not, it reformulates the query to enhance comprehension of e_1 with e_3, instead of e_2, where e_3 is an explanatory event of e_2 (and e_2 does not exist in the database). This process continues recursively until the first *transitively* indirect explanatory events are identified.

[4] S_{18} itself may be useful in this case when users forget information very soon.

5 Discussion

Since the quality of video annotations is difficult to be measured and its judgment is highly subjective, a critical question may arise for validity of our approach. The verb ontology can be used to bridge the semantic gap between user queries and annotations which are imprecise, due to subjectivity of annotator perception. For example, a video segment about *kill* (determined by an annotator) may be instead annotated with *die* (by another annotator). Nevertheless, such a video segment can be obtained for the query, *kill*, which is one of the surrounding concepts of *die* in the ontology. Therefore, the verb ontology serves two purposes: to relate events together and to bridge the semantic gap between human perception.

6 Related Work

[14] proposes a hypervideo authoring tool, called Hyper-Hitchcock. The tool is capable of generating hierarchical video summaries where users can access detailed information through hyperlinks being placed between different levels of summaries. The algorithm to identify and place links in video depends on heuristics for determining importance of video segments such as segment duration. Because of the nature of placing links hierarchically in a video, sometimes, users are required to search for detailed information from the top- until the bottom-level summaries (i.e. the whole video). Plot units are a knowledge representation technique used primarily for narrative summarization [6], and are recently exploited for hypervideo applications [1,13]. Associated with any two video segments, each plot unit is composed of an explicit cause-effect relationship linking between human affect states, each can express underlying motivation of actions or events being performed. Given a particular video with such hyperlinks visible to end-users, they can easily navigate such a video for the parts they do not understand [13]. Nevertheless, it is reported in [13] that the time required for manual annotating plot units is about 35 person hours per feature film.

7 Conclusions and Future Work

In this paper, we argue that both notions of similarity and relatedness are not appropriate for increasing user comprehension. To enhance user comprehension and to avoid information overload we propose three types of explanatory events: context, precondition and causal, which are based on the semantics of Fellbaum's verb entailments. Our explanation-on-demand hypervideo prototype employs XDD as the underlying querying mechanism capable of dynamically identifying video segments that are helpful in increasing user comprehension as well as performing query reformulation whenever necessary, making the query processing more efficient. Since our approach hinges on the availability of video annotations and the quality of verb ontologies, support for the annotation task and construction of the verb ontology is necessary. Experiments and user evaluations also form part of our future work.

Acknowledgement. This work was supported in part by (1) MEXT The 21st Century COE (Center of Excellence) Program "Informatics Research Center for Developmentt of Knowledge Society Infrastructure" (Leader: Katsumi Tanaka, 2002-2006), (2) MEXT Grant for "Development of Fundamental Software Technologies for Digital Archives", Software Technologies for Search and Integration across Heterogeneous-Media Archives (Project Leader: Katsumi Tanaka), (3) MEXT Grant-in-Aid for Scientific Research on Priority Areas: "Cyber Infrastructure for the Information-explosion Era", Planning Research: "Contents Fusion and Seamless Search for Information Explosion" (Project Leader: Katsumi Tanaka, A01-00-02, Grant#: 18049041), and (4) MEXT Grant-in-Aid for Scientific Research on Priority Areas: "Cyber Infrastructure for the Information-explosion Era", Planning Research: "Design and Development of Advanced IT Research Platform for Information" (Project Leader: Jun Adachi, Y00-01, Grant#: 18049073).

References

1. Allen, R., B., Acheson, J. Browsing the Structure of Multimedia Stories. In Proceedings of ACM Digital Libraries'00. (2000)
2. Arijon, D. Grammar of the Film Language. Silman-James Press. (1976)
3. Chklovski, T., Pantel, P. VerbOcean: Mining the Web for Fine-Grained Semantic Verb Relations. In Proc. of Conf. on Empirical Methods in NLP. (2004)
4. Fellbaum, C. A Semantic Network of English Verbs. WordNet: An Electronic Lexical Database, MIT Press, pp.69–104. (1998)
5. Haase, K., B. Context for Semantic Metadata. In Proceedings of ACM Multimedia'04 (2004)
6. Lehnert, W., G. Plot Units and Narrative Summarization. Cognitive Science, 4, 293–331. (1981)
7. Manjunath, B. S., Salembier, P., Sikora, T. Introduction to MPEG-7: Multimedia Content Description Interface. (2002)
8. Naphade, M., R., Smith, J., R. On the Detection of Semantic Concepts at TRECVID. In Proceedings of ACM Multimedia'04 (2004)
9. Pattanasri, N., Chatvichienchai, S., Tanaka, K. Towards a Unified Framework for Context-Preserving Video Retrieval and Summarization. In Proceedings of International Conference on Asian Digital Libraries (ICADL'05). (2005)
10. Pedersen, T., Patwardhan, S., Michelizzi, J. Wordnet::Similarity - Measuring the Relatedness of Concepts. In Proceedings of National Conference on Artificial Intelligence (AAAI'04). (2004)
11. Resnik, P., Diab, M. Measuring Verb Similarity, In Proceedings of the 22nd Annual Meeting of the Cognitive Science Society. (2000)
12. Salway, A., Tomadaki, E. Temporal Information in Collateral Texts for Indexing Movies. In Proc. of Third Int. Conf. on Language resources and evaluation. (2002)
13. Salway, A., Xu, Y. Navigating Stories in Films. Technical Report CS-05-04, Dept. of Computing, University of Surrey. (2005)
14. Shipman, F., Girgensohn, A., Wilcox, L. Generation of Interactive Multi-Level Video Summaries. In Proceedings of ACM Multimedia'03. (2003)
15. Snoek, C., G.M., Worring, M. Multimodal Video Indexing: A Review of the State-of-the-Art. Multimedia Tools and Applications, 25(1), 5–35. (2005)
16. Wuwongse, V., Akama, K., Anutariya, C., Nantajeewarawat, E. A Data Model for XML Databases. Journal of Intelligent Information Systems, 20(1), 63–80. (2003)

Low-Complexity Binaural Decoding Using Time/Frequency Domain HRTF Equalization

Rongshan Yu, Charles Q. Robinson, and Corey Cheng

Dolby Laboratories, 100 Potrero Ave. San Francisco, CA 94103, USA
{rzyu,cqr,cnc}@dolby.com

Abstract. Binaural rendering technology is used to generate a two-channel signal from one or more channel signals, where each channel signal has associated with it a position relative to the listener. The resulting binaural signal, when played back over an appropriate device such as headphones, gives the sensation of audio signal(s) originating from the assigned position. The binaural rendering process typically involves applying a pair of Head-Related Transform Function (HRTF) equalizer to each input channel signal. The left and right ear signals from each of the input channels are then combined to generate a binaural signal. In this paper, we introduce a Time/Frequency (T/F) domain HRTF equalization technique which can be used to accomplish HRTF-based binaural rendering. The proposed technique can be conveniently combined with multi-channel/spatial decoding systems, such as multi-channel HE-AAC or MPEG surround decoder for low-complexity binaural rendering of multi-channel program.

1 Introduction

As sound propagates, it interacts with the physical environment and is modified. Thus the exact acoustic path and the physical features along the path from a sound source to the ear (or other transducer) will result in particular sound modifications. When listening to live events, sound arrives at each of our ears ("binaurally") following slightly different acoustic paths, resulting in different modifications. In particular, the location of the ears, and the shape of the outer ear, head, and shoulders result in the sound arriving at each ear at different times, levels, and with different spectral shaping or filtering. The cumulative effect of these signal manipulations is called the Head Related Transfer Function (HRTF). The HRTF varies with individual and with the relative position of the sound source and the ear.

We are able to process the signals and the differences between the signals in each ear to determine spatial characteristics of the sound such as position (direction and distance) and image size. For example, a sound source located on our left hand side will be louder and arrive slightly sooner at our left ear than our right ear. Other more subtle cues (such as relative spectral shape) can give us information about the elevation, and help resolve front/back location.

Binaural sound is inherently a two-channel signal, with one channel applied to each ear. "Ordinary" two channel signals (with limited binaural cues) are commonly available (via radio, CDs etc) for reproduction over loudspeakers or headphones. In

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 545 – 556, 2007.
© Springer-Verlag Berlin Heidelberg 2007

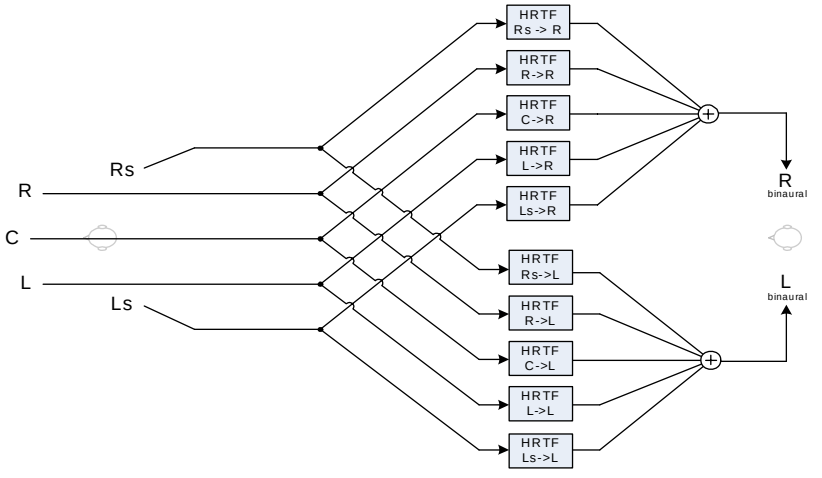


Fig. 1. Binaural rendering of a 5 channel program

such cases the reproduction of spatial information is limited, especially with headphones which can suffer from "inside the head" imaging. On contrary, binaural sound is inherently a two-channel signal, with one channel applied directly to each ear (e.g. over headphones, or traditional left-right speaker pair with crosstalk cancellation). As a result, a more spatial impression can be delivered.

The HRTFs (for left and right ear respectively) associated with a particular location "A" can be estimated by analyzing the relationship between signals received via a binaural recoding and a source signal played at this particular location. Such a recording is optimally made in an anechoic room to eliminate room effects and isolate the head-related effects (anechoic HRTF). A binaural signal can then be synthesized by applying the computed HRTF pair to a desired source signal. When played over headphones or other appropriate means, the signal will now appear to come from location "A." In order to represent other locations, a multiplicity of HRTF pairs can be derived. Furthermore, a synthetic room response can be added to the HRTFs to provide a more natural sense of space.

One application for binaural audio is for playback of multi-channel audio programs. Multi-channel audio content is increasingly common (DVD's, HDTV). However multi-channel reproduction typically requires a large room with multiple channels of amplification and loudspeakers, which may not be available all the time. An alternative to loudspeaker play back is to listen to a binaural rendering of the five channel program over headphones. In order to recreate the same spatial perception, the original speaker signals are processed using the appropriate HRTF pair that associated with each combination of different loudspeaker locations and each ear, and then summed to create a binaural signal. The above binaural rendering process for 5 channel program is illustrated in Fig. 1.

Digitizing audio is a very common means for storing and transmitting audio data. There are many advantages to digital representation of audio, but one disadvantage is the large amount of data required. This is especially true for multi-channel audio

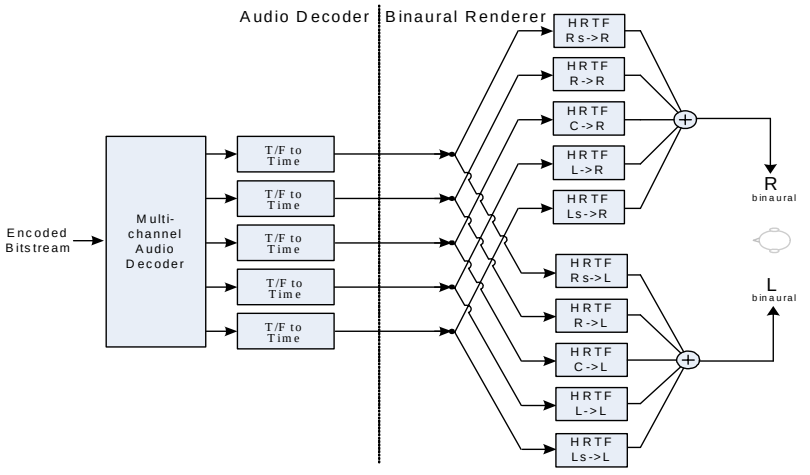


Fig. 2. Brute-force combination of multi-channel audio decoder and binaural renderer by cascading of these two processes

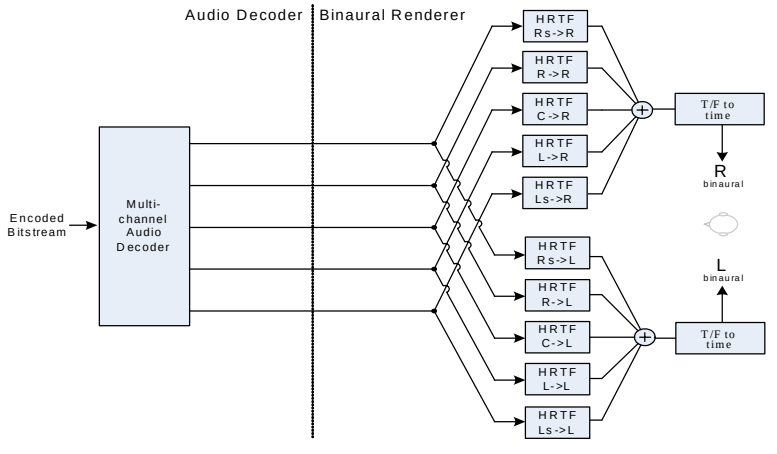


Fig. 3. Simplified Audio Decoder and Binaural Renderer with T/F domain HRTF processing

programs. As a result, digital audio is frequently encoded to decrease the amount of data required, while maintaining sound quality as much as possible. Common digital audio coding algorithms for achieving this include Dolby Digital™ [1], MPEG 1 Layer 3 (mp3) [2], MPEG 4 Advanced Audio Coding (AAC) [3], and High Efficiency AAC (HE-AAC) [4]. Multi-channel audio are supported in these audio compression algorithms by coding the waveform for each individual channel explicitly. More recently, more efficient multi-channel audio coding methods are being developed [5][6]. Those algorithms rely on some down-mix and up-mix processes where only the waveforms of the down-mixed channels are encoded explicitly, while the original channel configuration are regenerated at the decoder side by using an up-mix process. In order to more accurately model the original N channel program's sound field,

parameters are extracted during the down-mix/encode process which are use to help guide the up-mix process. Typical parameters include channel level differences (CLD), inter-channel time or phase differences (ITD or IPD), and inter-channel coherence (ICC) [5].

When coded audio is to be rendered binaurally, it is worth considering how the multi-channel audio decoding process and binaural rendering process interact in order to optimize performance. Typically, HRTF processing is implemented as an equalization process, where the HRTF is often represented in terms of its discrete time impulse response. The impulse response is then convolved (as an FIR filter) with the desired digitized source signal to create the binaural signal. Thus, binaural decoding of multi-channel coded program can be achieved by using the brute-force combination as shown in Fig. 2, where the time-domain waveforms are decoded with multi-channel audio decoder first, and then processed by the binaural renderer to generate the binaural audio. The disadvantage of this approach in terms of computational complexity is quite obvious, since both the filterbank operation in the audio decoder to transform the audio signal from certain Time/Frequency (T/F) domain to time domain, and the FIR filtering operation required for HRTF equalization in the binaural renderer have high computational complexities.

Alternatively, it is possible to reduce the computation complexity of binaural decoding of multi-channel program by implement the HRTF equalization in the T/F domain (Fig. 3). The advantages this approach are two-folds. First, the number of filterbank operations is reduced as they now perform on the down-mixed binaural signal, which has only two channels. Second, it is also possible to reduce the complexity of HRTF equalization process since it now works in T/F domain which has a lower sampling rate. Techniques for implementing the HRTF equalization in the T/F domain were previously proposed in [7-10]. In principle, those techniques try to find the T/F domain HRTF equalizers that could produce the mathematically identical results to those from time-domain HRTF equalizers. To achieve this goal, the impulse responses of the time-domain HRTFs are polyphase decomposed, and convoluted with the analysis/synthesis filterbank to produce impulse response of the T/F domain equalizer. The overall complexity of the resulting T/F domain equalizer, however, can still be pretty high in particular if the analysis/synthesis filterbank has long impulse responses. In addition, filtering operations across different T/F bands are in general inevitable in order to produce the mathematically identical result [7][9].

In most practical applications, we are in fact more interested in having a low-complexity binaural decoder that could produce the same binaural perceptual effect as that from the brute-force combination in Fig. 2. To this end, we notice that the perception of binaural audio is mainly determined by two factors: 1) the amplitude response of HRTF which controls the strength and timbre of the sound presented to both ears, as well as necessary location hints; and 2) the phase and delay response of HRTF which are also relevant to the perception of the sounds location. Clearly, instead of having a “numerically lossless” solution we only need to design the T/F domain HRTF equalizers that produce, in combination with the analysis/synthesis filterbank, similar amplitude and phase responses to those of the time-domain HRTF equalizers in order to achieve “perceptually transparent” binaural decoding.

In this paper, we follow this philosophy to derive a low-complexity T/F domain HRTF equalizer design. To simplify the problem, the proposed HRTF equalizer is

constructed by a cascade of three digital filters, namely, an amplitude filter, a fractional delay filter, and a phase filter, and each filter is then designed separately in order to produce the desirable amplitude and phase response. The performance of the resulting T/F domain HRTF equalizer is compared with time-domain HRTF equalizer through listening test. The listening test results show that the proposed algorithm produces identical perceptual binaural decoding as to the brute-force time-domain HRTF equalization process, and hence justifies the effectiveness of the proposed method.

2 Design of T/F Domain HRTF Equalizer

Consider the equalization process in the T/F domain as shown in the left-hand side of Fig. 4. Here $H_1(z),\ldots,H_M(z)$ is the analysis filterbank, $G_1(z),\ldots,G_M(z)$ is the synthesis filterbank. For each subbands k, $k=1,\ldots M$, in the T/F domain, the subband signal $x_k(n)$ is processed with the corresponding equalizer $S_k(z)$, and its output $y_k(n)$ is finally synthesized with the synthesis filterbank to generate the time-domain output signal.

Now for a given time-domain HRTF equalizer $F(z)$, our design goal here is to derive the T/F domain filter $S_1(z),\ldots,S_M(z)$ so that their aggregated amplitude and phase responses are as close to those of $F(z)$ as possible. To achieve this goal, we decompose the subband equalizer $S_k(z)$ into a cascade of three filters as follows:

1. Amplitude filter $A_k(z)$: The purpose of $A_k(z)$ is to ensure the composite amplitude response of the subband filter is as close as possible to the amplitude response of the target time-domain filter $F(z)$.
2. Delay filter $D_k(z)$: $D_k(z)$ is a fractional delay filter used to model the phase response of the time domain filter by assuming a linear phase response of $F(z)$ at each subband.
3. Phase filter $P_k(z)$: Phase filter $P_k(z)$ is to ensure continuous phase response over subband boundary to avoid undesirable "signal cancellation effects" when the subband signal are synthesized at the synthesis filter.

The detailed design methods for these filters are given in the subsequent subsections. To facilitate our derivation, we assume the filterbank is the odd-stack complex QMF filterbank [11] used in HE-AAC and MPEG surround, which is in turn the main application of the T/F domain HRTF equalization process proposed here.

2.1 Design of Amplitude Filter

For a given time domain input signal $X(z)$, the ouput signal $Y(z)$ from the subband equalization system in left-hand side of Fig. 4 is given by:

$$Y(z) = \frac{1}{M}\mathbf{x}^T(z)\mathbf{H}_{AC}(z)\mathbf{g}(z), \tag{1}$$

where,

$$\mathbf{x}^T(z) = \left[X(z), X(zW), \ldots, X\left(zW^{M-1}\right)\right], \tag{2}$$

$$\mathbf{H}_{AC}(z) = \begin{bmatrix} H_1(z) & \cdots & H_M(z) \\ H_1(zW) & \cdots & H_M(zW) \\ \vdots & \ddots & \vdots \\ H_1\left(zW^{M-1}\right) & \cdots & H_M\left(zW^{M-1}\right) \end{bmatrix}, \tag{3}$$

$$\mathbf{g}^T(z) = \left[G_1(z)\cdot S_1\left(z^M\right), \ldots G_M(z)\cdot S_M\left(z^M\right)\right]. \tag{4}$$

Here $W_M \triangleq \exp(j\pi/M)$, and the z^M in (4) follows from the noble identities for multi-rate system. For the complex QMF filterbank considering here the aliasing term in $\mathrm{H}_{AC}(z)\cdot\mathbf{g}(z)$ is negligible [11] [12]. Thus we have

$$\mathrm{H}_{AC}(z)\cdot\mathbf{g}(z) = \left[T(z), 0, \ldots, 0\right]^T \tag{5}$$

where

$$T(z) = \sum_{k=1}^{M} H_k(z) S_k\left(z^M\right) G_k(z). \tag{6}$$

Combining (5), (6) and (1) gives us

$$Y(z) = \sum_{k=1}^{M} H_k(z) S_k\left(z^M\right) G_k(z) X(z). \tag{7}$$

Similarly, for the time-domain equalization system as shown in the right hand side of Fig. 4, for the same input signal $X(z)$ the output signal is given by:

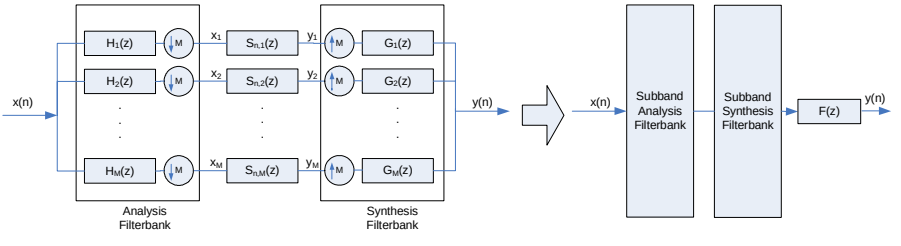


Fig. 4. T/F domain equalization vs. time domain equalization

$$Y'(z) = \sum_{k=1}^{M} H_k(z) G_k(z) F(z) X(z). \tag{8}$$

Now we denote the transfer function of the aggregated T/F domain HRTF equalizer as $T(z)$:

$$T(z) \triangleq \sum_{k=1}^{M} H_k(z) S_k(z^M) G_k(z) \tag{9}$$

and the transfer function of the time-domain HRTF equalizer as $T'(z)$:

$$T'(z) = \sum_{k=1}^{M} H_k(z) G_k(z) F(z) \tag{10}$$

Comparing (7) with (8) we have the following condition for the ideal $S_k(z)$

$$T(z) = T'(z), \tag{11}$$

To simplify the problem, we now only consider the elements in (11) that have significant energy. For odd-stack filterbank, at frequencies near subband boundary $k\pi/M$ only the contributions from subband k and k+1 have significant energy. Thus (11) can be further rewritten to:

$$H_k(\omega) S_k(M\omega) G_k(\omega) + H_{k+1}(\omega) S_{k+1}(M\omega) G_{k+1}(\omega) = T'(\omega) \tag{12}$$

Here the frequency response of each filter at frequency ω is obtained by replacing z in the transfer function with $z = e^{j\omega}$. Now, assuming both $D_k(z)$ and $P_k(z)$ are of unit amplitude response, and the phase response of the composite filters $S_k(z) = A_k(z) D_k(z) P_k(z)$ are phase aligned at subband boundary, (12) can be translated into the following equations for the amplitude filter $A_k(z)$:

$$\left|F_1(\Delta\omega) H_1(\Delta\omega)\right| \left|A_1(\Delta\omega)\right| = \left|T'(\Delta\omega)\right| \tag{13}$$

$$\begin{cases} \left|F_{2k-1}(W_M^{2k-1} - \Delta\omega) H_{2k-1}(W_M^{2K-1} - \Delta\omega)\right| \left|A_{2k-1}(\pi - M\Delta\omega)\right| + \left|F_{2k}(W_M^{2k-1} - \Delta\omega) H_{2k}(W_M^{2K-1} - \Delta\omega)\right| \left|A_{2k}(\pi - M\Delta\omega)\right| = \left|T'(W_M^{2k-1} - \Delta\omega)\right| \\ \left|F_{2k-1}(W_M^{2k-1} + \Delta\omega) H_{2k-1}(W_M^{2k-1} + \Delta\omega)\right| \left|A_{2k-1}(\pi - M\Delta\omega)\right| + \left|F_{2k}(W_M^{2k-1} + \Delta\omega) H_{2k}(W_M^{2k-1} + \Delta\omega)\right| \left|A_{2k}(\pi - M\Delta\omega)\right| = \left|T'(W_M^{2k-1} + \Delta\omega)\right| \end{cases}$$

for $k = 1, 2, \ldots, \frac{M}{2}$, (14)

$$\begin{cases} \left|F_{2k}(W_M^{2k} - \Delta\omega) H_{2k}(W_M^{2k} - \Delta\omega)\right| \left|A_{2k}(M\Delta\omega)\right| + \left|F_{2k+1}(W_M^{2k} - \Delta\omega) H_{2k+1}(W_M^{2k} - \Delta\omega)\right| \left|A_{2k+1}(M\Delta\omega)\right| = \left|T'(W_M^{2k} - \Delta\omega)\right| \\ \left|F_{2k}(W_M^{2k} + \Delta\omega) H_{2k}(W_M^{2k} + \Delta\omega)\right| \left|A_{2k}(M\Delta\omega)\right| + \left|F_{2k+1}(W_M^{2k} + \Delta\omega) H_{2k+1}(W_M^{2k} + \Delta\omega)\right| \left|A_{2k+1}(M\Delta\omega)\right| = \left|T'(W_M^{2k} + \Delta\omega)\right| \end{cases}$$

for $k = 1, 2, \ldots \frac{M}{2} - 1$, (15)

$$\left|F_M(\pi - \Delta\omega) H_M(\pi - \Delta\omega)\right| \left|A_M(\pi - M\Delta\omega)\right| = \left|T'(\pi - \Delta\omega)\right|. \tag{16}$$

Here $\Delta\omega \in [0, \pi/2M)$ and $W_M^k \triangleq k\pi/M$. Solving the above equations for a discrete set of $\Delta\omega_i \in [0, \pi/2M)$ will thus give us a set of the desirable amplitude response $A_k(\omega_i)$ at frequency $\omega_i = M\Delta\omega_i$ and $\omega_i = \pi - M\Delta\omega_i$, which can then be used as the amplitude response specification for designing $A_k(z)$. Detailed algorithms for designing FIR filter given the amplitude response specification can be found in [13].

The orders of the amplitude filters $A_k(z)$ for different subbands can be selected according to the accuracy requirement. Generally speaking, the amplitude response of $F(z)$ can be modeled more accurately by using larger order $A_k(z)$, while at the cost of increasing the overall complexity. It is also well-known that human ears don't perceive sounds of different frequencies with equal sensitivity. For example, the human ear has better frequency resolution at low frequencies compared with that at high frequencies. For this reason, lower order $A_k(z)$ should be used at higher frequency bands. In fact, it is found in our experiments that for subbands above 2kHz, using a zero order $A_k(z)$, i.e., a scalar, is sufficient to model the amplitude response of a short time-domain HRTF without significant room reverberation.

2.2 Design of Fractional Delay Filter

The phase response of $F(z)$ is approximated by a piece-wise linear function in our design by assuming the phase response is linear within each subband. In such a case, the phase response can be modeled by simply using a delay filter in each subband. Since for most the subband audio coding system, the subband signal is down-sampled in order to have a compact representation of the signal, the time-resolution in the T/F domain is very limited. As a result, fractional delay (FD) filter is in general necessary in order to have an accurate model for the phase response.

Theoretically, an ideal FD filter that provides constant fractional delays over all frequencies has an impulse response with infinite length and hence it is not practical. In most practical fractional delay filter designs [14], compromise is mode so that near constant fractional delay is achieved for only a certain frequency range $[-\omega_0, \omega_0]$ where $\omega_0 < \pi$. However, the distortion in delay at frequencies near the Nyquist frequency π will be very large. Although this is generally not a big issue for conventional full rate systems, where the high frequencies near the Nyquiest frequency are in most cases perceptually not important, for the subband equalization system the Nyquist frequency in the T/F domain is in fact corresponding to the subband boundaries at time-domain, which are normally perceptually relevant. For this reason, those FD filter design techniques are not directly applicable here.

To solve this problem, we notice that if the real-coefficient FD-filter with a near constant fractional delay range $[-\omega_0, \omega_0]$ is modulated with a complex sinusoidal $s(n) = e^{jn\theta}$, the near constant fractional delay will be shifted to $[-\omega_0 + \theta, \omega_0 + \theta]$ after the modulation. As an example, Fig. 5a) shows the group delay of a real- coefficient

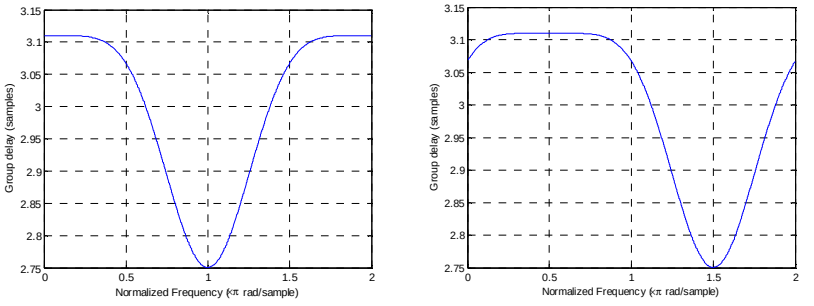


Fig. 5. a) Group delay of a real-coefficient FIR FD-filter; **b)** Group delay of the same FIR FD-filter modulated by complex sinusoidal $s(n) = e^{jn\pi/2}$

5-order FIR FD filter, which has a near constant fractional delay at $[-\pi/2, \pi/2)$. In Fig. 5b) group delay of the same FD filter however modulated by a complex sinusoid $s(n) = e^{jn\pi/2}$ is shown. Clearly the group delay response has been shifted by $\pi/2$ and the near constant delay range now overlaps frequency $[0, \pi)$.

In the context of T/F domain equalization, for each subband, we wish the near constant fractional delay covers the frequency range that has significant energy after the subband synthesis filter. For odd-stack filterbank, this is corresponding to frequency range $[(k-1)\pi, k\pi)$ for subband *k*. In the downsampled T/F domain this frequency range is mapped to $[0, \pi)$ for odd number subbands $k = 1, 3, 5, \ldots$, and $[-\pi, 0)$ for even number subbands $k = 2, 4, 6, \ldots$. Consequently, the desirable delay filters can be obtained by modulating a prototype real-coefficient FD-filter with complex sinusoidals of frequency $\pi/2$ or $-\pi/2$, respectively for odd and even number subbands.

In practical implementation, the fractional delay filter is used only for subbands blow 1.5kHz for complexity reduction. For subbands of higher frequencies it is replaced with an integer delay filter. It is found that this simplification only introduces negligible degradation in perceptual quality since human ears are insensitive to ITD at high frequencies.

2.3 Design of Phase Filter

The phase filter $P_k(z) = e^{j\varphi_k}$ is used to make ensure the alignment of the overall phase response of the combined subband filter $H_k(z)S_k(z)G_k(z)$ at subband boundaries to avoid undesirable signal cancellation when signals from different subbands are summed up to produce the time-domain waveform. For this purpose the phase correction angle φ_k is selected so that the overall phase response $\phi_k(\omega)$ of $H_k(z)S_k(z)G_k(z)$ satisfies $\phi_k(k\pi/M) = \phi_{k+1}(k\pi/M)$, $k = 1, \ldots M-1$.

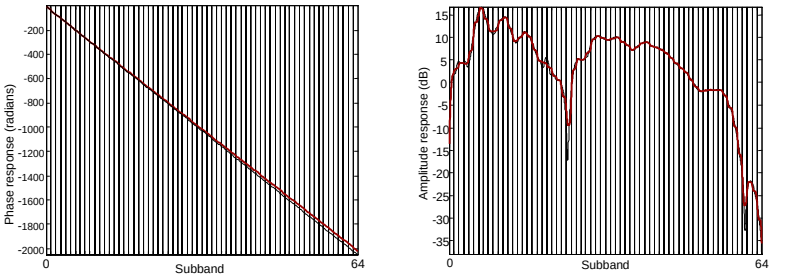


Fig. 6. Phase and amlitude response of the aggregated subband filter fors KEMAR. The thin-back lines in the above figures indicate the responses of the original KEMAR equalizer. The bold-red lines indicate the responses of the aggregated T/F domain KEMAR equalizer.

3 Experimental Results

We use the KEMAR HRTF available from [15] as an example to evaluate the performance of the proposed T/F domain HRTF equalizer. The impulse response of the KEMAR HRTF used in our test has a impulse response of 128 taps. In our test, we implement the KEMAR HRTF in the T/F domain of the MPEG surround filterbank by using the design procedure described in previous section. The orders of the amplitude filters are four for the first subband, and zero for the rest. The fractional delay filter is constructed by modulating a fifth-order Lagrange filter [14] with complex sinusoidal, and is used for subbands up to 1.5kHz. The T/F domain equalizer thus has a much shorter impulse response and consequently lower complexity compared with its time domain counter part. Despite its simplicity, the T/F domain equalizer still provides reasonable accuracy in modeling the amplitude and phase response of the KEMAR HRTF, which are shown in Fig. 6 in comparison to those of the time-domain KEMAR equalizer. Clearly, the T/F domain implementation provides an amplitude response that closely follows that of its time-domain counterpart. As to the phase response, accuracy modeling is also achieves for frequencies < 1.5kHz where the human ears are sensitive in detecting ITD. For frequencies above 1.5kHz, the phase response only approximates that of the time-domain equalizer since the fractional delay filter is not used for those subbands.

We further integrated the T/F-domain KEMAR HRTFs into a multi-channel MPEG surround decoder, and compared its performance with the brute-force approach by listening test. The bit-streams used in our test were generated by an MPEG-surround encoder running in 5-1-5 configuration, and have a bit-rate of 48 kbps. Totally eleven 5.1 multi-channel programs, which were previously used in the MPEG surround standardization process [16], were used in our test. The listening test was conducted in a quite listening room with STAX headphones. Six listeners participated in our listening test, and all of them are researchers working in audio coding or acoustics and hence they can be considered as “expert” listeners. Binaural signals generated from the original 5.1 programs with the KEMAR HRTF and their 3.5 kHz low-pass version were used in our test as the reference and anchor signals respectively. The results of the listening tests are given in Fig. 7 in MUSHRA score with 95% confidence interval. From the test results, it can be seen that the MUSHRA

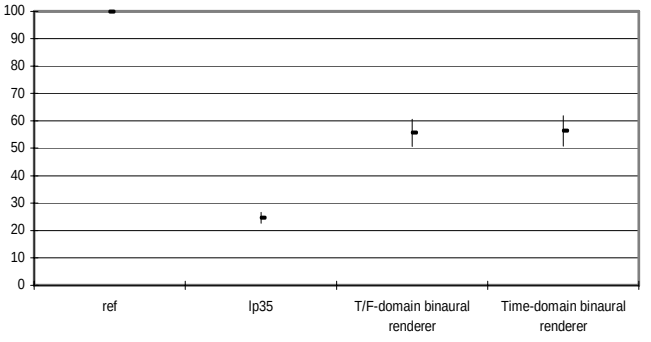


Fig. 7. Listening Test Results

scores for these two binaural decoders are in fact statistically indistinguishable. In other words, despite its much lower complexity, the T/F-domain HRTF approach achieves the same perceptual quality as that of the time-domain approach which is very desirable.

4 Conclusion

In this paper a low-complexity T/F domain HRTF equalizer is proposed and its detailed design method is described. The proposed HRTF equalizer is constructed by using a cascade of three filters that include an amplitude filter, a fractional-delay filter and a phase filter, and has amplitude and phase responses that closely resemble those of the corresponding time-domain HRTF equalizer. It is shown in our test that the proposed T/F domain HRTF equalizer achieves a similar perceptual quality compared to the computational more expensive time-domain brute-force implementation when it is integrated with an MPEG surround decoder. Therefore, it is preferable to use the proposed algorithm in combination with multi-channel audio decoding algorithms, such as multi-channel HE-AAC or MPEG surround, for low-complexity binaural decoding.

References

1. Steve Vernon, “Dolby Digital: Audio Coding for Digital Television and Storage Applications,” AES 17th Conference, Florence, Italy, 1999.
2. K. Brandenburg, G. Stoll, “ISO/MPEG-1 Audio: A Generic Standard for Coding of High-Quality Digital Audio,” JAES Volume 42 Number 10 pp. 780-792; October 1994 .
3. B. Grill, “The MPEG-4 General Audio Coder,” The AES 17th International Conference, August, 1999.
4. M. Wolters, K. Kjorling, D. Homm, H. Purnhagen, “A closer look into MPEG-4 High Efficiency AAC,” 115th AES convention, New York, USA, 2003.
5. F. Baumgarte, C. Faller, “Design and Evaluation of Binaural Cue Coding Schemes,” AES Convention Paper 5706, Oct. 2002, Los Angeles CA, USA.

6. J. Breebaart *et al*, "MPEG Spatial Audio Coding / MPEG Surround: Overview and Current Status," AES convention paper 6599, 119th AES Convention, October, 2005.
7. C. A. Lanciani and R. W. Schafer, "Subband-Domain Filtering of MPEG Audio Signals," Proceeding of IEEE International Conference on Acoustic, Speech, Signal Processing, 1999.
8. C. A. Lanciani and R. W. Schafer, "Application of Head-Related Transfer Functions to MPEG Audio Signals," Proceeding of 31st Symposium on System theory, March 21-23, 1999.
9. A. B. Touimi, "A Generic Framework for Filtering in Subband Domain," Proceeding of IEEE 9th Workshop on Digital Signal Processing, Hunt, Texas, USA, October 2000.
10. A. B. Touimi, M. Emerit and J.-M. Pernaux, "Efficient Method for Multiple Compressed Audio Streams Spatialization," 3rd international conference on Mobile and ubiquitous multimedia, 2004.
11. P. Ekstrand, "Bandwidth Extension of Audio Signals by Spectral Band Replication," IEEE Benelux Workshop on Model based Processing and Coding of Audio, Leuven, Belgium, November 15, 2002.
12. Shimada, Osamu et al., "A Low Power SBR Algorithm for the MPEG-4 Audio Standard and Its DSP Implementation," preprint 6048, 116 AES Convention, May 2004.
13. Parks, T. W., and C.S. Burrus, Digital Filter Design, John Wiley & Sons, New York:, 1987.
14. T. I. Laakso et. al., "Splitting the Unit Delay – Tools for fractional delay filter design," IEEE Signal Processing Magazine, pp. 30 – 60, Jan. 1996.
15. B. Gardner, K. Martin, "HRTF Measurements of a KEMAR Dummy-Head Microphone," Technical Report #280, MIT Media Lab Perceptual Computing, May, 1994.
16. ISO/IEC JTC 1/SC 29/WG 11, N6691, Workplan for the Evaluation of Spatial Audio Coding Responses, July 2004.

Complexity Reduction of Multi-frame Motion Estimation in H.264

Linjian Mo, Jiajun Bu, Chun Chen, Zhi Yang, and Yi Liu*

College of Computer Science, Zhejiang University, Hangzhou 310027, China
{molin,bjj,chenc,yangzh,yiliu}@zju.edu.cn

Abstract. H.264 video coding standard gains significant improvement in compression efficiency. However, the computational complexity also considerably increases. Many fast algorithms are introduced to speed up the encoding. Distinguishing from pervious multi-frame based fast motion estimation algorithms which only exploit temporal correlation, a novel algorithm both employing temporal and spatial correlations is proposed in this paper, which has much better prediction accuracy. To further speed up the encoding process, an adaptive mode reduction scheme which will discard small probability modes is also proposed. Combining the novel fast multi-frame based motion estimation algorithm and the adaptive mode reduction scheme, experimental results show that the proposed algorithms improve the encoding speed by an average of 9 times faster than fast full search and about 2.5 times faster than reference method which only employs temporal correlation, PSNR dropping and bitrate increasing are negligible.

1 Introduction

Video coding plays an important role in multimedia communication applications. H.264, the latest video coding standard, achieves significant improvement in compression efficiency than previous standards. However, H.264 has much higher computational complexity due to the use of many new features. One of them is multi-frame based variable block-size motion estimation (ME).

It's well known that ME is introduced to exploit the temporal redundancy between frames. Compared to previous coding standards, H.264 supports more flexibility in the selection of motion compensation block sizes (P16x16, P16x8, P8x16, P8x8, P8x4, P4x8, and P4x4). Moreover, the new standard allows ME to select the reference frame among a large number of previous stored pictures. It provides the best coding result but followed with linear increasing processing time. According to statistics, in H.264 reference software, multi-frame based variable block-size ME contributes about 70% of total computational load.

Many efforts have been made to explore the fast algorithms of ME for H.264. Some researchers try to reduce the search range, such as the early-stop technique, which early terminate the searching when some conditions are satisfied

* Project supported by National Basic Research Program of China (2006CB303000).

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 557–567, 2007.
© Springer-Verlag Berlin Heidelberg 2007

[1]. Others address this problem through a different approach, which named information reuse. Up-layer Prediction (ULP) method is introduced to reuse the up-layer MV in the quad-tree structure [1,2], which exploits the spatial correlation. Those works significantly speed up the ME processing. However, most of them were based on one reference frame rather than multi-frame, which can provide much better prediction results and will lead to higher coding efficiency.

Multi-frame based ME of macroblock requires deriving a set of motion vectors (MV) referring to each reference frame. There are high temporal correlations among these MVs. Utilizing the correlations, Youn et al. [3] propose the Forward Dominant Vector Selection algorithm (FDVS), which reuses the sum of existing MVs to generate a new MV referring to previous nonadjacent frames without performing ME. Adaptive variable block size activity dominant vector selection (VADVS) method [4] proposed by Chen et al. further improved the performance by using more flexible scheme and different dominant MV selection mechanism. Based on those MV reusing algorithm, many fast multi-frame ME methods are proposed [4,5]. The previous works of fast multi-frame based ME only employ temporal correlation to get the predictive MV (PMV) referring to previous nonadjacent frames. However, according to our experimental results, ULP by exploiting the spatial correlation, have better accuracy than that using temporal correlation, such as FDVS, VADVS etc. In other words, the former will consume less computational resource on the motion refinement stage.

Both employing temporal prediction method, FDVS, which can obtain the PMV on previous nonadjacent reference frames, and spatial prediction method, ULP, which has better prediction accuracy, a novel efficient multi-frame motion estimation algorithm in H.264 is proposed. The algorithm consists of three steps. Firstly, P16x16, P8x8 and Intra modes (I16x16 and I4x4) are full search for each macroblock and Rate-Distortion cost (RDC) will be calculated. According to these RDCs, small probability modes will be discarded. Secondly, temporal correlation is exploited. We adopt FDVS algorithm to achieve the MVs of P16x16 and P8x8 modes on each reference frames. Finally, utilizing the spatial correlation, ULP is performed to get the MVs of other modes on each reference frames.

The rest of the paper is organized as follow. In section 2 we analyze the statistics of modes reduction and performance of FDVS and ULP methods. Section 3 describes the detail of the proposed algorithm and then we give the experimental results in section 4. We conclude this paper in section 5.

2 Statistical Analysis

2.1 Reduction of the Potential Modes

In the reference software of H.264, ME is performed mode by mode to get the best coding efficient. Sometimes the coding efficient gain by searching more modes is very significant, but usually much computation is wasted without any benefit. Hence, we will discard some probability modes to reduce the computational load.

It's well known that high texture blocks are tended to be encoded by Small Size Modes (SSM), which are P4x4, P4x8, P8x4 and P8x8; and low texture blocks

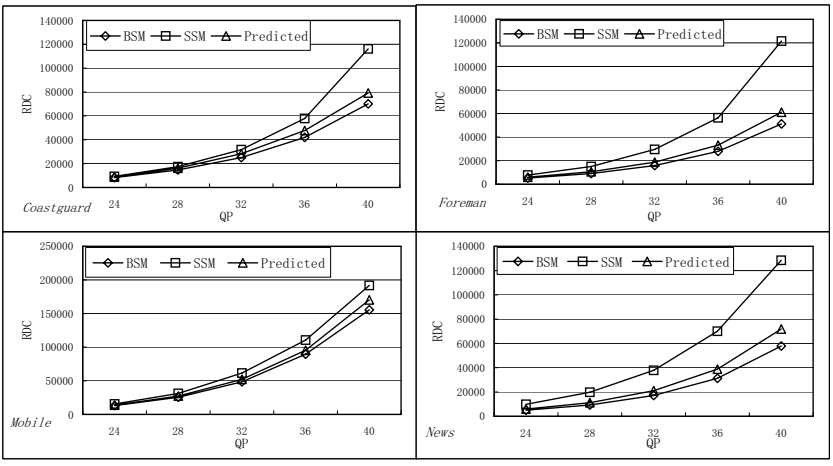


Fig. 1. Average RDC(I4x4) of BSM and SSM blocks and predicted threshold

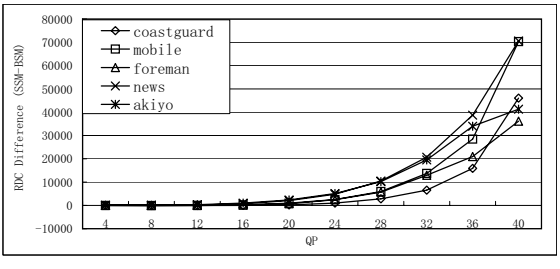


Fig. 2. RDC(I4x4) difference vs QP

prefer Big Size Modes (BSM), such as P16x16, P16x8 and P8x16. For simplicity, we use BSM block to denote a macroblock with the best mode appearing in BSM, and SSM block to represent that appearing in SSM. If we can predict the texture complexity for a block, and only use high probability modes (BSM or SSM) for ME, significant improvement of encoding time will be achieved.

In Table 2, the correlation of $RDC(P16\times16)$, $RDC(P8\times8)$ and best mode is analyzed. We find that when $RDC(P16\times16)$ is less than $RDC(P8\times8)$, almost all best mode will appear in BSM; when $RDC(P8\times8)$ is less, it's hard to decide. Here we introduce $RDC(I4\times4)$ to help the classification.

The average $RDC(I4\times4)$ of BSM and SSM blocks are figured in Fig.1 and the difference is in Fig.2. We find that when $QP \geq 24$, $RDC(I4\times4)$ of BSM block is much smaller than that of SSM block. A desirable threshold can be used to discriminate between BSM and SSM blocks. However, different sequences, even the same sequences with different QPs have different thresholds. Therefore, an adaptive threshold is required. We use the average value to denote the optimal threshold, which is:

$$T = 0.5 \times RDC(I4 \times 4\ of\ BSM) + 0.5 \times RDC(I4 \times 4\ of\ SSM) \quad (1)$$

where T means the threshold to discriminate between BSM and SSM blocks, $RDC(I4 \times 4\ of\ BSM)$ and $RDC(I4 \times 4\ of\ SSM)$ are average $RDC(I4 \times 4)$ of BSM and SSM blocks, respectively. Unfortunately, $RDC(I4 \times 4\ of\ BSM)$ and $RDC(I4 \times 4\ of\ SSM)$ can't be directly achieved before encoding.

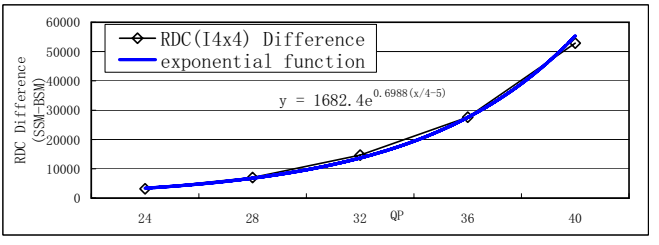


Fig. 3. Fitting for average RDC(I4x4) difference ($QP \geq 24$)

Further analysis shows that given a fixed QP value, $RDC(I4 \times 4)$ difference of BSM and SSM blocks alters following QP. An exponential function is used to fit the average differences($QP \geq 24$), as Fig.3. The analytical expression is:

$$RDC(I4 \times 4\ of\ SSM) - RDC(I4 \times 4\ of\ BSM) = 1682.4 \times e^{0.6988 \cdot (QP/4-5)} \quad (2)$$

Moreover, average $RDC(I4 \times 4)$ of frames on one IPPP... group is similar, which means we can use the average $RDC(I4 \times 4)$ of I-frame to predict that of following P-frames. Meanwhile, average $RDC(I4 \times 4)$ of a P-frame is the weighted average of $RDC(I4 \times 4\ of\ BSM)$ and $RDC(I4 \times 4\ of\ SSM)$, and the weighs are the percentages of their number. Hence we have:

$$RDC(I4 \times 4\ of\ I-frame) = \alpha \cdot RDC(I4 \times 4\ of\ BSM) + \beta \cdot RDC(I4 \times 4\ of\ SSM) \quad (3)$$

where $RDC(I4 \times 4\ of\ I-frame)$ means average $RDC(I4 \times 4)$ of I-frame and α, β denotes the percentages of BSM and SSM blocks, respectively. Experimental result shows that usually BSM blocks take up about 70% of the total macroblock, therefore $\alpha = 0.7$ and $\beta = 0.3$.

According to (1), (2) and (3), the adaptive threshold T is:

$$T = RDC(I4 \times 4\ of\ I-frame) + 0.2 \times 1682.4 \times e^{0.6988 \cdot (QP/4-5)} \quad (4)$$

where $RDC(I4 \times 4\ of\ I-frame)$ can be achieved after the I-frame coding.

The performance of adaptive threshold selection algorithm can be found in Fig.1. Experimental results show that it is close to the optimal threshold.

It's should be mentioned that when QP less than 24, $RDC(I4 \times 4\ of\ BSM)$ and $RDC(I4 \times 4\ of\ SSM)$ are very close, even the latter will less than the former. To ensure the predicting result, we restricted the condition: $QP \geq 24$.

Summarily, the mode reduction will be performed under following criterions:

Criterion 1: If $RDC(P16 \times 16) < RDC(P8 \times 8)$, SSM will be discarded.
Criterion 2: If $RDC(P8 \times 8) \leq RDC(P16 \times 16)$ and $RDC(I4 \times 4) > T$ and $QP \geq 24$, BSM will be discarded.

2.2 Performance Comparison of Temporal and Spatial Prediction

In H.264, ME algorithm with multiple reference frames will check all potential modes on each reference frame respectively. Usually, there is high correlation existed in the successive frames and the adjacent macroblocks.

Thus, exploiting the temporal correlation, some fast algorithm can be developed to accelerate the ME. FDVS is a MV composition scheme, which is used to fast get the MVs referring to previous nonadjacent reference frames. FDVS selects the dominant MV from the neighboring macroblocks. These dominant MVs are added to the current MV to compose the target MV referring to target reference frame. MV associated with the largest overlapping region is regards as the dominant MV.

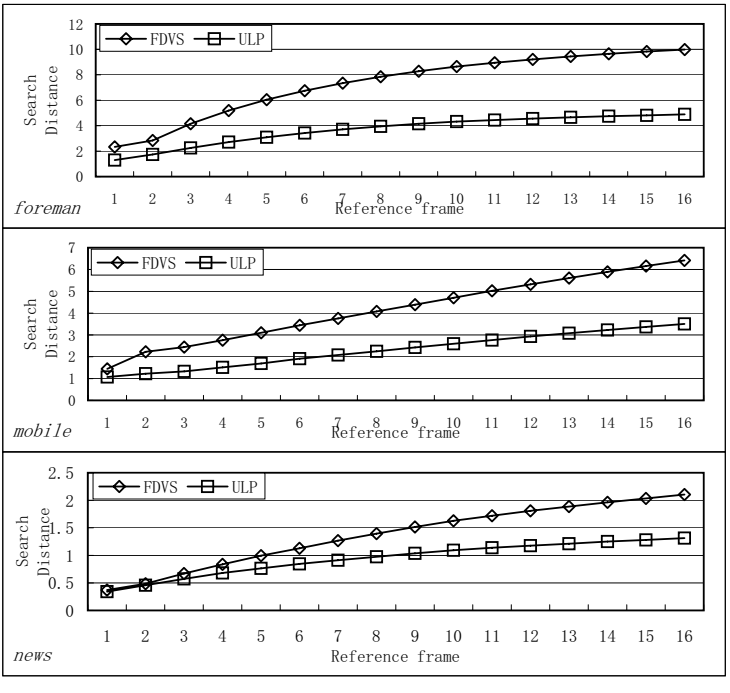


Fig. 4. Search distance comparison of FDVS and ULP

However, temporal prediction methods do not always perform best. Spatial prediction methods, such as ULP may have better prediction accuracy. ULP will directly takes the MV of up-layer as the PMV for low-layer blocks, such as mode 1 for mode 2 and 3, mode 4 for mode 5 and 6, mode 5 or 6 for mode 7.

$$PMV_{mx} = \begin{cases} MV_{m1} & , x = 2,3 \\ MV_{m4} & , x = 5,6 \\ (MV_{m5} + MV_{m6})/2 & , x = 7 \end{cases} \tag{5}$$

where PMV_{mx} means the PMV for mode x and MV_{mx} denotes the actual MV of mode x. It can be easily seen that compared to FDVS, the computational complexity is much lower.

To compare the prediction accuracy of FDVS and ULP, full search is performed to find the best MV, MV_{fs}, and then calculate the distances between it and two PMVs predicted by FDVS and ULP, which are PMV_{fdvs} and PMV_{ulp}. The distance is calculated by:

$$Dist(MV1, MV2) = |MV1_x - MV2_x| + |MV1_y - MV2_y| \tag{6}$$

where $Dist(MV1, MV2)$ is the distance of $MV1$ and $MV2$; $MV1_x$, $MV1_y$, $MV2_x$, $MV2_y$ denote the x and y components of $MV1$ and $MV2$, respectively.

Hence, $Dist(MV_{fs}, PMV)$ can be used to measure the search distance of a prediction method, which reflects the computational complexity. Search distances of FDVS and ULP are figured on Fig.4. It's should be mentioned that mode 1 is omitted, because ULP can not be performed on mode 1.

It can be seen that usually ULP is better than FDVS on prediction accuracy, especially with big reference frame number.

3 Proposed Algorithms

3.1 Mode Reduction Scheme

The adaptive mode reduction scheme can be performed as following steps:

1. For each IPPP... group, encoding and calculating the average $RDC(I4 \times 4)$ of each macroblock on I-frame.
2. Calculate the threshold T according to (4).
3. For each macroblock on following P-frame, discard the small probability modes with the criterions proposed in section 2.1.

This mode reduction scheme can be easily combined with other fast algorithm to achieve better performance.

3.2 Fast Multi-frame Based ME Algorithm

The main idea of the fast ME algorithm is the method of MV reusing by both exploiting the temporal and spatial correlation. P16x16 and P8x8 modes will be full searched in the first reference frame (REF_0). With the temporal predictive MV reusing algorithm and small range motion refinement scheme, the MVs of these two modes on other reference frames will be achieved. Finally, on each reference frame, the MVs of P16x16 or P8x8 modes blocks will be regarded as the search center for rest modes to finish the ME process.

With those PMVs, only small range motion refinement is needed. According to the experimental results, for simplification, we directly set search ranges of temporal predicted modes (1 and 4) to be 1, 2, 3, ..., 16 for reference frames REF_0, REF_1, REF_2,..., REF_15, respectively. For spatial predicted modes

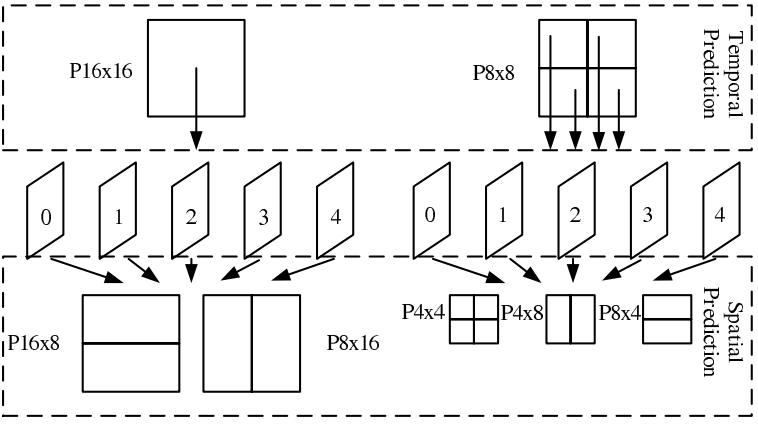


Fig. 5. Illustration of proposed temporal and spatial predictive ME

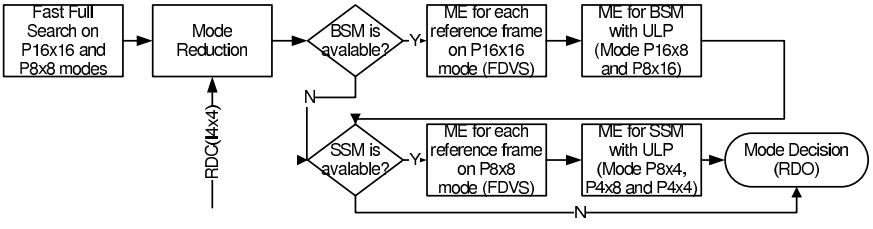


Fig. 6. Flowchart of the proposed ME algorithm and mode reduction scheme

(2, 3, 5, 6, and 7), search range is set to be half of that on each reference frame. The illustration of the predictive ME algorithm is in Fig.5.

Combining the proposed adaptive mode reduction scheme and the fast multi-frame based ME algorithm, the entire flowchart is in Fig.6.

4 Experimental Results

To evaluate the performance of the proposed mode reduction scheme and fast multi-frame based ME algorithm, we implement five methods, which are:

FFS: Fast Full search ME.
FDVS: MVs of all modes are predicted by FDVS on each reference frame.
Prop: MVs of P16x16 and P8x8 are predicted by FDVS, others are by ULP.
FDVS+MR: FDVS method plus proposed mode reduction scheme.
Prop+MR: Prop method plus proposed mode reduction scheme.

Four sequences are selected to be encoded by JM8.6 baseline profile, with 100 frames and frame rate is 30fps, reference frame number is 5. The hardware platform is P IV 2.4 CPU with 512M memory.

Table 1. Performance comparison of *FFS*, *FDVS+MR*, *Prop+MR*, *FDVS*, and *Prop*

QP		FFS	FDVS+MR	Prop+MR	FDVS	Prop
			coastguard			
24	ME Time(ms)	194051	74669	28574	125178	46007
	PSNR(dB)	37.484	37.415	37.417	37.479	37.48
	Bitrate(bps)	2286082	2290447	2287224	2293068	2290671
28	ME time(ms)	199424	71507	26257	120561	45601
	PSNR(dB)	34.442	34.387	34.39	34.433	34.429
	Bitrate(bps)	1328827	1334064	1335007	1333440	1331834
32	ME time(ms)	205583	69667	27119	115914	46032
	PSNR(dB)	31.391	31.374	31.363	31.386	31.387
	Bitrate(bps)	652279	656623	657158	655685	656162
36	ME time(ms)	208151	68282	27160	106632	44208
	PSNR(dB)	28.764	28.721	28.718	28.757	28.727
	Bitrate(bps)	277306	279698	281592	279389	279830
40	ME time(ms)	213961	64062	25424	100861	41083
	PSNR(dB)	26.544	26.507	26.463	26.514	26.474
	Bitrate(bps)	117154	117425	117763	117113	116890
			foreman			
24	ME Time(ms)	197696	61841	25259	97618	39985
	PSNR(dB)	39.25	39.208	39.215	39.233	39.245
	Bitrate(bps)	683942	702394	701213	693082	692340
28	ME time(ms)	202321	54648	24062	91281	38622
	PSNR(dB)	36.871	36.836	36.824	36.844	36.857
	Bitrate(bps)	382301	392743	394490	387725	387216
32	ME time(ms)	203184	51082	22732	83887	36629
	PSNR(dB)	34.496	34.469	34.437	34.468	34.44
	Bitrate(bps)	224170	230882	232303	226404	228506
36	ME time(ms)	206582	47233	20714	76500	34679
	PSNR(dB)	32.244	32.184	32.178	32.223	32.186
	Bitrate(bps)	139171	142788	143292	140045	141274
40	ME time(ms)	209760	41509	20813	69223	32867
	PSNR(dB)	30.004	29.964	29.887	29.933	29.88
	Bitrate(bps)	89837	91721	91502	90408	90518
			mobile			
24	ME Time(ms)	190592	70623	25314	114999	39019
	PSNR(dB)	37.501	37.422	37.431	37.481	37.497
	Bitrate(bps)	2947531	2995380	2975050	2976795	2958744
28	ME time(ms)	192341	65152	24178	110205	38863
	PSNR(dB)	34.233	34.162	34.174	34.217	34.222
	Bitrate(bps)	1672903	1712642	1699296	1699572	1682098
32	ME time(ms)	194947	61779	23587	105537	36729
	PSNR(dB)	30.878	30.798	30.81	30.849	30.873
	Bitrate(bps)	806045	830287	821371	820800	814685
36	ME time(ms)	197093	56088	21921	105319	37512
	PSNR(dB)	27.931	27.867	27.865	27.909	27.911
	Bitrate(bps)	376423	387583	382790	383969	380376
40	ME time(ms)	201479	55838	21870	97473	38128
	PSNR(dB)	25.132	25.07	25.078	25.098	25.101
	Bitrate(bps)	203688	206009	205274	205301	204314

Table 1. (*continued*)

QP		FFS	FDVS+MR	Prop+MR	FDVS	Prop
			news			
24	ME Time(ms)	195442	43495	18845	71106	29318
	PSNR(dB)	40.736	40.679	40.693	40.725	40.727
	Bitrate(bps)	351497	356801	357480	352514	352418
28	ME time(ms)	195812	38780	17788	65746	27337
	PSNR(dB)	38.141	38.123	38.123	38.138	38.131
	Bitrate(bps)	211445	213302	215544	211867	212383
32	ME time(ms)	200897	34636	16257	61235	24928
	PSNR(dB)	35.431	35.393	35.384	35.412	35.415
	Bitrate(bps)	128669	130140	131184	128798	129307
36	ME time(ms)	203494	31509	15504	58489	24105
	PSNR(dB)	32.751	32.745	32.708	32.749	32.748
	Bitrate(bps)	78451	80119	80566	78530	79788
40	ME time(ms)	207505	30189	14088	55175	23100
	PSNR(dB)	30.155	30.104	30.052	30.129	30.087
	Bitrate(bps)	48434	49378	49428	48394	48809
			AVERAGE			
	ME Time(ms)	201015.75	54629.45	22373.3	91646.95	36237.6
	PSNR(dB)	33.21895	33.1714	33.1605	33.19885	33.19085
	Bitrate(bps)	650307.75	660021.3	657976.35	656144.95	653908.15

We use Miss Rate and Failing Rate to measure the performance of the proposed mode reduction scheme. Miss Rate is defined as the percentage of macroblocks that not satisfying mode reduction criterions and Failing Rate denotes the percentage of wrong determined macroblocks. As Table 3, the average Miss Rate is 7.66% and average Failing Rate is 3.34%. Moreover, the scheme performs better when QP is larger. This is because that the number of BSM blocks will increase when QP increasing and more macroblocks will satisfy criterion 1. As Table 2, the failing rate of criterion 2 will be higher than that of criterion 1. Comparing the performance of *FDVS*, *FDVS+MR* and *Prop*, *Prop+MR* in Ta-

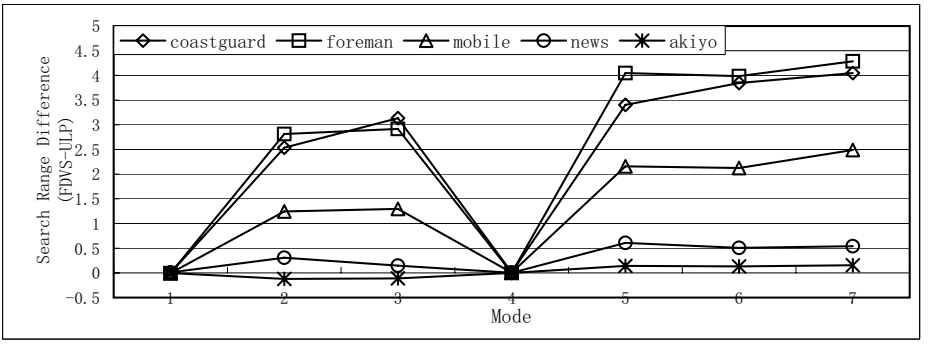


Fig. 7. Search distance difference of FDVS and ULP on each mode. (FDVS-ULP)

Table 2. Correlation of RDCs (P16x16 and P8x8) and best mode

Sequence	**Best Mode Appear in**	**Better RDC**	
		P16x16	P8x8
news	BSM	91.72%	3.02%
	SSM	0%	5.26%
foreman	BSM	78.03%	10.09%
	SSM	0%	11.88%
mobile	BSM	47.65%	14.09%
	SSM	0%	38.27%

Table 3. Performance of proposed mode reduction scheme

QP	sequences	coastguard	foreman	mobile	news	akiyo
24	Miss Rate(%)	29.15	17.95	25.02	4.01	0.72
	Failing Rate(%)	4.57	4.68	10.24	2.79	3.25
28	Miss Rate(%)	20.69	10.72	20.61	2.67	0.31
	Failing Rate(%)	4.37	3.97	10.71	2.30	2.15
32	Miss Rate(%)	13.33	5.93	14.55	1.68	0.14
	Failing Rate(%)	3.50	2.83	9.74	1.59	1.07
36	Miss Rate(%)	6.05	2.70	7.70	0.87	0.06
	Failing Rate(%)	2.38	1.61	5.38	1.03	0.35
40	Miss Rate(%)	1.07	1.18	3.85	0.48	0.03
	Failing Rate(%)	1.34	0.98	1.98	0.58	0.13

ble 1, we find that methods with mode reduction perform 39.33% faster than methods without mode reduction, PSNR dropping and bitrate increasing is negligible, which are 0.029 and 0.61%, respectively. The search distance comparison of FDVS and Prop methods are figured in Fig.7 and the performance of ME time, PSNR and bitrate are tabulated in Table 1. Compared to *FDVS* method, *Prop* method uses only 39.54% ME time, the PSNR and bitrate performance is similar (0.008 PSNR dropping and 0.3% bitrate decreasing). Even compared to the *FFS* method, PSNR dropping of *Prop+MR* method is only 0.058 and bitrate increasing is 1.18%, with about 9 times faster ME time.

5 Conclusion

We proposed a fast multi-frame based ME algorithm for H.264 in this paper. Both temporal and spatial correlations are exploited to improve the coding speed. Moreover, a mode reduction scheme with adaptive threshold is introduced to further improve the performance. The mode reduction scheme can be easily combined with other fast algorithm to achieve better performance. Experimental results show that the proposed algorithm improves the coding speed by an average of 9 times faster than *FFS* and about two times faster than *FDVS* method which only employs temporal correlation, the PSNR dropping and bitrate increasing are negligible.

References

1. Xu, J.F., Chen, Z.B., He, Y.: Efficient fast me predictions and early-termination strategy based on h.264 statistical characters. In: 4th Pacific Rim Conference on Multimedia. Volume 1. (2003) 218–222
2. Zhou, Z., Sun, M.T., Hsu, Y.F.: Fast variable block-size motion estimation algorithms based on merge and split procedures for h.264/mpeg-4 avc. In: Inter Sym Circuits and Systems. Volume 3. (2004) 725–728
3. Youn, J., Sun, M.T., Lin, C.W.: Motion vector refinement for high-performance transcoding. IEEE Trans. Multimedia **1** (1999) 30–40
4. Chen, M.J., Li, G.L., Chiang, Y.Y., Hsu, C.T.: Fast multiframe motion estimation algorithms by motion vector composition for the mpeg-4/avc/h.264 standard. IEEE Trans. Multimedia **8** (2006) 478–487
5. Kim, S.E., Han, J.K., Kim, J.G.: An efficient scheme for motion estimation using multireference frames in h.264/avc. IEEE Trans. Multimedia **8** (2006) 457–466

A Novel Intra/Inter Mode Decision Algorithm for H.264/AVC Based on Spatio-temporal Correlation

Qiong Liu[1], Shengfeng Ye[1], Ruimin Hu[1], and Han Zhen[2]

[1] Computer School of Wuhan University
430079 Wuhan, China
[2] The Key Laboratory of Multimedia and Network Communications Engineering
430079 Wuhan, China
Liuq_whu@163.com, yeshengfeng007@163.com

Abstract. The high computation burden of mode decision procedure is a challenge to extensive application of H.264/AVC. In this paper, we present a binary mode decision scheme that intra/inter coding mode is decided without performing the exhaustive spatial prediction and motion estimation search. This decision algorithm is based on spatio-temporal feature of current macroblock. And a fast adaptive intra prediction algorithm is provided for further timesaving. By avoiding a large amount of prediction processing, the computation complexity can be greatly reduced. Experimental results indicated that the proposed algorithm, compared by full-search algorithm, can achieve reduction of 54.60% encoding time on average, with a negligible average PSNR loss of only 0.0085 dB and a mere 0.28% bit-rate increase.

Keywords: video compression, H.264/AVC, intra/inter mode decision, spatio-temporal correlation, adaptive intra prediction.

1 Introduction

H.264/MPEG-4 Part 10 Advanced Video Coding (AVC) [1], which was developed by the Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG, is a powerful and high performance video compression standard. To achieve high coding efficiency, it employs many new techniques, such as spatial prediction in intra coding, adaptive block size motion compensation, multiple reference pictures, and so on. Especially, the complex encoding modes, including intra and inter modes, obtain notable coding gains. Fig.1 shows all the encoding modes of H.264/AVC, which are classified as three levels.

Intra prediction is necessary for inter frame to get more precise prediction and prevent the error drift. When the scene is changed suddenly or object motion is too large to get a precise temporal prediction, the amount of intra mode is increased a lot. H.264/AVC encoder checks all the intra prediction modes for intra frame, while all the intra and inter modes for inter frame. In order to choose an optimal coding mode for a macroblock (MB), it calculates the rate-distortion cost (RD cost) of every possible mode and chooses the mode having the minimum cost. And this process is repeatedly carried out for all MBs. So the computation complexity is extremely high.

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 568 – 575, 2007.
© Springer-Verlag Berlin Heidelberg 2007

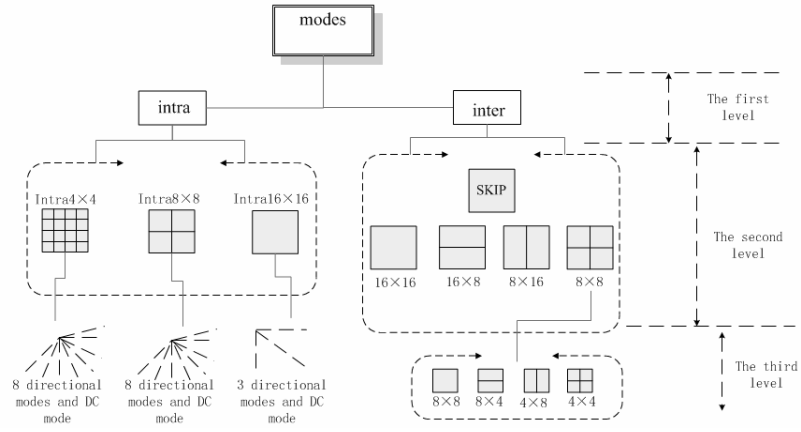


Fig. 1. The three-level structure of modes in H.264/AVC. The first level is a binary decision, intra or inter. And then in the second level, modes are classified by block size, respectively. For intra modes, there are intra4×4, intra16×16 and added intra8×8 in FRExt profiles, while there are SKIP, 16×16, 16×8, 8×16 and 8×8 block sizes for inter modes. In the third level, for each block type of intra modes, there are several directional modes. And for inter 8×8 block type, it can be further divided as 8×8, 8×4, 4×8 and 4×4.

Therefore, it is desirable to make a coarse-level mode decision about whether intra or inter mode to perform.

A few approaches have been proposed about fast intra/inter mode decision algorithm. Y.K. Chen *et al.* [2] proposed a fast intra/inter mode selection scheme by considering both prediction residual and motion vectors (MV). To facilitate video transmission over networks, D.S. Turaga and T. Chen [3] using the maximum likelihood (ML) criterion to make classification based mode decision. However, these schemes are not suitable for H.264/AVC since it has more complex modes. C.S. Kim and C.C. Kuo [4] decided the mode using the expected risk of choosing the wrong mode in a multidimensional feature space. It gets little R-D loss, but the computation reduction is limited.

In this paper, we propose an efficient intra/inter mode decision algorithm. We predict the proper class coding mode (intra or inter) and only perform the modes of chosen class. The classification is based on spatio-temporal correlation. Then, the specific intra type is chosen for a fine-level mode decision.

This paper is organized as follow: In section 2, spatio-temporal model is shown in detail. And the further adaptive fast intra prediction algorithm is proposed in section 3. The experiment result is shown and analyzed in section 4.

2 The Spatio-temporal Model and Intra/Inter Decision

H.264/AVC employs RDO procedure to estimate the cost by the following equation:

$$J(S,C,Mode|QP,\lambda_{mode}) = SSD(S,C,Mode|QP) + \lambda_{mode} \cdot R(S,C,Mode|QP) \quad (1)$$

where QP is the quantization parameter, λ_{mode} is the Lagrange multiplier for mode decision, *SSD* is the sum of the squared differences between the original block luminance (denoted by *S*) and its reconstruction *C*, and *R(S, C, Mode|QP)* represents the number of bits associated with the chosen mode. In most conditions, the amount of bits needed for DCT coefficients is increasing when *SSD* is increasing. So the difference between *S* and *C* is vital for the whole RDO cost. Inter mode exploits the temporal correlation across frames and intra mode is used to reduce the spatial correlation.

2.1 Spatial Correlation

There exist many effective techniques for detecting spatial features of images [5], [6]. But from our observation, the edge information has a correlation to encoding modes.

A region is homogeneous if the textures in the region have very similar spatial property. And the homogeneous region has a high spatial correlation. An effective way of determining homogeneous regions is to use the edge information, as the video object boundary usually exhibits strong edges. Another reason is when the moving object occupied a large area and moved fast, MBs in the edge are coded as intra block usually. So the edge information has a correlation to encoding modes.

Considering the reason above and computation complexity, we choose edge information to detect the spatial features of images. Furthermore, the JVT recommended fast intra and inter mode decision algorithms also used edge information [7] [8]. They can be performed in the second or third level mode decision without extra computation.

At first, apply the Sobel operator to each pixel of current image. It is calculated by the Equation (2), and then the amplitude (*Amp*) is derived by Equation (3). We select the 16×16 block as the basic unit to create edge histogram. $Amp_{16\times16}$ is obtained using Equation (4) to present the spatial characteristics in the following algorithm.

$$\begin{aligned} dx_{i,j} &= P_{i-1,j+1} + 2\times P_{i,j+1} + P_{i+1,j+1} - P_{i-1,j-1} - 2\times P_{i,j-1} - P_{i+1,j-1} \\ dy_{i,j} &= P_{i+1,j-1} + 2\times P_{i+1,j} + P_{i+1,j+1} - P_{i-1,j-1} - 2\times P_{i-1,j} - P_{i-1,j+1} \end{aligned} \tag{2}$$

$$Amp\left(\vec{D}_{i,j}\right) = \left|dx_{i,j}\right| + \left|dy_{i,j}\right| \tag{3}$$

$$f_{intra} = Amp_{16\times16} = \sum_{j=0}^{15}\sum_{i=0}^{15} Amp(\vec{D}_{i,j}) \tag{4}$$

2.2 Temporal Correlation

In this paper, we used the difference of current MB and reference MB to present the temporal correlation. How to find the reference MB is a key of the efficiency of our decision algorithm. It is the fact that the sum of absolute (SAD) values of current MB and the MB under proper predictive MVs (*PMV*) in the reference frame are much less

than others. We use *PMV* defined in [1] in order to avoid 16× 16 motion estimation. And then, the temporal domain feature is presented by $SAD_{16\times16}$ which is defined as difference between current MB and the MB referred by *PMV* in the reference frame. The location of the two MBs is dedicated in Fig. 2.

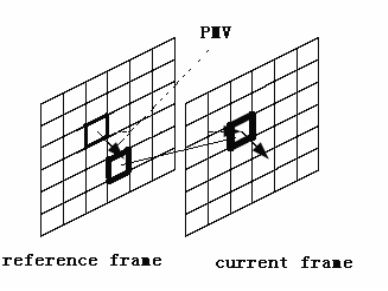


Fig. 2. Locations of current MB and *PMV* corresponding MB

$SAD_{16\times16}$ is calculated using Equation (5).

$$f_{inter} = SAD_{16\times16} = \sum_{j=0}^{15}\sum_{i=0}^{15}\left|x(i,j) - y(i+PMV_x, j+PMV_y)\right| \tag{5}$$

$x(i, j)$ and $y(i+PMV_x, j+PMV_y)$ denote the pixels at location (i, j) in the current frame and pixels at location $(i+PMV_x, j+PMV_y)$ in reference frame.

2.3 Spatio-temporal Correlation

For observing the features of intra and inter in spatio-temporal domain, we run H.264/AVC reference software JM9.5 [9] under the test conditions in Section 6. Part of results is depicted from Fig. 3, corresponding to 100 frames of "*Foreman*".

As shown in Fig.3, the whole feature space is divided into two areas by a line $y=k\times x$. These facts below can be observed:

(1)A majority of samples congregate in area (1) and almost the whole samples in area (1) are inter blocks. In area (1), sample's temporal correlation is much higher than its spatial correlation. If the block is coded by intra modes, the residues will be much larger than it coded by inter modes. If we can specify whether current block belongs to this area, the whole needless intra prediction process is avoided. The computation complexity can be reduced greatly.

(2) Only a minority of samples are distributed in area (2) in which inter blocks and intra blocks are mixed together. In this area, the difference between f_{intra} and f_{inter} is smaller than that in area (1). So other parameters of Equation (1) make a more impact on the final results. It is hard to predict whether intra or inter mode is the optimal one.

Based on analysis above, proposed intra/inter decision strategy is to filter out unlikely intra modes by spatio-temporal correlation, i.e. we only specify the area (1),

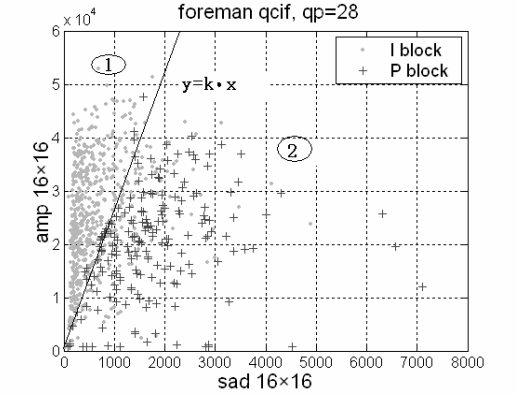


Fig. 3. Distribution of intra and inter blocks in spatio-temporal correlation domain

where $f_{intra} > k \times f_{inter}$. If spatio-temporal correlation of current block is satisfied below Equation (6), we think it must be locate in area (1) and only perform inter prediction modes on it. From our experiments, we found that $k = (52 + QP)/3$ is an empirical threshold to obtain a good trade-off between coding efficiency and complexity.

$$f_{\text{int}ra} - k \cdot f_{\text{int}er} > 0 \tag{6}$$

3 Adaptive Fast Intra Type Decision Algorithm

When Equation (6) is not satisfied, both intra and inter mode will be performed. If fast algorithm is developed for the sub-decision, the further time saving of the whole encoding procedure will be obtained.

In our previous work [10], we had developed a fast algorithm to decide whether intra4×4 or intra16×16 is optimal. As intra4×4 is well suit for blocks with detailed information and intra16×16 is suit for smooth ones, the decision algorithm is based on the smoothness detection. Usually, the block in edges is not smooth. So we use edge information to decide whether current block is smooth. The amplitude of the maximum cell of edge histogram is called Amp_{max}. The Fig.4 shows the relationship between Amp_{max} and intra types. In statistically, intra16×16 blocks have much lower Amp_{max} value than intra4×4 blocks. In this paper, the algorithm is improved by adding threshold $T_{16\times16}$ and changing static thresholds to adaptive ones. If $Amp_{max} < T_{16\times16}$, choose intra16×16. If $Amp_{max} > T_{4\times4}$, choose intra4×4. Else, perform the two block types.

As depicted from Fig. 4, the blocks with large Amp_{max} may be coded in intra16×16 under the condition of low bit-rate, whereas they could be coded in intra4×4 when in high bit-rate. In order to accurately apply our proposed algorithm in different bit-rate, the adaptive threshold is necessary. We do another group experiments to find out more

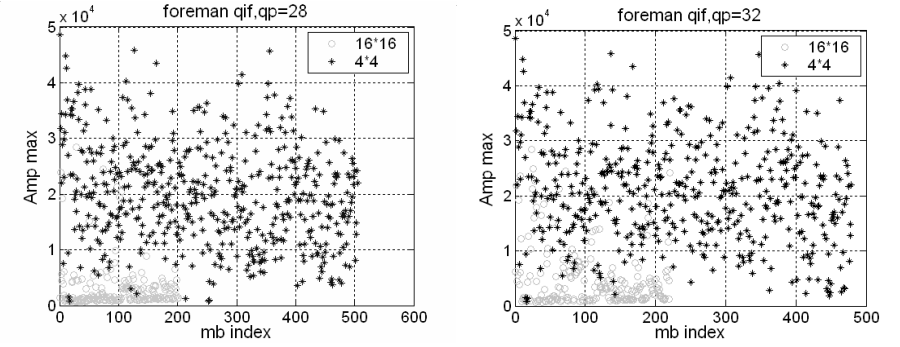


Fig. 4. Comparision of the Amp_{max} of intra4×4 blocks and intra 16×16 blocks. In statistically, intra16×16 blocks have much lower Amp_{max} value than intra4×4 blocks and larger QP will result in larger block type.

exact laws of threshold changing. For each QP condition, we test a range of thresholds, and get the best one with highest performance. Through a huge amount of experiments, the adaptive threshold $T_{16\times16}$ and $T_{4\times4}$ is generated by an empirical trend.

$$T_{16\times16} = 0.45\times QP^{3/8}, \quad T_{4\times4} = 27.585\times QP^{2} \tag{7}$$

4 Simulation Results

Our proposed algorithm was implemented into the reference software JM9.5 provided by JVT [9], tested on five standard QCIF (352×288) sequences (*Foreman*, *News, Silent, Container*, *and Paris*) and three 480i (704×480) sequences (*Mobile, News, and Teacher)*. Each QCIF sequence has 100 frames and 480i sequence has 50 frames. We compared our proposed technique with the original JM9.5. The simulation condition is as: main profile, search range = 16, reference frame =1, RDO=1, CABAC, and picture structure is IPPP (only first frame is intra frame). The performance is evaluated by the difference of coding time (△T), the PSNR difference (△PSNR_Y) and the bit-rate difference (△BITS).

Table 1 tabulate the comparison results (QP=24, 28, 32, 36) of QCIF and Table 2 is the results of 480i. In these tables, positive number means increasing, and negative number means decreasing. On average, our proposed scheme is able to achieve a reduction of 54.60% encoding time, with a negligible average PSNR loss of only 0.0085 dB and a mere 0.28% bit-rate increase. The result of each QCIF sequence in Table 1 is an average data of the four different QP. Furthermore, the result of Table 2 is more detailed to compare the experiment results under different QP. It can be seen that the algorithm performs efficiently varying from high bit-rate to low bit-rate.

However, the time saving is a little lower with large QP. The reason is that large QP counteracts the influence of spatio-temporal correlation. When QP is increasing, more and more inter and intra samples mixed together.

Table 1. Comparison results using five standard QCIF sequences

Sequence	△T	△PSNR_Y(db)	△BITS
Foreman	-46.81%	-0.0175	0.13%
News	-53.61%	-0.0275	0.42%
Container	-55.56%	-0.02	0.27%
Silent	-54.19%	0.01	0.13%
Paris	-60.36%	0.0075	0.03%
average	-54.11%	-0.0095	0.20%

Table 2. Comparision result of 480i sequences under different QP

QP	Sequence	△T	△PSNR_Y(db)	△BITS
24	*Mobile*	-69.34%	-0.01	0.00%
	News	-57.71%	-0.01	0.46%
	Teacher	-46.54%	0	-0.10%
28	*Mobile*	-66.92%	-0.01	0.06%
	News	-58.95%	-0.02	0.64%
	Teacher	-49.34%	0	-0.23%
32	*Mobile*	-64.42%	0	0.18%
	News	-54.80%	-0.02	0.69%
	Teacher	-42.42%	0	1.10%
36	*Mobile*	-57.46%	0	0.06%
	News	-57.71%	-0.02	1.13%
	Teacher	-58.95%	0	0.22%
average		-55.09%	-0.0075	0.35%

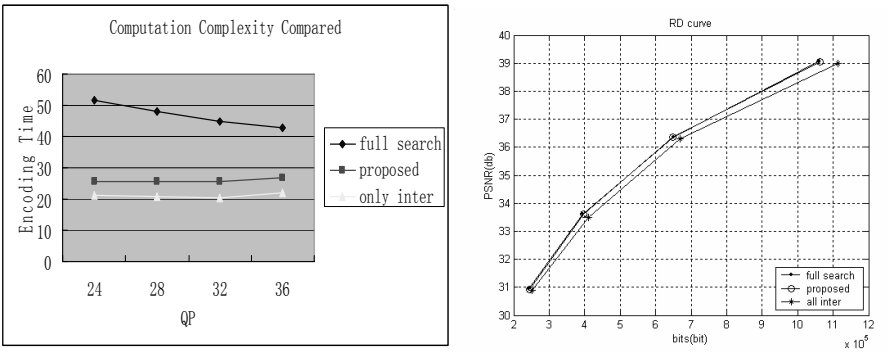


Fig. 5. Computation complexity compared on sequence *foreman*, 100 frames. RD performance compared with sequence *foreman*, 100 frames.

Fig.5 compared the performance of full search algorithm of JM9.5, our proposed algorithm and only-using-inter-mode strategy. The computation complexity of proposed algorithm is much less than the full search algorithm of JM and it is very close

to only-inter strategy. And the RD performances of the three strategies are compared. The curve of full search algorithm is the optimal one, and our proposed one is very closed to it and better than the performance of only-using-inter-mode strategy.

5 Conclusion

In this paper we proposed a binary intra/inter mode decision scheme. We analyze the correlation between spatio-temporal feature and prediction modes. By judging the attribute of current MB in feature space, we predict the encoding mode without performing the exhaustive spatial prediction and motion estimation search. Using a dynamic adjusted threshold, the proposed algorithm will be applied in different bits-rate environments. Experimental results indicate that the proposed algorithm, compared by full-search algorithm, can achieve 54.60% time saving, while maintaining the same rate distortion performance as full search algorithm. It should be note that the thresholds are generated by experiment, and it can be adjusted according to different requirement of trade-off between efficiency and complexity.

Acknowledgement

This work was supported by National Natural Science Foundation of China (NO. 60472040).

References

[1] "Draft ITU-T recommendation and final draft international standard of joint video specification (H.264/1SO/IEC 14496-10 AVC)", JVT-G050, Mar. 2003.
[2] Chen, Y.-K., Vetro, A., Sun, H., and Kung, S.Y., "Optimizing intra/inter coding mode decisions," in Proc. 1997 International Symposium on Multimedia Information Processing, Taipei, Taiwan, 1997.
[3] Turaga, D.S. and Chen, T., "Classification based mode decisions for video over networks," IEEE Trans. Multimedia, vol. 3, no. 1, pp. 41–52, 2001.
[4] Kim, C.S., Kuo, C.C., "A Feature-based Approach to Fast H.264 Intra/Inter Mode Decision", Circuits and Systems, 2005. 23-26 May 2005 Page(s):308 - 311 Vol. 1
[5] Uchiyama, T., Mukawa, N., and Kaneko, H., "Estimation of homogeneous regions for segmentation of textured images," in Proc. IEEE ICPR2000, pp. 1072–1075.
[6] Liu, X.W., Liang, D.L., and Srivastava, A., "Image segmentation using local spectral histograms," in Proc. IEEE ICIP2001, pp. 70–73.
[7] Pan, F., Lin, X., Susan, R., Lim, K.P., Li, Z.G., Feng, G.N., Wu, D.J., and Wu, S., "Fast Mode Decision Algorithm for Intra Prediction in JVT", JVT-G013, Pattaya, March 2003.
[8] Lim, K.P., Wu, S., et. al, "Block Inter Mode Decision for Fast Encoding of H.264", Proceedings of 29th International Conference on Acoustics, Speech, and Signal Processing (ICASSP), 2004, PP.III181-III184.
[9] JVT reference software version 9.5, http://bs.hhi.de/~suehring/tml/download/ reference_ software/
[10] Qiong, L., Ruimin, H., Li, Z., Xincheng, Z., and Zhen, H., "Improved fast intra prediction algorithm of H.264/AVC", Journal of Zhejiang University: Science, v7 SUPPL. 1, May 2006, pp 101-105.

Switchable Bit-Plane Coding for High-Definition Advanced Audio Coding

Te Li[1], Susanto Rahardja[1], and Soo Ngee Koh[2]

[1] Institute for Infocomm Research, 21 Heng Mui Keng Terrace, Singapore 119613
{lite,rsusanto}@i2r.a-star.edu.sg
http://www.i2r.a-star.edu.sg/
[2] Nanyang Technological University, 50 Nanyang Avenue, Singapore 639798
esnkoh@ntu.edu.sg

Abstract. Scalable audio coding technique such as MPEG-4 Scalable Lossless coding (SLS) is a unified solution for demands in high-compression perceptual audio and high-quality lossless audio. It provides a fine-grain scalable extension of the well-known MPEG-4 Advanced Audio Coding (AAC) perceptual audio coder up to fully lossless reconstruction at word lengths and sampling rates typically used for high-resolution audio. Recently, the combination of SLS and AAC is renamed as "High Definition Advanced Audio Coding" (HD-AAC). It is observed that HD-AAC can be further improved at intermediate enhancement bitrate for many audio sequences when the core bitrate is low. Based on this observation, a Switchable Bit-Plane Coding (SBPC) is proposed in this paper. The SBPC consists of a normal BPC and a Prioritized BPC (PBPC). The corresponding optimal bit-plane coding method is switched into action according to the residual energy distribution in different frequency ranges. With the SBPC, the bit-plane coding for scalable audio can be implemented in a perceptually more efficient manner, and the perceptual quality of the audio under aforementioned scenario is much improved and stable.

1 Introduction

With advances in broadband networking and storage technologies, the capacities of more and more digital audio applications are approaching those for delivery of high sampling rate, high resolution digital audio at lossless quality. On the other hand, there are still applications such as wireless devices that require high-compression audio. Envisioning of such variable needs, the international standardization body MPEG has recently released a scalable tool for lossless audio coding named MPEG-4 SLS [1] coding. The combination of SLS and MPEG-4 AAC [2] which is referred to as HD-AAC integrates the functionalities of lossless audio coding, perceptual audio coding, and fine granular scalable audio coding in a single framework. As such, it provides a universal digital audio format that can be used in a variety of application domains such as professional audio, internet music, consumer electronics and broadcasting.

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 576–585, 2007.
© Springer-Verlag Berlin Heidelberg 2007

Like most of general scalable audio coding schemes, HD-AAC employs two layers - an AAC core layer and an enhancement layer. The scalable enhancements are achieved through sequential bit-plane coding of the residual signals between the original and AAC encoded spectrum. This scalable coding scheme works efficiently when the core AAC layer is with high bitrate so that the quantization noise are all below the psychoacoustic mask, and the perceptual information is mostly reserved in the AAC coded format. However, when the AAC core bitrate is low, the spectral shape of the residual signal is far from the optimal as in these cases noise shaping is usually performed in only limited frequency range. This will directly result in non-optimal perceptual quality of output audio at intermediate bitrate.

With this concern, there were several approaches that target at improving the efficiency of bit-plane coding for scalable audio. It is mentioned in [3] that the psychoacoustic information like just noticeable distortion can be applied in the bit-plane coding process for a perceptually more efficient enhancement coding. An issue of this approach is the non-negligible amount of extra side information which can be risky for low bitrates. Another approach that gets rid of extra perceptual side information is Embedded Audio Coding (EAC)[4] with implicit auditory mask. However, its criteria to determine the priorities of bit-planes and implicit audio mask calculations involve considerable amount of coding complexity.

This paper presents a computationally simple yet efficient SBPC that is favorable for scalable audio with low core bitrate. In this approach, the whole frequency spectrum is divided into a Low Frequency (LF) and a High Frequency (HF) region. The SBPC is composed of two bit-plane coding methods: a normal BPC and the PBPC. When the residual energy distribution in the LF and HF are balanced, normal BPC is applied. Otherwise, the PBPC is switched into action. Since the SBPC fully utilizes the distribution feature of residual energy, the perceptual quality of the audio under aforementioned scenario is much improved with negligible amount of extra side information. The rest of this paper is organized as follows. Firstly, a short overview over HD-AAC and its corresponding issue for bit-plane coding are given. It is followed by the detailed description about the SBPC and the way it is integrated into HD-AAC. Finally, extensive results are provided to verify the efficiency of the SBPC.

2 Overview of HD-AAC

2.1 System Structure

The system diagram of the HD-AAC codec is shown in Fig. 1, which comprises of two distinguished layers in both the encoder and decoder namely a core layer and Lossless Enhancement (LLE) layer.

In the HD-AAC encoder, the input audio in integer PCM format is losslessly transformed to the frequency domain by using the Integer Modified Discrete Cosine Transform (IntMDCT) [5]. The resulting IntMDCT coefficients are passed

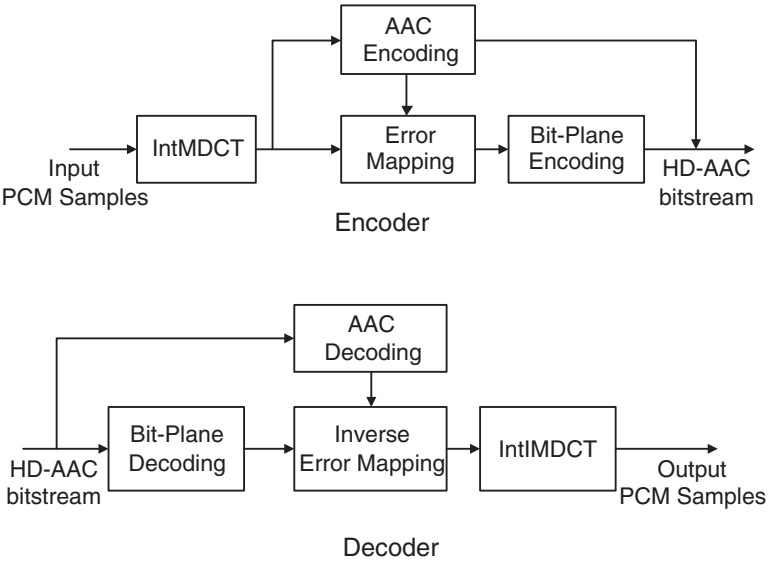


Fig. 1. Structure of HD-AAC encoder and decoder

to the AAC encoder to generate the core layer AAC bitstream. In the AAC encoder, transformed coefficients are first grouped into scalefactor bands (sfbs) which are then quantized with an non-uniform quantizer.

In order to efficiently utilize the information of the spectral data that has been carried in the core layer bitstream, the error-mapping procedure is employed to generate the residual spectrum coded in the LLE layer by subtracting the AAC quantized spectrum from the original spectrum. The residual spectrum is then coded using Bit-Plane Golomb Code (BPGC) [6] to generate the scalable LLE layer bitstream. As illustrated in Fig. 2, the Most Significant Bit (MSB) for spectral data from all sfbs are coded first. Subsequently, the coding process is progressed to the following bit-planes until it reaches the Least Significant Bit (LSB) for all sfbs by using arithmetic code with a structural frequency assignment

$$Q[j] = \begin{cases} \frac{1}{1+2^{2^{j-L}}}, & j \geq L \\ \frac{1}{2}, & j < L \end{cases} \tag{1}$$

where j is the current bit-plane to be coded, and Lazy plane parameter L can be selected using the adaptation rule. BPGC will enter into absolute lazy-mode coding from the fifth bit-plane afterwards. As the final step of the encoder, the output of LLE bitstream is multiplexed with the core AAC bitstream to produce the final HD-AAC bitstream.

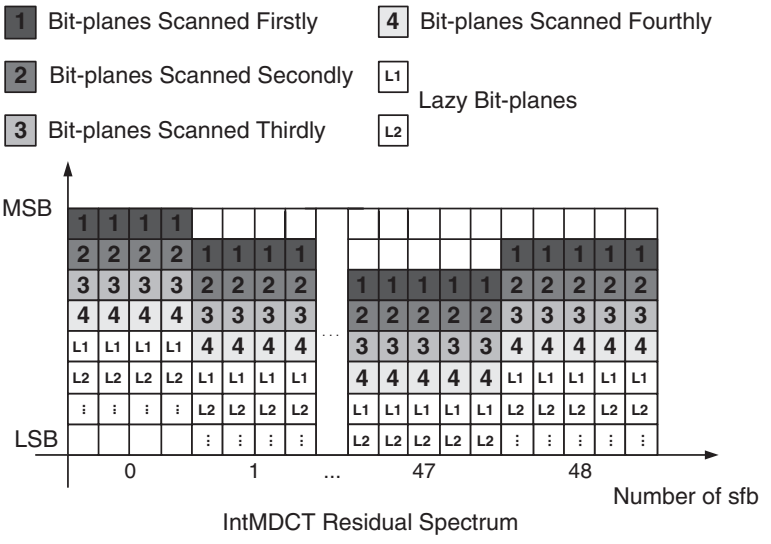


Fig. 2. Bit-plane coding order adopted in HD-AAC

2.2 Perceptual Issue for Bit-Plane Coding

When the bitrate for AAC core layer is relatively low, e.g., less than 64kbps, the quantization noise of most coded signal will roughly follow the shape of original signal. Fig. 3 depicts the signal, allowed distortion and quantization noise energy (one channel of one frame) for an excerpt named dcymbals.wav that coded at 32kbps. The adopted AAC coder is the one in Reference Model (RM) of SLS. The allowed distortion d is computed as

$$d[k] = \begin{cases} E[k] \times m[k], & E_k > 70\text{dB and } m[k] < 1 \\ E[k], & E_k > 70\text{dB and } m[k] \geq 1 \\ E[k] \times 1.1, & \text{otherwise} \end{cases} \tag{2}$$

$$k = 0, \ldots, K$$

where k denotes the sfb number, K is the maximum number of sfbs, $E[k]$ is the signal energy and $m[k]$ is the mask to signal ratio calculated by the psychoacoustic model. The whole sfbs are divided into the LF and the HF region. Normally for $K = 48$, sfbs $0 \sim 24$ and $25 \sim 48$ belong to the LF and the HF, respectively.

When the noise energy distributed in two frequency domains are almost balanced, e.g., the excerpt in Fig. 3, it is reasonable for a normal bit-plane scanning with the order from first sfb to the last sfb as mentioned in the previous section. However, when the residual energy of an audio sequence performs a certain level of unbalanced distribution in two frequency ranges as shown in Fig. 4, the sfbs should not be treated fairly anymore. Intuitively, it is perceptually more efficient to encode those sfbs contain dominant quantization noise, so that much noise

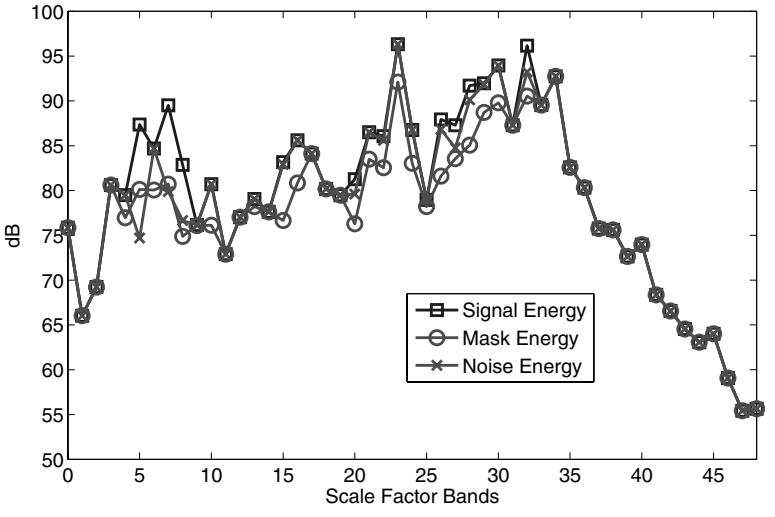


Fig. 3. Plot of signal, mask and quantization noise energy (1 frame) for dcymbals.wav coded by AAC at 32kbps

can be masked after enhancement coding. In other words, the PBPC method should be applied to achieve optimal quality in this case. This idea is further elaborated in next section.

3 Switchable Bit-Plane Coding

3.1 Basic Structure

In order to optimize the bit-plane coding for the case in Fig. 4, PBPC is proposed as an alternative coding method of the normal BPC. For each audio sequence, the optimal bit-plane coding should be switched into action according to the residual energy distribution. The basic structure of HD-AAC with SBPC is depicted in Fig. 5. For each frame, a decision will be made to determine the bit-plane coding method according to the AAC quantization information. Specifically, if the frame is identified as balanced, the residual signals from error mapping can be bit-plane encoded using the sequential order as shown in Fig. 2. Otherwise, this frame should be coded in a prioritized manner such that those sfbs with dominant quantization noise can be encoded first.

It should also be noted that thought one bit for each frame is needed to specify the corresponding coding scheme, the reserved bit for LLE coding can be adopted for this application and it incurs no extra payload. The detailed switching criteria and coding algorithm of the PBPC will be elaborated in later sections.

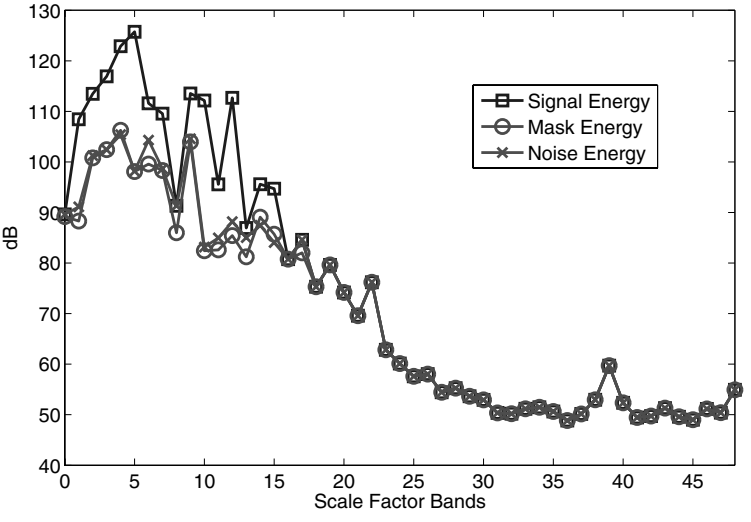


Fig. 4. Plot of signal, mask and quantization noise energy (1 frame) for mfv.wav coded by AAC at 32kbps

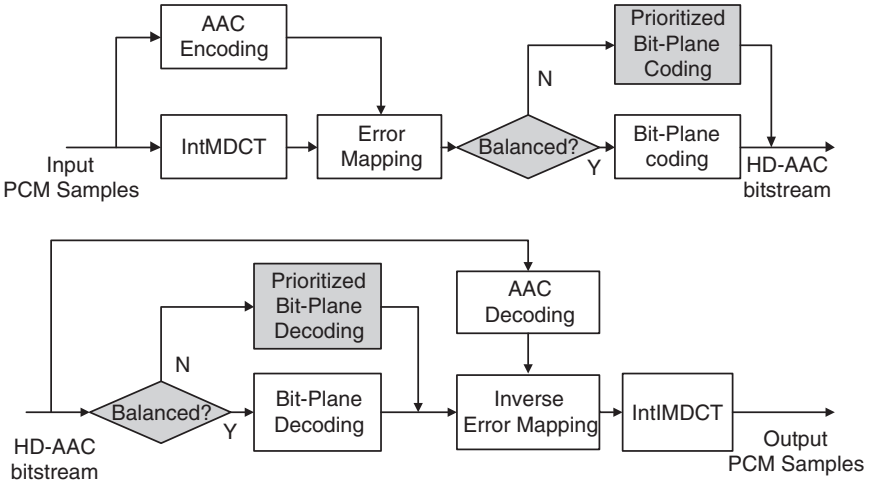


Fig. 5. Encoder and decoder of HD-AAC with SBPC

3.2 Prioritized Bit-Plane Coding

When the noise energy distribution is detected to be unbalanced, PBPC will be switched on and high priority are assigned to the non-lazy bit-planes in the LF.

The coding method of PBPC is depicted in Fig. 6. The LF and HF regions are treated as two independent objects to be coded for non-lazy bit-planes. The coding

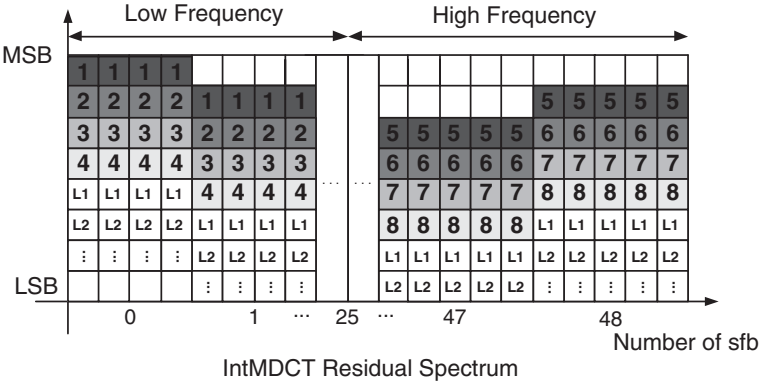


Fig. 6. Prioritized Bit-Plane Coding

starts from the MSB of LF sfbs, all the way till the Lazy plane of the LF is reached. The MSB of the HF region will start when the top 4 bit-planes of the LF sfbs are coded. When the top 4 bit-planes of HF sfbs are also finished, the lazy mode coding begins. For lazy bit-planes, the LF and HF sfbs are no longer considered as independent and the coding sequence will be the same as the one in Fig. 2.

It can be easily understood that for high coding bitrates that the lazy mode coding can be reached, the coding results will be almost the same for two bit-plane coding methods. However, for the bitrates that only non-lazy coding is available, the PBPC method will show considerable advantages for unbalanced audio sequences. This will be demonstrated in Section 4.

3.3 Switching Criteria

The criteria for determining whether a frame is balanced or not is crucial for the optimized quality. Let E_L be the sum of the quantization noise energy for LF sfbs, i.e.

$$E_L = \sum_{i=0}^{24} \sum_{k=O[i]}^{O[i+1]-1} (c[k] - a[k])^2 \tag{3}$$

where $O[i]$ is the offset spectrum number for sfb i, $c[k]$ is the IntMDCT coefficient and $a[k]$ is AAC encoded coefficient. Further define E_H as the remaining quantization noise energy in HF, i.e.

$$E_H = \sum_{i=25}^{48} \sum_{k=O[i]}^{O[i+1]-1} (c[k] - a[k])^2 \tag{4}$$

A frame is considered to be unbalanced when

$$\frac{E_L - E_H}{E_H} \geq T_B \tag{5}$$

Table 1. Performance of HD-AAC using SBPC Comparing with that using normal BPC

Items (.wav)	(16+128)kbps					
	NMR (dB)			ODG		
	HD-AAC	EN HD-AAC	Improvement	HD-AAC	EN HD-AAC	Improvement
avemaria	-0.50	-3.75	3.25	-2.52	-1.92	0.60
broadway	4.65	3.13	1.52	-3.16	-2.77	0.39
dcymbals	1.29	1.28	0.01	-2.20	-2.20	0.00
etude	0.20	-3.28	3.49	-2.72	-2.00	0.72
flute	-0.03	-2.08	2.05	-3.36	-2.67	0.69
haffner	0.02	-3.31	3.32	-1.96	-1.92	0.04
mfv	0.89	-2.97	3.85	-3.35	-1.70	1.65
average			2.50			0.59

Items (.wav)	(16+192)kbps					
	NMR (dB)			ODG		
	HD-AAC	EN HD-AAC	Improvement	HD-AAC	EN HD-AAC	Improvement
avemaria	-3.24	-5.95	2.71	-1.83	-1.14	0.69
broadway	3.58	-1.78	5.36	-2.70	-1.76	0.94
dcymbals	-0.18	-0.77	0.59	-1.63	-1.55	0.08
etude	-2.47	-5.63	3.16	-2.05	-1.22	0.83
flute	-3.23	-6.27	3.04	-2.74	-1.95	0.79
haffner	-2.55	-5.74	3.19	-1.12	-0.88	0.24
mfv	-1.98	-4.25	2.28	-2.45	-1.58	0.88
average			2.90			0.63

Items (.wav)	(32+128)kbps					
	NMR (dB)			ODG		
	HD-AAC	EN HD-AAC	Improvement	HD-AAC	EN HD-AAC	Improvement
avemaria	-1.48	-4.62	3.14	-2.50	-1.64	0.86
broadway	3.03	1.85	1.18	-3.04	-2.70	0.35
dcymbals	1.21	1.21	0.00	-2.16	-2.16	0.00
etude	-0.94	-4.21	3.27	-2.70	-1.70	0.99
flute	-2.31	-3.83	1.53	-3.24	-2.65	0.59
haffner	-0.42	-3.86	3.44	-1.91	-1.83	0.07
mfv	-1.74	-4.17	2.43	-3.11	-1.59	1.52
average			2.14			0.63

Items (.wav)	(32+192)kbps					
	NMR (dB)			ODG		
	HD-AAC	EN HD-AAC	Improvement	HD-AAC	EN HD-AAC	Improvement
avemaria	-4.44	-6.61	2.17	-1.68	-1.15	0.52
broadway	1.33	-2.66	3.99	-2.60	-1.74	0.86
dcymbals	-1.06	-1.38	0.33	-1.50	-1.50	0.00
etude	-3.98	-6.34	2.37	-1.89	-1.23	0.66
flute	-5.42	-7.78	2.36	-2.46	-1.85	0.61
haffner	-3.67	-6.03	2.36	-1.01	-0.88	0.13
mfv	-4.16	-5.73	1.58	-2.15	-1.37	0.78
average			2.16			0.51

where T_B is a global threshold value that is a non-increasing function of available LLE bitrate B

$$T_B = f(B) \qquad (6)$$
$$f'(B) \leq 0, \quad B_0^{25} < B < B_3^{48}$$

with $B_{j'}^{s}$ indicating the LLE bitrate that is able to encode the remaining spectrum up to sth sfb of the bit-plane that with distance of j' from the MSB. It can be easily understood from Fig. 6 that with above bitrate range, optimized quality of coded audio can be achieved through switching of suitable bit-plane coding method. Certainly the required bitrate for each frame is different, and the above bitrate range is just a rough estimation of average values.

4 Performance

We shall compare the perceptual quality of the enhanced HD-AAC with SBPC with that of the original HD-AAC at various intermediate rates by using Noise to Mask Ratio (NMR) and Objective Difference Grade (ODG) measurements. In our evaluation, we used the standard MPEG-4 audio test sequences, which include 7 stereo music files sampled at 48 kHz, 16 bits/sample. The results are illustrated in Table 1, where four bitrate combinations with AAC core bitrate at 16 and 32 kbps and LLE bitrate at 128 and 192 kbps are used for testing.

From these results, it can be seen that for all these bitrates combinations, HD-AAC with SBPC achieves improvements on both NMR and ODG values compared with the results of the original HD-AAC. The improvement is very significant for some unbalanced audio sequence such as mfv.wav. Moreover, the quality of different types of audio coded by the enhanced HD-AAC is more stable at same bitrate. It is also worth to note that for the sequences named dcymbals.wav and haffner.wav, the improvements are marginal. The reason is that dcymbals.wav is a very balanced audio sequence and PBPC will be seldom switched on. As a result, the coding of such a sequence is almost the same as the one using the original BPC. While for haffner.wav, as the quantization noise for this sequence is already quite small compared with normal sequences, the improvements is marginal due to the quality saturation.

5 Conclusions

In this paper, a novel bit-plane coding methodology named Switchable Bit-Plane Coding is proposed to study the perceptual optimization on the coding of enhancement layers for scalable audio with a low-bitrate core layer. By switching the coding method according to the quantization information from core layer, the LLE bit-plane coding is performed in a more efficient manner. The perceptual quality of output audio at intermediate bitrates is improved by using the proposed method comparing with the original HD-AAC for most of audio sequences. Meanwhile, this is achieved without introducing any overhead in terms of computational complexity or lossless coding efficiency.

References

1. R. Yu, X. Lin, S. Rahardja, and H. Huang: Technical description of I^2R's proposal for mpeg-4 audio scalable lossless coding (SLS): Advanced audio zip (AAZ).(2003)
2. Information technology - coding of audiovisual objects, part 3. audio, subpart 4 time/frequency coding. ISO/IEC 14496-3(1998)
3. R. Yu, T. Li, S. Rahardja: Perceptually enhanced bit-plane coding for scalable audio. IEEE Int. Conf. Multimedia and Expo (2006) 1153–1156
4. J. Li: Embedded audio coding (EAC) with implicit auditory masking. ACM Multimedia (2002) 592–601
5. R. Geiger, T. Sporer, J. Koller, and K. Brandenburg: Audio coding based on integer transform. 111th AES Convention Preprint 5471 (2001)
6. R. Yu, C.C. Ko, S. Rahardja, and X. Lin: Bit-plane golomb code for sources with laplacian distributions. IEEE Int. Conf. Accoustics, Speech, and Signal Processing (2003) 277–280

Neighborhood Graphs for Semi-automatic Annotation of Large Image Databases

Hakim Hacid

Lyon 2 University
ERIC Laboratory- 5, avenue Pierre Mendès-France
69676 Bron cedex - France
hhacid@eric.univ-lyon2.fr

Abstract. Images annotation is the main tool for associating a semantic to an image. In this article we are interested in the semi-automatic annotation of images data. Indeed, with the great mass of data managed throughout the world and especially with the Web, the manual annotation of these images is almost impossible. We propose an approach based on neighborhood graphs offering several possibilities: *content-based retrieval, key-words based interrogation, and the annotation* which concerns us in this article. The approach we are proposing offers, as the experiments section shows it, very interesting annotation results while satisfying the scalability criteria which is a very significant point in this context where the mass of data is very important.

1 Introduction

Among the multimedia data types, image undoubtedly constitutes the more used type. Indeed, its use is in various fields like medicine, museums, astronomy, etc. To be able to interact with these data, a lot of work was carried out for their automatic processing [22]. Large panoply of image segmentation, analysis, and interrogation tools exists offering interesting results. However, this does not solve definitely the involved problems in the imagery such as the segmentation problem which remains always an open problem [16].

Another challenge in this domain is the semantics association to an image. Indeed, image processing methods associate for each image a features vector (or vectors) calculated on the image. These features are known as "low level features" (color, texture, etc.). The interrogation of an image database is then done by introducing an image query into the system and its comparison to the available ones using similarity measures [22]. Thus, no semantics is associated to images with this process.

The common way for semantics assignment to an image is the annotation. Multimedia data annotation is the task of assigning, for each multimedia document or for a part of the multimedia document, a keyword or a list of keywords describing its semantic contents. This function can be considered as a mapping between the visual aspects of the multimedia data and their high level characteristics. In this article, we are interested in the semi-automatic annotation of

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 586–595, 2007.
© Springer-Verlag Berlin Heidelberg 2007

images in order to assign a semantic to images. We consider the semi-automatic annotation because it requires the user intervention to validate the system's decisions. The rest of this article is organized as follows: Section 2 presents some related work to images annotation context. Section 3 introduces neighborhood graphs. Our method for images annotation is discussed in Section 4. We present the performed experiments in Section 5. We will finish by a conclusion and give future directions of our work in Section 6.

2 Related Work

There are three types of image annotation techniques: manual, semi-automatic, and automatic. The first annotation type is carried out manually by human charged to allot for each image a set of keywords. The automatic annotation is carried out by a machine and aims to reduce the user's charge. The first type of annotation increases the precision and decreases the productivity. The last type as for it, decreases the precision and increases its productivity. In order to make a compromise between these two tasks, their combination became necessary. This combination is named the semi-automatic annotation. Furthermore, image annotation can be performed on two levels: the local level and the global level. In the local level, the image is regarded as a set of objects. The annotation aims to affect for each object a keyword or a list of keywords to describe it. The global level as for it, concerns the whole image and assigns a list of keywords to describe its general aspect. The two approaches, the first one more than the second one, depend considerably on the quality of the image segmentation. Unfortunately, image segmentation remains always a challenge and a lot of work concentrate on this topic [17].

There does not exist a lot of work on the automatic annotation of images. There are methods which apply a clustering of the images and their associated keywords in order to make it possible to attach a text to images [1][2][3]. With these methods, it is possible to predict the label of a new image by calculating some probabilities. Minka and Picard [13] proposed a semi-automatic image annotation system in which the user chooses the area to be annotated in the image. A propagation of the annotations is carried out by considering textures. Maron et al., [11] studied the automatic annotation using only one keyword at the same time. Mori et al., [19] proposed a model based on co-occurrences between images and keywords in order to find the most relevant keywords for an image. The disadvantage of this model is that it requires a large training sample to be effective. Dyugulu et al., [4] proposed another model, called translation model, which is an improvement of the co-occurrence model suggested by Mori et al., [19] by integrating a training algorithm. Probabilistic models such as Cross Media Relevance model [8] and Latent Semantic Analysis [12] were also proposed. Jia and Wang [10] use the two-dimensional hidden markov chains to annotate images. These works operate mainly on the local level (consider the objects of an image).

Our work, as the work of Barnard and Forsyth [1], concerns the global level. We use a prediction model (neighborhood graphs) to annotate an image collection. The method we are proposing can be adopted easily for a local level annotation.

3 Neighborhood Graphs

Neighborhood graphs are used in various systems. Their popularity is due to the fact that the neighborhood is determined by coherent functions which reflect, in some point of view, the mechanism of the human intuition. Their use is varied from information retrieval systems to geographical information systems.

In order to avoid some problems related to the use of the *k-NN*, the use of neighborhood graph was proposed in [15]. Neighborhood graphs, or proximity graphs, are geometrical structures which use the concept of neighborhood to determine the closest points to a given point. For that, they are based on dissimilarity measures [21]. We will use the following notations throughout this paper: Let Ω be a set of points in a multidimensional space R^d. A graph $\mathcal{G}(\Omega,\varphi)$ is composed by the set of points Ω and a set of edges φ. Then, for any graph we can associate a binary relation upon Ω, in which two points $(\alpha, \beta) \in \Omega^2$ are in binary relation if and only if the pair $(\alpha, \beta) \in \varphi$. In an other manner, (α, β) are in binary relation if and only if they are directly connected in the graph $\mathcal{G}$. From this, the neighborhood $\mathcal{N}(\alpha)$ of a point α in the graph $\mathcal{G}$, can be considered as a sub-graph which contains the point α and all the points which are directly connected to it.

Several possibilities were proposed for building neighborhood graphs. Among them we can quote the Delaunay triangulation [14], the relative neighborhood graphs [20], the Gabriel graph [5], and the minimum spanning tree [14]. In this paper, we'll consider only one of them, the relative neighborhood graph *RNG*(see Figure 1). Two points $(\alpha, \beta) \in \Omega^2$ are neighbors in this graph if they check the relative neighborhood property defined hereafter. Let $\mathcal{H}(\alpha, \beta)$ be the hyper-sphere of radius $\delta(\alpha, \beta)$ and centered on α, and let $\mathcal{H}(\beta, \alpha)$ be the hyper-sphere of radius $\delta(\beta, \alpha)$ and centered on β. $\delta(\alpha, \beta)$ and $\delta(\beta, \alpha)$ are the dissimilarity measures between the two points α and β. $\delta(\alpha, \beta) = \delta(\beta, \alpha)$. Then, α and β are neighbors if and only if the lune $\mathcal{A}(\alpha, \beta)$ formed by the intersection of the two hyper-spheres $\mathcal{H}(\alpha, \beta)$ and $\mathcal{H}(\beta, \alpha)$ is empty [20]. Formally:

$$\mathcal{A}(\alpha, \beta) = \mathcal{H}(\alpha, \beta) \cap \mathcal{H}(\beta, \alpha) \;\; Then \; (\alpha, \beta) \in \varphi \; iff \; \mathcal{A}(\alpha, \beta) \cap \Omega = \phi$$

4 Contribution

In this section, we will consider the following question:"*Having a set of annotated images, how can we proceed in order to annotate a new image introduced without annotations?*". Formally, let us consider Ω a set of n images $\Omega = \{I_1, I_2, ..., I_n\}$. Each image I_i is described by a set of features $< f_1, f_2, ..., f_m >$ representing

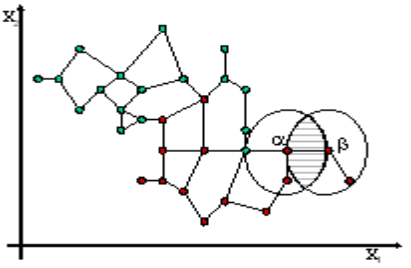


Fig. 1. Relative neighborhood graph

its low level characteristics and a list of keywords $W_i = w_1, w_2, ..., w_k$. Then, an image I_i can be described by a vector $I_i =< f_1, f_2, ..., f_m, W_i >$ where each image can have a different number of keywords.

From that, having a new unlabeled image $I_x =< f_1, f_2, ..., f_n >$, it is a question of finding a model able to assign to the image I_x the labels being able to describe its semantic aspect as close as possible. In other words, the goal is to pass from a representation in the form of $I_x =< f_1, f_2, ..., f_n >$ to a representation in the form of $I_x =< f_1, f_2, ..., f_n, W_x >$.

We believe that the problem of images annotation passes by two levels: the data modeling (indexing) and the decision-making (effective annotation). These two steps are described in the following.

4.1 Multidimensional Indexing

This is the first step in the annotation process. This step exploits the low level characteristics (color, texture, etc.) extracted from the images. The images are then represented in a multidimensional space R^p. The objective of this step, in addition to the fast access to the data which is the main objective of an index, is to make easy the neighbors location of an image (the images having rather similar low level characteristics) in the considered multidimensional space.

Concretely, during this step the images of the database are processed and transformed into a set of low level features vectors. Each image is then represented as a point in R^p and the graph is then built with respect to the corresponding neighborhood property. At the end of this step, we obtain a graph representation of the image database and the system can answer to queries in the Query by Image Content form. This step is described in [7].

4.2 Decision-Making (Effective Annotation)

In this phase, we assume that the data are indexed using a neighborhood graph. The goal here is the effective attribution of the annotations to a new unlabeled image. The main principle is the "*heritage*". Indeed, we make inherit an image, after its insertion in the neighborhood graph, from the annotations of its neighbors by calculating scores for each potential annotation.

We can consider different manners to calculate the scores for the inherited annotations. We give here a simple but a powerful function for illustrating our proposals. This possibility consists in the consideration of the distances between the query point and its neighbors. We use a weighting function and we give a more important decision power to the nearest neighbors. So, more an image is near (among the neighbors of the query image) more its decision power is important. The attribution of the weights is performed using the following formula:

$$W_i = 1 - \frac{\delta(\beta_i, \alpha)}{\sum_{j=1}^{l} \delta(\beta_j, \alpha)}$$

where W_i: The weight affected to the neighbor i, and $\delta(\beta_j, \alpha)$: The distance between the neighbor β_j and the query point α.

After the weights attribution, a score is calculated for each annotation in the neighborhood using the following function:

$$S_t = \frac{\sum_{j=1[t \in A(\alpha) \ and \ \beta_j \in V(\alpha) \ ]}^{l} W_j}{\sum_{j=1}^{l} W_j}$$

Where $V(\alpha)$ is the set of the neighbors of the query image α, t is a specific annotation in the neighborhood, l is the number of neighbors of the query image α, and β_j is the neighbor j of the query image α .

4.3 Scalability of the Proposed Method

With the approach we are proposing, the decisions are taken in a coherent manner thanks to the use of the neighborhood graphs (Use a similarity measure and the topology of the points in the multidimensional space). However, neighborhood graphs suffer from a major problem due especially to their high complexity. Indeed, neighborhood graphs are efficient only when deal in with a few datasets because of their complexity of $O(n^3)$.

The construction principle of the neighborhood graphs consists in seeking for each point if the other points in the space are in its proximity. The cost of this operation is of complexity $O\left(n^3\right)$ (n is the number of points in the space). Toussaint [21] proposed an algorithm of complexity $O\left(n^2\right)$. He deduces the RNG starting from a Delaunay triangulation [14]. Using the Octant neighbors, Katajainen [9] also proposed an algorithm of complexity. Smith [18] proposed an algorithm of complexity $O\left(n^{23/12}\right)$ which is less significant than $O\left(n^3\right)$. The major problem with these algorithms is the fact that they are not able to update the initial structure without rebuilding the whole structure.

The direct use of these algorithms is not suitable even with their low construction complexity compared to the standard algorithm. This is due to two reasons: (1) the large mass of the databases in the context of image databases, and (2) an image database is currently updated with new images (other images can be removed), unfortunately, the described algorithms do not offer a good

results with regards to this function. We describe in the following an interesting method for updating quickly and efficiently a neighborhood graph.

The neighborhood graph local update task passes by the location of the inserted (or removed) point as well as the points which can be affected by the update. To achieve this, we proceed in two main stages: initially, we look for an optimal space area which can contain a maximum number of potentially closest points to the query point. The second stage is done in the aim of filtering the items found beforehand in order to recover the real closest points to the query point by applying an adequate neighborhood property. This last stage causes the effective updating of the neighborhood relations between the concerned points.

The main stage in this method is the "search area" determination. This can be considered as a question of determining an hyper sphere of center α (the query point) which maximizes the chance of containing the neighbors of α while minimizing the number of items that it contains.

We exploit the general neighborhood graphs structure in order to establish the radius of the hyper sphere. We are focusing especially on the nearest neighbor and the farther neighbor concepts. So, two observations in connection with these two concepts seem to be interesting:

- The neighbors of the nearest neighbor are potential candidates for the neighborhood of the query point α.
  From this, by generalization, we can deduce that:
- All the neighbors of a point are also candidates to the neighborhood of a query point to which it is a neighbor.

With regard to the first step, the radius of the hyper-sphere which respect the above properties is the one including all the neighbors of the first nearest neighbor of the query point. So, considering that the hyper-sphere is centered in α, its radius will be the sum of the distances between α and its nearest neighbor and the one between this nearest neighbor and its further neighbor.

The content of the hyper sphere is processed in order to see whether there are some neighbors (or all the neighbors). The second step constitutes a reinforcement step and aims to eliminate the risk of losing neighbors or including bed ones. This step proceeds so as to take advantage of the second observation. So, we take all the truths neighbors of the query point recovered beforehand (those returned in the first step) as well as their neighbors and update the neighborhood relations between these points. An algorithm summarizing the different steps of the method is discussed in [6].

That is, let consider α be the query point and β its nearest neighbor with a distance δ_1. Let consider λ be the further neighbor of β with a distance δ_2. The radius SR of the hyper sphere can be expressed as:

$$SR = (\delta_1 + \delta_2) \times (1 + \epsilon)$$

$1+\epsilon$ is a wideness factor. ϵ is a relaxation parameter, it is included in the interval $[0, 1]$ and can be fixed according to the state of the data (their dispersion for example) or by the domain knowledge. We fixed experimentally this parameter to 0.1.

The complexity of this method is very low and meets perfectly our starting aims (locating the neighborhood of points in an as short as possible time). It is expressed by:

$$O(2n + n'^3)$$

With

- n: the number of items in the database.
- n': the number of items in the hyper sphere ($<< n$).

This complexity includes the two stages described previously, namely, the search of the radius of the hyper sphere and the seek of the correct neighbors which are in it corresponding to the term $O(2n)$. This step includes the search of the first-nearest neighbor ($O(n)$) and the search of the points contained in the hyper-sphere. The second term corresponds to the necessary time for the effective update of the neighborhood relations between the real neighbors which is very weak taking into account the number of candidates turned over. This complexity constitutes the maximum complexity and can be optimized by several ways. The most obvious way is to use a fast nearest neighbor algorithm. With this method, neighborhood graphs can be adopted to the context of large databases. Figure 2 and Figure 3 illustrate and summarize the principle of the method.

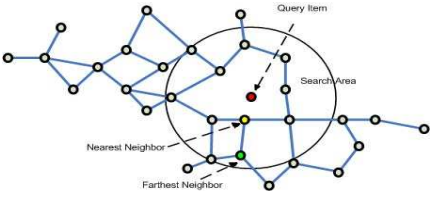


Fig. 2. Illustration of the first step of the local updating method(Retrieving of the search Area)

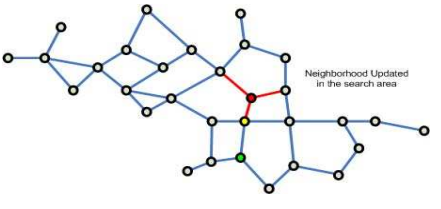


Fig. 3. Illustration of the second step of the local updating method (application of the desired neighborhood property in the search area)

5 Experiments and Results

The validity and the utility of the obtained results using the method were shown and discussed in [6] and its application to content based image retrieval is discussed in [7]. In this section, we will evaluate the images annotation.

The interest here is to show the utility of the suggested method for annotating real image databases. To achieve that, we use an image database containing 1.000 heterogeneous images[1]. We pre-process the image database to extract the low

[1] This database is available at http://wang.ist.psu.edu/docs/home.shtml

level characteristics. We extract here only the color histograms of the three bands of the RGB color space. We amputate from each histogram component the first thirty values (black) and the last thirty values (white). So, at the end, each image is represented by 588 color features. Figure 4 illustrates a visual example about the obtained results by retrieving the database using two queries in an image form.



Fig. 4. An example of a query by image content using two queries in an image form

After the low level features extraction, we annotate also the whole images and we divide the database into two subsets:

- *Training set:* This set consists 70% of the images taken arbitrary. This first dataset will be used to build the initial neighborhood graph. Let us announce that the graph is built exclusively with the low level characteristics extracted from the images, the annotations are not used during this stage.
- *Test set:* This set consists of 30% of the database's items (the remaining items), it is intended to be annotated using our approach. To annotate an image belonging to this set, we introduce it as a query image in the graph, previously built using the training set, and it is positioned (its neighborhood is located and is updated) by using its low level characteristics. Note that the initial annotations of these images are used only to compare the annotations affected by the system and those affected by the user in order to evaluate the general behavior of the approach.

The graphic of Figure 5 illustrates the obtained results.

So according to the results, we have some images which were not completely annotated with the good annotations. This represents 35% the image database. However, the results seem to be interesting since we obtain more than 60% of the items with more than 50% of good annotation rate. Note that we performed a simple experiments here and the results can be improved by using more low level features (texture features for example).

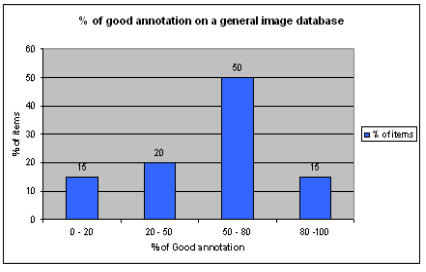


Fig. 5. Statistics of good annotation of the general image database

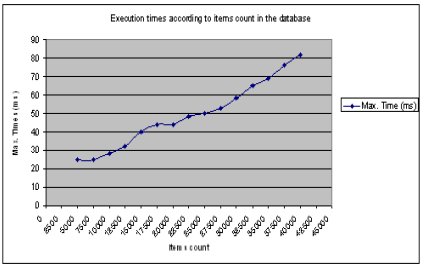


Fig. 6. Scability of the method

6 Conclusion and Future Work

Content based information retrieval in image databases is a complex task because of, mainly, the nature of the images data and the related subjectivity to their interpretation. The development of adequate techniques for the exploitation of the semantics of these data is necessary. Annotation constitutes the main tool for associating a semantic to an image. In this article we were interested in the semi-automatic annotation of images. Indeed, with the great mass of data managed throughout the world and especially with the advent of the Web, the manual annotation of these images is almost impossible. We proposed an approach based on neighborhood graphs offering several possibilities: Content based retrieval, keywords based retrieval, and the annotation with vote techniques. The approach that we have proposed offers very interesting annotation results.

As future work, we plan to use other image analysis algorithm to improve the annotation results since this is an important point as we showed it in the experiments. Also, we plan to use another more important general database to confirm and improve the results, this will enables us to compare our work with other methods such as those quoted in the state of the art. Also, our approach deals with the semi-automatic annotation, the next step is naturally the integration of the user's feedback in the annotation process. In another hand, some annotation conflicts appear, we plan to use an external knowledge (like ontology) to prevent these conflicts.

References

1. K. Barnard, P. Duygulu, and D. A. Forsyth. Clustering art. In *CVPR (2)*, pages 434–441, 2001.
2. K. Barnard and D. A. Forsyth. Learning the semantics of words and pictures. In *ICCV*, pages 408–415, 2001.
3. E. Celebi and A. Alpkocak. Semantic image retrieval and auto annotation by covering keyword space to image space. In *MMM. Beijing, China*, pages 153–160, 2006.

4. P. Duygulu, K. Barnard, J. F. G. de Freitas, and D. A. Forsyth. Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. In *ECCV (4)*, pages 97–112, 2002.
5. K. R. Gabriel and R. R. Sokal. A new statistical approach to geographic variation analysis. *Systematic zoology*, 18:259–278, 1969.
6. H. Hacid and A. D. Zighed. An effective method for locally neighborhood graphs updating. In *DEXA*, pages 930–939, 2005.
7. H. Hacid and A. D. Zighed. Content-based image retrieval using topological models. In *12th International MultiMedia Modelling Conference (MMM 06), Beijing, China*, pages 308–311, 2006.
8. J. Jeon, V. Lavrenko, and R. Manmatha. Automatic image annotation and retrieval using cross-media relevance models. In *SIGIR*, pages 119–126, 2003.
9. J. Katajainen. The region approach for computing relative neighborhood graphs in the lp metric. *Computing*, 40:147–161, 1988.
10. J. Li and J. Z. Wang. Automatic linguistic indexing of pictures by a statistical modeling approach. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(9):1075–1088, 2003.
11. O. Maron and A. L. Ratan. Multiple-instance learning for natural scene classification. In *ICML*, pages 341–349, 1998.
12. F. Monay and D. Gatica-Perez. On image auto-annotation with latent space models. In *ACM Multimedia*, pages 275–278, 2003.
13. R. W. Picard and T. P. Minka. Vision texture for annotation. *Multimedia Syst.*, 3(1):3–14, 1995.
14. F. Preparata and M. I. Shamos. *Computationnal Geometry-Introduction*. Springer-Verlag, New-York, 1985.
15. M. Scuturici, J. Clech, V. M. Scuturici, and D. A. Zighed. Topological representation model for image databases query. *Journal of Experimental and Theoritical Artificial Intelligence (JETAI)*, pages 145–160, 2005.
16. J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.
17. J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(8):888–905, 2000.
18. W. D. Smith. Studies in computational geometry motivated by mesh generation. *PhD thesis, Princeton University*, 1989.
19. Y. M. H. Takahashi and R. Oka. Image-to-word transformation based on dividing and vector quantizing images with words. In *Proceedings of the International Workshop on Multimedia Intelligent Storage and Retrieval Management*, pages 341–349, 1999.
20. G. T. Toussaint. The relative neighborhood graphs in a finite planar set. *Pattern recognition*, 12:261–268, 1980.
21. G. T. Toussaint. Some insolved problems on proximity graphs. *D. W Dearholt and F. Harrary, editors, proceeding of the first workshop on proximity graphs. Memoranda in computer and cognitive science MCCS-91-224. Computing research laboratory. New Mexico state university Las Cruces*, 1991.
22. R. C. Veltkamp and M. Tanase. Content-based image retrieval systems : A survey. Technical Report UU-CS-2000-34, Department of Computing Science, Utrecht University, 2000.

Bridging the Gap Between Visual and Auditory Feature Spaces for Cross-Media Retrieval*

Hong Zhang and Fei Wu

The Institute of Artificial Intelligence, Zhejiang University,
HangZhou, 310027, P.R. China
zhanghong_zju@yahoo.com.cn, wufei@cs.zju.edu.cn

Abstract. Cross-media retrieval is an interesting research problem, which seeks to breakthrough the limitation of modality so that users can query multimedia objects by examples of different modalities. In this paper we present a novel approach to learn the underlying correlation between visual and auditory feature spaces for cross-media retrieval. A semi-supervised Correlation Preserving Mapping (SSCPM) is described to learn the isomorphic SSCPM subspace where canonical correlations between original visual and auditory features are furthest preserved. Based on user interactions of relevance feedback, local semantic clusters are formed for images and audios respectively. With the dynamic spread of ranking scores of positive and negative examples, cross-media semantic correlations are refined, and cross-media distance is accurately estimated. Experiment results are encouraging and show that the performance of our approach is effective.

Keywords: Cross-media retrieval, canonical correlation, relevance feedback, dynamic cross-media ranking.

1 Introduction

Content-based multimedia retrieval attempts to provide an effective and efficient tool for searching interested media objects. Current approaches include image retrieval [1][2][3], audio retrieval [4][5], video retrieval [6], etc. However, cross-media retrieval [10][8] which breakthroughs the restriction of modality during retrieval process is rarely concerned and left open.

The fundamental challenge in cross-media retrieval lies in the heterogeneity of different low-level feature spaces. It's uneasy to judge the similarity between an image with 200-dimensional visual features and an audio with 500-dimensional auditory features. In this paper, we propose a novel approach to build mapping from heterogeneous visual and auditory feature spaces to a semantic subspace. The proposed Semi-supervised Correlation Preserving Mapping (SSCPM) algorithm not

* This research is supported by National Natural Science Foundation of China (No.60533090, No.60525108), Science and Technology Project of Zhejiang Province (2005C13032, 2005C11001-05), and China-US Million Book Digital Library Project (www.cadal.zju. edu.cn).

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 596 – 605, 2007.
© Springer-Verlag Berlin Heidelberg 2007

only solves the problem of heterogeneity but also fuses semantic information, namely cross-media correlations, into the mapping process. Moreover, we utilize user interactions to fuse prior knowledge into the system and globally refine cross-media semantic relationship. Cross-media distance is accurately estimated in semantic subspace.

The organization of this paper is as follows. First, section 2 describes how to discover cross-media correlation and construct SSCPM. Section 3 presents subspace learning methods from user interactions to improve cross-media retrieval. The experimental results are shown in section 4. Conclusions and future work are given in the final part.

2 Mining Cross-Media Correlation

Because of the well-known gap between low-level visual-acoustical features and high-level semantic concepts, traditional statistical methods, such as PCA and ICA, can't effectively enable cross-media retrieval by dimension reduction. In this section, we present a semi-supervised subspace mapping approach to discover underlying canonical correlations [7] between visual-acoustical features and solve the problem of heterogeneity effectively.

2.1 Canonical Correlation Analysis of Visual and Auditory Features

Intuitively, image samples and audio samples can be considered as two different representations of a certain semantic concept. If the correlation between two different representations is learned and modeled, images of similar semantics can be retrieved by query examples of audios according to this correlation clue.

The underlying idea of Canonical Correlation Analysis [7] is very intuitive: it looks for basis vectors for two sets of variables such that the correlation between the projections of the variables onto these basis vectors are mutually maximized. Formally, let $S_x = (x_1, ..., x_n)$ denote image dataset and $S_y = (y_1, ..., y_n)$ denote audio dataset, where $x_i = (x_{i1}, ..., x_{ip})$ represents image feature vector and $y_i = (y_{i1}, ..., y_{iq})$ represents audio feature vector. We do the following projection on S_x and S_y:

$$S_x \xrightarrow{W_x} S_x' =< x_1', ..., x_n' >,\ x_i' = (x_{i1}', ..., x_{im}');\ S_y \xrightarrow{W_y} S_y' =< y_1', ..., y_n' >,\ y_i' = (y_{i1}', ..., y_{im}') \quad (1)$$

Then the problem of correlation preserving boils down to finding optimal W_x and W_y, which makes the correlation between S_x' and S_y' is maximally in accordance with that between S_x and S_y. In other words the function to be maximized is:

$$\rho = \max_{Wx,Wy} corr(S_x W_x, S_y W_y) = \max_{Wx,Wy} \frac{(S_x W_x, S_y W_y)}{\|S_x W_x\| \|S_y W_y\|} = \max_{Wx,Wy} \frac{W_x' C_{xy} W_y}{\sqrt{W_x' C_{xx} W_x W_y' C_{yy} W_y}} \quad (2)$$

where C is covariance matrix. Since the solution of equation (2) is not affected by re-scaling W_x or W_y either together or independently, the optimization of ρ is

equivalent to maximizing the numerator subject to $W_x'C_{xx}W_x=1$ and $W_y'C_{yy}W_y=1$. Then with Lagrange multiplier method we can get equation (3):

$$C_{xy}C_{yy}^{-1}C_{yx}W_x=\lambda^2C_{xx}W_x \tag{3}$$

which is a generalized Eigenproblem of the form $Ax=\lambda Bx$. And the sequence of W_x's and W_y's can be obtained by solving the generalized eigenvectors.

2.2 Semi-supervised Correlation Preserving Mapping

In order to discover cross-media correlations, namely canonical correlation between visual features and auditory features, first we need to manually label images and audios with certain semantics. It is a tedious process if the training database is very large. We present a Semi-supervised Correlation Preserving Mapping (SSCPM) method based on partially labeled data. Given unlabeled image and audio database, and suppose the number of semantic categories is also given, SSCPM can be described as follows:

1. **Semi-supervised clustering.** We randomly label several image examples A_i for each semantic category Z_i; calculate image centroid $ICtr_i$ for each labeled example sets A_i; employ K-means clustering algorithm [11] on the whole image dataset (labeled and unlabeled) with $ICtr_i$ selected as initial centroids. Conduct above operations on audio dataset. Then the images (or audios) in the same cluster are considered to represent the same semantics, and grouped into the same semantic category.
2. **Correlation preserving mapping.** Let S_x denote visual feature matrix of category Z_i, and S_y denote the corresponding auditory feature matrix of Z_i; we find optimal W_x and W_y for S_x and S_y (see subsection 2.1); construct SSCPM subspace S^m that optimizes the correlation between corresponding coordinates by: $S_x'=S_xW_x$ and $S_y'=S_yW_y$.

In this way, visual features are analyzed together with auditory features, which is a kind of "interaction" process. For example, dogs' images are analyzed together with dogs' audios. Therefore, images affect the location of audios to a certain extent in the subspace S^m, and vice versa. Since "dog" auditory features differ from "bird" auditory features, "dog" visual features will be located differently in SSCPM subspace S^m from that of "bird" visual features.

3 Semantic Refinement from User Interactions

One problem of the SSCPM algorithm is: when partially labeled images and audios are projected into subspace S^m, the topology of multimedia dataset is not always consistent with human perception. Regarding the problem, we present solutions to discover local and global semantic structures based on user interactions, and construct

a semantic subspace containing both image and audio points. In the following description, X denote image database, and Y denote audio database.

3.1 Learning Local Semantic Cluster

SSCPM is based on the semi-supervised clustering, which requires much less manual effort but is not very robust compared with the supervised methods. Therefore, we refine image distance matrix and audio distance matrix in S^m, build local semantic clusters for images and audios respectively.

Let I denote image distance matrix in S^m, $I_{ij} = \left\| Pos(x_i) - Pos(x_j) \right\|, (x_i, x_j \in \mathrm{X})$ where $Pos(x_i)$ is x_i's coordinates in S^m obtained in section 2.2. Here we describe a simple method to update matrix I gradually. Intuitively, the images marked by user as positive examples in a query session share some common semantics. So we can shorten the distances between them by multiplying a suitable constant factor smaller than 1. Similarly, we can lengthen the distance between the positive images and negative images by multiplying a suitable constant factor greater than 1. In subspace S^m, audio points are represented in the same form of vectors as images are, so audio distance matrix A can be updated in the same way.

As users interact with the retrieval system, matrix I and A will gradually reflect the similarity within images and within audios in semantic level. Thus, we label the updated S^m as semantic subspace S^{m*}.

The topology of image dataset and audio dataset in semantic subspace S^{m*} differs from its initial topology in subspace S^m. We construct local semantic clusters in S^{m*} with three steps: (1) Employ Multidimensional Scaling (MDS) [9] to find meaningful underlying dimensions that explain observed image similarities, namely distance matrix I; (2) Employ MDS to find meaningful underlying dimensions that explain distance matrix A; (3) Use K-means clustering algorithm [11] to recover image semantic clusters and audio semantic clusters in S^{m*}.

3.2 Dynamic Cross-Media Ranking

There are two heuristic rules that are quite helpful for us to estimate cross-media distance in S^{m*}: (1) Positive examples are probably surrounded in a certain area with "less-positive" ones of the same modality. (2) Negative examples are probably surrounded in a certain area with "less-negative" ones of the same modality.

Let E denote Euclidean distance between images and audios in S^m, $E_{ij} = \left\| Pos(x_i) - Pos(y_j) \right\|, (x_i \in \mathrm{X}, y_j \in \mathrm{Y})$. Based on above two heuristic rules, we define cross-media similarity between x_i and y_j as:

$$F_{ij} = \alpha E_{ij} + (1-\alpha) R_{ij} \tag{4}$$

where α is a parameter in $(0,1)$, and R_{ij} is the cross-media ranking score which is initially obtained from user interactions and dynamically spreaded through local

semantic clusters. We introduce R_{ij} to refine cross-media similarity and make the system more efficient. Let r be an image query example, P denote the set of positive audios marked by the user in a round of relevance feedback, and N denote the set of negative audios. Given $p_i \in P$ or $n_i \in N$, suppose p_i or n_i belongs to the semantic cluster C_i in S^{m^*}, k_i is used to denote the number of audio points in semantic cluster C_i. The pseudo-code to calculate R_{ij} is shown below:

Dynamic Cross-media Ranking Algorithm:

Input: distance matrices I, A, and E
Output: cross-media ranking score matrix R and cross-media similarity matrix F

Initialize $R_{ij} = 0$;
Choose a constant $-\tau$ as the initial ranking score;
for each positive audio $p_i \in P$ **do**
  $R_{(r,\,pi)} = -\tau, \tau > 0$;
  $T : \{t_1, ..., t_k\} = k_i$ -nearest audio neighbors of p_i;
  rank T in ascending order by their distances to p_i;
  $d = \tau / k_i$;
  for each $t_j \in T$ **do**
    $R_{(r,\,tj)} = -\tau + j \times d$;
  end for
end for

Choose a constant τ as the initial ranking score;
for each negative audio $n_i \in N$ **do**
  $R_{(r,\,ni)} = \tau, \tau > 0$;
  $H : \{h_1, ..., h_k\} = k_i$ -nearest audio neighbors of n_i;
  rank H in ascending order by their distances to n_i;
  $d = \tau / k_i$;
  for each $h_j \in H$ **do**
    $R_{(r,\,hj)} = \tau - j \times d$;
  end for
end for

The spread of ranking scores reflects the semantic relationship between image points and audio points. And the resultant ranking score of an audio is in proportion to the probability that it is relevant to the query image, with small ranking score indicating high probability. The accumulated cross-media knowledge is incorporated into the cross-media ranking matrix. This will accordingly update cross-media similarity so that the system's future retrieval performance can be enhanced.

3.3 Introduction of New Media Objects

If a media object is out of semantic subspace, we call it a new media object. Since semantic subspace is built on the basis of SSCPM subspace S^m, we first need to locate the new media object into SSCPM subspace. Let v denote the extracted feature vector of the new media object. There are two options to obtain its coordinates in SSCPM subspace.

W_x and W_y have been obtained as the basis vectors of subspace S^m for each semantic category, including image and audio examples (see section 2.2). Thus, if the user gives semantic information of the new media object, we can identify the corresponding W_x or W_y, and map the new media object into S^m by $Pos(v) = v \cdot W_x$ (if it is a new image object) or $Pos(v) = v \cdot W_y$ (if it is a new audio object). In most cases the semantic information is unknown, then the choice of W_x or W_y is difficult. Thus we describe another mapping method consisting of two steps:

1. Find k-Nearest neighbors of the same modality from database for the new media object using content-based Euclidean distance, and return them to users.
2. Suppose $Z = \{z_1, ..., z_j\}$ are positive examples marked by user, then the coordinates of v in S^m are defined the weighted average of Z : $Pos(v) = Pos(z_1)\beta_1 + ... + Pos(z_j)\beta_j, (\beta_1 + ... + \beta_j = 1)$.

Once the new media object is projected into SSCPM subspace, we can group it into a corresponding local semantic cluster, then dynamic cross-media ranking algorithm would spread ranking score on the new media object during a round of relevance feedback. Thus the distance in semantic subspace between the new media object and all other points are obtained.

4 Experimental Results

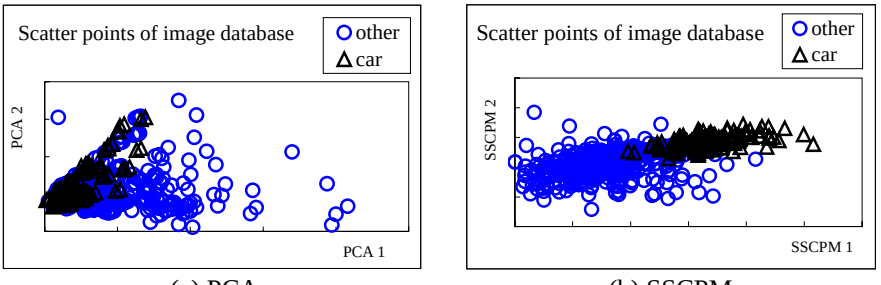
We performed several experiments to evaluate the effectiveness of our proposed methods over an image-audio dataset, which consists of 10 semantic categories, such as dog, car, bird, war, tiger, etc. The media objects are collected from Corel image galleries and the Internet. In each semantic category there are 70 images and 70 audios. The image dataset we use contains 700 images in all, and so does audio dataset. The 700 images are divided into two subsets. The first subset consists of 600 images, and each semantic category contains 60 images. The second subset consists of 100 images, and each semantic category contains 10 images. The 700 audios are grouped in the same way. Thus the first subset contains 600 images and 600 audios, and is used as training set for subspace learning. The second subset consists of 100 images and 100 audios, and is for testing. A retrieved image (or audio) is considered correct if it belongs to the same category of the query image (or audio).

Since audio is a kind of time series data, the dimensionalities of combined auditory feature vectors are inconsistent. We employ Fuzzy Clustering algorithm [4] on auditory features to do dimension reduction and get audio indexes. In our experiment,

a query is formulated by randomly selecting a sample media object from the dataset. For each query, the system returns 60 media objects as the results. We generate 5 random image queries and 5 random audio queries for each category, and conduct 5 rounds of relevance feedback for each query to form local semantic clusters.

4.1 Data Topology in SSCPM Subspace

We compare mapping results of our SSCPM method with dimensionality reduction method of PCA, which has been shown to be useful for feature transformation and selection by finding the uncorrelated components of maximum variance.



(a) PCA (b) SSCPM

Fig. 1. Scatter plots of the image dataset

Figures 1(a) and (b) show the scatter plots of the images that are projected to a two-dimensional subspace identified by the first two principal components and the first two SSCPM components. Dark triangles correspond to the category of "car" (one of the 10 categories), and the blue circles correspond to the other 9 categories. Compared with PCA in Figure 1(a), SSCPM in Figure1(b) can better separate data from different semantic classes. It can be concluded that SSCPM not only simultaneously projects heterogeneous visual and auditory features into isomorphic SSCPM subspace, but also implements data separation by semantics. Differently, PCA only removes noises and redundancies between feature dimensions of single modality. This observation confirms our previous intuition that the location of car images into SSCPM subspace is affected with the location of car audios.

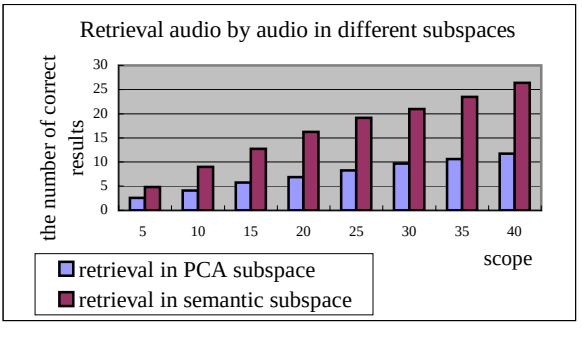
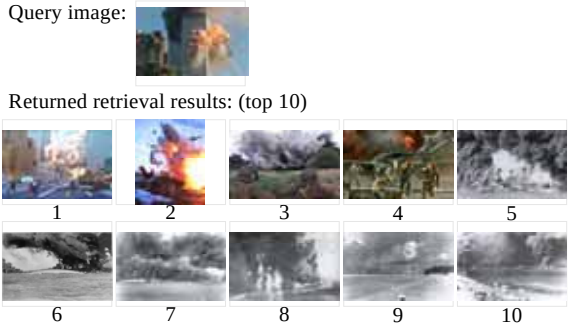


Fig. 2. Comparison of audio retrieval

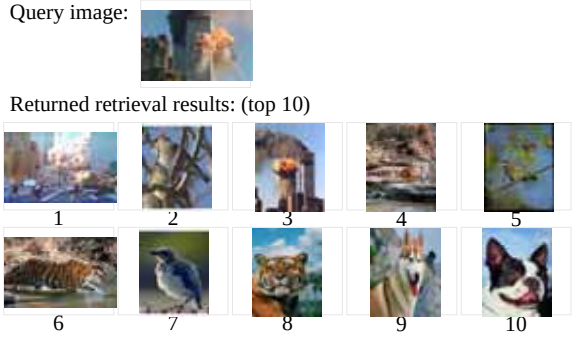
4.2 Single Modality Retrieval in Semantic Subspace

We provide the results of single modality retrieval in semantic subspace to evaluate the performance of local semantic clustering. Figure 2 shows the results of single modality audio retrieval in semantic subspace. As can be seen, our method gains great improvement compared with PCA based method. And it can be concluded that audio points cluster well in semantic subspace.

In the database most images in "war" category are black and white, which are quite difficult to be retrieved by an example of a colorized "war" image with Euclidean distance in low-level visual feature space. It's most interesting and encouraging that in our experiments with a colorful "war" image as the query example, black and white "war" images are returned to the precision about 0.64 when the number of returned images is 45. Figure 3 shows the comparisons of returned images with the same query example, but different retrieval methods. Local image semantic clusters and local audio semantic clusters are well formed with our methods.



(a) Retrieval in semantic subspace



(b) Retrieval in PCA subspace

Fig. 3. Comparison of image retrieval results in different subspaces

Overall performance (such as precision and recall) of single modality image retrieval in semantic subspace is not presented here for the following reason: the performance of single modality image retrieval in semantic subspace directly affects

that of cross-media retrieval, so the subsequent evaluations on cross-media retrieval results give an overall judgment.

4.3 Cross-Media Retrieval in Semantic Subspace

Parameter τ (see Algorithm 1) affects how deeply the ranking score R_{ij} and cross-media similarity F_{ij} are updated in a round of relevance feedback. We set τ as the difference between the maximum F_{ij} and the minimum F_{ij}, and assume two positive and two negative examples are provided at each round of relevance feedback.

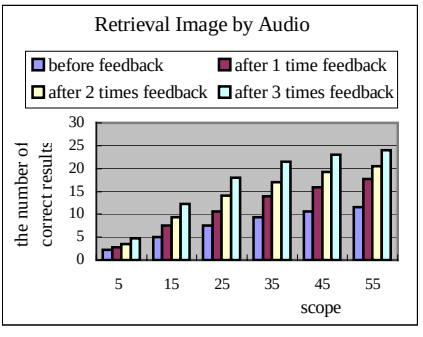


Fig. 4. Query images by examples of audios

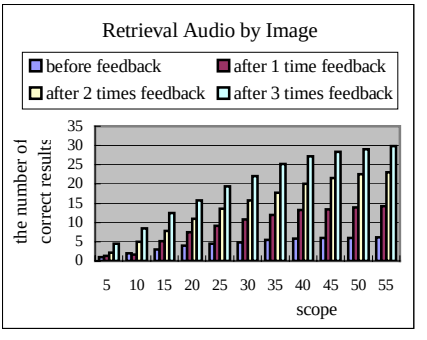


Fig. 5. Query audios by examples of images

Figure 4 shows retrieval results of querying images by examples of audios. At the third round of relevance feedback the number of correct results is 21.4 while originally it is 9.4 when the number of returned results is 35. Figure 5 shows experiment results of retrieving audios by image examples. The number of correct results is 27 when the number of returned results is 40 at the third round of relevance feedback. This observation confirms our previous intuition that the existence of image (or audio) points doesn't mess the distribution of audio (or image) points, instead, the semantic subspace gets more and more consistent with human perceptions as the user's relevance feedback is incorporated. And it can be concluded that our dynamic cross-media ranking algorithm is effective for discovering cross-media semantic relationship.

5 Conclusions and Future Work

In this paper we have investigated the problem of cross-media retrieval between images and audios with only partial information on training data labels. We develop discriminative learning methods to map heterogeneous visual and auditory feature space to an semantic subspace. Our approach gives a solution to the two fundamental problems in cross-media retrieval: how to understand cross-media correlations and how to judge the distance between media objects of different modalities. Although this paper proposes methods applied to cross-media retrieval between audio and

image objects, it is applicable to other problems of content-based multimedia analysis and understanding, such as the correlation analysis between web images and surrounding texts.

Our further work will focus on the following fields: (1) Seek for a more general cross-media model to represent media objects of more than three modalities. (2) Explore active learning strategies to better utilize informative relevance feedbacks.

References

1. Xin-jing Wang, Wei-Ying Ma, Gui-Rong Xue, Xing Li: Multi-Model Similarity Propagation and its Applications for Web Image Retrieval. 12th ACM International Conference on Multimedia, USA, 2004.
2. E. Chang, K. Goh, G. Sychay, G. Wu: CBSA: Content-based soft annotation for multimodal image retrieval using Bayes point machine, IEEE Trans on Circuits and Systems for Video Technology, vol. 13, No.1, 2003.
3. X. He, W.Y Ma, H.J. Zhang: Learning an image manifold for retrieval, ACM Multimedia Conference, pp.17-23, 2004.
4. Xueyan Zhao, Yueting Zhuang, Fei Wu: Audio clip retrieval with fast relevance feedback based on constrained fuzzy clustering and stored index table. The Third Pacific-Rim Conference on Multimedia, pp.237-244, 2002.
5. Guodong Guo; Li, S.Z.: Content-based audio classification and retrieval by support vector machines, IEEE Transactions on Neural Networks, Vol. 14, Issue 1, pp.209-215, 2003.
6. Jianping Fan, Elmagarmid, A.K., X.q. Zhu, Aref, W.G., Lide Wu: ClassView: hierarchical video shot classification, indexing, and accessing, IEEE Transactions on Multimedia, Vol. 6, Issue 1, pp.70-86, 2004.
7. D.R. Hardoon, S. Szedmak, J. Shawe-Taylor: Canonical correlation analysis; an overview with application to learning methods. Neural Computation, Vol.16, pp.2639-2664, 2004.
8. Hong Zhang, Jianguang Weng: Measuring Multi-modality Similarities from Partly Labeled Data for Cross-media Retrieval. The 7th Pacific-Rim Conference on Multimedia. pp. 979-988, 2006.
9. J.B. Tenenbaum, V.D. Silva, J.C. Langford: A global geometric framework for nonlinear dimensionality reduction, Science, Vol. 290, pp.2319-2323, 2000.
10. Fei Wu, Hong Zhang, Yueting Zhuang: Learning Semantic Correlations for Cross-media Retrieval. The 13th Int'l Conf. on Image Processing (ICIP) USA 2006.
11. E. P. Xing, A. Y. Ng, M. I. Jordan, S. Russell: Distance metric learning, with application to clustering with side-information. In NIPS 15, pp.505-512, 2003.

Film Narrative Exploration Through the Analysis of Aesthetic Elements

Chia-Wei Wang[1], Wen-Huang Cheng[2], Jun-Cheng Chen[1], Shu-Sian Yang[1], and Ja-Ling Wu[1,2]

[1] Department of Computer Science and Information Engineering
[2] Graduate Institute of Networking and Multimedia
National Taiwan University, Taipei, 10617, Taiwan, R.O.C.
{nacci,wisley,pullpull,pigyoung,wjl}@cmlab.csie.ntu.edu.tw

Abstract. In this paper, we propose a novel method for performing high-level narrative structure extraction of films. Our objective is to utilize the knowledge of film production for analyzing and extracting the structure of films. This is achieved by combining visual and aural cues on the basis of cinematic principles. An aesthetic model is developed to integrate visual and aural cues (aesthetic fields) to evaluate the aesthetic intensity curve which is associated with the film's narrative structure. Finally, we conduct experiments on different genres of films. Experimental results demonstrate the effectiveness of our approach.

1 Introduction

Film is one central part of the entertainment industry. Every year about 4,500 movies are released around the world, spanning over approximately 9,000 hours of digital movie contents, and the field is continuing to expand[1,2,6]. Since a film usually spans a long period of time and lacks organized metadata, extracting its content structures to facilitate user's access is a fundamental task in video analysis [7]. For film data, it is able to obtain the structures by analyzing the specific features called expressive elements (or aesthetic elements) that embedded in the film. Directors exploit the expressive elements to convey meanings, values and feelings during the production. Explicitly, directors create and manipulate expressive elements related to some aspects of visual or aural appeal to have perceptual or cognitive impact on the audience. Therefore, in this work, we utilize the knowledge of film production to analyze and extract the film structure.

1.1 Film Narrative Exploration

Narrative structure is the foundation upon which all stories are built to develop humans' cinematic literacy [1]. A classical narrative structure contains three basic parts called the beginning (exposition), the middle (conflict), and the end (resolution). The story intensity changes during different stages of the story. The term story intensity refers to the degree of tension that an audience feel about

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 606–615, 2007.
© Springer-Verlag Berlin Heidelberg 2007

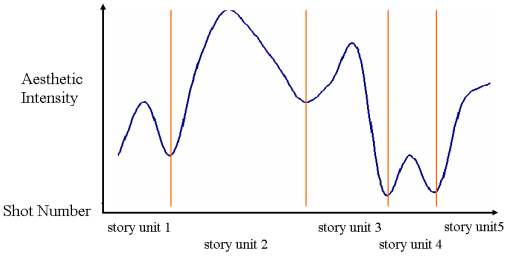


Fig. 1. A plot of story intensity curve and associated story boundaries

the story [4]. Generally, the story intensity is low at the beginning and then increases during the middle until reaches a climax. After the climax, the intensity diminishes at the end. In Figure 1, the three-part structure can be observed for each story unit. Later, this fact also helps to determine the corresponding story boundaries.

In film production, talented directors will purposely organize all movie shots to create a corresponding moods in a way that audiences will experience the same emotion enacted on the screen [8]. In addition, one of the director's major task is to emphasize a certain context or content, as for better expression of the situation of a story, in a manner such that audiences can naturally follow his way of story-telling. The storytelling now reflected through framing, light, color, objects, sounds, movement, and shot editing in a film [8]. A director applies the principle of media aesthetics to these basic aesthetic components to structure the visual and aural perception of a film [3]. For example, a director may use high energy colors to attract viewer's eyes and indicate the climaxes and emotional points of a story, etc. Therefore, directors can construct the aesthetic intensity structure that well corresponds to the story intensity structure [4].

Accordingly, it is able to detect and reconstruct such high-level mappings by extracting low-level computable features according to the principles of media aesthetics [3,9]. Zettl *et al.* [9] defined the media aesthetics as the study of how to apply the expressive elements to manipulate people's perception and helps media producers to translate significant ideas into significant messages efficiently, effectively, and predictably. Further, *Computational Media Aesthetics* proposed by Dorai *et al.* [3] provides a practical guidance for interpreting and evaluating expressive elements of films in an algorithmic way.

The rest of this paper is organized as follows. Section 2 illustrates the architecture of the proposed approach. In Section 3, we explain extractions of the adopted expressive elements and the associated aesthetic fields (light, color, movement, rhythm, and sound). The aesthetic model is presented in Section 4. Section 5 gives the experimental results and presents some applications. Finally, Section 6 concludes this paper and describes the directions of our future work.

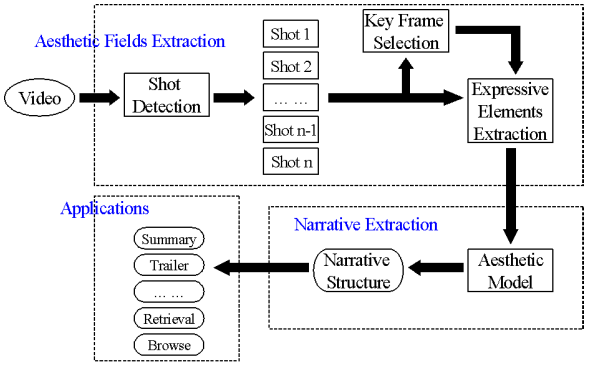


Fig. 2. The flowchart of the proposed algorithm

2 System Framework

We aim to extract the narrative structure of a film through computing the expressive elements used by film directors. Figure. 2 shows the flowchart of the proposed approach. The approach is composed of two stages: the aesthetic fields extraction and the narrative structure extraction through aesthetic modeling. In the aesthetic fields extraction stage, the first step is to explicitly detect the shot boundary between two consecutive frames. We compute the aesthetic fields associated with the expressive elements according to the principle of media aesthetics (see Section 3) on a keyframe basis. Next, in the stage of narrative structure extraction, we analyze the aesthetic fields extracted above. An aesthetic model is proposed (see Section 4) to evaluate the contribution of each field (denoted by the so-called aesthetic intensity) and obtain the narrative structure to realize some high-level video applications (see Section 5).

3 Aesthetic Field Extraction

According to the literatures [3,4,9], we identify and isolate five fundamental aesthetic fields (light, color, movement, sound, rhythm) that are computable and extractable for evaluating the aesthetic energy (intensity strength). The proposed framework is illustrated in Figure. 3. First, we compute the expressive elements, like color temperature and motion activity, directly from the keyframes of shots. Since the expressive elements themselves (such as shot length or motion activity) tell us nothing or little about the meaning expressed by directors, we further construct aesthetic fields by combining the extracted expressive elements. In this way, the so-obtained aesthetic fields are able to faithfully represent the semantic and perceptual importance of film events. Finally, we evaluate and combine the contributions of each aesthetic field and construct the aesthetic intensity structure through applying the aesthetic model to each shot. The adopted aesthetic fields are separately extracted for each keyframe and described as follows.

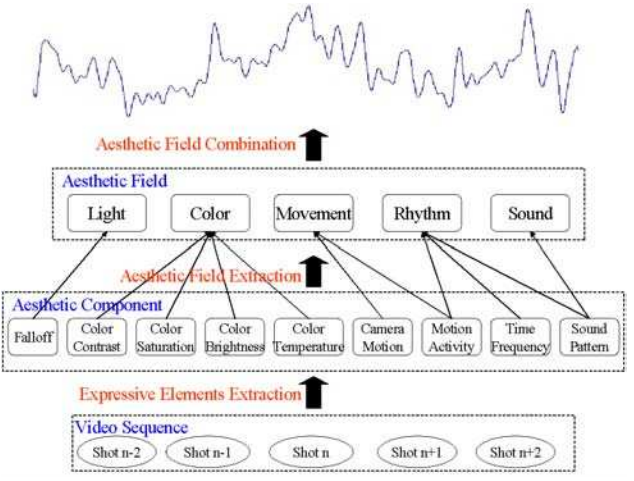


Fig. 3. The framework for extracting the aesthetic intensity curve

Light – It manipulates the perception of our environment and tells us how we would feel about a certain event. To structure the aesthetic field, light, many lighting instruments are generally used for the control of shadows than for just illuminating a scene. The brightness contrast between the light and the shadow sides of an object is referred as light falloff. To compute light falloff, we first coarsely classify the foreground and the background. Since the focused objects have more details within the object than the out-of-focus background, we adopt Wang's algorithm [12] to detect the focused objects in a frame using multi-resolution wavelet frequency method. After the classification of foreground and background, we use the idea of Mulhem *et al.* [11] to calculate the light falloff value. We calculate the luminance contrast along the boundary and linearly quantize the contrast values. Since the falloff edge often has the highest contrast, we use the average of the highest 10% contrast values along the edge as the light falloff value of the frame.

Color – It makes the audience feel in a specific way the content authors would like to communicate. We can translate colors into energies (or dynamics). The energy of a color presents the aesthetic impact on the audience, some colors seem to have high-energy and excite the audience while others seem to have low-energy and calm the audience down. Generally, it is common to use colors harmonically (high-energy color event matched by high-energy colors). The elements that influence the color energy are shown in [9].

Movement – It affects the viewers emotional reactions and impressions. When a human is watching a film, his emotional reactions and impression are often affected by the movement amount in the film. Generally, a larger movement will have greater visual intensity than a smaller one. We extract two kinds of movements (the object in front of the camera and the camera itself [4,9]) by

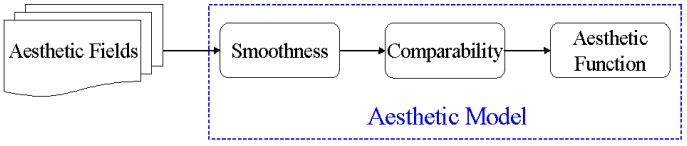


Fig. 4. The flowchart of aesthetic field modeling

using the motion activity descriptor defined in MPEG-7 [13]. The descriptor is compact and can be easily extracted in the compressed domain.

Sound – It helps to establish or supplement the visual effects of the screen event. Nonliteral sounds that refer to mainly the background and/or sound effects, can provide additional energy to a scene and quickly provide a desired mood. Since the semantic meaning and energy magnitude of literal sounds are not easy to measure, we focus on the energy of nonliteral sounds. We compute the loudness (volume) as the sound energy by the approximation of the root mean square value of the signal magnitude within a frame.

Rhythm – It is the perceived speed or felt time of an event [9]. For example, movement may produce a visual rhythm: when an actor slowly walk through a scene, the audience's felt time of this event is long and the rhythm is low; when the actor hurriedly run through the scene, the felt time is short and there is a high rhythm produced. Often the rhythm serves as a psychological guidance of audience. Generally, a faster (higher) rhythm is associated with excitement, and a slower (lower) rhythm suggests calm. Directors may control and present the rhythm by the techniques of montage (shot length), motion, and audio effects. We then adopt the formulation proposed in [14] to compute the rhythm elements.

4 Aesthetic Modeling

In this section, we explain in detail the process of evaluating the aesthetic intensity curve through integrating various aesthetic fields. Figure. 4 illustrates the procedure for modeling the aesthetic fields.

4.1 Smoothness

The aesthetic intensity of each aesthetic field is carefully smoothed via a smoothing window. The smoothing process is demanded for the following two reasons:

1) Memory is an important factor when considering the human perception. The emotional state of human is a function of a neighborhood of frames or shots and it does not change abruptly from one video frame to another.

2) Directors generally do not make the aesthetic intensity changing in a single or small number of shots. They often build a specific mood gradually from shot to shot.

In Hanjalic's original algorithm [10], a kaiser window is adopted to conduct the smoothing process. However, the memory is merely influenced by preceding

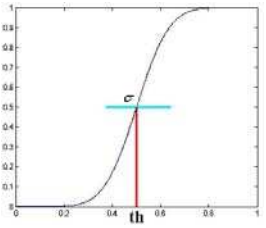


Fig. 5. Weighting function, the horizontal axis represents the value of the original curve while the vertical axis denotes the weighting value correspondingly

shots whereas the original kaiser window treats the preceding and posterior shots as equally important. Thus, we propose a modified kaiser window to reflect this property of human perception. To construct the modified kaiser window, two original kaiser windows are integrated together, both are of length 21, and the shape parameters are 3.5 and 31, respectively. We then combine the two kaiser windows into a new modified kaiser window which is then applied to conduct the smoothing process. Through the convolution with the modified kaiser window, we obtain the smoothed aesthetic intensity of each aesthetic field that takes account for the degree of memory retention of preceding frames and shots.

4.2 Comparability

This module ensures the aesthetic intensity of each aesthetic field is comparable and combinable. Each field is normalized by the shot with maximum magnitude in that field. Since the aesthetic intensities of all fields are scaled to the range between 0 and 1, they can be combined with each other on the same basis.

4.3 Aesthetic Function

As discussed previously, directors increase the energies or dynamics of aesthetic elements to emphasize the story conflicts or climax. According to the principle, we apply a filtering process to the intensity curve of each aesthetic field to provide highly distinguishable peaks at the segments of the curve corresponding to the story climax. The filtering process is performed through weighing the values of the aesthetic intensity curves. The weight function is defined as:

$$F(a(k), i = 1, ..., N) = \sum_{i=1}^{N} w_k a_k, \tag{1}$$

where

$$w_k = \frac{1}{2}(1 + erf(\frac{a_k - th}{\sigma})), \quad erf(x) = \frac{2}{\pi}\int_0^x e^{-t^2} dt. \tag{2}$$

As Figure. 5 depicts, the parameter **th** is the threshold of the original curve while σ denotes the spread factor determining the steepness of the middle curve

segment. The term a_k denote the aesthetic intensity curves that have been applied smoothing and normalizing processes as prescribed. The segments with intensity value larger than the threshold are strengthened while the other segments are lessened. Thus, we can distinctly identify the aesthetically emphasized segments and no possible climax segments are discarded.

5 Experiments and Applications

5.1 Experiments

In our experiments, five Hollywood movies are used, i.e., *Hitch (HT), Ghost (GH), Hero (HE), The Promise (TP),* and *Charlie's Angels II (CA)* to evaluate our performance. We compare the story units detected by our approach with those of both the ground truth taken from DVD chapter information and the human made results. Each of the videos was digitized at MPEG-1 format (352×240 pixels, 29.97fps). The data set is carefully selected to represent a variety of film genres such as action, drama, and romance. In addition, we examine the importance of each of the aesthetic fields.

Story Unit Segmentation. The chapter boundary detection is achieved based on the aesthetic intensity curve. As shown in Figure 1, a chapter boundary usually occurs at the point between the ending of the previous chapter unit and the opening of the next one. Since the chapter unit is usually with low aesthetic intensity at that point, we select the shot with the minimum intensity between two neighboring climax shots as the chapter boundary. We select those shots with the intensity value higher than a predefined threshold as the candidates for chapter unit climax since it can be found that the most impressive segments are often accompanied with a high aesthetic intensity values. Due to the fact that there is exactly one climax in a chapter unit and the shots near the climax are usually with higher values. For each pair of the candidate shots, if their distance is smaller than a threshold, ε_{high}, the shots with smaller intensity value are deleted from the candidate set.

Results. We compare the chapters detected by our approach with those of the ground truth (i.e., commercial DVD chapters, and manually labeled chapters). Table 2 and Table 3 show the statistics of our results as compared with the DVD chapter information and the manually labeled results, respectively. Note that a boundary would be identified as been correctly detected if it is within 35 seconds with a boundary in the ground truth. Since the chapter number in a commercial DVD is usually small to give viewers a rough idea about the video, it is reasonable that the overall recall is much higher than the precision. For real applications, the over-segmented chapters can be further grouped with further analysis. Overall, the experiment shows that our approach is successful in establishing the linkage between the computable aesthetic intensity and the abstract storyline of films.

Table 1. Comparisons with DVD chapters

Film	*HT*	*GH*	*TP*	*CA*	*HE*	Overall
Story units in ground truth	28	15	17	28	24	112
Story units detected	46	34	42	74	43	239
Correct detection	19	11	12	22	16	80
False negative	9	4	5	6	8	32
False positive	27	23	30	52	27	159
Recall	68%	73%	71%	79%	67%	71%
Precision	39%	29%	26%	28%	35%	33%

Table 2. Comparisons with Human Observers

Film	*HT*	*TP*	Overall
Story units in ground truth	36	30	66
Story units detected	44	42	86
Correct detection	26	23	49
False negative	10	7	17
False positive	18	19	37
Recall	72%	77%	74%
Precision	59%	55%	57%

Importance of Aesthetic Fields. We analyze the usefulness of each aesthetic field by removing one of the aesthetic fields at each time and re-evaluate the overall aesthetic intensity that is obtained from the reserved fields. For example, the weight parameter is 0.25 for each of the remaining four fields (note that the weight parameter is 0.2 for each of the five fields when no field is removed). From Table 3, it can be found that the overall performance drops while we remove any one of the aesthetic fields. These results show that it is essential to consider all of the aesthetic fields together.

Importance of Parameter. We also test other weighting schemes since each aesthetic field may not contribute equally to the human perception. Empirically, the weights of *rhythm*, *movement*, *sound*, *light*, and *color* are set to 0.2, 0.23, 0.11, 0.26, and 0.2, respectively. The results are shown in Table 4. It demonstrates that there is a notable gain in performance after tuning the weights. Besides, different film genres possess different art forms, and a certain weights may work the best for a particular film genre. For example, action films have more motion activity and faster rhythm than those of the other genres. The performance can be improved if taking this fact into account. We analyze each aesthetic field of *Charlie's Angel II* (an action movie) against the corresponding DVD chapter information. We found that the sound and the light fields do not work well and we decrease their weights. Empirically, the weights of *rhythm*, *movement*,

Table 3. Importance of different aesthetic fields

Feature (removed)	*rhythm*	*movement*	*sound*	*light*	*color*
Recall	63%	63%	62%	63%	49%
Precision	30%	30%	28%	31%	26%

Table 4. Performance gains from adjusting weights

Type	Linear	Tuned
Story units in ground truth	112	112
Story units detected	239	242
Correct detection	80	85
False negative	32	27
False positive	159	157
Recall	71%	76%
Precision	33%	35%

Table 5. Performances of different films for a given set of weights

Film	*CA*	*HT*	*GH*	*TP*	*HE*
Recall	+13.6%	-15.8%	-27.3%	-8.3%	-12.5%
Precision	+12.1%	-15.8%	-31.3%	-10.5%	-21.6%

sound, *light*, and *color* are set to 0.2, 0.22, 0.14, 0.14, and 0.3, respectively. The performances of each film under the given weights are listed in Table 5. There is a remarkable performance gain in *Charlie's Angel II* while the performances of the other films drop drastically. Therefore, automatic weights selection for different film genres is an important issue and will be the major direction of our future work.

5.2 Applications

As described in [5], identification and extraction of the high-level narrative structure associated with the expressive elements and the form of story in films opens the way for more sophisticated applications to meet the demands of the audience. For example:

1) It helps to automatically generate video indexes and makes it possible for query specification in semantic terms such as "Where is the most intense part of the movie?" or "How long is the first story unit last?", etc. Generally speaking, the higher the level of the structure is, the more efficient the search would be.

2) It locates the important boundaries of a film or a story segmentation to meet viewers' need to gain more control of what they see, e.g., DVD options are being made for users to randomly view a specific story unit of the movie.

3) It enables us to give the summaries of movies for efficiently browsing and previewing the movie.

6 Conclusion and Future Work

We proposed a method to perform high-level narrative structure extraction of films. We demonstrate that combining visual and aural cues with the aid of cinematic principles can provide significant performance for extracting the corresponding narrative structure. In the future, we are interested in concatenating small story units into longer and more meaningful ones for further applications.

References

1. N. Abrams, I. Bell, and J. Udris, *Studying Film.* London: Hodder Headline Group and NY: Oxford University Press, 2001.
2. J. Monaco, *How to Read a Film, 3ed.* NY: Oxford University Press, 2000.
3. C. Dorai and S. Venkatesh, *Media Computing: Computational Media Aesthetics.* Boston/Dordrecht/London: Kluwer Academic Publisher, 2002.
4. B. Block, *The Visual Story: Seeing the Structure of Film, TV, and New Media.* Boston: Focal Press, 2001.
5. B. Adams, C. Dorai, S. Venkatesh, and H. H. Bui, "Indexing narrative structure and semantics in motion pictures with a probabilistic framework," *IEEE International Conference on Multimedia and Expo* (ICME'03), vol. 2, pp. II 453-456, July 2003.
6. Z. Rasheed and M. Shah, "Detection and representation of scenes in videos," *IEEE Transactions on Multimedia,* vol. 7, no. 6, pp. 1097-1105, Dec 2005.
7. R. Yong, S. H. Thomas, and S. Mehrotra, "Constructing table-of-content for videos," *Multimedia Systems,* vol. 7, pp. 359-368, Sept 1998.
8. R. W. Picard, *Affective Computing.* MA: The MIT Press, 1997
9. H. Zettl, *Sight, Sound, Motion: Applied Media Aesthetics.* SF: Wadsworth, 1973.
10. A. Hanjalic, and L. Q. Xu, "Affective video content representation and modeling," *IEEE Transactions on Multimedia,* vol. 7, no. 1, pp. 143-154, Feb 2005.
11. P. Mulhem, M. S. Kankanhalli, Y. Ji, and H. Hassan, "Pivot Vector Space Approach for Audio-Video Mixing," *IEEE Multimedia,* vol. 10, pp.28-40, April-June 2003.
12. J. Z. Wang, J. Z, R. M. Gray, and G. Wiederhold, "Unsupervised multiresolution segmentation for images with low depth of field," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 23, no. 1, pp. 85-90, Jan 2001.
13. S. Jeannin and A. Divakaran, "MPEG-7 visual motion descriptors," *IEEE Trans. Circuits and Systems for Video Technology,* vol. 11, no. 6, pp. 720-724, June 2001.
14. H. W. Chen, J. H. Kuo, W. T. Chu, and J. L. Wu, "Action movies segmentation and summarization based on tempo analysis" *Proc. ACM SIGMM International Workshop on Multimedia Information Retrieval* (MIR'04), pp. 251-258, Oct 2004.

Semantic Image Segmentation with a Multidimensional Hidden Markov Model

Joakim Jiten and Bernard Merialdo

Institut EURECOM, BP 193, 06904 Sophia Antipolis, France
{jiten,merialdo}@eurecom.fr

Abstract. Segmenting an image into semantically meaningful parts is a fundamental and challenging task in image analysis and scene understanding problems. These systems are of key importance for the new content based applications like object-based image and video compression. Semantic segmentation can be said to emulate the cognitive task performed by the human visual system (HVS) to decide what one "sees", and relies on a priori assumptions. In this paper, we investigate how this prior information can be modeled by learning the local and global context in images by using a multidimensional hidden Markov model. We describe the theory of the model and present experiments conducted on a set of annotated news videos.

Keywords: Image Segmentation, Hidden Markov Model, 2D HMM, Block-based.

1 Introduction

Hidden Markov Models (HMM) have become increasingly popular in such diverse applications as speech recognition [1], language modeling, language analysis, and image recognition [3,9,12]. The reason for this is that they have a rich mathematical structure and therefore provide a theoretical basis for many domains. A second reason is the discovery of the Baum-Welch's training algorithm [2] which allows estimating the numerical values of the model parameters from training data.

Most of the current applications involve uni-dimensional data. In theory, HMMs can be applied as well to multi-dimensional data. However, the complexity of the algorithms grows exponentially in higher dimensions, so that, even in dimension 2, the usage of plain HMM becomes prohibitive in practice [4].

For this reason we have proposed an efficient sub-type of multi-dimensional hidden Markov model; the Dependency-Tree Hidden Markov Model [5] (DT-HMM) which preserves a reasonable computational feasibility and therefore enables us to apply it to multidimensional problems such as image segmentation.

In this paper, we explore the intrinsic ability of the DT-HMM to automatically associate pixels (or blocks of pixels) to semantic sub-classes which are represented by the states of the Markov model. To this end we enforce restrictions to the states during training, by having the training set labeled on pixel level. The performance of the model is demonstrated on a subset of the TrecVideo archive [16] which consists of 60 hours of annotated news broadcast.

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 616 – 624, 2007.
© Springer-Verlag Berlin Heidelberg 2007

The remainder of this paper is organized as follows: section 3 outlines our motivation and presents the theory of DT-HMM. We show how the training and decoding algorithms for DT-HMM keep the same linear complexity as in one dimension. Section 4 will describe the experimental setup conducted on TrecVideo 2003 data and in section 5 we conclude and suggest future work.

2 Related Work

A number of researches have introduced systems for mapping users' perception of semantic concepts to low-level feature values [8,10]. The probabilistic framework of multijects (multi-objects) and multinets by Naphade and Huang [10] maps high level concepts to low level audiovisual features by integrating multiple modalities and infer unobservable concepts based on observable by a probabilistic network (multinet). The Stanford SIMPLIcity system [13] uses a scalable method for indexing and retrieving images based on region segmentation. A statistical classification is done to group images into rough categories, which potentially enhances retrieval by permitting semantically adaptive search methods and by narrowing down the searching range in a database.

Motivated by the desire to incorporate contextual information, Li and Gray [3] proposed a 2D-HMM for image classification based on a block-based classification algorithm using a path constrained Viterbi. An attempt in associating semantics with image features was done by Barnard and Forsyth at University of California at Berkeley [14]. Using region segmentation in a pre-processing step to produce a lower number of color categories, image feature search becomes a text search. The data is modeled as being generated by a fixed hierarchy of nodes organized as a tree. The work has achieved some success for certain categories of images. But, as pointed out by the authors, one serious difficulty is that the algorithm relies on semantically meaningful segmentation which is, in general, not available to image databases.

In recent work by Kumar and Hebert at Carnegie Mellon University [15], a hierarchical framework is presented to exploit contextual information at several levels. The authors claim that the system encodes both short- and long-range dependencies among pixels respectively regions, and that it is general enough to be applied to different domains of labeling and object detection.

3 DT-HMM: Dependency-Tree HMM

For most images with reasonable resolution, pixels have spatial dependencies which should be enforced during the classification. The HMM considers observations (e.g. feature vectors representing blocks of pixels) statistically dependent on neighboring observations through transition probabilities organized in a Markov mesh, giving a dependency in two dimensions.

3.1 2D-HMM

In this section, we briefly recall the basics of 2D HMM and describe our proposed DT-HMM [5]. The reader is expected to be familiar with 1D-HMM. We denote by

$O=\{o_{ij}, i=1,\ldots m, j=1,\ldots,n\}$ the observation, for example each o_{ij} may be the feature vector of a block (i,j) in the image. We denote by $S = \{s_{ij}, i=1,\ldots m, j=1,\ldots,n\}$ the state assignment of the HMM, where the HMM is assumed to be in state s_{ij} at position (i,j) and produce the observation vector o_{ij}. If we denote by λ the parameters of the HMM, then, under the Markov assumptions, the joint likelihood of O and S given λ can be computed as:

$$\begin{aligned} P(O,S|\lambda) &= P(O|S,\lambda)P(S|\lambda) \\ &= \prod_{ij} p\left(o_{ij}|s_{ij},\lambda\right)p\left(s_{ij}|s_{i-1,j},s_{i,j-1},\lambda\right) \end{aligned} \tag{1}$$

If the set of states of the HMM is $\{s_1, \ldots s_N\}$, then the parameters λ are:

- the output probability distributions $p(o \mid s_i)$
- the transition probability distributions $p(s_i \mid s_j,s_k)$.

Depending on the type of output (discrete or continuous) the output probability distribution are discrete or continuous (typically a mixture of Gaussian distribution). We would like to point out that there are two ways of modeling the spatial dependencies between the neighbor state variables; by a causal or non-causal Markov random field (MRF). The former is referred to as Markov mesh and has the advantage that it reduces the complexity of likelihood functions for image classification [6]. The causality also enables the derivation of an analytic iterative algorithm to estimate states with the maximum a posteriori probability, due to that the total observation is progressively built from smaller parts. The state process of DT-HMM is defined by the Markov mesh.

3.2 DT-HMM

The problem with 2D-HMM is the double dependency of $s_{i,j}$ on its two neighbors, $s_{i-1,j}$ and $s_{i,j-1}$, which does not allow the factorization of computation as in 1D, and makes the computations practically intractable.

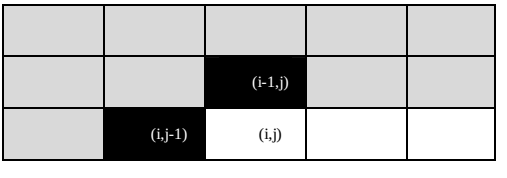


Fig. 1. 2D Neighbors

Our idea is to assume that $s_{i,j}$ depends on one neighbor at a time only. But this neighbor may be the horizontal or the vertical one, depending on a random variable t(i,j). More precisely, t(i,j) is a random variable with two possible values:

$$t(i,j) = \begin{cases} (i-1,j) & \text{with prob } 0.5 \\ (i,j-1) & \text{with prob } 0.5 \end{cases} \tag{2}$$

For the position on the first row or the first column, t(i,j) has only one value, the one which leads to a valid position inside the domain. t(0,0) is not defined. So, our model assumes the following simplification:

$$p(s_{i,j}|s_{i-1,j},s_{i,j-1},t)=\begin{cases}p_V(s_{i,j}|s_{i-1,j}) & if\ t(i,j)=(i-1,j)\\ p_H(s_{i,j}|s_{i,j-1}) & if\ t(i,j)=(i,j-1)\end{cases} \quad (3)$$

If we further define a “direction” function:

$$D(t)=\begin{cases}V & \text{if } t=(i-1,j)\\ H & \text{if } t=(i,j-1)\end{cases} \quad (4)$$

then we have the simpler formulation:

$$p(s_{i,j}|s_{i-1,j},s_{i,j-1},t)=p_{D(t(i,j))}(s_{i,j}|s_{t(i,j)}) \quad (5)$$

Note that the vector **t** of the values t(i,j) for all (i,j) defines a tree structure over all positions, with (0,0) as the root. Figure 2 shows an example of random Dependency Tree.

The DT-HMM replaces the N^3 transition probabilities of the complete 2D-HMM by $2N^2$ transition probabilities. Therefore it is efficient in terms of storage. We will see that it is also efficient in terms of computation. Position (0,0) has no ancestor. In this paper, we assume for simplicity that the model starts with a predefined initial state s_I in position (0,0). It is straightforward to extend the algorithms to the case where the model starts with an initial probability distribution over all states.

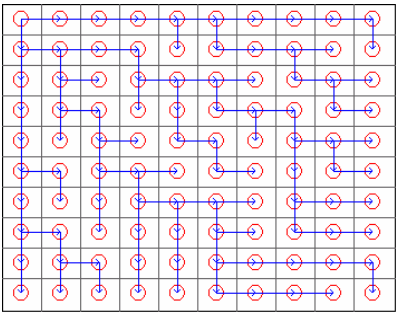


Fig. 2. Example of Random Dependency Tree

4 Application to Image Segmentation

4.1 Viterbi Algorithm

The Viterbi algorithm finds the most probable sequence of states which generates a given observation O:

$$\hat{S} = \underset{S}{\text{Argmax}}\ P(O, S|t) \tag{6}$$

The details of the algorithm for DT-HMM are given in [5][18].The algorithm is used for training the model, by iteratively reestimating the output and transition probabilities with the relative frequencies computed on the Viterbi sequences of states on the training images. It is also used for image segmentation on the test data, where each region is composed of the blocks which are covered by a given state in the Viterbi sequence.

4.2 States with Semantic Labels

We illustrate the use of DT-HMM for semantic segmentation on the example of segmenting *beach* images (class) into semantic regions (sub-classes). In principle, we should define one state of the model for each semantic region, however, to account for the variability of the visual appearance of semantic region, each semantic region (sub-class) is assigned a range of states. This potentially allows a sub-class such as *sky* to be represented by different states with dominant color blue, white, gray or yellow. The table below lists the sub-classes and their associated number of states.

Table 1. The number of states for each sub-class

Sub Class	**No. states**
Un-annotated	3
Sky	7
Sea	5
Sand	6
Mountain	3
Vegetation	3
Person	4
Building	3
Boat	2
8 sub-classes	36 states

One special class, called "*un-annotated*", is used for areas that are ambiguous or contain video graphics etc... Ambiguous areas are patches which contain several sub-classes or which are difficult to interpret.

4.3 Model Training

The training was conducted on the TrecVideo archive [16], from which we selected a wide within-class variance of 130 images depicting "Beach" (see Figure 3).

Each image is split into blocks of 16x16 pixels, and the observation vector for each block is computed as the average and variance of the LUV (CIE LUV color space) coding $\{L_\mu, U_\mu, V_\mu, L_\sigma, U_\sigma, V_\sigma\}$ combined with six quantified DCT coefficients



Fig. 3. Example of training images

(Discrete Cosine Transform). Thus each block is represented by a 12 dimensional vector. Those images have been manually segmented and annotated, so that every feature vector is annotated with a sub-class.

To define the initial output probabilities, a GMM (Gaussian Mixture Model) is trained with the feature vectors corresponding to each sub-class. We allow three GMM components for every state, so the GMM for the sub-class *sky* has 21 components and for *vegetation* (see Table 1). Then we group the components into as many clusters as there are states for this sub-class (using the k-means algorithm). Finally, the GMM model for each state is built by doubling the weight of the components of the corresponding cluster in the GMM of the sub-class. The transition probabilities are initialized uniformly. Then, during training we iterate the following steps:

- We generate a random dependency tree and perform a Viterbi alignment to generate a new labeling of the image. The Viterbi training procedure is modified to consider only states that correspond to the annotated sub-class at each position, thus constraining the possible states for the observations (the manual annotation specifies the sub-class for each feature vector, but not the state).
- We reestimate the output and transition probabilities by relative frequencies (emission of an observation by a state, horizontal and vertical successors of a state) with Lagrange smoothing.

4.4 Experimental Results

During training, we can observe the state assignments at each iteration as an indication of how the model fits the training data. For example, the first ten iterations on the training image to the left in figure 4 provide the following state assignments:



Fig. 4. State segmentation after 0, 2, 6 and 10 iterations

This shows that the model has rapidly adapted each sub-class to a particular set of observations. As such, the Viterbi labeling provides a relevant segmentation of the image. The graph below shows the evolution of likelihood of the training data during the training iterations. We can see that the likelihood for the model given the data has an asymptotic shape after 10 iterations.

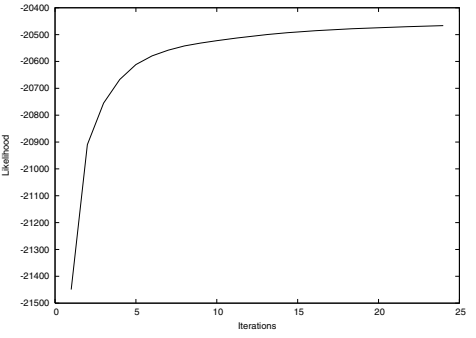


Fig. 5. Likelihood of the training data after N iterations

Once the model is trained, we can apply it on new images. Below is an example of the state assignment for an image in the test set; 70% of the blocks are correctly classified.

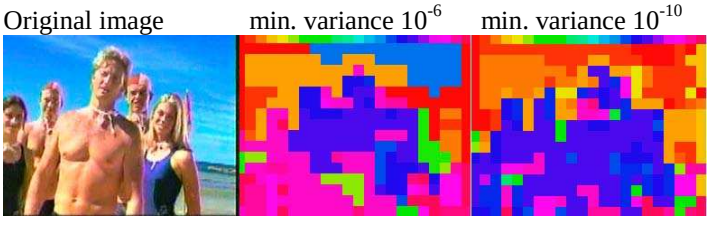


Fig. 6. State segmentations on test image

It should be emphasized that this is not just a simple segmentation of the images, but that each region is also assigned one of the 36 states (which belongs to one of the 8 sub-classes). The definition of those states has been done taking into account all training data simultaneously, and provides a model for the variability of the visual evidence of each sub-class.

During training, we impose a minimum variance for the Gaussian distributions, in order to avoid degeneracy. This minimum has an impact, as we noted that the number of correct labeled blocks in the example above increased to 72% when changing the minimum variance from 10^{-6} to 10^{-10}. An explanation for this is that if the selected minimum variance is too high, some Gaussians will be flattened out and collides with Gaussians from states representing similar observations.

Sometimes the result is degraded because of visually ambiguous regions, as in the examples below (looking through a window, or sky reflection on the sea). Because the output probabilities of model have generally a greater dynamic range than the transition probabilities, they often play the major contribution in the choice of the best state assignment.



Fig. 7. Test images with ambiguous regions

Still, to show the effect of transition probabilities, we used the model to semantically segment 40 test images. We compare the best state assignment obtained by the Viterbi algorithm (this takes into account both output and transition probabilities) with the assignment where each feature vector is assigned the state which has the highest output probability. The average rate of correctly labeled blocks was 38% when taking transition probabilities into account and 32% with only the output probabilities. Figure 8 shows an example, with the original example image, the sub-class assignment without transition probabilities (56% blocks correctly labeled), and the Viterbi assignment (72% correct).

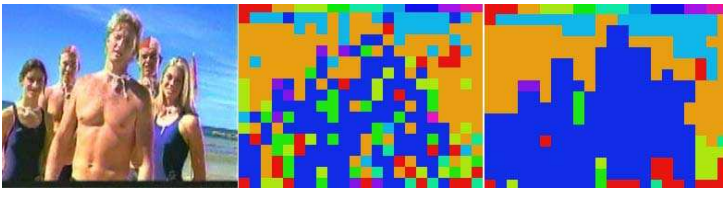


Fig. 8. Sub-class assignment without/with transition probabilities

5 Conclusions and Future Research

The contribution of this paper is to illustrate semantic segmentation of an image by a two dimensional hidden Markov model. We show how the model can be trained on manually segmented data, and used for labeling new test data. In particular, we use a modified version of the Viterbi algorithm that is able to handle the situation where a visual sub-class is represented by several states, and only the sub-class annotation (not the state annotation) is available. We investigated several properties of this process. The motivation for this approach is that it can be easily extended to an larger number of classes and sub-classes, provided that training data is available. Allowing several states per sub-class gives the model the flexibility to adapt to sub-classes which may have various visual evidence.

Acknowledgements

The research leading to this paper was supported by the Institut Eurecom and by the European Commission under contract FP6-027026, Knowledge Space of semantic inference for automatic annotation and retrieval of multimedia content - K-Space.

References

[1] Rabiner, L.R., S.E. Levinson, and M.M. Sondhi, (1983). On the application of vector quantization and hidden Markov models to speaker independent, isolated word recognition. B.S.T.J.62,1075-1105

[2] LE. Baum and T. Petrie, Statistical Inference for Probabilistic Functions of Finite State Markov Chains, Annual Math., Stat., 1966, Vol.37, pp. 1554-1563.

[3] J. Li, A. Najmi, and R. M. Gray, Image classification by a two-dimensional hidden markov model, IEEE Trans. Signal Processing, vol. 48, no. 2, pp. 517–533, 2000.

[4] Levin, E.; Pieraccini, R.; Dynamic planar warping for optical character recognition, IEEE International Conference on Acoustics, Speech, and Signal Processing, , Volume 3, 23-26 March 1992 Page(s):149 - 152

[5] Merialdo, B; Dependency Tree Hidden Markov Models, Research Report RR-05-128, Institut Eurecom, Jan 2005

[6] Kanal, L.N.: Markov mesh models in Image Modeling. New York: Academic, 1980, pp. 239-243

[7] P. F. Felzenszwalb , D. P. Huttenlocher, Image Segmentation Using Local Variation, Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, p.98, June 23-25, 1998

[8] F. Golshani, Y. Park, S. Panchanathan, "A Model-Based Approach to Semantic-Based Retrieval of Visual Information", SOFSEM 2002: 149-167

[9] O. Agazzi, S. Kuo, E. Levin, and R. Pieraccini. Connected and degraded text recognition using planar hidden Markov models. In Proc. of the IEEE Int. Conf. on Acoustics Speech and Signal Processing (ICASSP), volume 5, pages 113-116, 1993.

[10] M. R. Naphade, and T. S. Huang, "Extracting Semantics from Aduiovisual Content: The Final Frontier in Multimedia Retrieval", IEEE Transactions on Neural Network, Vol. 13, No. 4, 793--810, 2002.

[11] Merialdo, B.; Marchand-Maillet, S.; Huet, B.; Approximate Viterbi decoding for 2D-hidden Markov models, IEEE International Conference on , Acoustics, Speech, and Signal Processing, Volume 6, 5-9 June 2000 Page(s):2147 - 2150 vol.4

[12] Perronnin, F.; Dugelay, J.-L.; Rose, K.; Deformable face mapping for person identification, International Conference on Image Processing, Volume 1, 14-17 Sept. 2003 Page(s):I - 661-4

[13] J.Z. Wang, "Integrated Region-Based Image Retrieval", Dordrecht: Kluwer Academic, 2001

[14] K. Barnard and D. Forsyth, “Learning The Semantics of Words and Pictures,” Proc. Int’l Conf. Computer Vision, vol 2, pp. 408-415, 2001.

[15] S. Kumar and M. Hebert, "A Hierarchical Field Framework for Unified Context-Based Classification," Proc. ICCV, October, 2005.

[16] TREC Video Retrieval Evaluation, http://www-nlpir.nist.gov/projects/TrecVideo/

[17] J. Jiten, B. Mérialdo; “Probabilistic image modeling with dependency-tree hidden Markov models”, WIAMIS 2006, 7th International Workshop on Image Analysis for Multimedia Interactive Services, April 19-21, 2006, Incheon, Korea

[18] J. Jiten, B. Mérialdo, B. Huet;”Multi-dimensional dependency-tree hidden Markov models “, ICASSP 2006, 31st IEEE International Conference on Acoustics, Speech, and Signal Processing, May 14-19, 2006, Toulouse, France

Semi-supervised Cast Indexing for Feature-Length Films

Wei Fan[1], Tao Wang[2], JeanYves Bouguet[2], Wei Hu[2], Yimin Zhang[2], and Dit-Yan Yeung[1]

[1] Department of Computer Science and Engineering,
Hong Kong University of Science and Technology, Hong Kong
{fwkevin, dyyeung}@cse.ust.hk
[2] Intel China Research Center, Beijing, P.R. China, 100080
{tao.wang, Jean-yves.bouguet, wei.hu, yimin.zhang}@intel.com

Abstract. Cast indexing is a very important application for content-based video browsing and retrieval, since the characters in feature-length films and TV series are always the major focus of interest to the audience. By cast indexing, we can discover the main cast list from long videos and further retrieve the characters of interest and their relevant shots for efficient browsing. This paper proposes a novel cast indexing approach based on hierarchical clustering, semi-supervised learning and linear discriminant analysis of the facial images appearing in the video sequence. The method first extracts local SIFT features from detected frontal faces of each shot, and then utilizes hierarchical clustering and Relevant Component Analysis (RCA) to discover main cast. Furthermore, according to the user's feedback, we project all the face images to a set of the most discriminant axes learned by Linear Discriminant Analysis (LDA) to facilitate the retrieval of relevant shots of specified person. Extensive experimental results on movie and TV series demonstrate that the proposed approach can efficiently discover the main characters in such videos and retrieve their associated shots.

1 Introduction

The ongoing expansion of multimedia information in the world wide web and the entertainment industry has generated increasing requirements for semantic based video mining techniques, such as news/sports summarization, film/TV abstraction and home video retrieval. Among various contents in these video data, characters are always the major focus of interest to the audience. In this paper, we utilize one of the most important visual cues, human face, to discover active characters who frequently appear in the feature-length films and retrieve their associated shots for efficient browsing.

Over the past few decades, there has been a good deal of investigation into automatic face detection and recognition techniques in the field of computer vision and pattern recognition [9]. However, due to the large variation of pose, expression and illumination conditions, robust face recognition is still a challenging

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 625–635, 2007.
© Springer-Verlag Berlin Heidelberg 2007

goal to achieve, especially for the scenario of still images. Recently, a significant trend in performing video-based face analysis has emerged, which aims to overcome the above limitations by utilizing visual dynamics or temporal consistence to enhance the recognition performance. In [6] Arandjelovic and Zisserman apply affine warping to mitigate the effect of various poses. However, it is unable to deal with the out-of-plan face rotation problem. The person spotting system [4] associates multiple exemplars of each person in the shot as a compact face-track to cover a person's range and expression changes. The approach constructs multiple patterns to improve the performance, but may fail in some shots with insufficient exemplars, which is often the case in movies and TV series. The multi-view 3D face model is described in [3] to enhance the video-based face recognition performance. However, it is very difficult to accurately recover the head pose parameters by the state-of-art registration techniques, and therefore not practical for real-world applications.

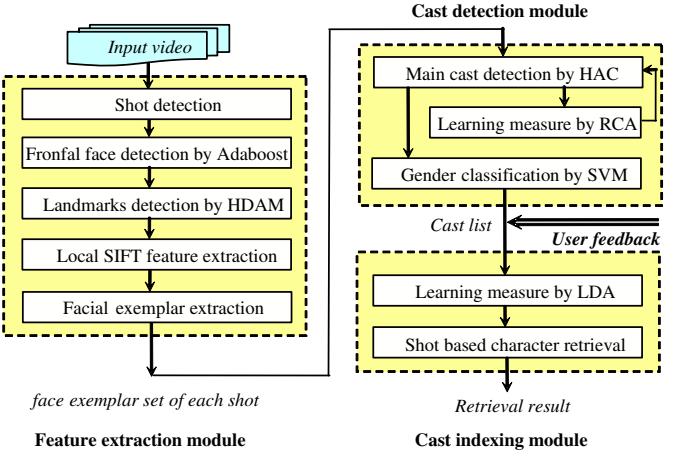


Fig. 1. Framework of the cast indexing system

As mentioned above, feature-length films contain multiple instances of each person's face that can be associated by visual tracking, speech identification and user feedback. Thus it is possible to improve the cast indexing performance by utilizing the complementary facial information under different pose, illumination and expression conditions. Motivated by this idea, we propose a novel semi-supervised cast indexing approach for feature-length films by hierarchical clustering, relevant component analysis and linear discriminant analysis. The framework consists of three modules as shown in Figure 1. In the feature-extraction module, near frontal faces are sequentially detected from sampling frames of the whole video, and then multiple facial exemplars in each shot are extracted by clustering and connected by tracking. We calculate the SIFT features in 5 local facial regions to jointly describe the face image. In the cast detection module, main characters are detected by partial Hierarchical Agglomerative Clustering

(HAC) [8] and a semi-supervised learning algorithm – Relevant Component Analysis (RCA) [10] iteratively. These face clusters are sorted by detected gender and appearing frequency (corresponding to the cluster size). Since faces of the same person with significant pose or expression variations may be unavoidably classified into a few separate clusters, it is necessary to utilize user feedback to further merge these duplicate clusters. Finally, the cast indexing module applies RCA and Linear Discriminant Analysis (LDA) [12] to learn a discriminative distance measure from the HAC output and the refined cast list, and then, in this discriminative feature space, retrieves associated shots for the characters of interest for the users.

The rest of this paper is organized as follows. In section 2, we describe the proposed method in detail, including feature extraction, main cast detection, and main cast retrieval. To evaluate the performance of this approach, extensive experiments are reported in section 3, followed by some concluding remarks in section 4.

2 Method Details

2.1 Shot Detection

Similar to document mining by parsing the textual content in the form of words, sentences, paragraphs and the whole document, video mining can be analyzed in four hierarchical levels – frame, shot, scene and the whole sequence. To well characterize the video content, shot detection is a prerequisite step and the basic processing unit of most video mining systems.

A shot is a set of video frames captured by a single camera in one consecutive recording action. According to whether the transition between shots is abrupt or not, the shot boundaries are categorized to two types, namely, Cut Transition (CT) and Gradual Transition (GT). In our work, we use a shot detection algorithm from Tsinghua University which achieved the best result in TRECVID 2004 and 2005 [5]. Its CT detector uses the 2nd order derivatives of color histogram, a flash light detector and a GT filter. Its GT detector uses motion vectors and the feature outputs from the CT detector.

2.2 Facial Feature Extraction

After shot detection, we use Viola and Jones' *'AdaBoost + Cascade'* face detector [7] to extract near frontal faces from temporal sampling frames in each shot. By automatic localization of four facial landmarks (centers of two eyes, nose and mouth) [2], each face is geometrically aligned into the standard normalized form to remove the variation in transition, scale, in-plane rotation and slight out-of-plane rotation. Then facial features are extracted from the normalized gray face images.

It is demonstrated that local features outperform global ones in most recognition and verification tasks, since they are more robust to partial occlusions, pose and illumination variations [1]. In our approach, we first apply Hierarchical

Direct Appearance Model (HDAM) [2] to detect facial landmark points and then extract the SIFT features [1] in five local rectangular regions, covering two eyes, central region of two eyes, nose, and forehead, as shown in Figure 2.



(a) (b) (c)

Fig. 2. Local SIFT feature extraction. (a) The original face image; (b) Detected 30 facial landmarks using HDAM; (c)Five local face regions for SIFT feature extraction.

As the basic processing unit in videos, a shot may contain NULL, one or more actors' faces. The faces of the same person in one shot can be easily detected by tracking the continuous positions of facial landmark points. To effectively characterize the variation of different poses, expressions and illumination conditions, we perform the *basic leader-follower clustering* algorithm [8] to generate multiple face exemplars for the same person in each shot. Thus a person appearing in one shot is represented by a representative face-exemplar set. The face-set distance measure between two shots S_i and S_j is defined by the shortest element-pair distance between the two sets as Eq(1):

$$d(S_i, S_j) = \min_{m,n} |x_{i,n} - x_{j,n}|/dim \tag{1}$$

where $x_{i,m} \in S_i, x_{j,n} \in S_j$ are the concatenated local SIFT feature vectors, $|.|$ is the L_1 distance and $dim = 5 \times 128$ is the dimension of the feature vector. The *basic leader-follower clustering* algorithm is described as following:

Algorithm (Basic leader-follower clustering)
1 begin initialization $\theta = threshold$
2 $C_1 = \{x\}, N = 1$
3 **do** accept new x
4 $j = \arg\min_i \|x - C_i\|$ $(i = 1, \ldots, N)$ //find the nearest cluster C_j
5 **if** $distance(x, C_j) < \theta$ //belong the same person
6 $C_j = C_j + \{x\}$
7 **else** create new cluster $C_{N+1} = \{x\}, N = N + 1$
8 **until** no more samples x
9 **return** $C_1, C_2, \ldots, C_N$
10 **end**

2.3 Main Cast Detection Using HAC

In most feature-length films, the main characters are the persons who frequently appeare in different shots, resulting in large numbers of similar face images, e.g.

frontal faces. Based on this observation, the main characters can be discovered by clustering all the selected shots using the distance measure proposed in Eq(1).

It is well known that facial features, represented as high-dimensional pixel arrays, often belong to a nonlinear manifold of intrinsically low dimensionality [11]. The variations between the facial features of the same person under different pose, illumination and expression are almost always larger than the variations due to changes in face identity. Therefore, in the clustering process, we do not partition all the shots by "flat" algorithms (e.g. K-means or spectral clustering) which will unavoidably group different persons into the same cluster. Instead, we perform Hierarchical Agglomerative Clustering (HAC) [8] to merge similar face shots whose distances are below a strict threshold, i.e. the clustering process will terminate once the merging face-set distance exceeds a pre-selected threshold. The threshold is set low enough to make sure that the two merged clusters are from the same person. As illustrated in Figure 3, the dendrogram shows how the shots are grouped by HAC, which well reflects the similarity relationship among different characters.

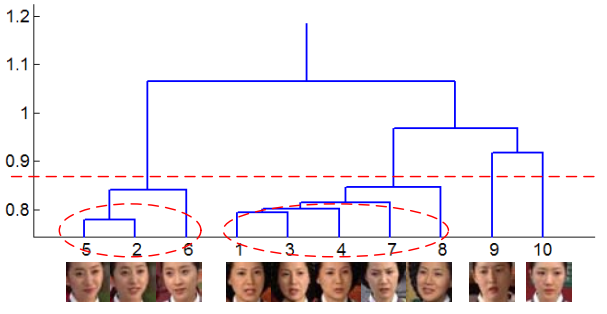


Fig. 3. Main cast detection by HAC on 10 shots. The HAC stops clustering when the face-set distance between shot 9 and shot 10 exceeds the threshold $\theta = 0.87$.

Algorithm (Agglomerative hierarchical clustering)

begin initialization $S_1, S_2, \ldots, S_n, \theta = threshold, N_{end}, F$
  $N = n, C_i = \{S_i\}, i = 1, 2, \ldots, N$
  do $N = N - 1$
    Find nearest clusters, say C_i and C_j
    if $\|C_i - C_j\| < \theta$ //make sure to be the same person by θ
      merge C_i and C_j
    else break
  until $N = N_{end}$
  return *sorted cluster with cluster size* $> F$ *(shots)*
end

After HAC procedure, the output clusters are sorted according to their sizes. Only clusters which contain more than F shots (i.e. the frontal face appears

at least in F shots) are selected as the main characters. Furthermore, genders of the main cast are detected by an SVM classifier using the local SIFT facial features (Figure 5). In our work, RBF kernel based SVM classifier is trained on a dataset of 2000 labeled samples and performs well for most of the videos with an averaged precision of 90%. According to the user's preference, the cast list can be also organized by their ages, poses or expressions for convenient browsing.

The main cast detection process is fully automatic. Although exemplars of each cluster belong to the same person, it is unavoidable that a person may appear in a few clusters due to the large variation of poses and expressions etc. The accuracy can be further refined by semi-supervised learning in section 2.4 and user's feedback.

2.4 Refine Main Cast Detection Using RCA

For many clustering and classification algorithms, such as K-means, SVM, and K nearest neighbor (KNN) etc., learning a good distance metric from training examples is the key to their success. Since exemplars of each cluster belong to the same person, each cluster is a *chunklet* [10]. We define "chunklet" as a subset of data points that are known to belong to the same although unknown class. From this kind of side-information in the form of *equivalence relations*, we learn a better distance metric in a semi-supervised manner and further perform the main cast detection using HAC.

In our approach, we employ Relevant Component Analysis (RCA) [10] to improve the feature space of HAC. The RCA algorithm has been theoretically shown to be an optimal semi-supervised learning procedure from the information theoretic perspective. By learning a Mahalanobis metric from chunklets, RCA transforms the original feature x into a new representation y, which assigns large weights to "relevant dimensions" and low weights to "irrelevant dimensions". Thus in the new feature space, the inherent structure of the data can be more easily unraveled for clustering.The RCA algorithm is described as following:

Algorithm (Relevant Component Analysis)

1 **Begin initialization** k chunklets $\{x_{ji}\}_{i=1}^{n_j}$ with means $m_j, j = 1, \ldots, k$

2 Compute the scatter matrix

- $C = \frac{1}{p} \sum_{j=1}^{k} \sum_{i=1}^{n_j} (x_{ji} - m_j)(x_{ji} - m_j)^T$

3 Compute the whitening transformation matrix by SVD

- $W = C^{-1/2}$

4 Transform the original feature x to the new feature $y = W \cdot x$

5 **end**

In the case of singular matrix C of high dimensional features, SVD is applied to calculate the transformation matrix W. Figure 4 (a) simulates the manifolds of facial features of two persons, where two chunklets are marked as red circles and blue circles respectively. Figure 4 (b) is the transformed features using RCA. It can be seen that transformed manifold becomes more separate. A constrained k-means clustering over the original feature space gives poor result with an

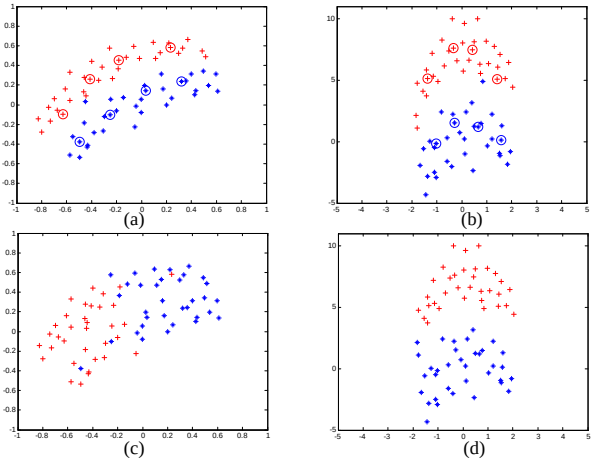


Fig. 4. (a) A 2-class clustering problem where each cluster has 4 labeled feedback samples as chunklets; (b) Data set after RCA transformation; (c) Constrained k-means clustering over the original space; (d) Constrained k-means clustering over the transformed RCA feature space

accuracy of 60% (Figure 4 (c)). However, through the RCA transformation, the constrained K-means achieves significant improved performance with an accuracy of 96% (Figure 4 (d)).

2.5 Main Cast Retrieval Using LDA

By main cast detection of section 2.3 and 2.4, we discovered main characters and most of their multi-view facial exemplars in the video. Since faces of the same person may be classified into a few different clusters, it is necessary to utilize the user's feedback to refine the finial cast list by indicating which clusters belong to the same person. To retrieval relevant shots of these main characters for efficient browsing, we apply a nearest neighbor matching in the Linear Discriminant Analysis (LDA) [12] subspace of the above feature space.

LDA is a well-known technique for dealing with the class separability problem and determining the set of the most discriminant projection axes. The most widely used LDA approach seeks an optimal projection from the input space onto a lower-dimensional discriminating feature space as Eq(2).

$$W_{opt} = \arg\max_{W} \frac{|W^T S_b W|}{|W^T S_w W|} \tag{2}$$

with the within class scatter matrix $S_w = \sum_{i=1}^{L} \sum_{x_k \in X_i} (x_k - m_i)(x_k - m_i)^T$ and the between class scatter matrix $S_b = \sum_{i=1}^{L} n_i (m_i - m)(m_i - m)^T$. Here m_i is the mean face of class X_i, m is the mean face of all classes, and n_i is

the number of samples in class X_i. The optimal projection matrix W_{opt} can be constructed by the eigenvectors of $S_w^{-1}S_b$. To avoid degeneration of S_w, we first reduce the feature dimensionality by PCA, and then perform discriminant analysis in the reduced PCA subspace. By applying this method, we find the projection directions that maximize the Euclidean distance between the face images of different classes and minimize the distance between the face images of the same class. An example of main cast retrieval is illustrated in Figure 6.

3 Experiment

To demonstrate the performance of the proposed cast indexing approach, extensive experiments were conducted on a story TV series of "Da ChangJin" and an action movie of "007 Die Another Day", totaling up to 3 hours of videos. "DaChangjin" is a hot Korea TV series with 594 shots and 67006 frames (45min). The main characters are Chang Jin, Jin Ying, Cui ShangGong, Shang Shan, Min ZhengHao etc. "007 die another day" is a famous action movie with 1652 shots and 237600 frames (132min). The main cast includes James Bond, Jinx Johnson, Gustav Graves, Miranda Frost, Zao etc.

In the experiments, we temporally sample each shot by 5 frames per second to reduce the duplicated images and computational burden. The detected main cast of "Da ChangJin" and "007 Die Another Day" are shown in Figure 5, which are organized according to their gender for convenient browsing. It can be observed that there are some duplicate faces which correspond to large pose, illumination and expression variations of the same character. The gender is detected by RBF

Table 1. Performance of Gender classification by SVM

Gender	Precision (%)	Recall (%)	F-score (%)
female	97	90	93
male	95	98	97



Fig. 5. Automatically detected main cast of "Da Changjin" and "007 Die Another Day" by HAC and RCA

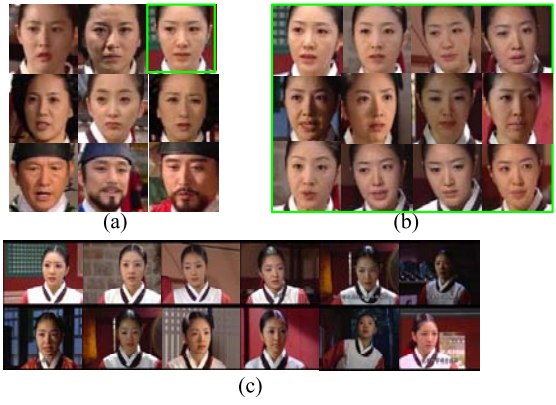


Fig. 6. An example of main cast retrieval. (a) The main cast list. (b) The face-exemplar set of one actress "Jin Ying". (c) Key frames of the retrieved shots for the query person "Jin Ying".

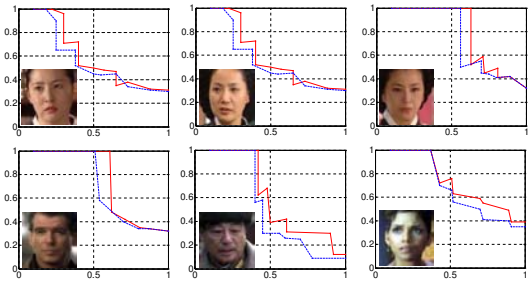


Fig. 7. The precision vs. recall curves of main cast retrieval of "Da ChangJin" and "007 Die Another Day". The red solid curves are the RCA based retrieval result. The blue dashed curves are the retrieval results without RCA.

kernel based SVM classifier on local SIFT features. Table 1 illustrates the gender classification performance with F-score above 93%. The F-score $= 2 \times Pr \times Re/(Pr + Re)$ evaluates the comprehensive performance.

According to user's feedback, we manually merge clusters of the same person to refine the finial cast list and get the multiple exemplars of each character. By these exemplars, LDA learns the discriminative transform W to retrieve relevant shots of the query person. Figure 6 illustrates one retrieval procedure of a main actress "JinYing" in the TV series "Da ChangJin". The curves of six main actors in "Da ChangJin" and "007 Die Another Day" videos are shown in Figure 7 and Table 2. It can be observed that LDA significantly improves the shot retrieval performance and achieves good cast retrieval result.

Table 2. Performance of the main cast retrieval of "DaChangJin" using RCA and LDA

Character	Precision (%)	Recall (%)	F-score (%)
CuiShangGong	85.7	93.1	89.25
HanShangGong	78.1	100	87.70
JinYing	85.2	100	92.01
ChangJin	95	57.1	72.69
LingLu	100	55.6	71.47
HuangShang	100	100	100
ShangShan	100	54.5	70.55

4 Conclusion

In this paper, we proposed a novel semi-supervised cast indexing approach using HAC, RCA and LDA. The method first detects near frontal faces from temporal sampling frames of each shot and then adopts partial hierarchical agglomerative clustering (HAC) and semi-supervised learning algorithm RCA to discover the main cast. To refine the accuracy of automatic main cast detection, user's feedback is employed by indicating which clusters belong to the same person. Then by these multiple exemplars of main characters, Linear Discriminant Analysis (LDA) algorithm learns a discriminative distance measure to retrieve relevant shots of the query person in the whole video. Extensive experimental results on movies and TV series demonstrate the effectiveness of the approach. In future work, we'll take advantage of multiple cues such as speech, music, clothing, close caption, and tracking etc. to improve the cast indexing performance and further retrieve the highlight scenes of main characters.

References

1. D. Lowe: Distinctive image features from scale-invariant keypoints. IJCV. **60** (2004) 315–333
2. G. Song, H. Ai, G. Xu: Hierarchical direct appearance model for elastic labeled graph localization. Proc of SPIE (2003) 139–144
3. J. Kittler, A. Hilton, M. Hamouz, J. Illingworth: 3D assisted face recognition: a survey of 3D imaging, modelling and recognition approaches. Proc. of IEEE CVPR (2005) 144–144
4. J. Sivic, M. Everingham, and A. Zisserman: Person spotting: video shot retrieval for face sets. Proc. of IEEE CIVR (2005) 226-236
5. J.H. Yuan, W.J. Zheng, L. Chen, etc.: Tsinghua University a TRECVID 2004: shot boundary detection and high-level feature extraction. NIST workshop of TRECVID. (2004)
6. Arandjelovic, G. Shakhnarovich, J. Fisher, R. Cipolla, and T. Darrell: Face recognition with image sets using manifold density divergence. Proc. of IEEE CVPR (2005) 581- 588
7. P. Viola, M. Jones: Rapid object detection using a boosted cascade of simple features. Proc. of IEEE CIVR (2001) 511–518

8. R. Duda, P. Hart, D. Stork: Pattern Classification. Wiley (2000)
9. W. Zhao, R. Chellappa, P. Phillips, A. Rosenfeld: Face recognition: a literature survey. ACM Comput. Surv. **35** (2003) 399-458
10. BarHillel, T. Hertz, M. Shental, D. Weinshall: Learning distance fucntions using equivalence relations. Proc. of ICML (2003)
11. S. Roweis and L. Saul: Nonlinear dimensionality reduction by locally linear embedding. Science **290** (2000)
12. P. Belhumeur, J. Hespanha, D. Kriegman: Eigenfaces vs. fisherfaces: recognition using class specific linear projection. IEEE Trans. on PAMI. **19** (1997) 711-720

Linking Identities and Viewpoints in Home Movies Based on Robust Feature Matching

Ba Tu Truong and Svetha Venkatesh

Department of Computing, Curtin University of Technology, Perth, Western Australia

Abstract. The identification of useful structures in home video is difficult because this class of video is distinguished from other video sources by its unrestricted, non edited content and the absence of regulated storyline. In addition, home videos contain a lot of motion and erratic camera movements, with shots of the same character being captured from various angles and viewpoints. In this paper, we present a solution to the challenging problem of clustering shots and faces in home videos, based on the use of SIFT features. SIFT features have been known to be robust for object recognition; however, in dealing with the complexities of home video setting, the matching process needs to be augmented and adapted. This paper describes various techniques that can improve the number of matches returned as well as the correctness of matches. For example, existing methods for verification of matches are inadequate for cases when a small number of matches are returned, a common situation in home videos. We address this by constructing a robust classifier that works on matching sets instead of individual matches, allowing the exploitation of the geometric constraints between matches. Finally, we propose techniques for robustly extracting target clusters from individual feature matches.

1 Introduction

The *aim* of this work is to extract the structure within a home video footage collection, and towards this goal three tasks are currently defined:

- *Shot matching.* We attempt to look for clusters of shots with overlapping fields of view, which often lie on the same side of a 180-degree axis. Identification of these shot clusters in the scene is important since each tends to depict one semantic component of the scene, both in terms of the structure and story.
- *Face matching.* Based on an initial set of faces returned by the face detector, we aim to extract a set of face clusters associated with different individuals. This is strongly desired in home videos, since they mainly focus on characters that appear in the video, for example, family and friends. In addition, the ability to link faces across different scenes is also relevant as the footage collection of a user often contains a small set of dominant characters, each appearing in separate footage captured at different times and locales.
- *Scene matching.* Apart from chronological organization of home videos, it is possible that, via scene matching, they can be organized on the basis of

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 636–648, 2007.
© Springer-Verlag Berlin Heidelberg 2007

where the event has actually taken placed and captured, enabling non-linear navigation of the footage collection.

However, in this paper, we will restrict our focus to the first two tasks since the scene matching can be seen as derivative from shot matching process.

Our approach to these three matching/clustering problems is based on the use of Scale Invariant Feature Transform (SIFT). First, syntactical analysis is performed on a home video to extract shots, keyframes and session boundaries. SIFT features are extracted for each frame and various matching techniques is applied on each frame pair to determine when they match. Matches at frame level can then propagate to shot and scene levels to form suitable clusters. With respect to face clustering, face detection is applied to each keyframe. Detected faces provide us with a set of subjects for clustering. Characters where faces are not captured is considered less important and it is acceptable if we fail to cluster their faces. Face matching is carried out by using only SIFT features associated with those faces.

To the best of our knowledge, this is the first work to investigate the use of SIFT feature to solve the difficult problem of clustering shots, faces and scenes in home videos. Although SIFT features have been known to be robust for object recognition, the standard SIFT matching method fails to deal adequately with the following complexities associated with the home video setting: objects lying in different planes, too few matches returned due to the large degrees of rotation in depth, and the high ratio of noisy matches to correct ones. Therefore, we propose novel techniques for adapting the use of SIFT features for our problem domain. First, we propose a distance-based verification procedure to produce the basic set of matches. An iterative version of RANSAC is used to extract all correct matches to robustly overcome the problem of objects being on different planes. For identifying correct matches when only a small number of matches are returned, we construct a robust classifier that works on matching sets instead of individual matches, allowing the exploitation of the geometric constraints between matches. To increase the variance in pose of the face set used for matching, we detect and track the presence of a face across keyframes in a shot. Finally, we propose methods for refining the cluster based on the knowledge about the cluster structure in the scene, e.g., two faces detected on the same frame cannot belong to the same cluster. We demonstrated the effectiveness of our techniques in various home video footage.

2 Previous Works

Shot clustering/grouping has been often used as an intermediate step in extracting scene boundaries [1,2,3,4]. Hence, these methods only demand that shots clustered together come from the same scene, instead of having overlapping views. They then use overlapping link reasoning to merge separated clusters into scenes. This work, in contrast, uses scene indices available through other methods as the temporal constraints in searching for shot clusters. Clustering of shots for the purpose of content browsing and presentation has been examined in [5]. Recently, [6] investigated the use of clustering to detect film scenes that

are coherent in time/space or mood and present them in a Scene-Cluster Temporal Chart that depicts the alternating themes in a film. The common problem with these works is that they tend not to explicitly specify what the extracted clusters represent, other than to describe them in terms of the results obtained (e.g., indoor, coffee shop scenes), and neither do they specify any consistent groundtruth.

The shot matching problem is more clearly stated in [7] and [8], which aim to detect 'identical' shots or shots that perceptually belong to the same production takes of a scene. However, they exclude the linking of shots where the camera setup has changed, although the focus of the shot remains the same. In addition, their use of color histogram as the feature for matching is not robust due to its sensitivity to changes in light condition and movement of local objects, while not being sufficiently distinctive for scenes that are under-lit. While it has not been thoroughly investigated, the matching of video shots using invariant features is not completely new and has been addressed in [9]. However, their method is different to ours in terms of the selection of features and the matching process. Moreover, they do not to report concrete results in terms of the precision and recall of matches.

3 SIFT Keypoints and Matching

Prior to this step some existing techniques are employed to extract shots, shot-based keyframes and session boundaries of a home video. We refer readers to the technical report [10] for details.

3.1 Extraction of Keypoints

Scale Invariant Feature Transform (SIFT) [11] is an approach for detecting and extracting local feature descriptors that are reasonably invariant to changes in image noise, illumination, rotation, scaling and viewpoint.

The SIFT algorithm produce a set of keypoints, each is associated with the following set of values:

- (x, y): The real-valued position of the keypoint.
- σ: The orientation of the keypoint. Note that multiple values may be associated with the same (x, y).
- s: The scale of the keypoint.
- d: The descriptor, a vector of 128 real values.

There are various methods for extracting invariant features. We select the SIFT approach due to its superiority in discrimination and stability as demonstrated in the the comparative evaluation reported in [12].

3.2 Filtering and Enlargement of Matching Set

In traditional object recognition, given a database of keypoints stored for object models, the set of keypoints found on the current image is matched to keypoints in the database on the basis of individual best matching, and the best match of

a particular keypoint in the image is the closest one in the database. Euclidean distance is used to measure the closeness between two keypoints. The matching is directional, i.e., image to the database. In our problem domain, the matching is non-directional; therefore we need to modify the matching process accordingly.

Basic Matching Based on Descriptor Distance. The nearest point matching will find a match for every keypoint, regardless of whether matching keypoints actually correspond to the same point in 3D scene. Here, we describe some methods for removing incorrect matches using only the distance between keypoint descriptors.

- *Matches not distinctive to the second closest match.* This is proposed in the SIFT paper [11] as a better method for broadly examining the correctness of a match based on the distance between descriptors. Instead of setting the threshold on the distance between the keypoint to its nearest match, the threshold (=0.75) is applied on the ratio of this distance against the distance to the second closest match. This technique is illustrated in Figure 1b, where (2) and (3) are two keypoints in the source image that is the closest and the second closest to the keypoint (b) in the target image, since their distance ratio (0.3/0.35) is large, the match $2 \mapsto b$ is discarded.
- *Multiple targets matches to the same source point.* In this case, multiple keypoints in the target image are matched to the same point in the source image. Only the maximum of one match is correct. Here, we greedily discard all matches. For example, in Figure 1c, all three keypoints in the target image is matched to the same keypoint (4) in the source image. Hence, all matches $4 \mapsto c, 4 \mapsto d, 4 \mapsto e$ are removed.
- *Matches with different returns.* This is illustrated in Figure 1d, (5) is the closest keypoint to (f), however, when the source and target images are swapped, a keypoint different to (5) is matched to (f). Like the above case, it is impossible that both matches are correct. Thus, we remove them both.

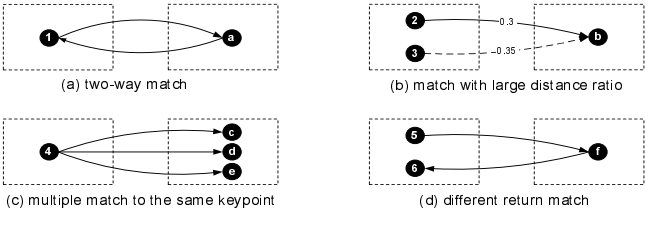


Fig. 1. Different cases of keypoint matches

After applying above techniques both ways to each image pair, we are left with matches of two types: perfect two-way matches (Figure 1a) and one-way matches. While the first ones are generally very reliable, the latter are less so. Therefore, a smaller threshold (=0.65) on the distance ratio is applied.

RANSAC for Verification and Enlargement of the Matching Set. Random Sample Consensus (RANSAC) [13] is a technique for fitting a model to

experimental data. The idea behind the RANSAC procedure is simple. A model is constructed from a set of samples randomly selected, and the rest of the data is checked for agreement with the proposed model to form the consensus set. The model is claimed if a good consensus set is found. In our work, RANSAC is used not only for deciding if two images match, but also for verifying the correctness of individual matches, crucial for the correct construction of keytracks. However, we observe that in the natural setting of home videos, objects lie in different image planes and each plane tends to produce different good consensus sets. Therefore, we propose the iterative execution of RANSAC, which removes a good consensus set each time till no more good consensus set can be found. More details of this algorithm can be found in [10].

Match Classifier. While RANSAC can robustly address the case where a lot of matches are returned, it is more difficult to deal with the situation when a few matches are returned for two overlapping shots due to significant rotation in depth. For these cases, we need to exploit all information available to determine when individual matching of keypoints is correct. However, features associated with individual matching are limited to the Euclidian distance and attributes of two associated keypoints, which is not sufficient for reliable verification of matches. Since a correct match needs to be geometrically consistent with other correct matches, we can classify a match by considering its relationship with other matches. Here, we pose the problem of individual match verification as the problem of verifying a set of matches together where interacting features can be exploited as follows: Given a predefined value k, we would like to construct a classifier to differentiate between correct and incorrect sets of k matches. A correct set is the set with all correct individual matchings. Let $\mathbb{M} = (\mathsf{m}_1, \mathsf{n}_1), ..., (\mathsf{m}_k, \mathsf{n}_k)$ be a set of k matchings of keypoints from frames f_i and f_j. First, we compute the similarity transform ϕ from $\mathbb{M}$. Via ϕ, the following set of features is extracted to construct the classifier.

- e. This is the error associated with the similarity transform and matching set $\mathbb{M}$. A perfect matching has Zero error, while a large value indicates an incorrect matching. This feature is irrelevant (=0) when $k = 2$.
- $\Delta(\sigma)$. This is the average difference in orientation of individual keymatches to the orientation in the similarity transform:

$$\Delta(\sigma) = |\frac{\sum_{i=1}^{k} (\sigma(\mathsf{n}_i) - \sigma(\mathsf{m}_i) - \sigma)}{k}|. \tag{1}$$

  $\Delta(\sigma) = 0$ for a perfect match.
- $\Delta(s)$. This is similar to the average difference in orientation but we take the log of the scale instead.

$$\Delta(s) = |\frac{\sum_{i=1}^{k} (\log(s(\mathsf{n}_i)) - \log(s(\mathsf{m}_i)) - \log(s))}{k}|. \tag{2}$$

  Just like orientation, $\Delta(s) = 0$ for a perfect match.
- σ. One expects little rotation along the z-axis in typical home video footage, so its value tends to be around 0 for correct matches.

– *dist*. This represents the average Euclidean distance between keypoints of all matching in $\mathbb{M}$. A low value indicates a correct match.

Training the classifier. Data used for generating the incorrect matching set is from keyframe pairs associated with shots that are not overlapping in the field of view. This set is relatively large. The construction of the correct matching sets requires manual annotation of individual keymatches. We have constructed a visual tool that works on keyframes associated with shot pairs belonging to the same cluster in the groundtruth, which allows incorrect key matches to be visually identified and eliminated. All combinations of size k from remaining matches is added to the data set. All data for training the classifier comes from only one single scene in our footage collection. The classification model used in our work is currently the decision tree, chosen for its simplicity and speed. We build two decision trees, with $k = 2$ and $k = 3$ respectively. Figure 2 shows some examples of matches (white lines) that are correctly picked up by the proposed match classifier. These matches can not be verified as correct via RANSAC.

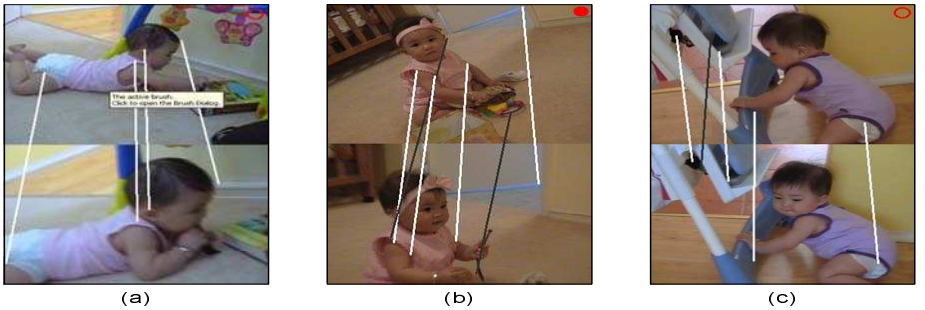


Fig. 2. Examples of matches correctly identified by the classifier

4 Clustering and Refinements

In Section 3, we only address the problem of keypoint matching and propose techniques that can classify the correctness of a key match. Here we discuss how individual key matches can be used to cluster video units of interest: shots, faces, and scenes.

From Keymatch to Image Match. First we need to roughly decide if two keyframes (or a face detected on these keyframes) match in terms of field of view (or the identity). Theoretically, we need only one correct keymatch to conclude if two keyframes match. Practically, we need to allow rooms for errors and inrelevant matches. Here we define the strength of match between two set of keypoints M and N as the number of correct matches between them:

$$\mathcal{M}(\mathsf{M}, \mathsf{N}) = |\{(\mathsf{m}_i, \mathsf{n}_j) | \mathcal{M}(\mathsf{m}_i, \mathsf{n}_j) = 1, \mathsf{m}_i \in \mathsf{M}, \mathsf{n}_j \in \mathsf{N})\}|. \quad (3)$$

We only claim two keyframes (or faces) match if their matching strength passes a threshold (=6) and (=3) respectively.

This can then be aggregated to shot-to-shot matching by having two shots matched if they contain at least one matching pair of keyframes, one from each shot. The strength of a shot-to-shot match is determined as the average of the strength of all keyframe level matches:

$$\mathcal{M}(s_1, s_2) = \mathsf{E}\{\mathcal{M}(f_i, f_j) | \mathcal{M}(f_i, f_j) > 0, f_i \in s_1, f_j \in s_2\},$$

where $\mathsf{E}\{.\}$ denotes the average of all values in the set. We only consider those keyframe pairs that actually match.

Consistency with Object Transform. When matching two objects, faces in our case, the overall similarity transform for all keymatches should roughly be the same as the similarity of the object bounding box itself. Otherwise, these keymatches, albeit correct, are not associated with the object features, but the background. This is very similar to the concept of the match classifier described in Section 3.2. Here, we can treat the centre of the face bounding box as a keypoint of the face, with the rotation and scale defined by the bounding box angle and size respectively. Using the similarity transform computed from keymatches, we can project the centre of the source face region to the target face. Ideally, the projected point should be identical with the centre of the target face both in location, scale and orientation.

Connected Components. For each scene, the clustering is then extracted based on these individual matching pairs. We consider each shot (or face) as a node in a graph with matching shot pairs forming its edges, and shot clusters can be easily identified by searching for connected components of the graph.

Cluster Splitting Based on Weak Links. Clusters formed above are relatively crude; one false positive match at keyframe level may lead to the merging of two separate shot clusters. These false positives often lead to the situation as depicted in Figure 3a. The detection of connected components means all shots (1),(2),(a) and (b) are considered to be from the same cluster. However, this cluster is actually formed by two separate clusters connected via weak links (1,b) and (2,a), and they should be split. The most important issue here is deciding when the linking is sufficiently weak to warrant a split. Currently, we heuristically search for a cut of the graph with a maximum of k (=2) edges between them and the strength of each edge between the two partitions are less than a threshold, (=7) for shot matching and (=3) for face matching.

Cluster Splitting Based on Distinctive Items. This is currently applicable to face matching only. For example, when multiple faces are detected from one single keyframe, these faces must be associated with different persons. However, they can be matched to the same person in a different shot and face clusters of these two persons are merged as they belong to the same connected component of the graph. This is also useful when the user wants to manually refine clusters; he only needs to pick two items that should belong to two different clusters, and the algorithm will automatically assign other items.

Given that two faces are known to be associated with two different persons, the splitting of clusters is done in a greedy manner as illustrated in Figure 3b.

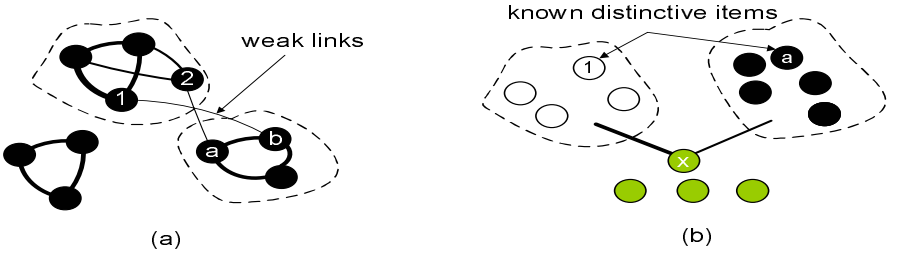


Fig. 3. Cluster splitting based on weak links and known distinctive items

For known distinctive faces (1) and (a), two clusters are created and one face is assigned to each cluster. For each face in the original cluster, the face with the largest total matching strength to these faces are considered next (the most likely match to either of the existing clusters) and it is added to the cluster with larger average strength. We exclude strength 0 from calculating the average, and hence are biased towards individual strong matches. In Figure 3b, the face (x) is the face currently having largest total strength to two clusters. It is added to the cluster containing (1) which has stronger average strength.

5 Implementation and Results

We have tested our matching techniques on a set of 10 typical home videos, consisting of two main themes: Baby & Family and The Office. In this section, we opt to demonstate typical results using sample video sequences. For more detailed results in terms of the clustering precision and recall, we refer readers to the technical report [10]. The overall results are very good.

5.1 Shot Matching

Given various matching techniques described in the precious section, the extraction of shot clusters is relatively straight forward.

- For successive keyframes of a shot, RANSAC is applied to identify correct matches and construct keytracks for the shot.
- Via keytracks, individual keyframes are matched and verified by the basic matching process based on the descriptor distance described in Section .
- For a pair of shot, if number of matches from any two of their keyframes is large ($\geq$ 10), they are considered to be correct matches and RANSAC is applied to extract the number of correct matches. Otherwise, the match classifier is applied.
- Shot clusters are formed by detecting connected components, which can then be split via the detection of weak links as described in Section 4.

This procedure is applied for every scene in the footage collection. Figure 4 show shot clusters detected for home video footage of an outdoor activity and an indoor baby crawling scene. Each shot is presented by its first keyframe. The figure shows that our method is very successful. In the first footage (a),

it correctly divides water body shots into two groups and so too for the table shots, as two camera position are used for each of these setups. The two single-item clusters are also correct for this footage. For the second footage (b), two main clusters of shots, the cot and baby on the ground, are both extracted. The close-up of the baby forms its own cluster, which can also be treated as correct.



Fig. 4. Example of Shot Clusters

5.2 Face Matching

Intra-Shot Matching. We first would like to examine the ability of our method to perform intra-shot matching of faces detected in individual keyframes. The extraction of face clusters within a shot should be more accurate than inter-shot extraction. These clusters are useful in two ways. Each cluster can be used to search for more faces in keyframes not overlapping with it, which produces a large set of faces of different poses associated with the same individual. Subsequently, enlarged face clusters within each shot can be used as a single unit for matching with other frames or face clusters in other shots. Given SIFT feature have already been extracted for each keyframe of the shot and a set of faces have been extracted by the face detector, the procedure for intra-shot clustering of faces is as follows.

1. Define a region around each detected face, and extract a subset of SIFT keypoints that lie in the region.
2. Perform basic matching and verification.
3. Apply RANSAC and extend the match set if possible.
4. Apply Match Classifier if required.

5. If two faces are matched, check the consistency between the face transform parameter and the similarity transform parameter computed from the matching set.
6. Compute the connected components to extract the initial set of face clusters.
7. For each cluster that contains two faces that are detected from the same keyframe, the splitting procedures described in Section 4 is applied.

Figure 5 show the detected face clusters in various shots[1], in which faces from the same cluster are grouped in the figure without any space separation. The face detector gives a total of four false alarms. The first row shows that for two shots (a & b) face matching based on SIFT feature can overcome the occlusion and distortion of faces due to subject movements, which can be very problematic for standard face recognizers. In the second row, two persons appear in the same shot. However, the two associated face clusters were actually detected as one after Step 6 due a couple of features being matched across two persons. Since the two faces 4 and 5 (marked with the dotted line) are detected in the same keyframe, Stage 6 correctly splits the cluster. Similarly in row 3, matching of calendar features initially places the calendar in the same face group as the subject. However, as two 'faces' come from the same frame, they are used to correct the cluster. For row 4, some correct feature matches are found between face 3 and 5. However, these matches are not associated with the face, and the consistency check in Step 4 correctly discards the match. If Step 4 is omitted, only 1 single cluster is produced.

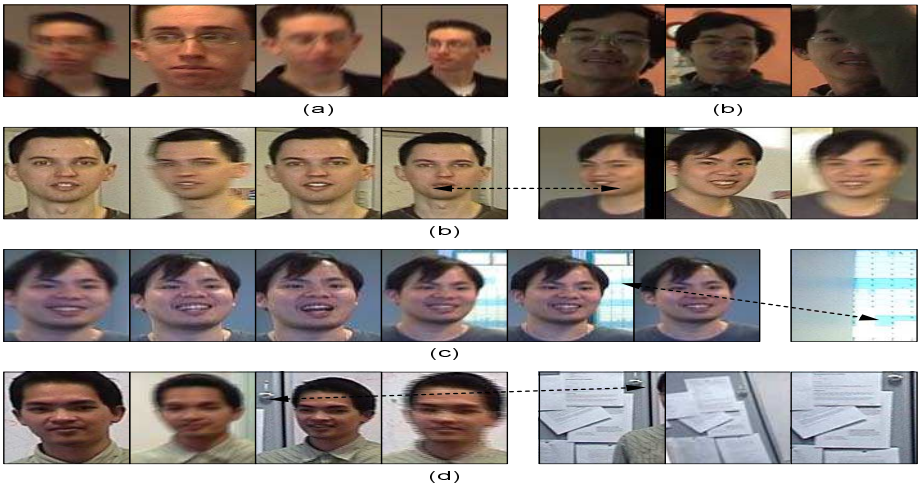


Fig. 5. Example of Intra-Shot Face Clusters

Extending the Face Set. Faces detected by the standard face detector are mainly limited to frontal views. Since we would like to find and match as many faces and people as possible in the video sequences, we need to generate more face

[1] We remove some faces from each cluster for display.

exemplars in various views. This is achieved by matching faces in each cluster formed by the intra-shot matching to the remaining keyframes of the shot with respect to each cluster.

For each detected face, we define two regions associated with it, F^0 and F^1, with $\mathrm{F}^0 \subset \mathrm{F}^1$. Using these face regions, we extract a set of keypoints to represent the face extracted from a keyframe f as:

$$\mathsf{K}(\mathrm{F}^{(.)}) = \{k_i = (x_i, y_i, \sigma_i, s_i, d_i) | k_i \in \mathsf{K}(f), (x_i, y_i) \in \mathrm{F}^{(.)}\},$$

where $\mathsf{K}(X)$ denotes the set of keypoints associated with entity X. We use $\mathsf{K}(\mathrm{F}^0)$ to match face regions across video shots, while $\mathsf{K}(\mathrm{F}^0)$ is used to generate more face exemplars as explained next. Note that although $\mathsf{K}(\mathrm{F}^1)$ may contain some features that belong to the background, the possibility of having faces of two different people in close approximity to the same feature point of the background is very low. This allows us to define $\mathsf{K}(\mathrm{F}^1)$ as relatively large, without generating false matches in background regions.

For a cluster C_i detected via the intra-shot cluster method described above, if we have a keytrack intersect with the current frame and some of regions F^0 of faces in C_i and there is no face of cluster C_i detected in the current frame, then a new face exemplar of cluster C_i is claimed. The region F^1 of the new exemplar is defined by projecting the matching face in C_i according to the similarity transform produced by the matches (i.e., on the same track).

Inter-Shot Matching. After extending the number of face exemplars, for each scene, we put all faces through the same procedure as intra-shot face clustering described above, which produces a set of face clusters. Finally, some of clusters are merged if they contain faces that come from the same intra-shot face clusters. This is possible due to the high precision obtained with intra-shot clustering.

Figure 6 shows 11 face clusters detected for a scene. Seven of them contain one single face. Row 1 and 2 shows our algorithm can cluster faces of different poses

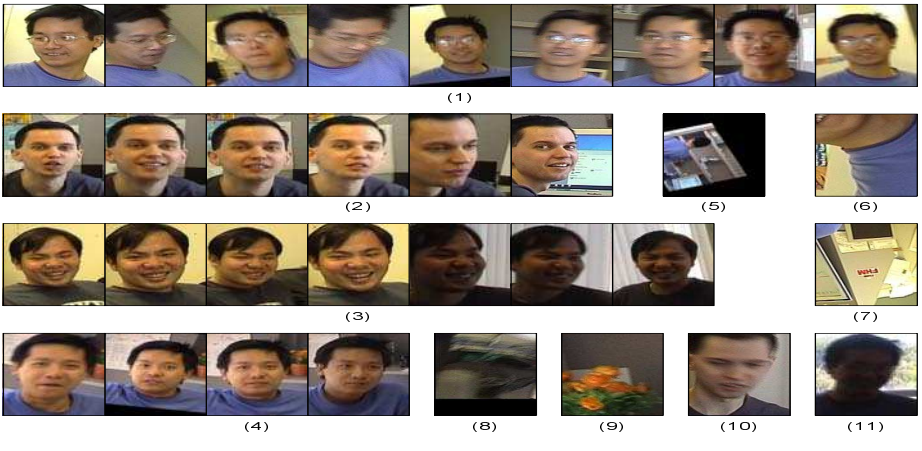


Fig. 6. Example of inter-shot face clusters

correctly. Ideally, main cluster in row 4 should be merged with the cluster in row 1. Yet, a close-examination shows that the difference between these two clusters is the subject wearing and not wearing glasses, emphasizing the importance of SIFT features around the eyes for face matching. Row 3 shows our clustering is insensitive to medium lighting changes. Single-face clusters mainly contain non-face objects. There are two cases in row 4, where they should have been merged to the main cluster. However, these faces involve severe lighting changes and motion distortion.

6 Conclusions

We have presented a SIFT based solution to the challenging problem of clustering shots, faces and scenes in home videos. We adapt the SIFT based matching process to deal the complexity of home video: objects lying in different planes, large rotations in depth and large viewpoint differences. We demonstrate the results of our algorithm on a set of 10 typical home videos. Future work can explore the use of alternatives such as PCA-SIFT to increase robustness. Another important issue to address is the computational complexity and methods to address this issue could explore the use of different levels of granularity levels for different matching tasks.

References

1. Rui, Y., Huang, T.S., S., M.: Constructing table-of-content for videos. ACM Multimedia System Journal: Special Issue in Multimedia Systems on Video Libraries **7** (1999) 359–368
2. Veneau, E., Ronfard, R., Bouthemy, P.: From video shot clustering to sequence segmentation. In: ICPR'00. Volume 4., Barcelona (2000) 254–257
3. Yeung, M., Yeo, B.L., Liu, B.: Segmentation of video by clustering and graph analysis. Computer Vision and Image Understanding **7** (1998) 94–109
4. Zhao, L., Qi, W., Yang, S., Zhang, H.: Video shot grouping using best-first model merging. In: Proc. 13th SPIE Symposium on Electronic Imaging - Storage and Retrieval for Image and Video Databases, San Jose (2001) 262–267
5. Gatica-Perez, D., Loui, A., Sun, M.T.: Finding structure in home videos by probabilistic hierarchical clustering. IEEE Transactions on Circuits and Systems for Video Technology **13** (2003) 539–548 IDIAP-RR 02-22.
6. Truong, B.T., Venkatesh, S., Dorai, C.: Application of computational media aesthetics methodology to extracting color semantics in film. In: ACM Multimedia (ACMMM'02), France Les Pins (2002) 339–342
7. Satoh, S.: News video analysis based on identical shot detection. In: Multimedia and Expo, 2002. ICME '02. Proceedings. 2002 IEEE International Conference on. Volume 1. (2002) 69–72
8. Truong, B.T., Venkatesh, S., Dorai, C.: Identifying film takes for cinematic analysis. Multimedia Tools and Applications **26** (2005) 277–298
9. Schaffalitzky, F., Zisserman, A.: Automated location matching in movies. Computer Vision and Image Understanding **92** (2003) 236–264

10. Truong, B.T., Venkatesh, S.: Sift feature for home video analysis. Technical report, IMPCA - Curtin University of Technology (2006)
11. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision **60** (2004) 91 – 110
12. Mikolajczyk, K., Schmid, C.: A performance evaluation of local descriptors. IEEE Transactions on Pattern Analysis and Machine Intelligence **27** (2005) 1615–1630
13. Fischler, M.A., Bolles, R.C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Commun. ACM **24** (1981) 381–395

An Efficient Automatic Video Shot Size Annotation Scheme*

Meng Wang[1], Xian-Sheng Hua[2], Yan Song[1],
Wei Lai[2], Li-Rong Dai[1], and Ren-Hua Wang[1]

[1] Department of EEIS, University of Sci&Tech of China
Huang Shan Road No.4 Hefei Anhui 230027, China
wangmeng@mail.ustc.edu.cn,
{songy, lrdai, rhwang}@ustc.edu.cn
[2] Microsoft Research Asia, 5F Sigma Center
49 Zhichun Road, Beijing 100080, China
{xshua, Lai.Wei}@microsoft.com

Abstract. This paper presents an efficient learning scheme for automatic annotation of video shot size. Instead of existing methods that applied in sports videos using domain knowledge, we are aiming at a general approach to deal with more video genres, by using a more general low- and mid- level feature set. Support Vector Machine (SVM) is adopted in the classification task, and an efficient co-training scheme is used to explore the information embedded in unlabeled data based on two complementary feature sets. Moreover, the subjectivity-consistent costs for different mis-classifications are introduced to make the final decisions by a cost minimization criterion. Experimental results indicate the effectiveness and efficiency of the proposed scheme for shot size annotation.

1 Introduction

With the rapid proliferation of digital videos and development in storage and networking technologies, content-based video organization and retrieval have emerged as an important area in multimedia community. This leads to an increasing attention on the detection and management of semantic concepts for video. However, existing works mainly focus on the concepts defined by the scenes and objects in the video frames, such as high level feature extraction task in TRECVID [1], while ignoring the fact that camera shot sizes also convey important information, especially in film grammar [5].

Generally, camera shot size is decided by the distance from camera to objects (here we don't take camera parameters into account). We argue that the shot size information is useful in at least the following three aspects:

(1) It is known that shot size variation of consecutive shots has some patterns in professional film editing, which can be regarded as one of the "editing grammars" in videos [5, 6, 7, 9]. If shot size can be automatically annotated, more

* This work was performed when the first author was visiting Microsoft Research Asia.

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 649 – 658, 2007.
© Springer-Verlag Berlin Heidelberg 2007

compelling editing results may be obtained by automatic video editing methods with shot size information.

(2) Shot size patterns can be regarded as semantic concepts, which are useful in video retrieval. As shown in Fig. 1, generally such three pictures are all regarded as with the semantic concept *building*. However, there are large differences among them in appearance. If we combine shot size patterns with these semantic concepts, we can obtain more accurate retrieval results.

(3) Shot size information facilitates semantically analyzing videos in higher level, such as tracking the *intention* of home videographers [11].

Shot size classification has already been extensively studied in sports video as they are useful to identify different views, such as field, audience and player [15]. However, these methods are mainly based on domain knowledge, such as detecting the ratio of playfield in frames [15], thus they can not be applied to other video genres. In [8] the authors annotate shot size patterns based on strict assumptions of video editing structure. Recently, Ferrer *et al.* [5] attempt to classify shot size patterns based on several general audiovisual features. However, their work mainly focuses on films and their features are based on the analysis of Hollywood films. Thus these methods can not be easily extended to other video genres (such as home videos, which are not with so high quality). Although automatic shot size annotation is appealing, how to obtain satisfied annotation accuracy for general videos still remains as a challenging issue.

In this paper, we propose an efficient learning scheme for automatic annotation of video shot size. Here we demonstrate our scheme by annotating shot size as three categories, including close-up (CU), medium shot (MS), and long shot (LS), as shown in Fig. 1. It is worthy noting that our scheme is extensible – we can easily introduce more categories, such as medium close-up and medium long shot. In our scheme, besides widely applied low-level features, we develop a mid-level feature set to depict the homogeneous color-texture regions after image segmentation, since it is observed that shot size patterns are closely related to the number, sizes and shapes of the objects in video frames. To deal with the fact that training data are usually limited and consequently classifiers learnt on training data are not accurate, we employ co-training to boost the accuracies of these classifiers by leveraging unlabeled data. Then, we make the final decisions by taking subjectivity-consistent costs of different mis-classifications into account: the cost for confusion of CU and LS is twice larger than other mis-classifications.

The organization of this paper is as follows. Section 2 briefly introduces the proposed scheme. In Section 3, we detail the features employed for shot size annotation. In Section 4, we introduce our classification approach, including co-training and cost-sensitive decision. Experimental results are provided in Section 5, followed by concluding remarks in Section 6.

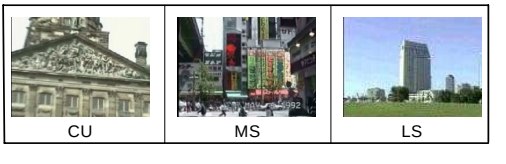


Fig. 1. Examples of semantic concept *building* with different shot size patterns

2 Scheme Overview

The proposed video shot size annotation scheme is illustrated in Fig. 2. Firstly, from video data we extract features, including low- and mid-level feature sets. The detailed employed features are introduced in the next Section. Then two Support Vector Machine (SVM) models are trained on several pre-labeled data. After that, we apply the co-training process to the two SVM classifiers with the help of unlabeled samples, and the two refined SVM classifiers are then combined to generate preliminary results. Finally we make the final decisions according to cost-sensitive criterion based on the truth that different mis-classifications are with different costs in subjectivity.

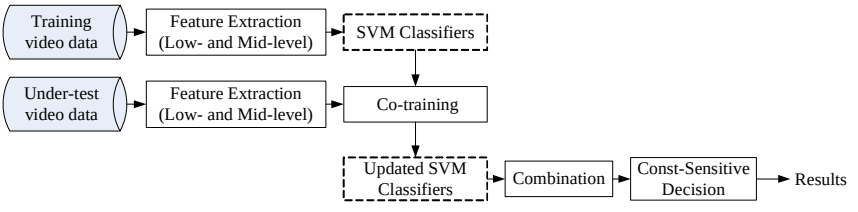


Fig. 2. Proposed video shot size annotation scheme

3 Feature Sets

3.1 Low-Level Feature Set

To discriminate different shot size patterns, the first step is to select the feature sets closely related to shot size. Here we choose a 95D low-level feature set, which consists of 45D block-wise color moment features, 15D edge distribution histogram features, 15D TAMRUA texture features, and 20D MRSAR texture features (as shown in Fig. 3). Experimental results in Section 4 indicate that such a low-level feature set is effective for shot size classification.

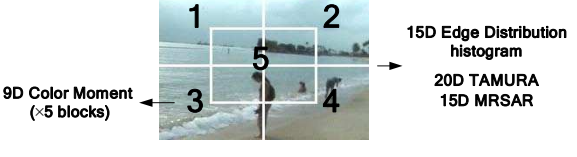


Fig. 3. The form of 90-dimensional feature set

3.2 Mid-Level Feature Set

As aforementioned, shot size patterns are related to the number, sizes, and shapes of the objects that are being captured. Consequently, we develop a mid-level feature set to depict these properties of color-texture homogeneous regions based on image segmentation. To sufficiently explore information, as shown in Fig. 4, the image is segmented with three different scales in a pyramid form (this can be easily achieved

by adjusting segmentation threshold for general image segmentation algorithms). The features introduced below are separately extracted with all three segmentation scales.

After image segmentation, the following features are extracted: (1) the number of regions; (2) variance of the region sizes; (3) mean of the region centers; and (4) covariance of the region centers. In addition, following features are extracted to depict each of the first three largest regions: (1) size of the region; (2) center of the region; and (3) minimal rectangular box that covers the region. In this way we can obtain an 84D mid-level feature vector.

For clarity, we illustrate all of the low- and mid-level features in Table 1.

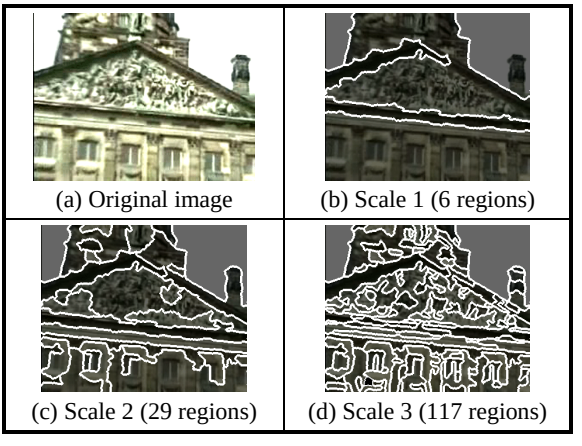


Fig. 4. Example of image segmentation with different scales

Table 1. Features for shot size annotation

Level	Type	Features	Dim
Low (95D)	Color	Color Moment	45
	Edge	Edge distribution histogram	15
	Texture	TAMRUA	15
		MRSAR	20
Mid (84D)	About all regions	Number of regions	3
		Variance of region sizes	3
		Mean of region centers	6
		Covariance of region centers	9
	About individual region	Size of largest region	3
		Location of largest region	18
		Size of 2-nd largest region	3
		Location of 2-nd largest region	18
		Size of 3-rd largest region	3
		Location of 3-rd largest region	18

4 Classification Approach

Insufficiency of training data is a major obstacle in many learning and mining applications. The video shot size annotation task may also encounter this problem as

manually labeling training data is a labor-intensive and time-consuming process. To tackle the training data insufficiency problem, many different semi-supervised algorithms have been proposed to leverage unlabeled data [2, 12, 14]. Among existing semi-supervised methods, co-training is widely acknowledged for its potential to learn from complementary feature sets. In this study, we apply co-training to exploit unlabeled data based on the low- and mid-level features. Then we introduce a set of subjectivity-consistent costs for different mis-classifications, and make the decisions by cost minimization instead of error minimization criterion.

4.1 Co-training on Low- and Mid-Level Features

Co-training [2] is a semi-supervised learning algorithm that is designed to take advantage of complementary descriptions of samples. It starts with two initial classifiers separately learnt from two feature sets. Each classifier is then iteratively refined using an augmented training set, which includes original training samples and additional unlabeled samples with highest classification confidences from the other classifier.

We apply co-training to the shot size annotation task based on the low- and mid-level features. It is worthy mentioning that co-training is only effective when the two feature sets are nearly independent [12]. To confirm this condition, we illustrate in Fig. 5 the correlation map of the two feature sets calculated from 4,000 samples. As we can see that there is little correlation between the low- and mid-level feature sets, it is rational for us to employ co-training on these two feature sets (encouraging experimental results also support it).

Detailed co-training process is shown in Fig. 6. We adopt Support Vector Machine (SVM) as the classifiers. To estimate classification confidences in co-training, as well as make the cost-sensitive decisions detailed in next sub-section, we have to map outputs of SVM classifiers to posterior probabilities. Here we apply the method proposed in [13], which achieves the mapping based on a parametric form of sigmoid.

Denote the posterior class probabilities from the two SVM classifiers by $P^1(l_i|x)$ and $P^2(l_i|x)$, where i=1, 2, 3 (here l_1, l_2, l_3 are corresponding to CU, MS, and LS respectively). Based on the mapped posterior probabilities, the classification confidences are estimated according to [9] as follows

$$\psi^j(x) = \sqrt{P_1^j(P_1^j - P_2^j)},\ j = 1,2 \tag{1}$$

where P_1^j and P_2^j are the largest and 2-nd largest posterior probabilities respectively among $P^j(l_1|x)$, $P^j(l_2|x)$, and $P^j(l_3|x)$.

After co-training, the combined posterior probabilities can be easily derived by assuming that the outputs from the two SVM classifiers are independent. They are calculated as follows

$$P(l_i \mid x) = \frac{P^1(l_i \mid x)P^2(l_i \mid x)}{\sum_{i=1}^{3} P^1(l_i \mid x)P^2(l_i \mid x)}. \tag{2}$$

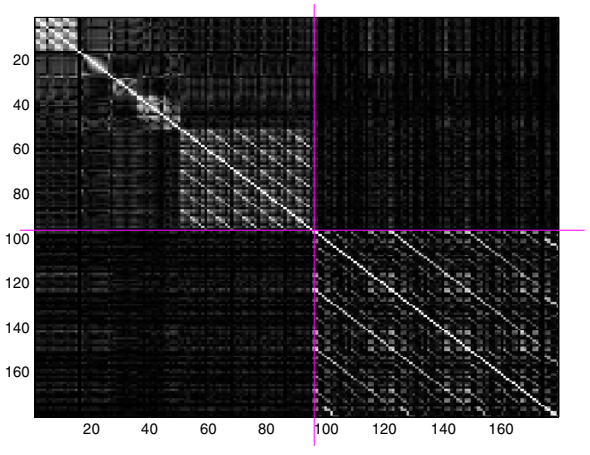


Fig. 5. Feature Correlation Map

Input:

Two feature sets V_1 and V_2; a set of labeled samples L; and a set of unlabeled samples U.

While U is not empty **Do**

C_1 teaches C_2:

(a) Train classifier C_1 based on feature sets V_1 on training data set L.
(b) Classify all samples in U using classifier C_1.
(c) Move the top-n samples from U on which C_1 makes the most confident predictions to L with their predicted labels.

C_2 teaches C_1:

(a) Train classifier C_2 based on feature sets V_2 on training data set L.
(b) Classify all samples in U using classifier C_2
(c) Move the top-n samples from U on which C_2 makes the most confident predictions to L with their predicted labels.

End While

Output:

Classifiers C_1 and C_2

Fig. 6. A typical co-training scheme

4.2 Cost-Sensitive Decision

Although we regard CU, MS, and LS as three different classes, relationships exist between them: typically in subjectivity we consider the difference between CU and

LS is larger than the difference between CU and MS as well as MS and LS. Thus, it is rational for us to introduce costs for different mis-classifications.

It is widely known that classification by maximum posterior probability is derived from the criterion of error minimization. Meanwhile, we can also introduce costs for different mis-classifications and make decision by the criterion of cost minimization. This is generally called cost-sensitive learning, which is an extensively studied topic in machine learning [4]. Although many different cost-sensitive learning methods have been proposed, we adopt a simple but efficient method to make the decisions as follows

$$L(x) = \max_i \sum_j P(l_j \mid x) C(i, j), \tag{3}$$

where $P(l_j|x)$ is the posterior class probability estimated as Eq. (2) and $C(i, j)$ is the cost of predicting label l_i when the truth is l_j. If $P(l_j|x)$ is accurate, then Eq. (3) achieves optimal decisions.

In our study, we adopt the costs illustrated in Table 2, i.e., the mis-classification between CU and LS has the twice cost than the other mis-classifications.

Table 2. Cost Table

***C*(*i*, *j*)**	*j*=1	*j*=2	*j*=3
i=1	0	1	2
i=2	1	0	1
i=3	2	1	0

C(*i*, *j*): cost of predicting class *i* when truth is class *j*

5 Experimental Results

To evaluate the performance of our approach, we conduct several experiments on 20 home videos, which are about 20 hours in duration. These videos are captured by several different camcorder users and include diverse content, including wedding, journey, conference, etc. Here we choose home video to evaluate our scheme due to the fact that home videos usually contain diverse content and they are with relatively low visual quality compared with other video genres.

These videos are segmented to about 1000 shots according to timestamps recorded in DV. Then we further segment these shots into 4000 sub-shots, since a shot may contain different shot size patterns. Each sub-shot is assumed to have an identical shot size pattern. We find three volunteers to manually identify the shot size pattern for each sub-shot, and decide its truth by voting between the three labels. Figure 7 illustrate several snap shots of class examples.

After that, a key-frame is selected from each sub-shot, and the features introduced in Section 3 are extracted from this key-frame (JSEG [3] is adopted to do the image segmentation). In the experiments, 20% of the samples are randomly selected to be training data, and others are test data. All experimental results are the average of 10 runs.

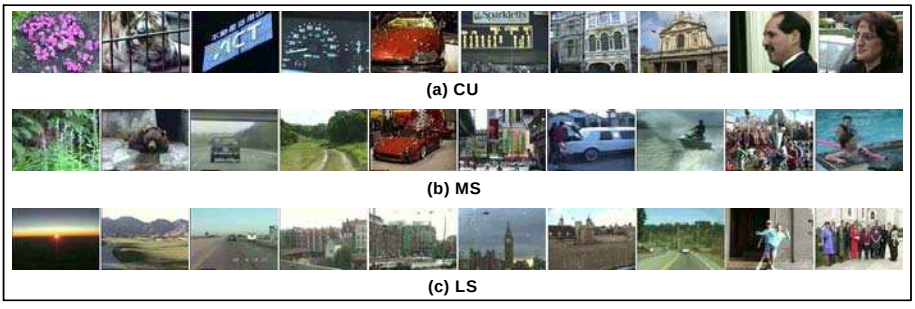


Fig. 7. Several snapshots of class examples

A. Experiments on Different Feature Sets

Firstly we compare contributions of different feature sets in a baseline scheme (i.e., SVM classifiers are learnt on training data and then they are used to do classification for test data). Here we adopt SVM classifier with a RBF kernel, where the parameters are optimally selected by 5-fold cross-validation method. We illustrate the results in Table 3.

Table 3. Classification results of baseline scheme based on different feature sets

Feature Set	Test Error
95D low-level features	0.276
84D mid-level features	0.327
complete features	0.265

From Table 3 we can find that both low- and mid-level features are discriminative for shot size patterns (note that the random test error should be 0.667 for the classification of three classes). However, the performance based on direct combination of low- and mid-level features only has limited improvement over the individual low-level feature set.

B. Experiments on Co-training

To demonstrate the effectiveness of co-training in our approach, we illustrate in Fig. 8 the learning curves of the two SVM classifiers and their combined results. They run up to 10 iterations, and every iteration 150 samples with highest confidences are added to training set for each classifier. From the figure we can see that co-training is effective to boost performances of the two independently learnt SVM classifiers.

C. Experiments on Cost-Sensitive Decision

We list in Fig. 9 the detailed results of classification by error minimization and cost minimization criteria introduced in Section 3.2. Here $n(i, j)$ stands for the number of shots classified to be l_i while its truth is l_j, and "P" and "R" indicate *precision* and *recall* respectively. From the two tables in the Figure, we can see that our approach can significantly reduce the cost by reducing the mis-classifications between CU and LS.

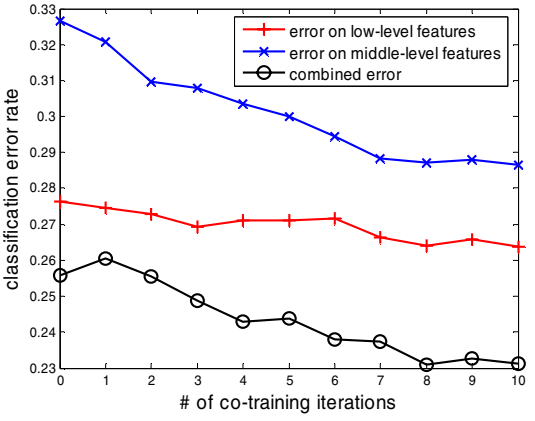


Fig. 8. Learning curves of co-training

Error Minimization
(error=0.232, cost=0.296)

$n(i, j)$	j=1	j=2	j=3
i=1	851	201	90
i=2	232	903	90
i=3	167	146	1320

	CU	MS	LS
P	0.745	0.737	0.808
R	0.681	0.722	0.880

Cost Minimization
(error=0.243, cost=0.274)

$n(i, j)$	j=1	j=2	j=3
i=1	718	94	48
i=2	452	1080	221
i=3	80	76	1231

	CU	MS	LS
P	0.835	0.616	0.888
R	0.574	0.864	0.821

Fig. 9. Experimental results of error minimization and cost minimization criteria

6 Conclusions and Future Work

This paper proposes an efficient learning scheme for automatic video shot size annotation. Different to existing methods developed for specific video genres using corresponding domain knowledge, our scheme is towards general video genres by adopting general features and learning methods. Encouraging experiments prove that our approaches in the scheme are effective: proposed features are discriminative for different shot size patterns, co-training can significantly boost the accuracies of classifiers learnt on training data, and cost-sensitive decision is effective to reduce misclassifications between CU and LS.

It is worthy mentioning that all the features used in our study are extracted from key-frames. Thus our shot size annotation scheme can be applied to images as well. Although in the study it seems that the low-level features outperform the mid-level

features, it is partially due to the fact that our experiment dataset is still not large enough, so that the variation of the low-level features is not very large. We argue that mid-level features can be comparative or even outperform low-level features if large scale of video data is incorporated. Meanwhile, we will try to introduce more features, such as camera motion. These works will be discussed in our future work.

References

1. TRECVID: TREC Video Retrieval Evaluation, http://www-nlpir.nist.gov/projects/trecvid
2. Blum, A., and Mitchell, T. Combining labeled and unlabeled data with co-training. In *Proceedings of COLT*, 1998
3. Deng, Y. and Manjunath, B. S., Unsupervised segmentation of color-texture regions in images and video. *IEEE Trans. Pattern Analysis and Machine Intelligence (PAMI)*, 2001
4. Elkan, C. The foundation of cost-sensitive learning. In *Proceedings of IJCAI*, 2001.
5. Maria, Z. F., Barbieri, M. and Weda, H., Automatic classification of field of view in video, In *Proceedings of ICME*, 2006
6. Hua, X. S., Lu, Lie and Zhang, H. J., AVE – Automated Home Video Editing, In *Proceedings of ACM Multimedia*, 2003
7. Kumano, M., Ariki, Y., Amano, M. and Uehara, K. Video editing support system based on video grammar and content analysis. In *Proceedings of ICPR*, 2002.
8. Kumano, M., Ariki, Y., Tsukada, K and Shunto., K., Automatic shot size indexing for a video editing support system, In *Proceedings of CBMI*, 2003
9. Li, B., Goh, K. and Chang, E. Confidence based dynamic ensemble for image segmentation and semantic discovery, In *Proceedings of ACM Multimedia*, 2003
10. Matsuo, Y., Amano, M. and Uehara, K. Mining video editing rules in video streams. In *Proceedings of ACM Multimedia*, 2002.
11. Mei, T. and Hua, X. S. Tracking users capture intention: a novel complementary view for home video content analysis, In *Proceedings of ACM Multimedia*, 2005
12. Nigam, K. and Ghani, R. Analyzing the effectiveness and applicability of co-training. In *Proceedings of CIKM*, 2000
13. Platt, J. C., Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods, *In Proceedings of Advances in Large Margin Classifiers*, 1999
14. Seeger, M. *Learning with Labeled and Unlabeled Data*. Tachnical report, Edinburgh University, 2001.
15. Tong, X. F., Duan, L. Y, Lu, H. Q., Xu, C. S., Tian, Q and Jin, J. S. A mid-level visual concept generation framework for sports analysis. In *Proceedings of ICME*, 2005.

Content Based Web Image Retrieval System Using Both MPEG-7 Visual Descriptors and Textual Information

Joohyoun Park and Jongho Nang

Dept. of Computer Science and Engineering, Sogang University, 1, ShinsuDong, MapoGu, Seoul 121-742, Korea
{parkjh, jhnang}@sogang.ac.kr

Abstract. This paper introduces a complete content based web image retrieval system by which images on WWW are automatically collected, searched and browsed using both visual and textual features. To improve the quality of search results and the speed of retrieval, we propose two new algorithms such as a keyword selection algorithm using visual features as well as the layout of web page, and a k-NN search algorithm based on the hierarchical bitmap index [17] using multiple features with dynamically updated weights. Moreover, these algorithms are adjusted for the MPEG-7 visual descriptors [14] that are used to represent the visual features of image in our system. Experimental results of keyword selection and image retrieval show the superiority of proposed algorithms and a couple of visual interfaces of the system are presented to help understanding some retrieval cases.

Keywords: Content based image retrieval, auto-annotation.

1 Introduction

Advent of new technologies in WWW (World Wide Web) and personal devices such as digital camera and mobile phone lead to increase the number of images on the WWW dramatically. Consequently, the needs of efficient searching by example or keyword have been increased as well. To fulfill these needs, there are three main issues should be considered carefully.

The first issue is how to annotate images collected from WWW automatically. There were some researches [1-3] which describe the problem of the image auto-annotation as a supervised or an unsupervised learning problem which builds up the relationship between visual features and concepts (textual features). Unfortunately, the annotations which generated by this approach would not describe the image content accurately because of the problem called "Semantic Gap [4]". Even though the images in web pages can be annotated and assigned to the images automatically by analyzing the layout on web pages where the descriptive texts are staying close to the images [5-8], it would produce many irrelevant annotations as well as relevant

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 659 – 669, 2007.
© Springer-Verlag Berlin Heidelberg 2007

ones because of the lack of measures which could evaluates the degree of relevance between the surrounding texts and the images. Another issue would by the way to define the similarity of images, which is the basis of CBIR (Content-Based Image Retrieval). This issue may include which features are used – in broad sense, features may include both textual and visual features – and how to calculate the distance between images. Several studies [9-12], which proposed their own visual features and similarity measures, have been made on CBIR. Final issue is how to reduce the search time which is incurred by the high dimensionality of features. To make the system scalable to large set of images, the use of efficient high dimensional indexing method needs to be considered seriously.

In this paper, the content based web image retrieval system using both MPEG-7 visual descriptors [14] and textual information with sufficient consideration for the above three issues will be introduced. There are three main components in the system such as the web image miner, the search server, and the search client. The web image miner periodically collects images on the WWW and extracts the visual and textual features from those images. The textual features are selected using both the visual features and the layout of web pages in order to improve the correctness of keyword selection. The collected images and the features extracted from those images are delivered to the database manager in the search server, which manages the three databases such as an image database, a keyword database, and a visual feature database. For efficient retrieval by combining visual and textual features, they are indexed together by the HBI (Hierarchical Bitmap Indexing) [18], an efficient high dimensional indexing method. Since all features must be represented as vector form to index it, the way to convert each feature to vector form should be considered. Based on theses databases, every image in the image database is ranked by the search engine according to the query object which is generated by the search client.

2 System Architecture

The system consists of three major components as shown in <Fig. 1>. The first component is the web image miner consists of three tools such as an image collector and a keyword extractor. The image collector periodically crawls in the WWW and collects image and the words around that image. Then MPEG-7 visual descriptors [14] would be extracted from the images and some keywords for the images are selected by the keyword extractor. The second component is the search server which consists of a database manager and a search engine. The database manager manages visual features, textual features, and images and indexes them for efficient retrieval. Based on these databases, the images in the Image database are ranked by the search engine according to the visual or textual query which is sent from the search client. The third component is the search client which generates a query object and helps to browse the image from the results.

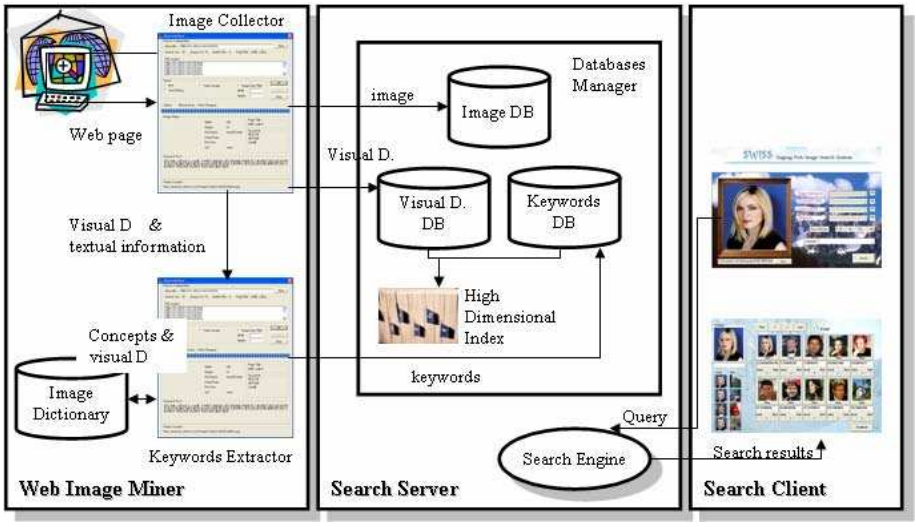


Fig. 1. The architecture of content based web image retrieval system which consists of 3 components such as web image miner, search server, and search client

3 Keyword Selection Algorithm

3.1 The Use of Image Dictionary

The meaning of Image Dictionary is the data structure which represents the relationship between the visual information and the concept (textual information). This relationship could be built up by the following learning process, which is similar to [3].

First, many sample images with manual annotations were collected in order to learn the concepts associated with the visual information. To remove the noises which were incurred by complicated images with multi-objects, each sample image is segmented into 3x3 uniform blocks, which are defined in MPEG-7 visual descriptors [14] such as *dominant color*, *color layout*, and *edge histogram* are extracted from. Based on these features, each block is clustered by *k-means clustering* algorithm with equal weights. Then each cluster has the blocks with similar visual properties and with the words annotated manually at the image preparation step. Finally, the representative keywords of each cluster are selected by the frequency of the words annotated to the blocks in the cluster.

3.2 Keyword Selection Algorithm

All words in the web page may not be evenly relevant to the image content. That is, the words with specific HTML tags could be more relevant than all other words in the web page. For example, according to the weighting scheme in [5], the words closer to the image or appearing with *src*, *alt* fields of the *img* tag, *title*, and *headers* may have higher importance as compared to other words. However, some words with higher

weights may not be relevant to the image content because the weights are evaluated by analyzing the layout of web page not the image content.

<Fig 2> shows the process of the proposed keyword selection algorithm to cope with the above problem. Initially, a HTML document is parsed into an image and its textual information (surrounding texts, pairs of word and its tag). The candidate keyword selector generates the pairs of candidate keyword and its weight from the textual information based on the weighting scheme in [5]. Furthermore, the image concept extractor analyzes the image to find the concepts associated to the image. Finally, the keyword selector with WordNet [16] filters out some irrelevant candidate keywords by comparing with the concepts associated to the image. The detail of the filtering process is as follows;

Assume that the number of the candidate keywords and the number of concept is l and m respectively. For each candidate keyword k_i $(1 \le i \le l)$, its final weight w'_i is calculated as follows;

$$w'_i = (1-\alpha)\cdot w_i + \alpha \cdot s_i \quad , (0 \le \alpha \le 1, 1 \le i \le l)$$
$$where, s_i = \max\left\{ \frac{w_j^c}{d_{i,j}} \middle| 1 \le j \le m \right\} \tag{1}$$

Note that w_i is the weight for the i-th candidate keyword and w_j^c is the weight for the j-th concept. $d_{i,j}$ means the length of the shortest path between k_i and the j-th concept in the word graph of WordNet[16]. Also, α controls relative importance of the visual features compared to the layout of web page. Top 5 words with higher weights will be selected as the final keywords for the image.

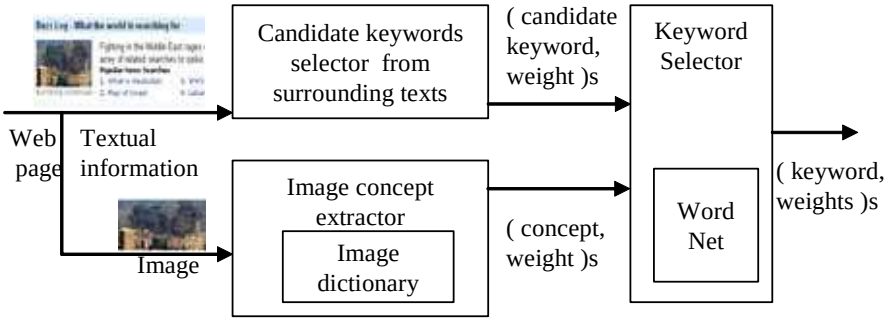


Fig. 2. The process of keyword selection

4 Content Based Image Retrieval

In this section, we will show how to represent textual features and visual features to vector form and how to index these feature vectors. We also discuss how to retrieve images based on the hierarchical bitmap index using multiple features with dynamically updated weights.

4.1 Vector Representation of Textual and Visual Features

4.1.1 Generating Textual Feature Vectors

As the results of the web image miner, each collected image has some keywords with their weights. Then, we can easily create the term matrix A (m x n), of which an element a_{ij} represents the weight of the j-th word in the i-th image. Note that m is the number of collected images and n is the number of words which can be used as keyword. From this matrix, we can use the vector a_i as the textual vector for the i-th image. It works but it can not resolve two problems that different words can be used to express the same concepts and the dimensionality of vector is too high. As the solution of these problems, we use an existing method known *Latent Semantic Analysis* (LSA) [5], which is commonly used in text analysis.

LSA decomposes the matrix A into three matrices U, S, and V by the singular value decomposition (SVD), $A = USV^T$, where $U \in \Re^{m \times k}$, $S \in \Re^{k \times k}$, $V \in \Re^{n \times k}$, and $U^T U = V^T V = I$. This operation reduces the dimension of the term vector by k dimension and captures statistically the semantic association across the terms in the set of terms with size n. Then the vector u_i $(1 \le i \le m)$, the i-th row of the matrix U, can be used as the textual vector for the i-th image with k dimension.

4.1.2 Generating Visual Feature Vectors

As Visual features, an image is represented as a subset of 9 visual descriptors which are defined in the visual part of the MPEG-7[14]. According to [15], the best descriptors for these combinations are *dominant color*, *color layout*, *edge histogram*, and *texture browsing* in terms of statistical properties for the judgement of the quality of descriptors such as redundancy, sensitivity, and completeness. *Texture browsing* is excluded from these descriptors because the general usage of it is not comparing of two images but browsing of images with similar perceptual properties. Finally, *dominant color*, *color layout*, and *edge histogram* are used.

In the MPEG-7 visual part of eXperience Model (XM) [13], the special metric of each descriptor is also defined. Therefore, it is necessary to check whether the data space where each descriptor is represented as vector space or not to index it. *Color layout* and *edge histogram* can be indexed without any modification because their metrics are *Euclidean distance* or *Manhattan distance* respectively. However, *dominant color* can not be indexed because its metric do not satisfy the properties of vector space or metric space. Consequently, it has necessitated a slight modification. Even though the definition and the metric function of *dominant color* looks complicate, it could be represented as the form of quantized color histogram with *Euclidean distance* [18].

4.2 Content Based Image Retrieval Using Visual and Textual Feature Vectors

To describe the way to calculate the distance of two images, it is necessary to formalize an image as visual and textual features. Consider an image database

Λ ($\Lambda = \{ o_i \mid 1 \le i \le n \}$, where o_i is the i^{th} image object.) with n image objects. An image object o_i is represented as a combination of feature vector as follows;

$$o_i = [t_i, d_i, c_i, e_i] \tag{2}$$

Note that t_i is a vector of textual feature and d_i, c_i, e_i are the vectors of *dominant color*, *color layout*, and *edge histogram* respectively associated with the image o_i. Then, total distance between the two images o_i and o_j, $D(o_i, o_j)$ is could be defined as follows;

$$\begin{aligned} &D(o_i, o_j) = \sum_{k=t,d,c,e} w_k \cdot GausNorm(D(k_i, k_j)) \quad (w_t + w_d + w_c + w_e = 1) \\ &where,\ D(t_i, t_j) = L_2(t_i, t_j),\ D(d_i, d_j) = L_2(d_i, d_j),\ D(c_i, c_j) = L_2(c_i^Y, c_j^Y) + L_2(c_i^{Cb}, c_j^{Cb}) + L_2(c_i^{Cr}, c_j^{Cr}), \\ &and \quad D(e_i, e_j) = L_1(e_i^l, e_j^l) + 5 \cdot L_1(e_i^g, e_j^g) + L_1(e_i^s, e_j^s) \end{aligned} \tag{3}$$

Note that *GausNorm* means Gaussian Normalization which normalized the distance of each feature within [0, 1]. To keep the original metrics defined in the MPEG-7 visual part of XM, the vectors of *color layout* and *edge histogram* must be split into 3 sub-vectors respectively before the distances are calculated. That is, c_i is split into the DCT coefficients for the luminance c_i^Y, and c_i^{Cb}, c_i^{Cr} for the chrominance. e_i is also split into the local edge histogram e_i^l, the global edge histogram e_i^g, and the semi global histogram e_i^s.

Similarity search problem in Λ can be formulated as a *k*-NN (Nearest Neighbor) problem because the distance measure between two images is defined. Also, the hierarchical bitmap indexing (HBI) [17] method is applied to solve the problem incurred by high dimensionality of features. With HBI, each feature vector is represented as a compact approximation and it reduce the time to calculate the distance of two images. The most irrelevant images can be filtered out during the process of scanning these approximations.

Let $B_p(\cdot)$ be the approximation of $L_p(\cdot)$ calculated using bitmap index. Then $D'(o_i, o_j)$, the approximation of the distance between the two images o_i and o_j, can be calculated as follows;

$$\begin{aligned} &D'(o_i, o_j) = \sum_{k=t,d,c,e} w_k \cdot GausNorm(D'(k_i, k_j)) \quad (w_t + w_d + w_c + w_e = 1) \\ &\quad where,\ D'(t_i, t_j) = B_2(t_i, t_j),\ D'(d_i, d_j) = B_2(d_i, d_j), \\ &\quad D'(c_i, c_j) = B_2(c_i^Y, c_j^Y) + B_2(c_i^{Cb}, c_j^{Cb}) + B_2(c_i^{Cr}, c_j^{Cr}), \\ &\quad and \quad D'(e_i, e_j) = B_1(e_i^l, e_j^l) + 5 \cdot B_1(e_i^g, e_j^g) + B_1(e_i^s, e_j^s) \end{aligned} \tag{4}$$

According to [17], $L_p(v_1, v_2)$ is always bigger or equal than $B_p(v_1, v_2)$ for any vector v_1, v_2. It implies $D(k_i, k_j) \ge D'(k_i, k_j)$, where *k*=*t*, *d*, *c*, *e*. Therefore, it always satisfies the condition $D(o_i, o_j) \ge D'(o_i, o_j)$. From this property, *k*-NN search algorithm for this CBIR system as shown in <Fig. 3> can be created. In this algorithm, the candidate set could not be generated completely during the filtering process because objects should be selected which distance to the query *relatively*

small. It forces us to keep a set of potential nearest objects, and the real distance of an image object is calculated only when its approximation of distance is less than the largest real distance among the distances of image objects in this set. If its real distance is less than the currently largest one, it is inserted and the image object whose real distance is the largest among the image objects in the set is deleted.

```
// o_q : the query image object    // w : the vector of weights associated with features
// o_i : the i-th image objects in the database Λ
// C_kNN−search : a set of candidate image objects for k-NN search
// kNNDist: the maximum distance between the query and the objects in C_kNN−search
// SelectMaxObject( C_kNN−search ) : a function that selects the image object from C_kNN−search
//                  that has the maximum distance to query image object
// FindMaxDist( C_kNN−search ) : a function that find the maximum distance between the query
//                  image object and the objects in C_kNN−search
Procedure k-NN Search( o_q , k,  w ){   //  k  is the number of nearest objects to find
   C_kNN−search ={}; kNNDist = MaxDist;
  for  ∀o_i  (1 ≤ i ≤ n)  do {
     if ( |C_kNN−search| < k ) { // if the number of candidate objects is less than k,
          C_kNN−search  =  C_kNN−search ∪ {o_i}  ; // insert  o_i  into the candidate set
     }
     else{
        apxDist = D'(o_i, o_q);
   // Filtering Process ; Compute real distance D(o_i, o_q)  only when  D'(o_i, o_q) < kNNDist
        if ( apxDist < kNNDist) {
           realDist = D(o_i, o_q);
           if ( realDist < kNNDist) {
              o_max = SelectMaxObject( C_kNN−search ) ;
              C_kNN−search  =  C_kNN−search −{o_max} ∪ {o_i}  ;  // replace  o_max  with  o_i
              kNNDist = FindMaxDist( C_kNN−search ) ;
           }
        }
     }
  }
}
```

Fig. 3. A *k*-NN search algorithm with HBI

5 System Implementation and Experiments

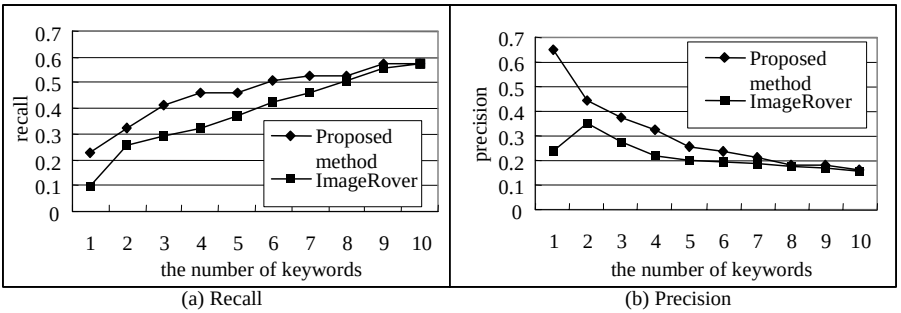
A fully functional web image retrieval system were implemented and tested based on the proposed algorithms. Every component of the system is tested under Windows XP on a Pentium 4 (3.0GHz) with 1GB memory.

5.1 Web Image Miner

In the web image miner, once the image collector starts to find images on the site specified by user, it continuously visits the web pages which are hyperlinked from the current page by breadth first search (BFS). If a visited page includes an image file, it downloads the image and passes it to the MPEG-7 visual descriptor extractor module which is programmed based on XM codes [13]. Three visual descriptors such as *dominant color* with 5 colors, *color layout* with 18 coefficients (6 for both luminance and chrominance), and *edge histogram* with 80 bins will be extracted from the image. After that, these visual descriptors and the HTML code of web page are passed to the keyword extractor module to extract keywords associated with the image by the proposed keyword selection algorithm.

To show the superiority of the proposed keyword selection algorithm compared to ones without use of visual features, experimental results were evaluated using precision and recall. To build up the image dictionary, 500 images are labeled manually and collected with 50 concepts such as landscape, animals, vehicles, and so on. And also the number of labels were restricted manually annotated for each image to 2~6 and set the number of clusters to 10.

80 web pages were collected to evaluate the proposed method where the page includes images associated with the concepts used in the learning stage. As shown in <Fig. 4>, both recall and precision of the proposed method are higher than those of ImageRover [5] and the difference of recall and precision between the two methods are decreased as the number of keywords increase. It implies that more relevant words to the image content get higher weights by the proposed algorithm.



(a) Recall (b) Precision

Fig. 4. The recall and precision of the proposed method compared to that of ImageRover[5] as a function of the number of keywords

5.2 Search Server and Search Client

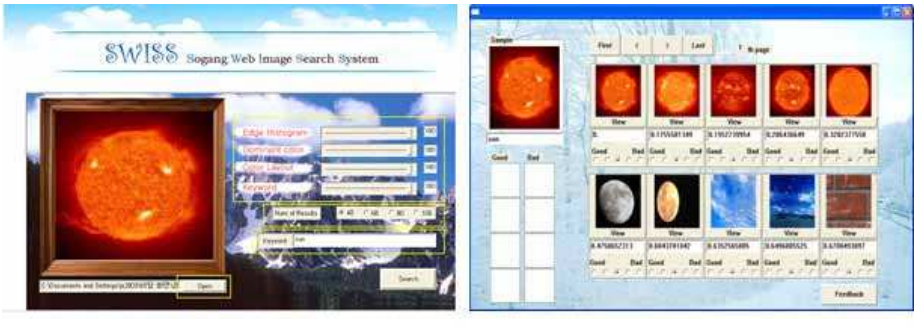
In our system, the search server consists of a database manager and a search engine. Whenever the database manager accepts an image and its features from the web image miner, those image and features will be saved to the temporary folder before inserting them into the database. The reason for it is that the vector representation and creation of index are CPU-consuming jobs. Therefore, the database manager is

designed to automatically trigger the insertion process when the number of collected images come up to the threshold user specifies (this threshold is set to 500 in our experiments).

Once the insertion process has triggered, all features are transformed to vector from and the unique identifier will be assigned to each image, which is used as the linker between an image and its index. Based on these identifiers, an index file per each feature is created respectively. Consequently, the system will have 8 index files for textual vectors, *dominant color* vectors, *Y* , *Cb*, *Cr* coefficient vectors of *color layout*, and local, global, semi global *edge histogram* vectors respectively.

Based on these index files, the search engine ranks the images in the database with regard to the query with the weights of features from the search client by the *k*-NN algorithm as mentioned in section 4.2 as the search results. To show the efficiency of the proposed *k*-NN algorithm using HBI, after 100,000 images were collected on the WWW and inserted into the search server, the total search time for 100 randomly generated query objects were evaluated. The meaning of total search time is that the time only for images ranked in the search server. According to our experiments, the total search time of the *k*-NN search using the proposed algorithm takes 960 ms, while the brute force search is about 2,500 ms on average. It implies the proposed *k*-NN search method is about 2.5 times faster than the brute force search. The detail of the performance of HBI, please refer to [17].

The search client provides convenient way of querying, browsing, and feedback. <Fig. 5>-(a) shows the querying interface of search client that it supports both query by example and keywords and also weight of importance could be specified by user. As the start of search, visual features extracted from example image and query for keywords will be sent to the search server and the search results will be shown as <Fig.5>-(b).



(a) Querying interface (b) Browsing interface

Fig. 5. Querying and browsing interface of the search client

<Fig. 6> is a good example of retrieval by combined visual and texture features. In <Fig. 6>-(a), both images of “star” and “Hollywood starts” are shown because only the textual features are used with the query string “star”. On the contrary, some odd

images are shown in <Fig. 6>-(b) because the images are retrieved by only visual features. Finally, the Hollywood star images could be retrieved by combination of visual and textual features as shown in <Fig. 6>-(c).



(a) The results of the query by keyword "star"

(b) The results of the query by example

(c) The results of the combined query by example and keyword "star"

Fig. 6. Comparison results of query by keyword and query by example

6 Conclusion

Our content based web image retrieval system was designed and implemented using both textual and visual features. To improve both the quality of results and the speed of retrieval, a new keyword selection algorithm based on both the visual features extracted from images and the layout of web pages, and an efficient k-NN search algorithm based on the hierarchical bitmap index using multiple features with dynamically updated weights was proposed. Also, these algorithms are adjusted to be well-suited with MPEG-7 visual descriptors such as dominant color, color layout, and edge histogram. Based on these algorithms, we built up a complete image retrieval system which provides the functionality for collection, management, searching, and browsing for images effectively. Upon experimental results, recall and precision of the proposed keyword selection algorithm were ranked higher than the existing algorithms. And it also shows that some examples of retrieval were enhanced by combination of visual and textual features. In terms of the efficiency of the system, the proposed *k*-NN search algorithm using HBI was about 2.5 times faster than brute force search when 100,000 images were stored in the server.

References

1. S. Rui, W. Jin, and T. Shua, "A Novel Approach to Auto Image Annotation Based on Pair-wise Constrained Clustering and Semi-naïve Bayesian Model," *Proc. of IEEE Int. Conf. on Multimedia Modeling*, pp.322-327, 2005.
2. L. Wang, L. Liu, and L. Khan, "Automatic Image Annotation and Retrieval using Subspace Clustering Algorithm," *Proceedings of the ACM international workshop on Multimedia Databases*, 2004.
3. Y. Mori, H. Takahashi, and R.Oka, "Image-To-Word Transformation based on Dividing and Vector Quantizing Images with Words," *Proc. of Int. Workshop on Multimedia Intelligent Storage and Retrieval Management*, 1999.

4. R. Yates and B. Neto, *Modern Information Retrieval*, Addison Wesley, pp. 74-84, 1999.
5. M. Cascia, S. Sclaroff, and L. Taycher, "Combining Textual and Visual Cues for Content-Based Image Retrieval on the World Wide Web," *Proc. of IEEE Workshop on Content-based Access of Image and Video Libraries*, pp. 24-, 28, 1998.
6. J. Smith and S. Chang, "WebSeek: An Image and Video Search Engine for the World Wide Web," *IS&T/SPIE Proc. of Storage and Retrieval for Image and Video Database V*, 1997.
7. C. Frankel, M. Swain, and V.Athitsos, "WebSeer: An Image Search Engine for the World Wide Web," *Technical Report 96-14*, University of Chicago Computer Science Department, 1996.
8. N. Rowe and B. Frew, "Automatic Caption Localization for Photographs on World Wide Web Pages," *Information Processing and Management*, Vol.34, No.1, 1998.
9. M. Flickner, et.al., "Query by Image and Video Content : the QBIC System," *IEEE Computer*, Vol.28, pp.23-32, 1995.
10. J. Smith and S. Chang, "VisualSeek : A Fully Automated Content Based Image Query System," *Proceedings of ACM Multimedia 96*, pp.87-98, 1996.
11. J. Bach, et.al., "The Virage Image Search Engine: An Open Framework for Image Managemant," *Proceedings of SPIE Storage and Retrieval for Image and Video Databases*, pp.76-87, 1996.
12. Y. Rui, T. Huang, and S. Mehrota, "Content based Image Retrieval with Relevance Feedback in MARS," *Proceedings of International Conference on Image Processing*, pp.815-818, 1997.
13. ISO/IEC JTC1/SC29/WG11 *MPEG-7 Visual part of eXperience Model Version 11.0*, 2001.
14. ISO/IEC JTC1/SC29/WG11 *Information Technology Multimedia Content Description Interface-Part3: Visual*, 2001.
15. H. Eidenberger, "Statistical Analysis of Content-based MPEG-7 Descriptors for Image Retrieval," *ACM Multimedia Systems Journal*, Vol.10, No.2, 2004.
16. C. Fellbaum, *WordNet: An Electronic Lexical Database*, MIT Press, pp.265~283, 1998.
17. J. Park and J. Nang, "A Hierarchical Bitmap Indexing Method for Content Based Multimedia Retrieval," *Proceedings of the IASTED International Conference on Internet, Multimedia systems, and Application*, pp.223-228. 2006.
18. J. Park and J. Nang, "Analysis of MPEG-7 Visual Descriptors for Data Indexing," *Proceedings of the Korean Information Science Society Conference*, pp. 175-177, 2005.

A New Method to Improve Multi Font Farsi/Arabic Character Segmentation Results: Using Extra Classes of Some Character Combinations

Mona Omidyeganeh[1], Reza Azmi[2], Kambiz Nayebi[3], and Abbas Javadtalab[4]

[1] Iran Telecommunication Research Center (ITRC), Tehran, Iran
momid@itrc.ac.ir
[2] Computer Dep., Azzahra University, Vanak, Tehran, Iran
razmi@alzahra.ac.ir
[3] Electrical Eng. Dep., Sharif University, Teharan, Iran
knayebi@sharif.edu
[4] Computer Eng. Dep., Sharif University, Tehran, Iran
Javadtalab@ce.sharif.edu

Abstract. A new segmentation algorithm for multifont Farsi/Arabic texts based on conditional labeling of up and down contours was presented in [1]. A preprocessing technique was used to adjust the local base line for each subword. Adaptive base line, up and down contours and their curvatures were used to improve the segmentation results. The algorithm segments 97% of 22236 characters in 18 fonts correctly. However, finding the best way to receive high performance in the multifont case is challengeable. Different characteristics of each font are the reason. Here we propose an idea to consider some extra classes in the recognition stage. The extra classes will be some parts of characters or the combination of 2 or more characters causing most of errors in segmentation stage. These extra classes will be determined statistically. We have used a learn document of 4820 characters for 4 fonts. Segmentation result improves from 96.7% to 99.64%.

Keywords: Farsi/Arabic text; Multi font; Character segmentation; Extra classes; Statistical methods.

1 Introduction

OPTICAL character recognition is an attractive branch of pattern recognition with many applications in man ± machine interface and document processing. Intensive research has been done and commercial systems are now available [9]. However, Farsi/Arabic texts have main specifications which make them difficult to recognize. Farsi/Arabic texts are cursive and are written from right to left. A Farsi/Arabic character might have several shapes –from 1 to 4 shapes- depending on its relative position in the word. In addition, some Farsi/Arabic characters have the same shape and differ from each other only by existing of dots or zigzag bar. Each word, machine-printed or handwritten, may consist of several separated subwords. A subword is either a single character or a set of connected characters. Although, seven

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 670 – 679, 2007.
© Springer-Verlag Berlin Heidelberg 2007

Farsi characters out of 32 do not join to their left neighbors, others join to the neighboring characters to make a word or a subword. The neighboring characters, separated or connected, may overlap vertically. These characteristics of Farsi script are shown in Fig. 1. There are many works reported on the recognition of Arabic and Farsi texts e.g. [3, 4, 6, 7, 8, 11, 12, 14 and 15]. There are two main approaches to word recognition: segmentation-based and segmentation-free and hybrid e.g. [2, 5, 10 and 13]. The main problem in Farsi/Arabic segmentation-based systems is character segmentation where each word or subword is first split into a set of single characters and then is recognized by its individual characters.

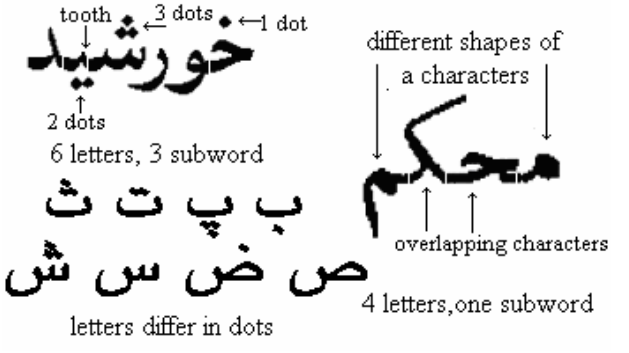


Fig. 1. Some characteristics of Farsi/Arabic script

In this paper, we present a new method to improve the results of multi font Farsi/Arabic text segmentation, by using extra classes in recognition stage. These extra classes may be combination of some characters – which cause problems in segmentation – or part of a character – which is over segmented during the segmentation. Working with Farsi/Arabic multifont texts, is difficult because each font has its own characteristics; and making the algorithm more precise to segment more characters in a font, causes over segmentation in others. Therefore, some errors in segmentation stage are inevitable. So we can use this method to get better results. We worked with 4 different fonts: Yagut, Yekan, Mitra and Nazanin. We prepared a document of 4820 characters for learning stage -for each font-, and examine this idea on these samples statistically. It is important to mention that our test set is separated from our learn one. The paper is organized in 5 sections; In Section 2, our segmentation algorithm [1] is explained. Section 3 describes how we chose new classes. The experimental results are presented in Section 4. Finally, the conclusion is given in Section 5.

2 Character Segmentation Algorithm

Here we have used the algorithm introduced in [1] as our segmentation algorithm. We will describe the algorithm in this section briefly. To learn more about the algorithm see [1]. This algorithm was tested on a data set of printed Farsi texts, containing 22236 characters of 18 different fonts and 97% of characters were correctly

segmented. The test and learn sets were different. In the preprocessing step, the text lines and their words/subwords are segmented by finding the valleys of the horizontal and vertical projection profile [15]. The most frequent size of the black-pixel runs in the vertical histogram of each line columns is adopted as the pen size, ***w***. The global base line will be the horizontal line, all across a text line, with ***w*** width, that covers the maximum number of black pixels in that text line. Each subword is the combination of regions, including bodies, points, zigzag bars, etc. If a region overlaps with base line in some pixels, it is a body (Fig. 2). The pen size is calculated for the bodies of each subword again. Then, the contour of each subword is extracted using a convolution kernel with Lapacian edge detection method. Up and down contours are extracted by moving from right top black pixel to left down black one, and from left down black pixel to right up one clockwise through the contour, respectively. To locate the base line accurately, a technique is used to locally adjust it for each one-fifth of the base line. To do so, the up and down contours of subwords of the determined length of line, traced in CCW, are represented by the eight-directional Freeman code (Fig. 2). Within a distance of ***w/2*** around the upper edge of the global base line, the row of the up contour image having the maximum instances of the code 4, say ***n4***, is considered as the upper bound of the local base line, ***iup***. The lower bound, ***idown***, is found in a similar way, searching for a row with maximum instances of the code 0 in the image of down contour image, say ***n0***, around the lower edge of the global base line. If the width of the resulting local base line is greater than ***1.25w***, then if ***n4*** > ***n0***, the ***iup*** is retained and the ***idown*** is shifted upward, so that the width of the base line becomes ***w***. Otherwise, the ***iup*** is shifted downward in the same way. By local adjustment of the base line, the performance of the segmentation algorithm improves. The pre-processing procedure is shown in Fig. 2.

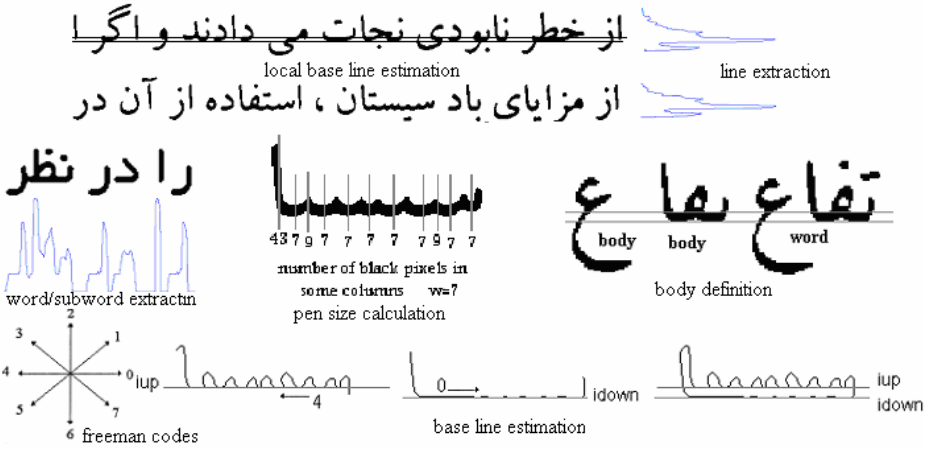


Fig. 2. Pre-processing

The segmentation step is based on the conditional labeling of the up and down contours of each subword (Fig. 3). Tracing the up and down contours from right to left in CCW, each point is labeled -1, 0 and -1 standing for up, middle and down,

respectively - depending on its distance from the base line and the label of its preceding point (fig. 5). The label of the first point of a contour is always up. Fig. 4 shows a sample word and it's labeled up contour. The neighboring points having the same label make a path. A path shorter than *(w/2+1)* is linked to the preceding path. Since in some cases the curves and bends are just in up contour or down contour of subwords, in our algorithm, we label down contours, too.

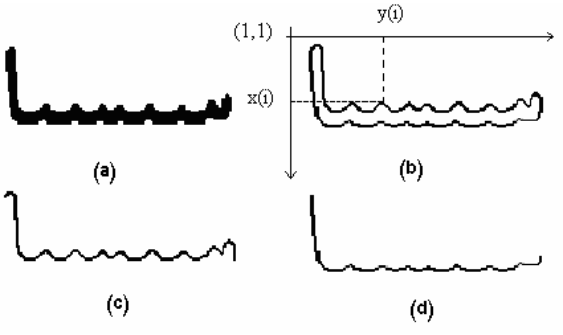


Fig. 3. (a) Body of word (b) its contour (c) up contour (d) down contour

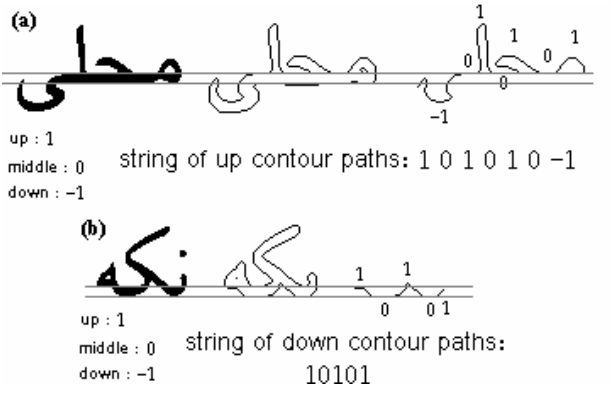
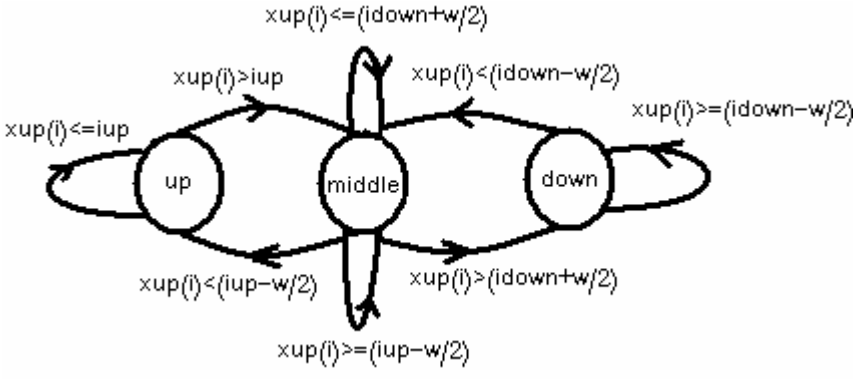
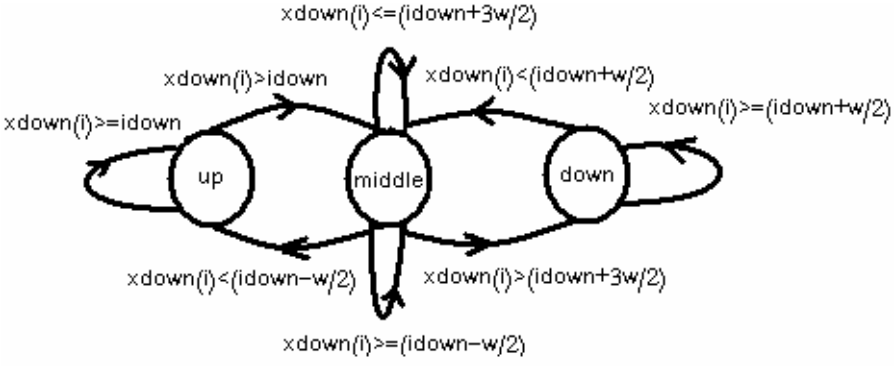


Fig. 4. (a)A word, its contour, and its labeled up contour. (b) A word, its contour and labeled down contour.

Using contour curvature of subwords will improve the segmentation results. Specifically soft bends in subwords are hard to determine with labels. Up contour and down contour of the subword, traced in CCW, are represented by the eight-directional Freeman code, numbered from 0 to 7. The neighboring points having the same number make a group. To smooth the codes, a group shorter than ***w/2*** is linked to the preceding one.



(a)



(b)

Fig. 5. State diagram of (a)up contour (b) down contour labeling process

Character segmentation is done as follows:

- For both up and down contours, if the 0 path (median) is longer than ***w***, and: The previous path and the next path are 1 path and the next path is longer than ***1.5w,*** or the next path is -1 and its length is more than ***2.5w***; or The next path is -1 path, longer than ***4w*** and the last path, the end point of the path is segmentation point.
- If the previous path is a 1(up) path longer than ***w***; and the point in the up contour is in a group with number 2, 3 or 4; and the point in down contour with the same column, is in a group with number 6 or 7, The point is segmentation point.

We divide the length of subword by the number of segmentation points, and compare the result ***r*** with a threshold ***t***. If ***r*** is less than ***t*** the local line will vary by

some conditions and the procedure will be repeated. This step is useful, especially when our base line is determined by mistake. To avoid over segmentation, points nearer than ***w/2*** is gathered to a one point. Some characters, when occurring at the end of a subword, may have a u path that causes a false segment. Some other characters have a similar u path that produces a correct segment. The second group of character is detectable by their height or loop. Therefore, the false segment is recognized and connected to its right neighbor. Using dots and their information such as position, number, etc will be useful, too [15]. It is worth mentioning that this segmentation algorithm is not sensitive to slant and overlapping characters.

3 Using Extra Classes

In this section the idea of using extra classes is introduced. After segmentation of learning texts, we grouped segmented images –characters, combination of characters and logical parts of a character- in several classes. 344 classes were obtained. 124 classes were characters depending on their positions in the word and some signs used in document –point, comma, semicolon, etc. We name these classes "necessary classes".220 classes were unnecessary classes - combination of

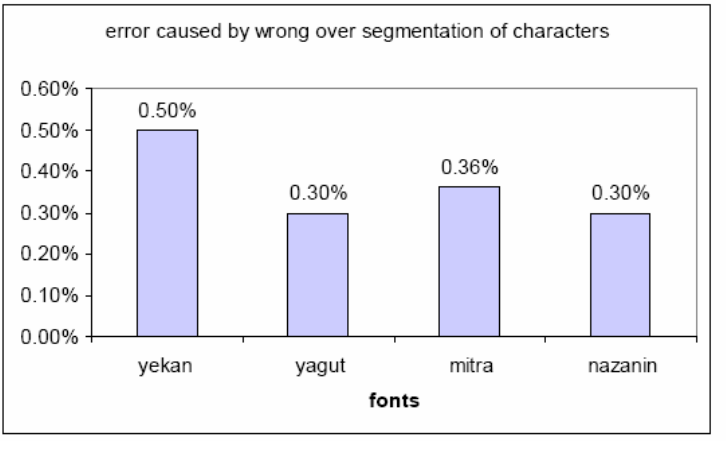


Fig. 6. Error caused by over segmentation of characters (%)

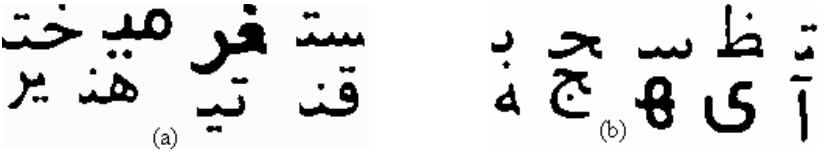


Fig. 7. Examples for (a) unnecessary and (b) necessary classes

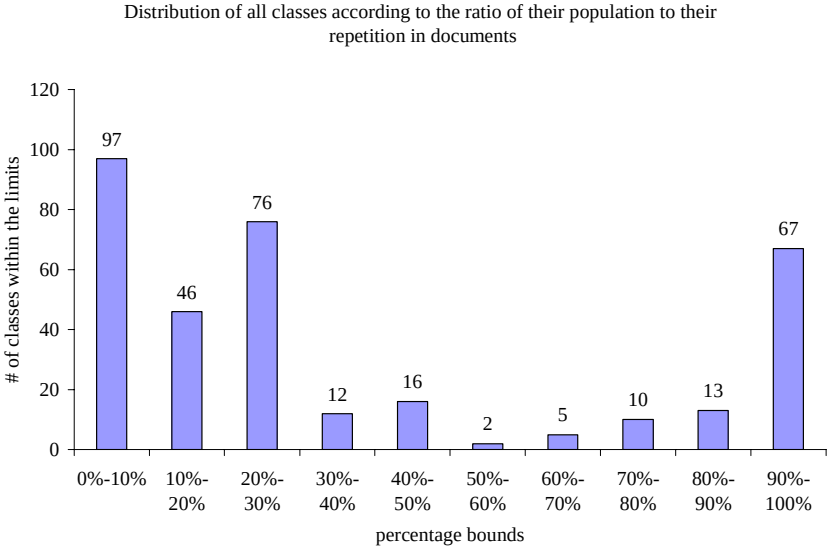


Fig. 8. Plots of class distribution

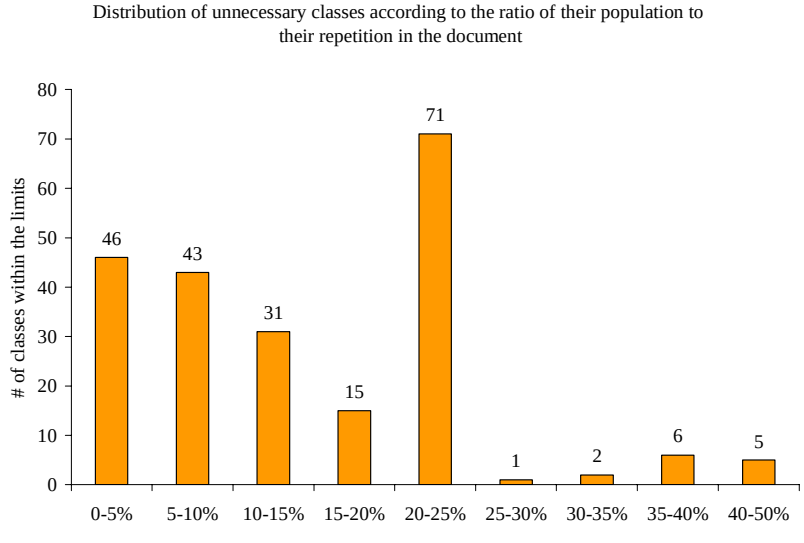


Fig. 9. Plots of unnecessary classes

some characters and a part of a character. If we use these 344 classes our result will be 99.64%. The error is for over segmentation of characters (Fig.6). Some of these classes are shown in Fig. 7.

According to the mean value of population of extra classes (about 108), we can eliminate some unnecessary classes with population less than 3. With this modification the number of total classes will reduce to 178. We will show the improvement procedure in segmentation results by using extra classes in the next section.

4 Experimental Results

The segmentation algorithm was tested on a set of printed texts in 4 different mentioned fonts. The test set includes 17920 characters. The training samples are not included in the test set. The errors caused by wrong over segmentation of characters are shown in Fig. 10.

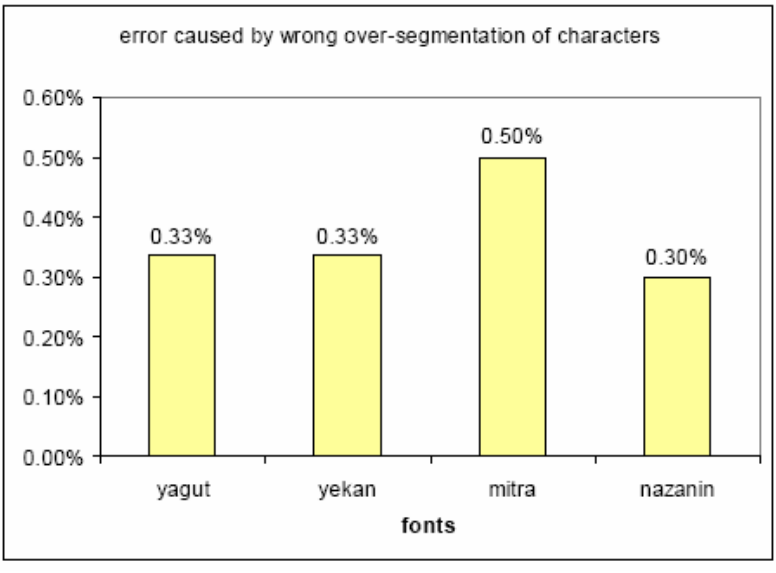


Fig. 10. Error caused by over segmentation of characters (%)

The plot in Fig. 11 shows the improvement of segmentation results by using more classes in recognition stage. As it is shown, using 124 necessary classes, the segmentation result, will be 96.37%. When necessary classes and unnecessary classes with population more than 3 are used -160 classes- the result will improve to 97.93%. Using 178 classes including necessary classes and unnecessary ones with population more than 2, 98.24% of characters will be considered to be segmented correctly. The result will be 98.73%, if 216 classes – necessary and unnecessary classes with more than one member- are used. Finally with all necessary and unnecessary classes the 99.64% will be achieved. As mentioned above, the error is due to wrong over segmentation shown in Fig. 10.

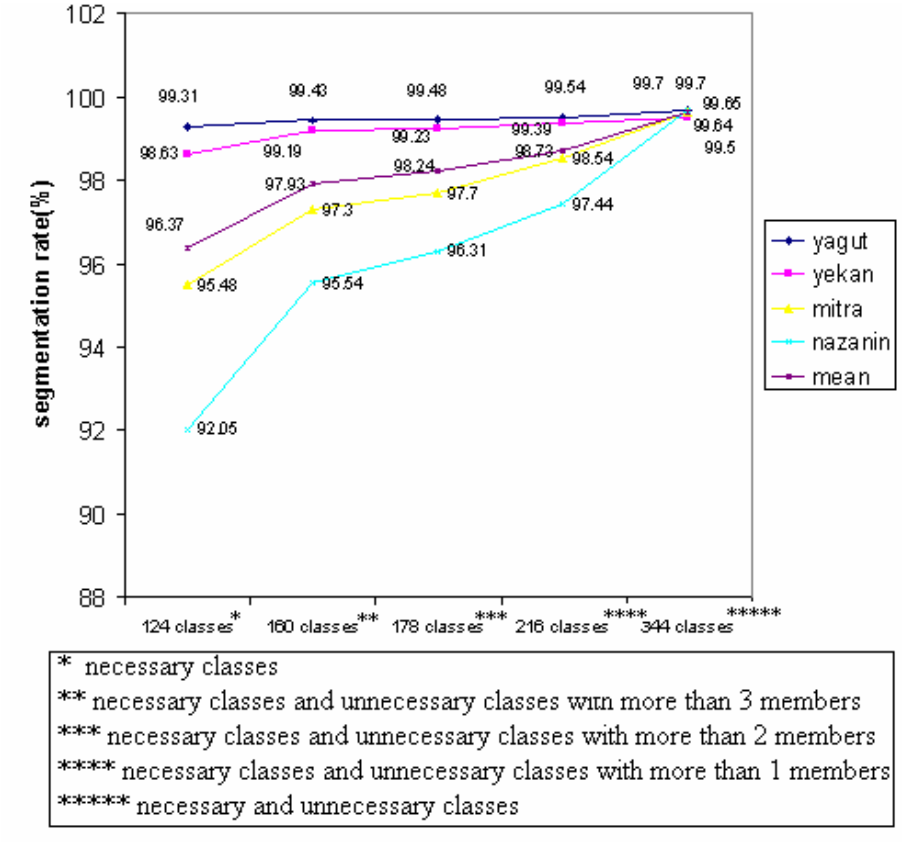


Fig. 11. Advantages in results by using extra classes

5 Conclusion

In this paper, a new method to improve multi font Farsi/Arabic text segmentation results was presented. Finding a best way to get a good result for segmentation of multi font Farsi/Arabic texts, is difficult. Different characteristics of each font are the reason. Here we proposed an idea of having some extra classes in recognition stage. The extra classes will be some parts of characters or combination of 2 or more characters. These extra classes will be determined statistically .Our segmentation algorithm -for multi font Farsi/Arabic texts- is based on the conditional labeling of the up contour and down contour. A pre-processing technique is used to adjust the local base line for each subword. This algorithm uses adaptive base line for each subword to improve the segmentation results. This segmentation algorithm uses up contour and down contour curvature, too. We have used a learn document of 4820 characters for 4 fonts (19280 characters). The test set had 17920 characters and was separate from learn one. Segmentation result improved from 96.7% to 99.64% when all extra classes are used. Using 178 classes, the segmentation result, will be 98.24%, which seems to

be suitable to be chosen as the number of classes in recognition stage comparing with 124 classes in ordinary systems. It is clear that if this idea can be implemented on a real and complete Farsi/Arabic database, the OCR results will improve considerably.

References

1. M. Omidyeganeh, K. Nayebi, R. Azmi, A. Javadtalab, " A New Segmentation Technique for Multi Font Farsi/Arabic Texts", Acoustics, Speech, and Signal Processing, 2005. Proceedings. (ICASSP '05). IEEE International Conference on Volume 2, (2005), pp. 757 - 760
2. H. Al-Muallim, S. Yamaguchi, "A method of recognition of Arabic Cursive Handwriting", IEEE Trans. Pattern Anal. Mach. Intell. PAMI – 9, (1987),pp 715-722.
3. A. Amin, "Off-line Arabic character recognition: the state of the art". Pattern Recognition 31, (1998), pp. 517-530.
4. A. Amin, G. Masini, "Machine Recognition of Multifont printed Arabic Texts", Proc. 8th Int. Conf. on Pattern Recognition, Paris, (1986), pp 392-295.
5. R. Azmi, E. Kabir, "A New Segmentation Technique for Omnifont Farsi Text", Pattern Recognition Letters 22, (2001), pp. 97-104.
6. R. Azmi, "Recognition of omnifont printed Farsi text". Ph.D. Thesis, Tarbiat Modarres University, Tehran, (1999).
7. R. Azmi, E. Kabir, "A recognition algorithm for hand printed Farsi characters". Proceedings of the International Conference on Telecommunication, ICT '96, Istanbul, (1996), pp. 852-855.
8. T.S. El-Sheikh, R.M. Guindi, "Computer recognition of Arabic cursive scripts". Pattern Recognition 21, (1988), pp. 293-302.
9. J.J Hull, S.N. Srihari, "A computational approach to visual word recognition: hypothesis generation and testing", Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, CVPR'86, Washington, DC, (1986), pp. 156-161, 1986.
10. B.M. Kurdy, A. Joukhadar, "Multi font recognition system for Arabic characters". Proceedings of the Third International Conference and Exhibition on Multi-Lingual Computing, Durham, (1992), pp. 731-739.
11. Y. Lu, M. Shridhar, "Character segmentation in handwritten words - an overview". Pattern Recognition 29, (1996), pp. 77-96.
12. K. Massruri, E. Kabir, "Recognition of hand-printed Farsi characters by a Fuzzy classifier". Proceedings of the Second Asian Conference Computer Vision, ACCV '95, Singapore, Vol. 2, (1995), pp. 607-610.
13. S. Mori, C.Y. Suen, K. Yamamoto, "Historical review of OCR research and development". Proc. IEEE 80, (1992), pp. 1029-1058.
14. B. Parhami and M. Taraghi, "Automatic recognition of printed Farsi text", Pattern Recognition 14, (1981), pp. 395-403.
15. M. Altuwaijri, M. Bayoumi, "Arabic text recognition using Neural Networks", Proc. Int. Symp. on Circuits and Systems – ISCAS, (1994), pp. 415 – 418.

Modeling Television Schedules for Television Stream Structuring

Jean-Philippe Poli[1,2] and Jean Carrive[1]

[1] Institut National de l'Audiovisuel,
4 Avenue de l'Europe,
94366 Bry sur Marne Cedex, France
{jppoli,jcarrive}@ina.fr
www.ina.fr
[2] Université Paul Cézanne, LSIS (UMR CNRS 6168),
Avenue Escadrille Normandie-Niemen,
13397 Marseille Cedex, France
www.lsis.org

Abstract. TV stream structuring is very important for huge archives holders like the French National Institute, the BBC or the RAI, because it is the first necessary step to describe the various telecasts broadcast on various channels. It is also necessary for television ratings in order to isolate the telecasts that must be rated.

One can think this structuring is a simple alignment of the TV guides on the stream itself. But TV guides present in average only 35% of the telecasts that represent in average less than 18 hours by day.

We propose in this article a method to predict TV schedules by modeling the past ones in order to boil down the television stream structuring problem to a simple alignment problem.

1 Introduction

The French National Audiovisual Institute is in charge of the legal deposit of the television. In order to provide an efficient way to consult its huge archives, it is used to describing manually both the structure and the content of French TV channels' broadcasts. So does Médiamétrie, which provides in France the television ratings of every channels. For many years, the TV streams are digitally acquired. That allows automating many kinds of treatments like video indexing, transcription or restoration. Our work consists in an automation of the TV stream structuring.

The video indexing community interests in structuring or summarizing videos by shot or scene detecting, by determining their genres and by extracting objects from it [1]. Shots and scenes of a television stream are not useful because they are too numerous. It would also be hard to merge them into telecasts. Video structuring and indexing methods cannot be applied to television stream structuring because of the huge computations. Video structuring is often based on video and audio features extraction[2,3]. The features are then integrated in

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 680–689, 2007.
© Springer-Verlag Berlin Heidelberg 2007

order to find *homogeneous semantic events*. More other, these methods depend on the genre of the video (for instance, the various scenes of a tennis video). Finally, genre recognition is also based on features extraction [4] that would be costly if applied on a TV stream that lasts at least 24 hours. Genre recognition is useful to determine whether a movie is a comedy, a drama or an horror film like in [5]. But there is no method that works with more than 3 genres. Hence the leading edge methods from the video indexing community cannot be easily applied to TV stream structuring.

Television stream structuring could be seen as a simple alignment problem because TV guides provide a structure for it. But all the TV guides are incomplete and imprecise views of the stream. We compared 3 months of TV guides for a channel with its real structure: telecasts presented in TV guides represent only 35% of the telecasts that have really been broadcast. Small magazines with sponsorship, weather forecast, lotteries, inter-programs (advertisings, coming next and previews), races results and pronostics are not announced. Even if these telecasts have short durations, they represent more than 2 hours by day in average. Hence, it is not possible to realign TV guides on TV streams. Since existing methods are not applicable, we propose to model TV schedules in order to statistically improve program guides. We call a TV schedule one day of TV broadcasts. The goal is that each telecast supposed to be broadcast appears in the improved program guide. Then an alignment of the improved guide can be performed on the stream. We consider 36 different genres of telecasts. A broadcast day is composed by 120 telecasts in average. There are hence $36^{120} \approx 5.7 \times 10^{186}$ possible schedules. The TV schedules modeling will decrease the number of possible schedules by deleting impossible successions (for instance a day composed by 120 telecasts of the same genre). The next step is to combine the predictions of the model and the program guides to revise the predictions. These improved guides will be used to guide detectors (e.g. audio and video jingles detection, logo recognition) providing the genre that must be detected and a temporal window within the telecast transition occurs.

In order to model the TV schedules, we have introduced an extension of Markov models which probabilities are context-dependent. The durations of the various telecasts are regressed by a regression tree. We present in the first section of this paper our Contextual Hidden Markov Model and then how we apply it to TV schedules modeling. We then present our use of regression trees to predict durations. We discuss how we compute improved program guides from existing program guides and from the statistical model. Finally, we present some results before concluding.

2 Modeling TV Schedules with Contextual Hidden Markov Model

In order to predict TV schedules, we need a statistical model to represent the past schedules. Markov models are very useful to represent sequences of observations. They have already been used for video contents modeling[6,7,8]. We show in the next section that classical Markov models are not suitable for this modeling.

2.1 Inadequacy of Classical Markov Models

Hidden Markov models (HMM) are defined by [9] a state-space S, a set of observable symbols Σ, a stochastic vector π that represents the probability for each state to start the sequence of observations, a matrix A that represents the transition probabilities from a state to another and a matrix B that represents for each state the probability to observe each symbol of Σ. Let M be a HMM, $O = O_1, ..., O_n$ a sequence of observations and $s_1, ..., s_n$ a state sequence; then the probability of O can be easily written:

$$P(O, s_1, ..., s_n|M) = P(s_1)P(O_1|s_1)\prod_{i=2}^{n} P(s_i|s_{i-1})P(O_i|s_i). \quad (1)$$

We can take, for example, a channel which broadcasts 3 news by day. The first one at 6 a.m., is followed by cartoons. The second one at 1 p.m. is followed by a soap opera. Finally the last one at 8 p.m. is followed by a movie. Let the state-space S of M be the set of telecasts genres. Then it is not possible to determine easily which telecast will follow the news and it will be necessary to test the 3 genres cartoons, soap opera and movie. To address this ambiguity problem, the author of [9] proposes to multiply the states. In our case, we need to consider morning news, noon news and evening news instead of a single state for news. But it cannot be done for inter-programs or for the short magazines with sponsorship because the number of occurrences is not constant through the days of the week. Another problem is met with inter-programs. They will be represented by a state that every other state will point to, and that will point to every other state. Thus, the most probable genre following an inter-program will be in fact the genre that has the most occurrences on a day.

These problems can be bypassed with a contextualization of the HMM probabilities. We propose in the next section an extension of classical HMM we called Contextual Hidden Markov Models.

2.2 Definition of Contextual Hidden Markov Models

Definition 1 (Context). *A context θ is a set of variables $x_1, ..., x_n$ with values in continuous or discrete domains, respectively $\{D_1, ..., D_n\}$. An instance θ_i of this context is an instantiation of each variables x_i:*

$$\forall\, i \in \{1, ..., n\}, x_i = v_i \text{ with } v_i \in D_i. \quad (2)$$

From this point, we also call θ_i a context.

Example 1 (Example of context). For the representation of television schedules, the context θ for our model can be a variable *Time* which represents beginning time of a telecast by an integer in the range $\{0, ..., 86399\}$, and a variable *Day* which represents the broadcast day of week with an integer in the range $\{0, ..., 6\}$:

$$\theta = \{Time, Day\} \text{ and } D_{Time} = \{0, ..., 86399\}, D_{Day} = \{0, ..., 6\}.$$

It is possible to update a context θ_i into a context θ_{i+1} with an evolution function.

Definition 2 (Evolution function). *Let Θ be the set of all possible instances of a context θ. An evolution function F for θ is defined by:*

$$\begin{array}{l} F : \Theta \times D_{p_1} \times ... \times D_{p_m} \rightarrow \Theta \\ \theta_i, p_1, ..., p_n \rightarrow \theta_{i+1} \end{array} \tag{3}$$

where D_{p_i} is the domain of the external parameter p_i.

Example 2 (Example of evolution function). Let $\mathbb{D}$ be the set of all possible durations and l a particular duration. In the case of the television stream structuring, if we consider the context θ defined in example 1, the evolution function F we want to use is defined by:

$$F : \Theta \times \mathbb{D} \rightarrow \Theta$$
$$\left\{ \begin{array}{l} Time = H \\ Day = D \end{array} \right\} \rightarrow \left\{ \begin{array}{l} Time = (L + H) \mod 86400 \\ Day = \left(D + \lfloor \frac{L+H}{86400} \rfloor\right) \mod 7 \end{array} \right\}.$$

We introduce now Contextual Hidden Markov Models (CHMM) which are basically a Markov model where the probabilities are not only depending on the previous state but also on a context. This context is updated every time a state of the model is reached.

Definition 3 (Contextual hidden Markov models). *A contextual hidden Markov model is totally defined by the 7-uplet $< S, \Sigma, \Theta, F, \pi_\theta, A_\theta, B_\theta >$, where:*

- *S is a state space with n items and s_i denotes the i^{th} state in the state sequence,*
- *Σ is an alphabet with m items and ϵ_j denotes the j^{th} observed symbol,*
- *Θ is the set of all instances of the context θ,*
- *F denotes the evolution function for instances of θ,*
- *π_θ is a parametrized stochastic vector and its i^{th} coordinate represents the probability that the state sequence begins with the state i:*

$$\forall\, \theta \in \Theta, \sum_{i=1}^{n} \pi_i(\theta) = 1. \tag{4}$$

π_i is a function of θ which represents the initial distribution in the context θ :

$$\forall i \in \{1, ..., n\}, \pi_i(\theta_1) = P(s_1 = i|\theta_1), \tag{5}$$

- *A is a stochastic matrix $n \times n$ where a_{ij} stands for the probability that the state i is followed by state j in the state sequence. Each a_{ij} is a function of θ:*

$$\forall \theta \in \Theta, \forall i \in \{1, ..., n\}, \sum_{j=1}^{n} a_{ij}(\theta) = 1. \tag{6}$$

$$\forall k, t \in \mathbb{N}, \forall i, j \in \{1, ..., n\}, \;\; a_{ij}(\theta_k) = P(s_{t+1} = j|s_t, \theta_k), \tag{7}$$

- *B is a stochastic matrix $n \times m$ where b_{ik} represents the probability of observing the symbol k from state i:*

$$\forall \theta \in \Theta, \forall i \in \{1, ..., n\}, \sum_{k=1}^{m} b_{ik}(\theta) = 1. \tag{8}$$

$$\forall k, t \in \mathbb{N}, \forall i \in \{1, ..., n\}, \forall j \in \{1, ..., m\}, \; b_{ij}(\theta_k) = P(\epsilon_t = j|s_t, \theta_k). \tag{9}$$

For this extension of the classical Markov models, the Markovian assumptions and properties must be updated.

Property 1 (Contextual Markovian Assumptions). Let $T \in \mathbb{N}$ be the length of the sequence of the observed symbols. The contextual Markovian assumptions are:

$$\begin{array}{l} P(s_t|s_1,...,s_{t-1},\theta_1,...,\theta_t,...,\theta_T) = P(s_t|s_{t-1},\theta_t) \\ P(s_t,\epsilon_t|s_1,...,s_t,\epsilon_1,...,\epsilon_{t-1},\theta_1,...,\theta_t) = P(s_t,\epsilon_t|s_t,\theta_t). \end{array} \tag{10}$$

In other words, probabilities in a contextual semi-Markov model depend only on the current context (not the previous or following ones). The observed symbols are all independent and transition probabilities depend only on the previous state.

Property 2. Let $\Lambda =< S, \Sigma, \Theta, F, \pi_\theta, A_\theta, B_\theta >$ be an instance of a contextual semi-Markov model. Let O be a sequence of symbols such as $O = O_1, ..., O_T$. Let θ_1 be the initial context. Then the probability of observing O is:

$$\begin{array}{rl} P(O|\Lambda) &= P(s_1,...,s_T,\epsilon_1,...,\epsilon_T) \\ &= P(s_1|\theta_1)P(\epsilon_1|s_1,\theta_1)\prod_{i=2}^{T} P(s_i|s_{i-1},\theta_i)P(\epsilon_i|s_i,\theta_i) \\ &= \pi_{s_1}(\theta_1)b_{s_1\epsilon_1}(\theta_1)\prod_{i=2}^{T} a_{s_{i-1}s_i}(\theta_i)b_{s_i\epsilon_i}(\theta_i). \end{array} \tag{11}$$

The context permits to resolve certain ambiguities in the transitions and eliminates impossible transitions. We can expand the context to seasons and vacations to be closer to the reality. But presently, we only regard broadcast times and days.

In order to represent the TV schedules, we chose to attribute at each state of the CHMM a telecast genre. We chose a continuous distribution for the emission probabilities : this means that observations are not discrete in our case. When we are on a state of our CHMM, for example the state representing magazines, we have a continuous distribution over its possible durations.

Example 3. Let <Monday, 6:30, Magazine, 10 minutes> denote a magazine that starts on Monday at 6:30 a.m. and that lasts 10 minutes. Let M be a CHMM. Then, the probability of the schedule S = <Monday, 6:30, Magazine, 10 minutes>, <Monday, 6:40, IP (inter-programs), 3 minutes>, <Monday, 6:43, News, 20 minutes> can be written:

$$\begin{array}{rl} P(S|M) = & P(magazine|\theta_1) \times P(d = 600s|magazine,\theta_1) \\ & \times P(IP|\theta_2, magazine) \times P(d = 180s|IP,\theta_2) \\ & \times P(news|\theta_3, IP) \times P(d = 1200s|news,\theta_3) \end{array} \tag{12}$$

where $\theta_1 = \{monday, 23400\}$, $\theta_2 = \{monday, 24000\}$, $\theta_3 = \{monday, 24180\}$.

As shown in the example 3, it is necessary to estimate the probability of a particular duration. We present in the next section our method to predict durations of a particular telecast.

2.3 Durations Regression

Regression trees. We discuss about two close concepts: decision and regression trees [10]. They are tools for predicting continuous variables or categorical

variables from a set of mixed continuous or categorical factor effects. The principles of decision trees and regression trees are the same except that regression trees are used to predict continuous values from one or more predictor variables. Their prediction are based on few logical *if-then* conditions. A regression tree is a tree where each decision node in it contains a test on some predictor variables' value. The leaves of the tree contain the predicted forecast values. There are many kinds of leaves: generally, they contain a mean value and a standard deviation. But sometimes they can contain an interval I and a function f: f is then a local regression on I of the input variable.

Regression trees are built through a recursive partitioning. This iterative process consists in splitting the data into partitions (generally two partitions), and then splitting them up further on each of the branches. Categorical predictors are easy to use because the partitioning can be done regarding their different values. For continuous predictors, the algorithm must choose a particular value to split the data into two sets. The chosen test is the one which satisfies a user-defined criteria.

Application to television schedules modeling. We use a regression tree in order to resolve two different problems. Firstly, we use it to predict a range of durations for a telecast from its context (i.e. broadcast days and times, previous telecast). It is very useful to know that between the minimum duration and the maximum duration a telecast transition may occur in order to only look for it in this temporal window. But this problem is directly resolved by regression trees. Secondly we want to deduce a probability from a leaf of the regression tree. We represent the distribution of the durations on a leaf with the asymmetric gaussian presented in [11]. Let μ and σ be respectively the mean value and $|Min(Duration) - Max(Duration)|$. Then the probability of a given duration d is given by:

$$A(d, \mu, \sigma^2, r) = \frac{2}{\sqrt{2\pi}} \frac{1}{\sigma(r+1)} \begin{cases} e^{\left(-\frac{(d-\mu)^2}{2\sigma^2}\right)} \text{ if d} > \mu \\ e^{\left(-\frac{(d-\mu)^2}{2r^2\sigma^2}\right)} \text{ otherwise} \end{cases} \tag{13}$$

where $r = \frac{|\mu - \min(Duration)|}{|\mu - \max(Duration)|}$.

2.4 Training the Model

Each training example of the INA's database gives the start time and day, the duration and the genre of the telecast.The training of this model is a two phases process.

Phase 1. The first step consists in building the regression tree. Classical leaves of regression trees contain generally a mean value and a standard deviation. Many papers provide criteria to pursue the building of the tree. For example, [12] proposes to maximize the expected error reduction. Let T be the set of training examples, T_i be the subset of examples that have the i^{th} outcome of the

potential test. [12] considers the standard deviation $sd(T_i)$ of the target values in T_i as a measure of error; the expected error reduction can thus be expressed by:

$$\Delta error = sd(T) - \sum_i \frac{|T_i|}{|T|} \times sd(T_i). \tag{14}$$

In our case, considering the standard deviation will cause bad predictions of minimal and maximal durations. Instead of considering the standard deviation in the expected error computation, we can use the distance between the greatest duration and the lowest (that can be referred to as width of the subset). Then the expected error is:

$$\Delta error = |\max(T) - \min(T)| - \sum_i \frac{|T_i|}{|T|} \times |\max(T_i) - \min(T_i)|. \tag{15}$$

This criterion can cause overfitting of the learning data: the regression tree will predict perfectly the learning durations but it will not be efficient for new examples. To avoid overfitting, we stop the building of a branche when the width of its subset is lower than a threshold ω. We can also impose a minimum value υ for a leaf.

Phase 2. The second phase is the evaluation of the probabilities of the CHMM. Since the computation of the emission probabilities (matrix B) is performed by the regression tree, the training of the model boils down to a simple Markov chain training. The problem of the contextualization of the probabilities is that during learning, every context must be represented. This needs a huge number of training examples, that grows proportionally to the number of possible contexts.

Hence, for each telecast, we predict its range of durations with the regression tree. This permits to have several contexts from a unique example. The computation of the probabilities in a context θ is simply the frequency of the genre in the context θ.

3 Combining Program Guides and Model'S Predictions

In the previous section, we have introduced a model that can represent past TV schedules. But the TV guides, which are delivered at least one week before the broadcast, can be seen as a revision of the schedule. We can differentiate three cases:

- the program guide is included in schedules predicted by the model: there is no need to revise the schedule,
- the program guide is in contradiction with the predicted schedules: they need to be combines,
- the program guide does not correspond with what has been broadcast (a special and unforeseeable event occurs): there is nothing to do, the structuring will not work.

The difficulty of combining both the predictions and the program guide is the telecast matching. A telecast that appears in the prediction must fit a telecast in

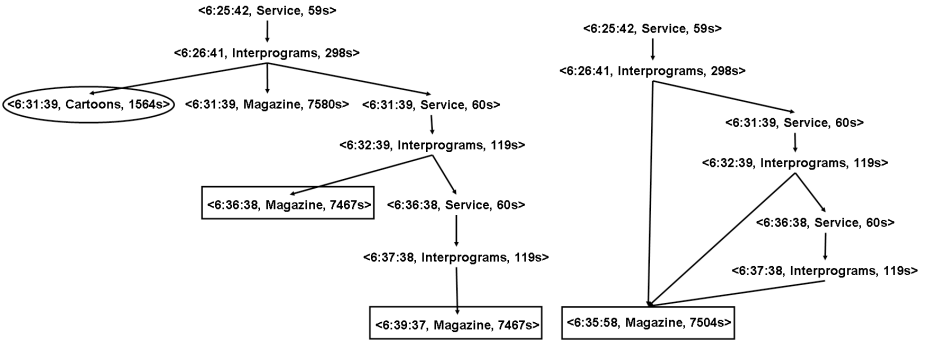


Fig. 1. Left : an example of prediction tree for a program guide which stops with a magazine. <6:31:39, Magazine, 7580s> represents a magazine that starts at 6:31:39 a.m. and that lasts 7580 seconds. Right : the same tree after the application of heuristics.

the program guide while do not have the same duration and the same start time. To perform this matching, we use an elastic partial matching method [13]. The proposed algorithm resolves the best matching subsequence problem by finding a cheapest path in a directed acyclic graph. It can also be used to compute the optimal scale and translation of time series values. The algorithm needs a distance to compare the values; in their case, they use the euclidean distance between two real values. We have used the following measure d between two telecasts E_1 and E_2:

$$d(E_1, E_2) = \begin{cases} \infty \text{ if } \mathrm{E}_1 \text{ and } \mathrm{E}_2 \text{ have the same genre} \\ |E_1.Start - E_2.Start| + |E_1.Duration - E_2.Duration| \text{ otherwise.} \end{cases} \tag{16}$$

In order to make the combination, we consider that the first telecast of both the program guide and the prediction are synchronized with the real start time of the telecast. The method consists then in predicting telecasts from a telecast of the program guide to the next one. If we consider the predicted schedules as a graph, it maps with browsing the graph in depth-first order until a telecast matches with the next telecast of the program guide. We introduced a threshold Δ which specifies the maximal delay between a telecast from the prediction and a telecast from the program genre. If the algorithm passes this delay, we consider a matching telecast will not be found. We then add the unmatched telecast from the program guide to the graph of predictions and the CHMM is reinitialized with the new context. This algorithm selects the possible paths in the prediction tree regarding the program guide. In order to decrease the combinatory aspect of the algorithm, heuristics can be used.

Heuristic 1: Pruning the impossible branches. We made a list of telecast genres that must appear in a program guide. For example, movies and TV shows always appear in a program guide, contrary to weather forecast, short magazines which can be omitted. If a path between two successive telecasts in the program guide

passes by a telecast which genre always appears in program guides, then the path must be pruned.

Heuristic 2: Merging matching telecasts. Several paths can lead from one telecast of the program guide to following one. Thus, there are several matching telecasts which differ from start times and sometimes from durations. However, they represent the same one and they will all be developed. We can merge all these matching nodes in order to have only one node to browse. The next section presents some results.

4 Experiments

In order to test the model, we trained it on telecasts broadcast on France 2 in 2004 (it represents more than 50000 telecasts) and we test the model on one week (because we needed the program guides) in 2005.

The regression must be efficient because it is necessary for the CHMM learning phase. We fixed $\omega = \upsilon = 300$. That means the minimum width of a temporal window is 300 seconds. We have 97% of good predictions. Good predictions are durations that are between the minimum and maximum values given by the leaf of the regression tree.

With the CHMM, it is possible to represent 83% of the days in 2005. The others present special events.

We fixed $\Delta = 1800$, i.e. a delay of 30 minutes is authorized. The improvement of 7 schedules from a program guide gives from 3 to 6 possible schedules. Only one of them is correct if we compare them to the ground truth. With all heuristics, when at least one path exists between two consecutive telecasts, only few nanoseconds are necessary. Otherwise, if there is no path and if a telecast from program guide must be added, it takes up to 20 seconds in average. For the prediction of a TV schedule, it takes less than 2 minutes in average.

Results could be ameliorated by cleaning up the training and the testing sets. In fact, special events like the Pope's death and Olympic Games have not been removed and change certain probabilities.

5 Conclusion

We present in this article an original approach for structuring TV streams. This approach is based on knowledge about TV schedules obtained by combining both past schedules and program guides. The program guides permit to revise the predictions that can be made with the statistical model.

In order to model the TV schedules, a new extension of Hidden Markov Models has been introduced, called Contextual Hidden Markov Models. Regression trees are used to complete CHMM by computing the durations' probabilities with an asymmetric gaussian. The results we obtained are totally satisfying but they can surely be improved.

The improvement of program guides is a first step of an automatic TV stream structuring system. The next step of our work is to guide detectors (e.g. jingle

detectors, advertisings detectors...) in function of the improved schedules. Maybe it will be necessary to revise again the transitions possibilities with the detectors' outputs.

Another improvement of the current work could be to take scholar vacations and summer vacations into account in the context.

References

1. Snoek, C.G., Worring, M.: Multimodal video indexing: A review of the state-of-the-art. Multimedia Tools and Applications **25**(1) (2005) 5–35
2. Kijak, E., Oisel, L., Gros, P.: Audiovisual integration for tennis broadcast structuring. In: International Workshop on (CBMI'03). (2003)
3. Gatica-Perez, D., Sun, M., Loui, A.: Probabilistic home video structuring: Feature selection and performance evaluation. In: Proc. IEEE Int. Conf. on Image Processing (ICIP). (2002)
4. Roach, M., Mason, J., Pawlewski, M.: Video genre classification using dynamics. In: IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2001. Volume 3. (2001) 1557–1560
5. Rasheed, Z., Shah, M.: Movie genre classification by exploiting audio-visual features of previews. In: Proceedings 16th International Conference on Pattern Recognition. Volume 2. (2002) 1086–1089
6. Kijak, E., Oisel, L., Gros, P.: Hierarchical structure analysis of sport videos using hmms. In: IEEE Int. Conf. on Image Processing, ICIP'03. Volume 2., IEEE Press (2003) 1025–1028
7. Duong, T.V., Bui, H.H., Phung, D.Q., Venkatesh, S.: Activity recognition and abnormality detection with the switching hidden semi-markov model. In: CVPR '05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1, Washington, DC, USA, IEEE Computer Society (2005) 838–845
8. Huang, J., Liu, Z., Wang, Y.: Joint scene classification and segmentation based on hidden markov model. Multimedia, IEEE Transactions on **7**(3) (2005) 538–550
9. Norris, J.: Markov chains. Cambridge series in statistical and probabilistic Mathematics (1997)
10. Breiman, L., Friedman, J., Olshen, R., Stone, C.: Classification and regression trees. Technical report, Wadsworth International, Monterey, CA, USA (1984)
11. Kato, T., Omachi, S., Aso, H.: Asymmetric gaussian and its application to pattern recognition. In: Lecture Notes in Computer Science (Joint IAPR International Workshops SSPR 2002 and SPR 2002). Volume 2396. (2002) 405–413
12. Quinlan, J.R.: Learning with continuous classes. In: Proceedings of 5th Australian Joint Conference on Artificial Intelligence. (1992) 343–348
13. Latecki, L.J., Megalooikonomou, V., Wang, Q., Lakaemper, R., Ratanamahatana, C.A., Keogh, E.: Partial elastic matching of time series. icdm **0** (2005) 701–704

Automatic Generation of Multimedia Tour Guide from Local Blogs

Hiroshi Kori, Shun Hattori, Taro Tezuka, and Katsumi Tanaka

Graduate School of Informatics, Kyoto University
Yoshida-Honmachi, Sakyo, Kyoto, 606-8501, Japan
{kori,hattori,tezuka,tanaka}@dl.kuis.kyoto-u.ac.jp

Abstract. It has recently become a common practice for people to post their sightseeing experiences on weblogs (blogs). Their blog entries often contain valuable information for potential tourists, who can learn about various aspects not found on the official websites of sightseeing spots. Bloggers provide images, videos and texts regarding the places they visited. This implies that popular travel routes could be extracted according to the information available in blogs. In this paper, we describe a system that extracts typical visitor's travel routes based on blog entries and that presents multimedia content relevant to those routes. Typical travel routes are extracted by using a sequential pattern mining method. We also introduce a new user interface for presenting multimedia content along the route in a proactive manner. The system works as an automatically generated tour guide accessible from a PC or a mobile device.

1 Introduction

Tourists planning their trips usually rely on a guidebook or Web sites to help them choose destinations and travel routes, but guidebooks often do not list the newest topics and official Web sites provide much less information than is available on the Web. In this paper, we propose a system that extracts typical travel routes from local blog entries and presents relevant multimedia content along these routes in a proactive manner.

To develop such a system, we crawled through local blogs and inferred the blogger's activity by focusing on sentences containing the place names in the blog text. We obtain typical travel routes by using a sequential pattern mining method, PrefixSpan, and we extract keywords that indicate the context of bloggers' movements along a typical route. In our system, the user can search for typical routes by specifying a starting point, end point, or context keyword.

Our system also presents multimedia content in a proactive manner. Once the user selects one of the typical travel routes, the system presents relevant images and texts collected from blog entries as a multimedia tour guide. The user can thus learn about the region almost effortlessly.

The rest of the paper is organized as follows. Section 2 discusses related work. We describe a formal model for a multimedia tour guide system in Section 3. Section 4 explains the method we use to extract a typical route and its context, and Section 5 explains how to generate a multimedia tour guide based on extracted routes. Section 6 shows examples of extracted routes and contents and evaluates the result. Section 7 concludes the paper by briefly summarizing it.

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 690–699, 2007.
© Springer-Verlag Berlin Heidelberg 2007

2 Related Work

2.1 PrefixSpan

There are various data mining methods that extract frequent combination of items from a database [1]. There are also methods that extract patterns with "order" between items [2,3,4,5]. The system we present in this paper extracts sequences of place names by using PrefixSpan, which is a fast mining method for extracting sequential patterns [5]. We regard a typical travel route as a major route for visitors. Visitors' routes are expressed by sequences of place names. We extract frequent sequential patterns as major routes for visitors by sequential pattern mining.

2.2 Local Blogs

There are various services that unify blogs with geographical information [6,7]. The user can search the blog entries that are mapped in a specific area. The user can search the location-specific blog entries provided by these systems, but those systems do not provide any other information. The user cannot learn anything more than what is those blog entries. On the other hand, there has been some research on spatial blog mining. Kurashima et al.[8] extract experiences from blogs based on time, space, action and object by association rule. They attempt to construct summary and search function.

2.3 Passive Interface of Local Contents

Our system's function includes passive interface of local contents. Tezuka et al. [9] introduce a passive browser of geographical Web pages based on landmark mining from Web. This system shows Web pages, which are usually contents generated in information provider. In contrast, Local blogs are contents generated by consumers. They are useful for visitors (consumers) because the contents include the same viewpoint for consumers. Our system shows such consumer-generated contents. Some researches introduce the method to generate a tour guide. Schilling and Zipf developed a system that generates 3D tour animation using a VRML city model [10]. Their system has a function to present Web pages relevant to each building along the tour route. However, Web pages are not dynamically mapped to buildings, and contexts of the user traveling the route are not considered. They have not performed information extraction from Web pages either.

3 Model of Multimedia Tour Guide System

In this section, we describe a formal model for a multimedia tour guide system. The characteristic of our model is that contents for the multimedia tour guide are collected based on a route selected by the user and its context. In Subsection 3.1, we model the tour guide system. In Subsection 3.2, we model contexts for routes.

3.1 Multimedia Tour Guide Model

In this subsection, we formulate elements of a multimedia tour guide system. A route r is formulated as sequences of place names p_i. R is a set of routes.

$$r := < p_1, p_2, ... >, \qquad R := \{r_1, r_2, ...\} \tag{1}$$

A sequential pattern α is a sequence of items, in this case place names. If all items in a sequential pattern α are contained in another sequential pattern β, with the order between items being preserved, We express the relationship as $\alpha \sqsubseteq \beta$. When the user selects a sequential pattern ξ, a set of relevant blogs $B(\xi)$ is defined as follows. In the formula, $r(b)$ represents a route contained in a blog b. B is a set of blog entries which contain any of the routes.

$$B(\xi) := \{b \in B \mid \xi \sqsubseteq r(b)\} \tag{2}$$

When the user selects a sequence ξ, a set of typical travel routes $R(\xi)$ which is adjacent to the user selected route ξ is formulated as follows.

$$R(\xi) := \{r(b) \mid b \in B(\xi)\} \tag{3}$$

A set of blogs $B'(\xi_j)$ relevant to the user specified route $R(\xi_j)$ is defined as follows. In the definition, the function $incl(b, x)$ indicates whether or not an object x is included in the blog b. Context X_j for a sequence ξ_j is formulated as follows. The *context* is discussed in Subsection 3.2.

$$B'(\xi_j) := \{b \mid incl(b, x) \wedge \xi_j \sqsubseteq r(b) \wedge x \in X_j\}, \;\; X_j := context(R(\xi_j)) \tag{4}$$

The function $ord(b, x, y, z)$ indicates whether or not the objects x, y, z appear in the blog b in this order, as follows.

$$ord(b, x, y, z) := true \;\; \texttt{iff} \;\; pos(b, x) < pos(b, y) < pos(b, z) \tag{5}$$

The function $pos(b, x)$ indicates the word position of the first appearance of the object x in the blog b. A set of contents $C(\xi_j)$ presented on the multimedia tour guide is formulated as follows.

$$C(\xi_j) := \{c_k \mid b \in B'(\xi_j) \wedge ord(b, p_1(j), g_k, p_l(j)) \wedge near(b, g_k, t_k)\} \tag{6}$$

c_k is a content, which is a pair of an image g_k and text t_k. $p_1(j)$ and $p_l(j)$ are the first and last appearances of place names in the sequence ξ_j. The function $ord(b, p_1(j), g_k, p_l(j))$ indicates whether or not g_k appears between *route elements* (a part of place names) in the blog b. A route element is described in subsection 4.2. The function $near(b, g_k, t_k)$ indicates whether or not the image g_k and the text t_k appear near each other (under certain criterion) in the blog b. A multimedia tour guide $T(\xi)$ consists of a set of routes $R(\xi)$, a set of blog entries $B'(\xi_j)$ and their contents $C(\xi_j)$.

$$T(\xi) := \{R(\xi_j), B'(\xi_j), C(\xi_j)\} \tag{7}$$

3.2 Context Model

In this subsection, we discuss contexts of routes. We define a context as a common topic or interest shared by tourists who actually traveled along the route. In our definition, a context can be expressed by a set of keywords. The concrete method to extract these keywords is described in the following sections. A context keyword extracted by the system is expected to be classified into one of the three types indicated below.

Edge type: Context is relevant to one of the edges between the nodes.
Multiple nodes type: Context is relevant to more than one node in the route.
Single node type: Context is relevant to one of two nodes in the route.

The *edge type* is defined as a context that is relevant to the route itself. This is the most preferred information for our multimedia tour guide system. In the *multiple nodes*

type, the context is relevant to more than one node in the route. This is useful for some tourists who plan to travel with interests in certain subjects. Lastly, in the *single node type*, the context keyword is relevant to only one of the nodes. Such information is useful but is already provided by many local information search systems.

4 Route Mining

In this section we explain our method to extract typical routes and their contexts.

4.1 Local Blog Crawler

A system cannot extract frequent sequential patterns that represent typical travel routes without mining a large set of local blog entries. Since it is difficult to obtain a large number of relevant blog entries each time the user sends a query, we built a local blog crawler that collects blog entries periodically. It sends place names from a manually created list as queries and collects blog entries from conventional RSS blog search engines.

4.2 Extraction of Visitors' Routes

In this subsection, the aim is to extract a tourist route from each blog entry, whenever there is one. We cannot expect the author of a blog entry to have visited all of the place names appearing in the entry, so a filtering mechanism is needed. In discussing the place-name filter that estimates whether a place name was actually visited by the blogger, we will call the place names that were visited by the author *route elements*. The order of appearances of the route elements are used as the order of the sequence. This is because many of the blog entries written by tourists are written in a diary style, and therefore the order in which place names appear reflects the order in which they were visited. In addition, if one place name appears in a sequence some times as route elements, we remove all but the last occurrence of items that occur more than once. Such patterns occur when a blogger is discussing about these locations before or after the trip.

In the filtering step the system judges whether the author of a blog entry has actually visited the location specified by the place name, and it adds the place name as a node of the route only if the judgment is yes. The criterion used in the judgment is whether the place name accompanies some actions performed by the author. Action verbs, such as "eat" and "see", and gerunds by a term indicating activity, such as "go fishing" and "go shopping," were extracted. We also considered the deep structure of a sentence and the dependency structure between noun phrases. We used CaboCha [11] to analyze dependency, and we used a Japanese Lexicon [12] as a dictionary for action verbs. We performed morphological and dependency analysis on sentences containing place names, and extracted place names that are followed by *spatial case particles* and then by an action verb. Place names followed by *spatial case particles* and an action verb are equivalent to English phrases like "going to Kyoto Station" and "arrive at Kiyomizudera Temple," and are found in sentences directly expressing actions. We use these place names as nodes for constructing typical travel routes. Another pattern is one in which

where a part of a sentence indicates action, but the whole sentence indicates a state. Examples are "the place where we arrived is the Silver Pavilion." and "Kiyomizudera Temple, to which we went next, ...". In these cases the place name does not accompany a spatial case particle but does accompany an action verb. The following two patterns can be formalized as follows.

Pattern 1: $\{place'\} \Rightarrow \{verb\}$
Pattern 2: $\{verb\} \Rightarrow \{place\}$

place': a place name + a spatial case particle
place: a phrase containing a place name, except the elements of *place'*
verb: an action verb
$\Rightarrow$ indicates a dependency relationship

In Pattern 1 we are only looking at the direct dependency relationship, but we also look at indirect dependencies in order to deal with sentences such as "we went to the Golden Pavilion *and* the Silver Pavilion" and "we visited a building *in* Kiyomizudera Temple." In addition, Blogs are often expressed in colloquial language. We used the following countermeasure to avoid these problems. For the abbreviation of a verb, we judged a place name to be a route element if it accompanied the case particles "-*kara* (from)," "-*e* (to)," and "-*made* (to)," which represent deep case for "source" or "goal." We assumed that appearances of place names in these deep cases are usually relevant to the author's visit. For the abbreviation of a case particle, we judged a place name to be a route element if it accompanied an action verb.

We tested the effectiveness of the above-mentioned method (Filtering) in a preliminary experiment measuring the precision of trials. When the route extracted by the filter was correct, we judged the route to be correct. The results are illustrated in Figure 1, where the line labeled "No filter" shows the results obtained without using the filter (i.e., when all the place names contained in a blog entry were used as route elements). The average precision plotted in this figure was calculated by considering the number of place names contained in each blog entry. The graph indicates that the revisions have increased the precision of the resulting routes.

4.3 Typical Route Mining

The system applies PrefixSpan [5] to the sequences of place names extracted by the revised method described in Subsection 4.2. The minimum value of items and the minimum value of support are both set to 2. We define the extracted patterns as *typical tourist routes*.

4.4 Context Extraction

In this subsection, we describe the method of obtaining contexts for the extracted routes. We define a context for a route as a common topic or interest shared by tourists who actually traveled along the route. In our definition, a context can be expressed by a set of keywords. The system estimates contexts from blog entries containing place names consisting the route. In the first step of the extraction, the system gathers blog entries that contain the route. In the second step, the system obtains a feature vector V_i for each blog entry b_i, in which each dimension represents a noun, based on wether or not the

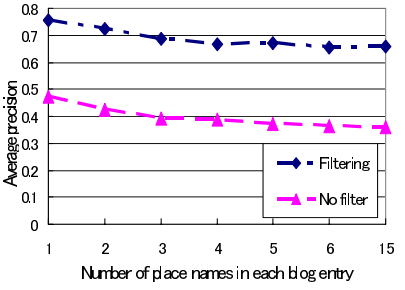


Fig. 1. Average precision of the extracted routes

noun exists in the blog entry e_i. In the third step, the system obtains a feature vector $V(r)$ for a route r by the following formula. In the formula, $N(r)$ represents the number of blog entries that contains the route r.

$$V(r) = \frac{1}{N(r)} \sum_{i=1}^{N(r)} V_i \tag{8}$$

Large components of the feature vector $V(r)$ indicate terms that are frequently used in blog entries containing the route r.Such terms contain not only context keywords that we are looking for, but also common words such as "person" and "group". Therefore, we use the *inverse document frequency* of the term to remove such noise. We obtain a feature vector $C_x(r)$ for a term x in the following way.

$$C_x(r) = V_x(r) \cdot \log \frac{N}{DF(x)} \tag{9}$$

$V_x(r)$ and $C_x(r)$ are x's components of the feature vector $V(r)$ and $C(r)$. N is the total number of blog entries which contain any of the routes. $DF(x)$ is the document frequency of a term x. We use m largest components of the feature vector $C(r)$ as contexts for the route r. In this way, the system obtains unique terms that are frequently used in blog entries that contain the route r. We define the $C_x(r)$ as the *contextual value* for a term x.

5 Generation of a Multimedia Tour Guide

The user either selects a location on the map interface or types a context keyword into the query box. The system searches through the database and presents typical travel routes that are relevant to the user's query. Once the user selects one of the presented routes, the system obtains blog entries whose travel routes are connected to the selected route. The system starts presenting multimedia content extracted from blog entries. Most of this multimedia content consists of images and texts provided by bloggers who actually traveled along the route. The continuously presented content enables the system user to experience the tour virtually. The user can obtain up-to-date information about the region because new blog entries are often posted frequently.

5.1 Tour Guide Interface

The generated guide content is browsed by using a Web browser, so it can be accessed from either a PC or a mobile device such as a PDA. As illustrated in Figure 2, the

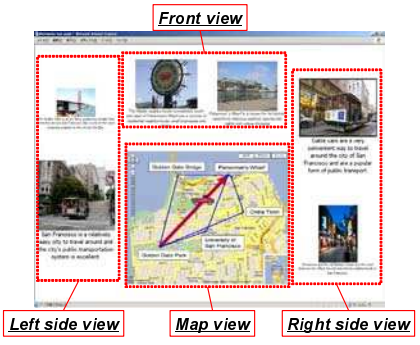


Fig. 2. Tour guide interface

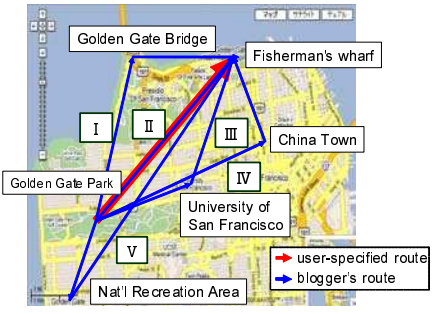


Fig. 3. Example of map view

interface comprises a map view, a front view, a left side view, and a right side view. The map view presents several routes that are relevant to the route specified by the user. The example is indicated in Figure3. The blue arrows indicate travel routes extracted from blog entries. As the vehicle icon moves along the selected route (red arrow), the views except map view present multimedia content extracted from blogs containing the names of places neighboring the current location of the vehicle icon. Depending on the relative location of these place names to the vehicle icon's location, the content is presented on the left side view, right side view, or front view. If the place is far from the vehicle icon's location, then the content is presented in a small size. Through this interface, the user can browse images and texts along the travel route continuously, without sending further commands.

5.2 Contents Extraction

In this subsection, we explain the method of extracting contents presented on the interface. Contents that we extract are mostly a set of tuples consisting of image and text, since blogs to this day mainly consist of these two media. However, the same method can be applied to videos and auditory contents. Images in local blogs are mostly photographs taken by bloggers. Through these images, the user can view his travel destination at a glance. We extract images and neighbored texts from a part of blog entries which are relevant to the user selected route. The system specifies the text contents about an image by html tags. We consider the number of html tags between text and an image. Then, the nearest one sentence from an image is extracted as neighbored text.

After collecting multimedia content from blog entries, the system must map it to geographic locations. This is a very difficult task because the location of the item presented in the content is usually not explicitly indicated in the blog entry. We therefore approximate the locations by the positions at which the content (e.g., images) and place names occur in the blog entry. In this subsection, we assume a place name as a route element discussed in Section 4. The estimation consists of two types, *place type* and *route type*. If the image is found between two occurrences of a place name A in a blog entry, we call it a *place type* image. In this case, we map the image to the coordinates of A. If, on the other hand, the image appears between two different place names A and

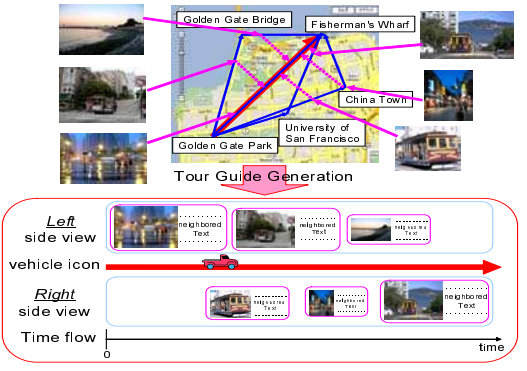


Fig. 4. Tour guide generation in right and left views

B, we call it a *route type* image. In this case, we assume that the image refers to some location between A and B. We approximate the location according to the length of the text between the image and the place names. We express coordinates of the images by a vector. The coordinates of the image C_i are estimated as follows:

$$C_i = \frac{p_A C_A + p_B C_B}{p_A + p_B} \tag{10}$$

C_A and C_B are respectively vectors indicating the coordinates of place names A and B, p_A is the text length between the image and place name A. Correspondingly, p_B is the text length between the image and place name B.

After estimating the image coordinates, we generate the multimedia tour guide. The method is illustrated in Figure 4. The system maps the images and their surrounding texts on the basis of the images' estimated locations. When the vehicle icon passes near the location, the image is presented in either view except map view. If the image location is far from the route, the image is presented in a small size.

6 Evaluation

In this section, we present the result of route mining and examples of contents extracted for the multimedia tour guide system. The source data contains 16,142 blog entries collected by an RSS search engine, sending 74 major place names in Kyoto, Japan, as search queries. Posting date of the blog entries ranged from May 22nd, 2006 to July 23rd, 2006.

6.1 Route Mining

Table 1 indicates the extracted context of the route. The six routes are the most frequent routes from Kiyomizudera Temple and to Kiyomizudera Temple. "observation deck" is extracted because Kiyomizudera Temple is famous for it. "Shop" is found on [Kiyomizudera Temple → Gion District] because there are many eating places between Gion Distinct and Kiyomizudera Temple. "Shinkansen Express" on [Kiyomizudera Temple → Kyoto Station] is a limited express which stops at Kyoto Station. Many school boys

Table 1. Extracted contexts

Route	freq.	extracted contexts with contextual values
Kiyomizudera Temple → Gion District	23	observation deck (0.9), shop (0.9), sightseeing (0.9)
Kiyomizudera Temple → Kyoto Station	17	observation deck (1.3), Shinkansen Express (1.2), leaving (1.1)
Kiyomizudera Temple → Golden Pavilion	17	school excursion (1.2), group (1.0), length (0.9)
Golden Pavilion → *Kiyomizudera Temple*	18	group (1.9), school excursion (1.8), act (1.4)
Yasaka Shrine → *Kiyomizudera Temple*	16	school excursion (0.9), position (0.8), place (0.7)
Kyoto Station → *Kiyomizudera Temple*	14	bus (1.2), arrival (1.0), tour (1.0)



Fig. 5. Contents extracted from blogs

and girls visit Kiyomizudera Temple, Yasaka Shrine, and Golden Pavilion in "school excursion". Contexts for routes, [Golden Pavilion → Kiyomizudera Temple] and [Yasaka Shrine → Kiyomizudera Temple], express it. "Bus" is a major means of transportation in Kyoto. Contexts for Kiyomizudera Temple and Golden Pavilion in different orders are similar. However, contexts for the route between Kiyomizudera Temple and Kyoto Station are not alike for different orders. It shows that the direction of movement affects the visitor's context in some cases.

6.2 Tour Guide Contents

In this subsection, we discuss retrieval results of images and texts used in the multimedia tour guide system. We performed manual extraction to evaluate the method described in Section 5. The result of the extraction is illustrated in Figure 5. Figure 5 shows contents extracted for the route, [Kiyomizudera Temple → Gion District] and its most frequent context "observation deck" and "shop". Some of contents in the figure are relevant to the context. The image at a bottom right corner in the figure is a landscape from the observation deck in Kiyomizudera Temple. Since local blogs are contents generated by consumers, they are useful for potential visitors (consumers). The contents include the same viewpoint for consumers. Contents include photographs of the route [Kiyomizudera Temple → Gion District], because there are many sightseeing spots between Kiyomizudera Temple and Gion District.

7 Conclusion

In this paper, we described a system to extract typical travel routes based on the blog entries of visitors and to present multimedia content relevant to these routes. We extracted typical travel routes by using a sequential pattern mining method. We also introduced a user interface for presenting multimedia content along the route in a proactive manner. The system works as an automatically generated tour guide accessible from a PC or a mobile device.

Acknowledgments. This work was supported in part by the Japanese Ministry of Education, Culture, Sports, Science and Technology by a Grant-in-Aid for Scientific Research on Priority Areas "Cyber Infrastructure for the Information-explosion Era", Planning Research: "Design and Development of Advanced IT Research Platform for Information" (Project Leader: Jun Adachi, Y00-01, Grant#: 18049073) and "Contents Fusion and Seamless Search for Information Explosion" (Project Leader: Katsumi Tanaka, A01-00-02, Grant#: 18049041), and by The 21st Century COE (Center of Excellence) Program "Informatics Research Center for Developmentt of Knowledge Society Infrastructure" (Leader: Katsumi Tanaka, 2002-2006).

References

1. Agrawal, R., Srikant, R.: Fast algorithms for mining association rules. In: Proc. 20th Int. Conf. Very Large Data Bases, VLDB. (1994) 487–499
2. Agrawal, R., Srikant, R.: Mining sequential patterns. In: the 11th Int. Conf. on Data Engineering, Taipei, Taiwan, IEEE Computer Society Press (1995) 3–14
3. Srikant, R., Agrawal, R.: Mining sequential patterns: Generalizations and performance improvements. In: Proc. 5th Int. Conf. Extending Database Technology, EDBT. Volume 1057. (1996) 3–17
4. Zaki, M.J.: SPADE: An efficient algorithm for mining frequent sequences. Machine Learning **42**(1) (2001) 31–60
5. Pei, J., Han, J., Mortazavi-Asl, B., Pinto, H., Chen, Q., Dayal, U., Hsu, M.C.: PrefixSpan mining sequential patterns efficiently by prefix projected pattern growth. In: Proc. 2001 Int. Conf. Data Engineering (ICDE'01), Germany (2001) 215–224
6. Uematsu, H., Numa, K., Tokunaga, T., Ohmukai, I., Takeda, H.: Ba-log: a proposal for the use of locational information in blog environment. The 6th Web and Ontology Workshop (2004)
7. maplog, http://maplog.jp/.
8. Kurashima, T., Tezuka, T., Tanaka, K.: Mining and visualization of visitor experiences from urban blogs. In: Proc. of the 17th Int. Conf. on Database and Expart Systems Applications(DEXA2006). (2006)
9. Tezuka, T., Tanaka, K.: Traveling in digital archive world: Sightseeing metaphor framework for enhancing user experiences in digital libraries. In: Proc. of The 8th Int. Conf. on Asian Digital Libraries(ICADL2005), Bangkok (2005)
10. Schilling, B.A., Zipf, A.: Generation of vrml city models for focus based tour animations: integration, modeling and presentation of heterogeneous geo-data sources. In: Proc. of the 8th Int. Conf. on 3D web technology (Web3D '03), France (2003) 39–48
11. CaboCha, http://chasen.org/ taku/software/cabocha/.
12. Japanese Vocabulary System, http://www.ntt-tec.jp/technology/C404.html.

A Robust 3D Face Pose Estimation and Facial Expression Control for Vision-Based Animation

Junchul Chun[1], Ohryun Kwon[1], and Peom Park[2]

[1] Department of Computer Science, Kyonggi University, Yui-Dong Suwon, Korea
{jcchun,kor5663}@kyonggi.ac.kr
[2] Department of Industrial Engineering, Ajou University/Humintec, Wonchun-Dong, Suwon, Korea
ppark@ajou.ac.kr
http://giplab.kyonggi.ac.kr

Abstract. This paper presents a new approach to estimate 3D head pose from a sequence of input images and retarget facial expression to 3D face model using RBF(Radial Based Function) for vision-based animation. The exact head pose estimation and facial motion tracking are critical problems to be solved in developing a vision based human computer interaction or animation. Given an initial reference template of head image and corresponding 3D head pose, full the head motion is recovered by projecting a cylindrical head model to the face image. By updating the template dynamically, it is possible to recover head pose robustly regardless of light variation and self-occlusion. Moreover, to produce a realistic 3D face model, we utilize Gaussian RBF to deform the 3D face model according to the detected facial feature points from input images. During the model deformation, the clusters of the minor feature points around the major facial features are estimated and the positions of the clusters are changed according to the variation of the major feature points. From the experiments, the proposed method can efficiently estimate and track the 3D head pose and create a realistic 3D facial animation model.

1 Introduction

The requirements of a realistic and feasibly animated facial model have been increased because facial modeling has been an important field of diverse application areas such as virtual character animation for entertainment, 3D avatars in the internet, 3D teleconferencing, and face recognition. Moreover, a growing interest in developing more intuitive and natural interaction between user and computer using vision-based facial expression. The vision-based face motion tracking and facial expression recognition is an attractive input mode for better human-computer interaction [1]. However, face pose estimation and tracking are tough challenge particularly in varying lighting conditions and a moving, clustered background image[2,3,4,5]. Meanwhile, the analysis of facial features has been one of the challenging problems in computer vision field. Especially, the facial expression retargeting is considered a critical work for human-centered interface design and even facial expression cloning [9,11,12].

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 700 – 708, 2007.
© Springer-Verlag Berlin Heidelberg 2007

Many studies have been done for recovering face motion from image sequences [2,3,4,5]. One is to use distinct image features [2], which work well when the features may be reliably tracked over the image sequence. When good feature correspondences are not available, tracking the entire facial region using a 3D head model is more effective. Both generic and user-specific model have been used for head motion recovery [3]. Much simpler geometric models such as planner model and ellipsoidal model, which is effective and robust against initialization errors, have been introduced [4].

In this paper we propose an automated model-based 3D head pose estimation and facial expression control for vision-based animation. Figure 1 illustrates the overall procedures for 3D face pose estimation and facial expression generation. In the initial stage, head region from video input image is detected by template matching between a given template face image and input video frame. To generate a template face image, an average face image is generated from training images and principal component analysis (PCA) is applied to the average face image. Then a cylindrical head model is created and projected onto the detected face image. The head motion is tracked by using optical flow and the exact head pose is recovered by dynamically updating the projected template. At the same time the detected facial points and the other feature features around the facial points are retargeted to a 3D face model (avatar) according to the facial variation of the input image.

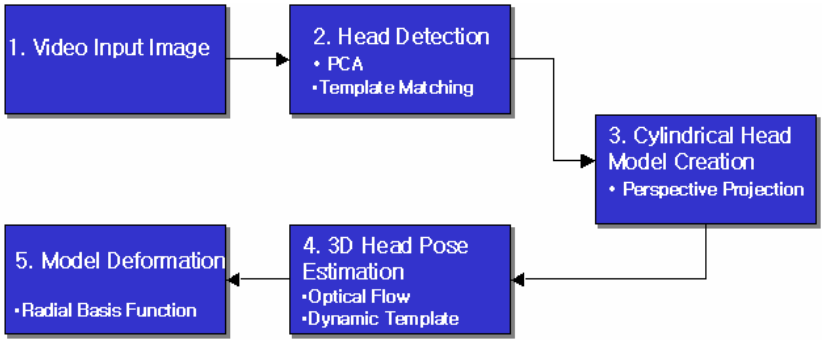


Fig. 1. An Overview Diagram of 3D Head Pose Estimation and Facial Expression Control

The rest of the paper is organized as follows. Section 2 explains how to estimate and track head pose from sequential input images. Section 3 describes the way to generate facial expression from facial feature points with Gaussian RBF. The results of head pose estimation and facial expression cloning based on the proposed approach are provided in section 4. Conclusion and future works are given in Section 5.

2 Dynamic Head Pose Estimation Technique

The proposed 3D face pose tracking consists of two major phases: face detection and cylindrical model-based head pose estimation. Figure 2 shows the details for head pose estimation.

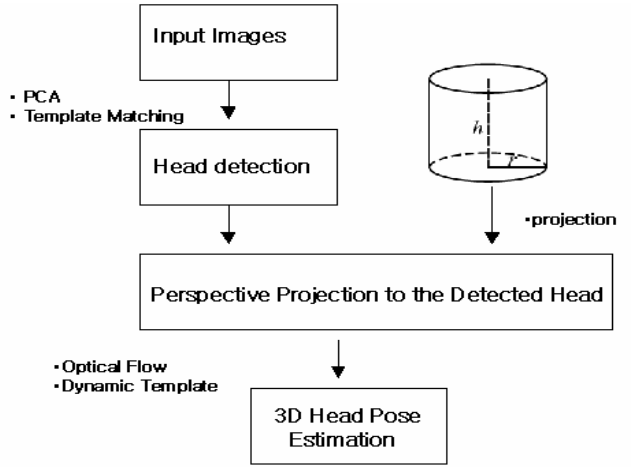


Fig. 2. Phases for 3D head pose estimation

Before tracking the varying head motion from the sequential input images, the candidate face region should be extracted. In general, color information is known efficient for identifying skin region. However, in computer vision, selecting color space is very important factor for face detection since every color space has different properties. The authors propose a nonparametric HT skin color model to detect facial region efficiently rather than using existing parametric skin color model [6,7]. With the HT skin color model we can extract the candidate face region and detect face by use of template matching.

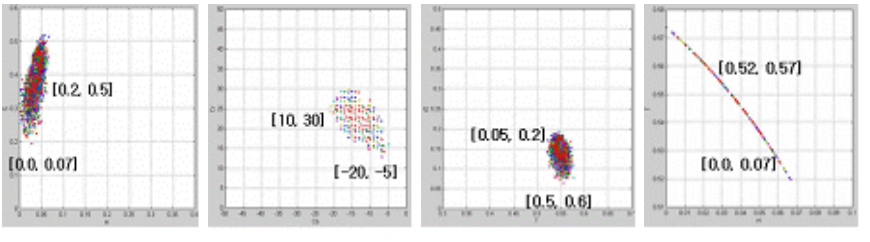


Fig. 3. Skin color distribution with H-S, Cb-Cr, T-S and H-T from left to right

In order to reduce the dimension of the facial data, Principal Component Analysis (PCA) is applied to the facial region. From the facial candidate region we can extract exact face using template matching based on L_2 norm defined as follow:

$$Error = \sum_{i=0}^{79} \sum_{j=0}^{79} \sqrt{(I_{ij} - T_{ij})^2} \tag{1}$$

Once face is detected from input video image, for face pose estimation we project the cylindrical 3D model to the detected. Given an initial reference template of the face image and the corresponding head pose, the cylindrical head model is created and the full head motion is traced from the sequential input images.

The head motion tracking using a template can be described as follows. If an image $I(u,t)$ at time t where $u=(x,y)$ is a pixel in the image is given, at $t+1$, u moves to $u'=F(u,\mu)$, where μ is the motion parameter vector and $F(u,\mu)$ is the parametric model, which maps u to the new position u'. The motion vector μ can be obtained by minimizing following function when the illumination condition is unchanged.

$$\min E(\mu)=\sum_{u\in\Omega}(I(F(u,\mu),t+1)-I(u,t))^2 \tag{2}$$

where Ω is the region of template at t. By using Lucas-Kanade method [8], the problem of equation (2) can be solved as follows:

$$\mu=-\left(\sum_{\Omega}(I_u F_\mu)^T(I_u F_u)\right)^{-1}\sum_{\Omega}(I_t(I_u F_u)^T) \tag{3}$$

where I_t and I_u respectively are the temporal and spatial image gradients. F_μ means the partial differential of F with respect to μ, which depends on the motion model and is computed at $\mu=0$. To present the geometry of the entire head, 3D cylindrical model is projected to the input face model and the head pose is estimated using the projected face model. If the location of the head pose t is $X=[x,y,z]^T$ then the locations of the head pose at $t+1$ becomes

$$X(t+1)=M\bullet X(t)=\begin{bmatrix}R & T\\ 0 & 1\end{bmatrix}\bullet X(t) \tag{4}$$

when R is the rotation matrix with 3 degree of freedom and T is the 3D translation vector. Then the image projection μ of $X=[x,y,z]^T$ at $t+1$ can be defined

$$u(t+1)=\begin{bmatrix}x-y\omega_z+z\omega_y+t_x\\ x\omega_z+y-z\omega_x+t_y\end{bmatrix}\cdot\frac{f_L}{-x\omega_y+y\omega_x+z+t_z} \tag{5}$$

where $\lfloor\omega_x,\omega_y,\omega_z\rfloor$, $\lfloor t_x,t_y,t_z\rfloor$, f_L represents the rotation, translation and the focal length respectively. Consequently, the motion model $F(u,\mu)$ with the parameter $\mu=\lfloor\omega_x,\omega_y,\omega_z,t_x,t_y,t_z\rfloor$ is defined by

$$F_\mu\big|_{\mu=0}=\begin{bmatrix}-xy & x^2+z^2 & -yz & z & 0 & -x\\ -(y^2+z^2) & xy & xz & 0 & z & -y\end{bmatrix}\cdot\frac{f_L}{z^2}(t). \tag{6}$$

However, the presence of noise and light variation can cause a problem of losing pixel data in the template while tracking the variation of head pose. To maintain the accuracy of the head pose estimation under such conditions, as preprocessing light

compensation is done by use of min-max normalization of the light and it is defined as follow:

$$y = \left(\frac{y - \min_1}{\max_1 - \min_1} \right) (\max_2 - \min_2) + \min_2 \tag{7}$$

where $\min_1$, $\max_1$ $\min_2$ and $\max_2$ are minimum and maximum values of input image and those of the desired value, respectively.

Meanwhile, the self-occlusion problem can be solved by dynamically updating the template while tracking the head pose. The single template through the entire image sequence is not enough to cope with the problems like light change and self-occlusion. Once the head pose is recovered the detected facial region is used as a template. However, if occlusion is occurred at certain frame, the current template is removed and the last template is considered as a new template for the next frame. This makes the robustness of the head pose estimation improved.

3 Facial Expression Control of a 3D Avatar Using RBF

In order to retarget the facial expression to a specific facial model, we can make use of various deformation methods such as scattered data interpolation, anthropometry techniques, and projection onto the cylindrical coordinates incorporated with a positive Laplacian field function [9,10]. In this work, we have used scattered data interpolation.

We have to consider two fitting process; the one fits estimated feature points in generic model to corresponding feature points and the other modify non-feature points in generic model using interpolation technique. Scattered data interpolation refers to the problem of fitting a smooth surface through a scattered or non-uniform distribution of data points. We have considered the problem of scattered data interpolation as follow:

Given

$$(p_i, q_i) \in \Re^3 \times \Re^3, \qquad i = 1, .., N \tag{8}$$

we can find a continuous function $f : \Re^3 \to \Re^3$

$$f(p_i) = q_i, \qquad i = 1, \ldots, N \tag{9}$$

The points (p_i, q_i) are corresponding feature points pair and the points in $\Re^3$ are denoted either by $\vec{x}$, or $\vec{x} = (x, y, z)$. Radial basis function (RBF) is to define the interpolation function as a linear combination of radialy symmetric basis functions, each centered on a particular feature point. A RBF generally consist of two functions. Given *N* corresponding feature point pairs, they can be described by the following equation, where $\vec{x} = (x, y, z)$;

$$f_k(\vec{x}) = P_{mk}(\vec{x}) + \sum_{i=1}^{N} A_{ik} \Phi(\|\vec{x} - \vec{x}_i\|), \qquad k = 1,2,3 \tag{10}$$

A_{Nk} is the weight associated with the *Nth* RBF, centered at $\vec{x}_i$. $P_{mk}(\vec{x})$ is a polynomial of degree *m*, or is not present. Φ is a radial function, $\| \ \|$ denotes the Euclidean norm, such that:

$$\|\vec{x}-\vec{x}_i\| = [(x-x_i)^2+(y-y_i)^2+(z-z_i)^2]^{\frac{1}{2}} \tag{11}$$

It is necessary to decide a proper basis function, weight, centers, and width parameter for interpolation. The choice of a basis function is determined by the dimension of the interpolation problem, the interpolation conditions, and the desired properties of the interpolation [10]. Gaussian function can exist without polynomial precision and be used to deform a complex structure like a face. In addition, Gaussian is localized in a neighborhood near the center in comparison to other functions that have a global response. Thus for facial feature control, we use Gaussian function as a basis function of RBF. The basis function of the Gaussian can be expressed by:

$$\Phi(\|\vec{x}-\vec{x}_i\|) = e^{(-(\vec{x}-\vec{x}_i)^2/\sigma)} \tag{12}$$

In this research, we consider feature points as center. Therefore, we only decide weights and width parameter. Since we know 3D coordinates of feature points $\vec{x}$ and vertices positions $\vec{y}$ in 3D face model corresponding to feature points, we can evaluate weights by solving the following equations:

$$f_k(\vec{x}_i) = \vec{x}_i - \vec{y}_i$$
$$f_k(\vec{x}_i) = \sum_{j=1}^{N} A_{ik}\Phi(\|\vec{x}_i-\vec{x}_j\|), \qquad k=1,2,3 \tag{13}$$

The clusters of feature points around the detected points are made under the influence of each width parameter. We use the mahalanobis distance between feature points and furthest points from feature points in each cluster as width parameters.

$$\sigma_i = \max_i([(\vec{x}_k-\vec{x}_i)'S^{-1}(\vec{x}_k-\vec{x}_i)]^{\frac{1}{2}}) \tag{14}$$

$\vec{x}_k$ is a point in k^{th} cluster and *S* is the covariance matrix. Following figure 4 shows clustering results based on the major 13 feature points.

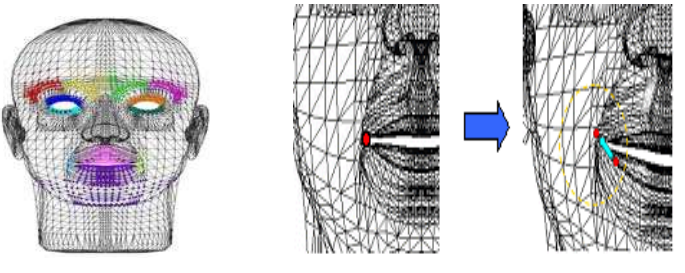


Fig. 4. Clustering results of 13 major feature points (left) and an example of local variation of a feature point (right)

4 Experimental Results

Figure 5 illustrates the results of face tracking and head pose estimation based on the cylindrical head model and optical flow method. The experiments show head motion is fully recovered using three different types of head pose variation. Moreover, the head pose is recovered even when the facial region is partially occluded by hand or paper during tracking the facial region as illustrated in figure 6.

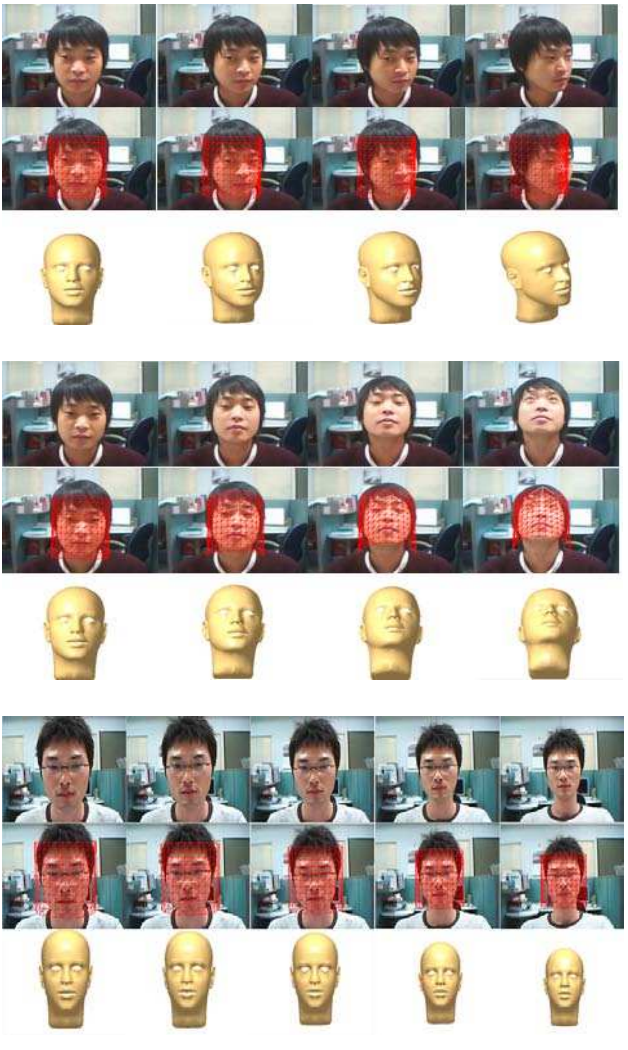


Fig. 5. Three different types of head pose estimation

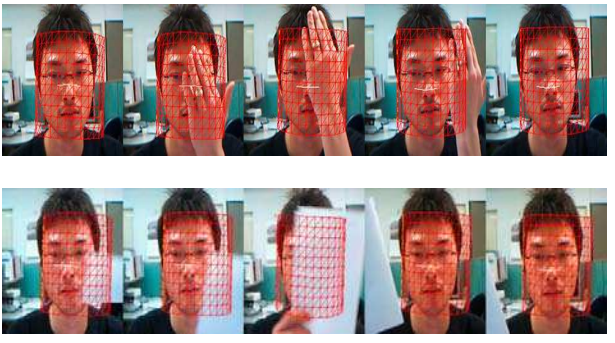
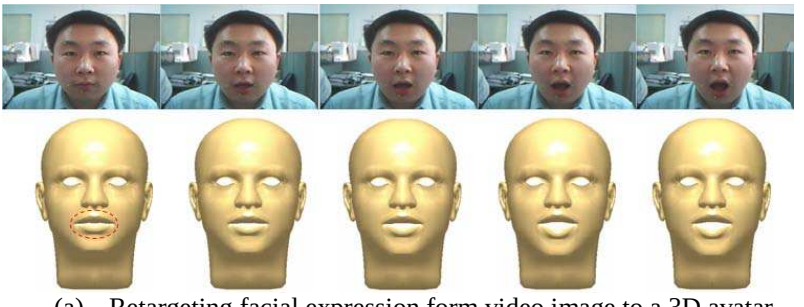
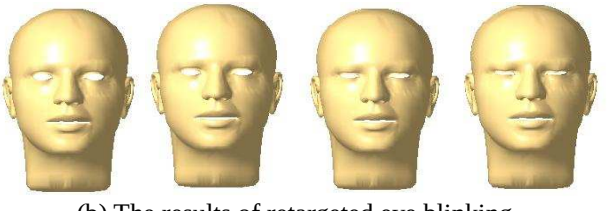


Fig. 6. Results of head pose recovery with self-occlusion

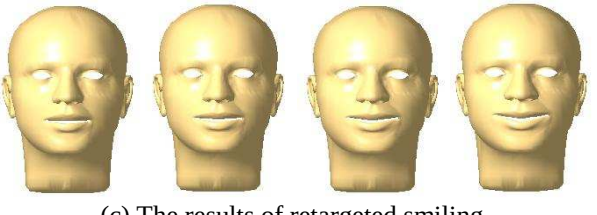
The facial feature variation of the target 3D model according to the change of facia l features of the video input image using the proposed facial feature clustering and RB F is illustrated in figure 7. Smiling, eye blinking and mouse shape variations from a s equence of input video images are instantly retargeted to a 3D animation face model.



(a) Retargeting facial expression form video image to a 3D avatar



(b) The results of retargeted eye blinking



(c) The results of retargeted smiling

Fig. 7. Facial expression control of a 3D avatar

5 Concluding Remarks

In this work, we propose a robust approach to track and estimate 3D head pose from a sequence of input face images. For this, we utilize a cylindrical head model and optical flow. Form the experiments, we can show the proposed method can effectively recover head pose fully even when self-occlusion is occurred in the sequences of input images. Consequently, the result of face tracking and pose estimation will be used for real time facial expression retargeting to a virtual 3D avatar. For facial expression control, we use scattered data interpolation with RBF to solve reconstruction problem. In this stage, it is necessary to decide a proper basis function, weight, centers, and width parameter for interpolation. Thus, we adopt Gaussian function as basis function and propose a new width parameter decision rule, which makes clusters of feature points to the detected major feature points under the influence of each width parameter. From experiments, the proposed method is also proved to be suitable to generate realistic facial expression of 3D avatar according to the variation of the facial expression from a sequence of input images.

References

1. J. Preece, *Human-Computer Interaction*, John Wiley, 1998.
2. Liu and Z. Zhang, “Robust Head Motion Computation by Taking Advantage of Physical Properties,” HUMO 2000, 2000
3. I.A. Essa and A.P. Pentland, “Coding analysis, interpretation, and recognition of facial expressions,” PAMI, Vol. 19, No. 7, pp. 757-763, 1997.
4. G.D. Hager and P.N. Belhumeur,“Efficient region tracking with parametric models of geometry and illumination,” PAMI, Vol. 20, No. 10, pp. 1025-1039, 1998.
5. Chien-Chia Chien, Yao-Jen Chang, and YC Chen, “Facial Expression Analysis Under Various Head Poses,” Proceedings of 3rd IEEE Pacific-Rim Conf. on Multimedia (PCM2002), Hsinchu, Taiwan, Dec. 16-18, 2002.
6. K. Min, J. Chun, G. Park “A Nonparametric Skin Color Model for Face Detection from Color Images,” LNCS 3320 (PDCAT), pp. 115-119, 2004..
7. J. Chun, K Min, “A Realistic Human Face Modeling from Photographs by Use of Skin Color and Model Deformation,” LNCS 3480, pp 1135-1143, 2005.
8. J.L. Barron, D.J. Fleet and SS. Beauchemin, "Performance of Optical Flow Techniques", Int. Journal of Computer Vision, pp. 43-77, 1994.
9. Noh, J., Neumann, U: A survey of facial modeling and animation techniques. Tech. rep., USC 99-705. 1998.
10. Wirth, M.A: A Nonrigid Approach to Medical Image Registration: Matching Images of the Breast. Ph.D. Thesis. RMIT University Melbourne Australia., 2000
11. A. Wojdel, L. J. M. Rothkrantx, “Parametric Generation of Facial Expressions Based on FACS”, Computer Graphic Forum, Vol. 24, pp. 743-757, 2005.
12. J.Y. Noh and U. Neumann, “Expression Cloning”, Computer Graphics, Proceedings of ACM SIGGRAPH, pp. 277-288, 2001

Hierarchical Shape Description Using Skeletons

Jong-Seung Park

Department of Computer Science & Engineering, University of Incheon,
177 Dohwa-dong, Nam-gu, Incheon, 402-749, Republic of Korea
jong@incheon.ac.kr

Abstract. This article presents a skeleton-based shape description scheme called a skeleton tree. A skeleton tree represents an object shape as a hierarchical structure where high-level nodes describe parts of coarse trunk of the object and low-level nodes describe fine details. Each node refines the shape of its parent node. Most of the noise disturbances are limited to bottom level nodes. The similarity of two shapes is measured by considering the best match of a skeleton tree to a subtree of another skeleton tree. This partial matching is particularly useful when the shape of an animated object is deformed and also when a part of an object is occluded. Several experimental results are presented demonstrating the validity of our scheme for the shape description and indexing.

1 Introduction

Description of image contents has been a fundamental issue for further processing in image analysis and content-based image retrieval. Shape is one of key visual features in characterizing image semantics. Two most important issues in a shape-based image retrieval is shape description and shape matching. Shape description characterizes image contents and shape matching determines the relevancy of shapes based on similarity measures of features. For the representation of 2D objects in images, many description schemes have been proposed, e.g., chain codes, signatures, and skeletons. Among them, skeletons provide an efficient way to represent high-level semantics but they are unstable and sensitive to shape details. Recently, there have been works to handle shape features effectively such as curvature scale space[1]. There also have been advances in the skeleton approach to overcome critical problems associated with the structural weakness of skeletons: skeleton abstraction schemes and similarity measures [2,3], shock graph[4,5], and skeletal graph[6].

This paper presents a new shape description scheme, called a *skeleton tree*, which is a hierarchical shape description scheme using skeletons. A skeleton tree represents an object in a hierarchical manner such that higher levels describe parts of coarse trunk of the object and lower levels describe fine details. Each low-level node refines the shape of the parent node. Most of the noise disturbances are limited to the bottom levels. The boundary noise is controlled by decreasing weights on the bottom levels.

The image retrieval process constructs a skeleton tree for a query image and compares the skeleton tree to the archived skeleton trees stored in a local

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 709–718, 2007.
© Springer-Verlag Berlin Heidelberg 2007

database. We assume that images are already segmented using a segmentation tool which could be either automatic or interactive. For each region, a skeleton is obtained using a distance transform of a segmented image. Then, using the skeleton image, a skeleton tree is constructed. Steps of our shape retrieval algorithm for a query image is as following:

Step 1: Skeletonize the query shape region.
Step 2: Extract skeleton segments from the skeletons.
Step 3: Select nodes and links from the skeleton segments and construct a skeleton tree.
Step 4: Compute the shape feature vector for the skeleton tree.
Step 5: Compute similarities between the feature vector of the skeleton tree for the query shape and those of the archived skeleton trees that are already saved in a local database.
Step 6: Rank the best matched n shapes using the similarities.

As well as the image retrieval process, an image archiving process is required to establish a shape database. The image archiving is an off-line process which constructs skeleton trees and shape feature vectors and then saves them to a local database, similar to those of image matching process corresponding to Step 1–4.

In the following, in Section 2, our proposed skeleton tree description scheme is introduced. Section 3 describes the shape matching of skeleton trees in computing shape similarities. Then experimental results are presented in Section 4. We conclude this paper with discussion and future works in Section 5.

2 Skeleton Tree for Shape Representation

An image could be considered as a set of regions where each region corresponds to a shape of an object. For each region, a skeleton image is obtained using a distance transform and, from the skeleton image, a skeleton tree is constructed. The skeleton representation is a natural way of shape description especially for articulated or deformable objects such as human, animals, fishes and insects. Beside its naturalness, the shape can be reconstructed from the skeleton by taking an inverse skeleton transform.

The major drawback of skeleton representation is that it is sensitive to noise in shape boundaries. Most segmentation algorithms without a priori information or user interference yield unsuccessful object shapes. They frequently contain small holes and ragged boundaries. In that case, approaches of medial axis may cause spurious branches and shape distortions for the jagged boundaries. Our proposed skeleton tree description supplements such a weakness. The sensitivity to the boundary noise is prohibited by employing a hierarchical tree representation called the skeleton tree. A set of nodes in a level represents the object in a certain coarseness.

2.1 Extracting Skeleton Segments

The skeleton $skel(obj)$ of an object obj is defined as the locus of the centers of the maximal disks that are contained within the shape. To extract the skeleton,

numerous methods has been proposed such as methods using a distance transform[7], methods using a morphological operator[8], and methods using the Voronoi graph[9]. Among them, we use a distance transform method for the purpose of simplicity. For each pixel, the method assigns a number that is the distance between the point and its closest border pixel.

To avoid too many skeletal branches, we remove any skeleton pixels having distances less than a given threshold t. The choice of a certain threshold value only affects the complexity of the constructed structure. For the simplification and the fast computation of the skeleton tree, t is typically chosen between 7 and 10 in pixel unit.

To make the skeleton more descriptive we convert it to a set of skeleton segments. A *skeleton segment* is defined as a curve segment which has only two end points without any junctions. It is represented as a list of skeleton pixels where each skeleton pixel has a distance value to the closest border pixel. A set of skeleton segments can be obtained by following and linking skeleton edges. A point is classified as an *end point* if it is connected to a single direction in its neighborhood. If the point is connected to more than two directions in its neighborhood, it is classified as a *junction point*. The edge following process starts at an end point and follows a connected pixel until all the connected pixels are visited. If the current point is a junction point, the curve segment is added to the set of skeleton segments and the edge following process is restarted at the junction point. We use the 8-connected neighborhood and the visited points are excluded for the further consideration.

Once all the skeleton segments are prepared, we construct a skeleton tree. After the construction, the skeleton segments are also kept so that the boundary can be easily reconstructed from the skeleton tree. Note that a reconstructed boundary may be different from the original one since we ignored all the skeleton pixels of distance less than the given threshold value t.

2.2 Constructing Skeleton Trees

Using the extracted skeleton segments, the *skeleton tree* is constructed. A skeleton segment introduces a link and its two end points introduce two nodes. When pixel coordinates of end points are equal, their corresponding nodes are regarded as the same node. The set of nodes is denoted by $V(obj)$ and the set of links is denoted by $E(obj)$. The two sets $V(obj)$ and $E(obj)$ defines a skeleton tree. A node is a leaf node if it has only one link. A link connected to a leaf node is called a *skin link* and a link which is not a skin link is called a *bone link*. The set of skin links is denoted by $E_S(obj)$ and the set of bone links is denoted by $E_B(obj)$.

Each link defines an *influence zone* which is a set of pixels whose nearest skeletal pixels are on the link. The zone corresponds to the union of sweep regions with a moving disk of various radii. Once $V(obj)$ and $E(obj)$ are given, we determine a path called the *maximum influence path*, $(v_1, \ldots, v_n)$. All nodes in the path must be non-leaf nodes. The maximum influence path is a path that the union of influence zones of all the links in the path is maximum among all

possible paths between node v_1 and node v_n. For a pair of non-leaf nodes we compute the maximum influence path by inspecting all possible paths for the two nodes and record the path and the area of its influence zone as attributes of the pair. A link is called a *spine link* if the link is on the maximum influence path. The set of spine links is just a subset of $E_B(obj)$.

A skeleton tree represents a shape in three different scales: a coarse level shape, an intermediate level shape, and a fine level shape. A *coarse level shape* is a description of the shape only by the set of spine links. An *intermediate level shape* is a description of the shape by the set of bone links. A *fine level shape* is a description of the shape using both the set of bone links and the set of skin links.

Once the maximum influence path is determined, we construct the coarse level skeleton subtree which is a one-level tree. The nodes in the path $(v_1, \ldots, v_n)$ constitutes level-one in the order they appear from left to right where all the remaining nodes are attached to the tree recursively. The tree after appending all the bone links and their nodes corresponds to the intermediate level skeleton subtree and the tree of all bone links and skin links corresponds to the fine level skeleton tree or just the skeleton tree. The skeleton tree construction steps are summarized as follows:

Step 1: *Extract skeleton segments*: For the given binary image of an object *obj*, do the distance transform and make a skeleton. Then extract all skeleton segments from the skeleton.

Step 2: *Construct nodes and links*: Compute the set of nodes, $V(obj)$, and the set of links, $E(obj)$, from the skeleton segments.

Step 3: *Construct a coarse level tree*: Determine the maximum influence path and construct a one-level tree $T(V_1, E_1)$ from the spine links and their nodes.

Step 4: *Construct an intermediate level tree*: For each bone link $e = (v, v')$ or $e = (v', v)$ in $E_B(obj)$ which is not contained in the skeleton tree, if one node v is already a node in V_l $(l \geq 1)$, then insert the node v' as a child of v. The node v' becomes a node in V_{l+1}. This process is repeated until every non-leaf nodes are included in the skeleton tree. The result tree becomes an intermediate level tree.

Step 5: *Construct a fine level tree*: For each skin link $e = (v, v')$ in $E_S(obj)$ where v' is a leaf node, find the non-leaf node v in the tree and insert the node v' as a child of v. The node v' is marked as a leaf node of the tree. The final skeleton tree is denoted by $T(V, E)$.

Note that a cyclic graph cannot be created even when the given shape has holes since the node always becomes a child of a node in the immediate previous level.

2.3 Simplifying Skeleton Trees

Often, there are too many extra nodes due to the discrete property of region boundaries or noisy jagged boundaries. To achieve fast computation without losing object information, we prune unnecessary nodes and links before the

structure construction. The slight deformations on a boundary cause a node to be split into several nodes and the final tree structure would contain too many nodes. To prevent the tree construction of an excessive number of nodes, we introduce three operations: *cut*, *delete*, and *merge*. Those operations greatly simplify nodes and links. The behaviors of the operations are defined as follows:

- *Cut operation*: If the sum of gradient magnitudes of a distance image along a skin link is too small, we cut off the link from $E_S(obj)$ and also remove the leaf node connected to the link from $V(obj)$. If the gradient magnitudes are small around a skeleton segment, the corresponding link was created due to a small noise in a boundary. Most of the influence zone of such a link is already contained in other influence zones and removal of such links and nodes does not change the shape significantly.
- *Delete operation*: If influence zones of two skin links nearly overlap, we remove one of the duplicated skin links and also the isolated leaf node on it. The redundancy of influence zones is prohibited by removing such nodes and links.
- *Merge operation*: If two non-leaf nodes are too close, we merge them into a single node by extending one node to the other. This rule considerably simplifies the skeleton tree of an object.

By applying the above three operations, a skeleton tree with reduced sets of nodes and links is generated without significantly changing the shape.

3 Shape Matching Using Skeleton Trees

3.1 Comparing Shapes by Skeleton Trees

The similarity of two shapes is computed by comparing the similarity of the corresponding skeleton trees. Frequently, a query shape is just a partial portion of a target shape or a coarse version of a target shape. To provide the partial shape matching functionality, we match the shape of the first object obj_1 to all possible partial shapes of the second object obj_2 and we choose the best matched partial shape of obj_2. Let $T(V, E)$ and $T(V', E')$ be the skeleton trees of two objects obj_1 and obj_2, respectively. First, we generate all possible subtrees of $T(V', E')$. Then, the similarity measure for $T(V, E)$ and $T(V', E')$ is computed. If the similarity measure gives the maximum similarity value up to now, it is stored for the further references. The similarity for the most similar subtree becomes the similarity for obj_2.

A partial shape of obj_2 is a subtree of $T(V', E')$. For the simplicity of the implementation, we always include all non-leaf nodes to a subtree. We generate the sum of all combinations of all leaf nodes. The union of the set of non-leaf nodes and a subset of leaf nodes gives a shape representation. Generally for m leaf nodes, the number of all the combinations is 2^m. The same number of subtrees are considered for the tree $T(V, E)$.

Each skeleton tree defines a region which is the union of influence zones of their links. We denote the shape from a tree T as $S(T)$. When we compare

a given object obj_1 to another object obj_2, we generate all the possible shape representations $S(T^1(obj_2))$,...,$S(T^m(obj_2))$ where T^i is a subtree of $T(V', E')$. Then, we compare $S(T(obj_1))$ to each $S(T^i(obj_2))$, $1 \leq i \leq m$. The shape similarity measure for two objects is the maximum similarity among all the representations.

3.2 Computing Similarity Measures by Invariant Shape Features

As well as the skeleton tree description scheme, we also provide a proper similarity measure for them. Generally, there are two types of shape measures: boundary-based measures and region-based measures. Fourier descriptors and curvature scale space descriptors are typical boundary-based measures. Zernike moment descriptors and grid descriptors are region-based measures. Among them, moment invariants and Fourier descriptors are considered as two most representative features in 2D shape matching. Both of the measures are invariant to translation, scale change, and rotation in 2D space.

Mehtre et. al [10] compared the retrieval efficiency of several methods: reduced chain code, Fourier descriptors, moment invariants, Zenike moments, and Pseudo-Zenike moments. Though, in there experiments[10], the measure using both Fourier descriptors and moment invariants gave the best average retrieval efficiency, there are also cases when a measure using a single type of features is practical for some reasons such as a restricted computational power.

It is required to define the similarity of two skeleton trees. Due to the flexibility of the skeleton tree scheme, both boundary-based measures and region-based measures are applicable to the tree representation. Three types of features can be used for the similarity measures of the skeleton tree or subtree: Fourier descriptors alone, moment invariants alone, and the combination of both Fourier descriptors and moment invariants.

In the case of articulated objects such as human or animals, the shape transformation due to the motion of articulations makes the shape matching to original shape fail and the system may regard them as different objects. To handle the hard problem, we compare a skeleton tree of an object to all possible skeleton subtrees of another object. This partial matching approach overcomes the shape deformation problem of an animated object and also the shape occlusion problem. A tree of the first object is compared to one of possible subtrees of the second object by their shape representations. Among them, the similarity of the best matched pair of subtrees is regarded as the similarity of the two objects.

The skeleton tree description provides both the shape region and its boundary. From a skeleton tree, the corresponding shape region is directly generated by computing the union of all influence zones of tree links. The shape boundary is easily obtained by linking boundary pixels.

For a given tree, two feature vectors, the moment invariant vector $\mathbf{f}_Z$ and the Fourier descriptor vector $\mathbf{f}_H$, are computed using the shape boundary. The similarity of two skeleton trees means the similarity of the two feature vectors. Let $\mathbf{f}_Z$ and $\mathbf{f}_F$ be feature vectors of obj_1 and $\mathbf{f}'_Z$ and $\mathbf{f}'_F$ be feature vectors of obj_2.

Then the distance of moment invariants d_Z and the distance of Fourier descriptors d_F are computed by the following equations:

$$d_Z = d(\mathbf{f}_Z(obj_1), \mathbf{f}'_Z(obj_2)),$$
$$d_F = d(\mathbf{f}_F(obj_1), \mathbf{f}'_F(obj_2))$$

where d is the Euclidean distance of two vectors. The combined similarity measure could be defined as the average of two distances. There is a trade-off between the overall rough matching and the partial exact matching. The similarity measure gives a better matching result for the case when the most part of two shapes are roughly matched then for the case when relatively small part of two shapes are exactly matched. To control the trade-off, we introduce the weighted distance measure d_W:

$$d_W = \frac{1}{2}(w_Z d_Z + w_F d_F) \tag{1}$$

where w_Z and w_F are the weights of the distances d_Z and d_F, respectively, which are determined proportional to the region area and the number of region boundary pixels, respectively.

4 Experimental Results

We tested our method on several shape databases that are constructed from our manual segmentation works or collected from other research works. The top left figure in Fig. 1 shows a sample fish shape from a database and the top right figure is the corresponding distance transform image. To avoid too many skeletal branches caused by the jagged boundary, we remove the branches having distances less than a given threshold. The threshold value is determined

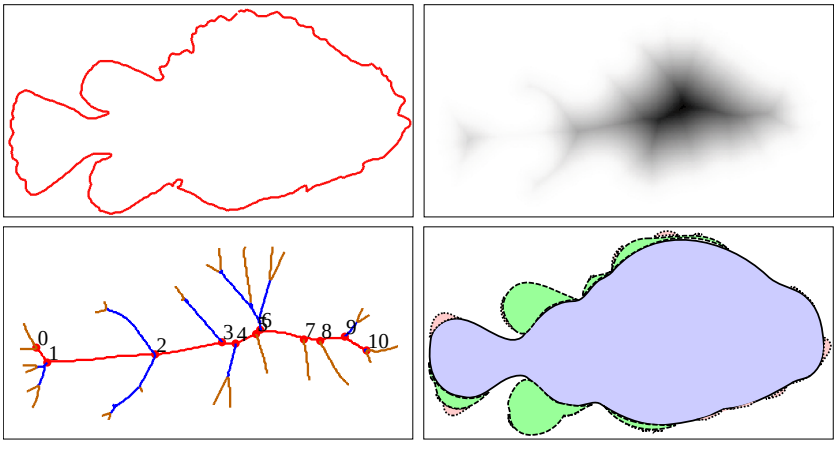


Fig. 1. A shape boundary of a fish (top left), the distance transform image (top right), skeleton segments from skeleton pixels (bottom left), and the recovered shapes with different levels of a skeleton tree (bottom right)

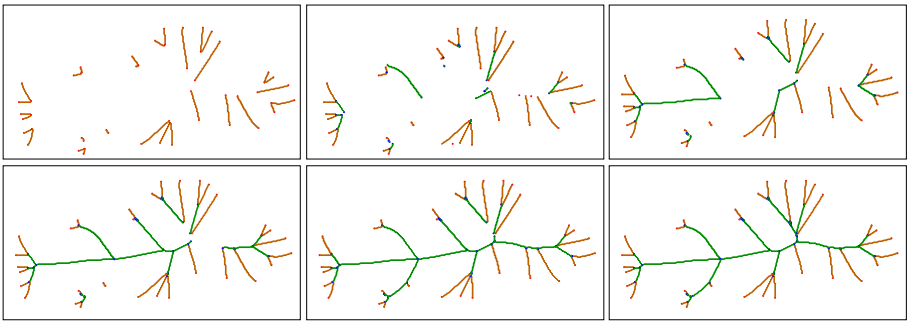


Fig. 2. Intermediate results of tree construction: Constructed trees after 1, 2, 3, 5, 7, and 13 iterations, in left-to-right and top-to-bottom order

in the interval from 5 pixels to 15 pixels depending on the image resolution. There are many unnecessary skeleton segments with a low threshold value less than 5 pixels. A threshold value more than 15 pixels is not preferred since the shape description could become obscure. In this example, we chose 10 pixels as a threshold value.

Once a skeleton is computed, we follow the skeleton pixels and generate a set of skeleton segments. Each skeleton segment has a link and two nodes. An end point or a junction point defines a node. A skeleton segment connecting two nodes defines a link. Fig. 2 shows the construction of skeleton segments with intermediate results of the process. The size of the fish image is 299×155 with 1223 skeleton points. To obtain all skeleton segments, we find all end points in the skeleton. For each end point we follow skeleton pixels until a junction point or an end point is encountered. The followed segments are added to a set of skeleton segments and removed from the skeleton. We repeat the process until there are no more pixels to follow in the skeleton. Three simplification operations are also applied to reduce the skeleton segments. All the skeleton pixels are followed in 13 iterations. The extracted segments are shown in the bottom left figure in Fig. 1. Each skeleton segment corresponds to one of a spine link, a bone link, and a skin link. The bottom right figure shows the recovered shapes in three different scales from the skeleton segments.

From skeleton segments, we construct the set of nodes and the set of links and finally generate a skeleton tree. Fig. 3 shows the constructed skeleton tree for the fish shape. The left figures show the nodes and links for the tree construction with the reconstructed region boundaries corresponding to the skeleton trees in the right figures. The first step is to determine the spine links. Among non-leaf nodes, the path from node 0 to node 10 is picked for the maximum influence path and they constitute level-one nodes (upper row in the figure). Then, other non-leaf nodes are attached to the tree recursively (middle row). Finally, all leaf nodes are added to the tree (lower row). Note that the number of nodes in level-one is just 11 while the total number of nodes in the tree is 48. But the area of the influence zone of level-one nodes is more than 90% of the total area of

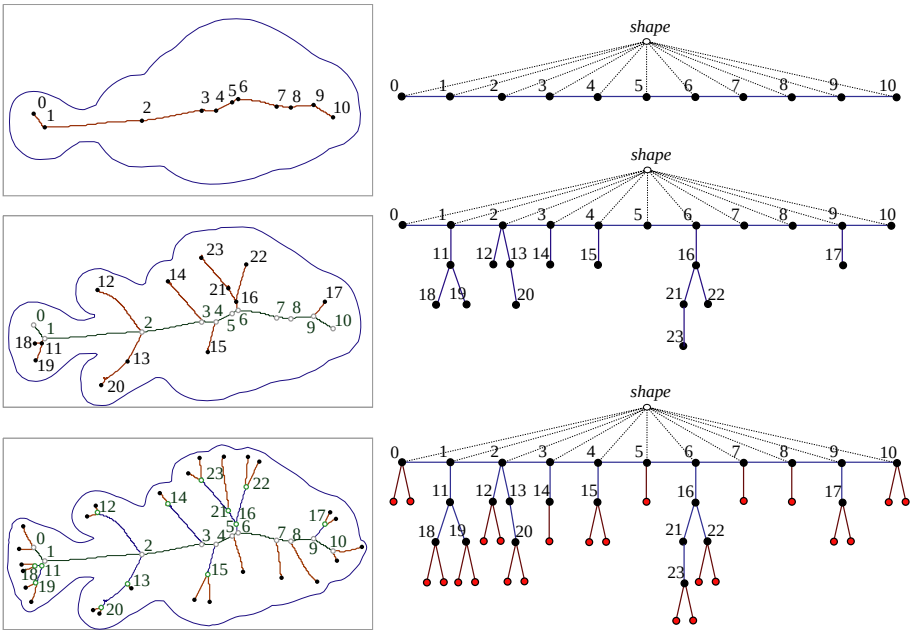


Fig. 3. A skeleton tree construction process. a tree from spine links (top), a tree from bone links (middle), a tree from all links (bottom).

the shape. Generally, the shape from level-one nodes is the major part of the shape and the set of other non-leaf nodes refines the shape. The set of leaf nodes describes the detailed part of the shape.

5 Conclusion

Many shape-based similarity retrieval methods perform well when the segmentation is adequate. However, most segmentation algorithms without a priori information or user interference yields unsuccessful object shapes. This paper proposed a novel shape description scheme termed the skeleton tree. Skeleton trees are not sensitive to noise in object boundaries. Once the skeleton tree is constructed, it is possible to do partial shape matching of two structures as well as reconstruction of original shape. The set of spine nodes describes deformable objects in a flexible manner. The description scheme also has the partial matching capability. A shape of the given query object is compared to all possible partial shapes of the target object. The best matched subtree is chosen and it is regarded as the match of two skeleton trees. This property makes the method overcome the shape deformation of an animated object.

Beside the novel property of our method, unexpected results may appear when there is a perspective effect in the shape since the invariance holds only when the deformation is a kind of 2D affine transformation. Irrelevant results may also

appear when the two boundaries are from the same object but a boundary was too much smoothed by a region extraction module.

As future works of our research, we are going to develop an automatic image segmentation algorithm which extracts only objects of interest regardless of the complexity of the environment where the object is located in.

Acknowledgements. This work was supported in part by grant No. RTI05-03-01 from the Regional Technology Innovation Program of the Ministry of Commerce, Industry and Energy (MOCIE) and in part by the Brain Korea 21 Project.

References

1. Mokhtarian, F., Abbasi, S., Kittler, J.: Robust and efficient shape indexing through curvature scale space. In: Proceedings of British Machine Vision Conference. (1996) 53–62
2. Telea, A., Sminchisescu, C., Dickinson, S.J.: Optimal inference for hierarchical skeleton abstraction. In: 17th International Conference on Pattern Recognition (ICPR 2004), IEEE Computer Society (2004) 19–22
3. Torsello, A., Hancock, E.R.: A skeletal measure of 2d shape similarity. Computer Vision and Image Understanding **95**(1) (2004) 1–29
4. Siddiqi, K., Shokoufandeh, A., Dickinson, S.J., Zucker, S.W.: Shock graphs and shape matching. Int. J. Comput. Vision **35**(1) (1999) 13–32
5. Sebastian, T., Klein, P., Kimia, B.: Recognition of shapes by editing their shock graphs. IEEE Trans. Pattern Analysis and Machine Intelligence **26**(5) (2004) 550–571
6. Ruberto, C.D.: Recognition of shapes by attributed skeletal graphs. Pattern Recognition **37**(1) (2004) 21–31
7. Arcelli, C., di Baja, G.S.: Euclidean skeleton via centre-of-maximal-disc extraction. Image and Vision Computing **11** (1993) 163–173
8. Haralick, R.M., Shapiro, L.G.: Computer and Robot Vision I. Addison-Wesley (1992)
9. Attali, D., Montanvert, A.: Modeling noise for a better simplification of skeletons. In: Proceedings of Int'l Conf. on Image Processing. Volume 3. (1996) 13–16
10. Mehtre, B.M., Kankanhalli, M.S., , Lee, W.F.: Shape measures for content based image retrieval: a comparison. Information Processing and Management **33**(3) (1997) 319–337

Motion Structure Parsing and Motion Editing in 3D Video

Jianfeng Xu[1], Toshihiko Yamasaki[2], and Kiyoharu Aizawa[2,3]

[1] Dept. of Electronics Engineering
[2] Dept. of Frontier Informatics
[3] Dept. of Information and Communication Engineering
The University of Tokyo
Fac. of Eng. Building # 2, 7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, Japan
{fenax, yamasaki, aizawa}@hal.k.u-tokyo.ac.jp

Abstract. 3D video, which is captured by multiple synchronized cameras and stored in mesh models, is emerging in recent years. However, the generation of 3D video is time-consuming and expensive. In this paper, we present an editing system to re-use 3D video efficiently. The hierarchical motion structure in 3D video is observed and parsed. Then, the representative motions are selected into a motion database, where the user can choose the desired motions. When synthesizing those chosen motions, the motion transition is optimized by a cost function. Some other information is also displayed in the interface to ease the editing. It should be mentioned that all the analysis and processing in our system are done in feature vector space.

1 Introduction

3D video, which consists of a sequence of 3D mesh models, is attracting increased attention recently. 3D video can reproduce not only the 3D spatial information such as shape and color of real-world 3D objects, but also the temporal information such as motion. Therefore, a dynamic 3D object can be rendered in high accuracy from an arbitrary viewpoint. The applications of 3D video include movies, games, medical system, broadcast, heritage documentation, etc.

Several 3D video generation systems have been developed in the last decade [1,2,3,4]. In these systems, many synchronized cameras were installed in a studio to capture the motion of the object such as dance or sports. Each frame in 3D video was generated independently frame by frame. Therefore, the geometry and topology in mesh models vary frame by frame, which is different from computer animation. Two characteristics of 3D video are that 3D video data are very large and the generation of 3D video is time-consuming and rather expensive.

Many technologies in 2D video have been developed to (semi)automatically edit the home video such as AVE [5]. In the professional field of film editing, video editing such as montage is necessary, which is manually implemented by experts. Similarly, 3D video editing will be useful and necessary to re-use 3D video due to the cost of 3D video generation. Another merit of 3D video editing is that some impossible motions for human beings can be generated by editing.

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 719–730, 2007.
© Springer-Verlag Berlin Heidelberg 2007

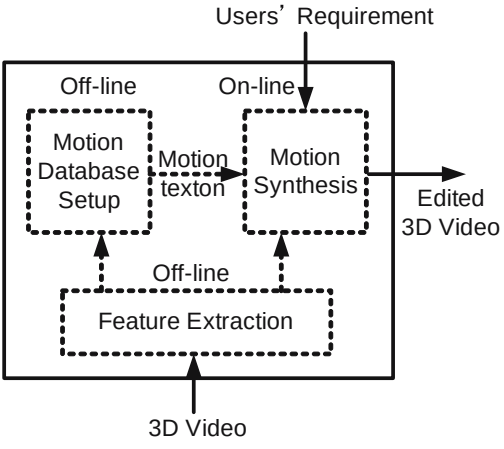


Fig. 1. Framework of our 3D video editing system

In this paper, a framework for editing 3D video is proposed as shown in Fig. 1. The feature vectors in [6], which is based on histograms of vertex coordinates, are adopted. Like video semantic analysis [7], several levels of semantic granularity are defined and parsed in 3D video. Then, we can construct the motion database by the parsed motion structure. Lastly, the user can edit the 3D video by selecting the motions in the motion database according to his/her wishes. A next motion is recommended, whose key frames are shown in the interface. The transition frames are optimized by a cost function. And the editing operation is on the motion level so that the user can edit 3D video easily.

2 Related Work

There are very few works on 3D video editing. Starck *et al.* proposed an animation control algorithm based on motion graph and a motion blending algorithm based on spherical matching in geometry image domain [8]. However, only genus-zero surface can be transfered into geometry image, which limits the adoption in our 3D video. Our previous work [9] presented a framework of motion editing in 3D video, which is similar to this work but much more simple. In this section, we mainly survey some works on 2D video editing and motion capture data editing.

The CMU Informedia system [10] was a fully automatic video editing system, which created video skims that excerpted portions of a video based on text captions and scene segmentation. Hitchcock [11] was a system for home video editing, where original video was automatically segmented into the suitable clips by analyzing video contents and the user dragged some key frames to the desired clips. Hua *et al.* [5] presented another video editing system for home video, where temporal structure was extracted with an *importance* score for a segment.

Besides 2D video editing systems, a number of algorithms on motion capture data editing have been proposed [12,13,14,15]. They can be classified into editing in spatial domain and temporal domain. In [12], Boulic *et al.* edited the

Table 1. The number of frames in 3D video sequences (10 frames per second)

	Person A	Person B	Person C	Person D
Walk	105	105	117	113
Run	106	107	96	103
BroadGym	1981	1954	1981	1954

motion capture data in spatial domain, where a prioritized Inverse Kinematics (IK) algorithm was proposed. The system presented by Chao *et al.* [13] was an example for editing in temporal domain, where the editing operation was done in several frames. Kovar *et al.* [14] proposed a concept called "Motion Graphs", which consisted of both original motion clips and generated transition clips. And a statistical method was proposed by Li *et al.* [15], where a synthesized motion was statistically similar to the original motion. Many other works have been done and surveyed by Geng *et al.* in [16]. Among these algorithms, some re-generated new motion capture data which were not contained in the original database and others only re-organized motion capture data in the database. The former is difficult and time-consuming for mesh models. Therefore, we will only consider the latter in this paper.

The works both in 2D video editing and motion capture data editing share some common characteristics in their systems. For example, it is necessary to segment the original video sequences and reassemble the basic units such as video clips by the user's requirements to realize his/her purposes. In this paper, we will also parse the motion structure in 3D video. It is also observed that the original data and the user's purposes have a great influence on the editing operations provided by their editing systems.

3 Feature Vector Extraction

Our 3D video sequences are generated in a 22-camera studio. Each frame in a sequence is stored in mesh model and has three types of information including vertex positions in Cartesian coordinate system, vertex connection in triangle edges, and the color attached to its corresponding vertex. Different from motion capture data, mesh model provides no structural information in spatial domain. Both the number of vertices and the topology change frame by frame in 3D video, thus 3D video has no corresponding information in temporal domain.

Our test sequences were generated from four persons as listed in Table 1. In these sequences, the number of vertices in a frame is about 16,000, the number of edges is about 32,000, and the number of colors is the same as vertices. "Walk" sequence is to walk for about 10 seconds. "Run" sequence is to run for about 10 seconds. And "BroadGym" sequence is to do the broadcast gymnastics exercise, which lasts about 3 minutes. Total time of these sequences lasts 872.2 seconds.

As mentioned above, 3D video has huge data without structural information in spatial domain or corresponding information in temporal domain, which makes geometry processing (such as model-based analysis and tracking) difficult and time-consuming. On the other hand, strong correlation exists in the statistical

point of view. Therefore, statistical feature vectors are preferred, which is the base of our system as shown in Fig. 1. We directly adopt the feature vectors in [6], where the feature vectors are the histograms of vertices in spherical coordinate system. A brief introduction is given as follows.

To find a suitable origin for the whole sequence, the vertex center of 3D object in (and only in) the first frame is calculated by averaging all the Cartesian coordinates of vertices in the first frame. Then, the Cartesian coordinates of vertices are transformed to spherical coordinates frame by frame by Eqs. (1)–(3) after shifting to new origin.

$$r_i(t) = \sqrt{x_i^2(t) + y_i^2(t) + z_i^2(t)} \tag{1}$$

$$\theta_i(t) = sign(y_i(t)) \cdot \arccos\left(\frac{x_i(t)}{\sqrt{x_i^2(t) + y_i^2(t)}}\right) \tag{2}$$

$$\phi_i(t) = \arccos\left(\frac{z_i(t)}{r_i(t)}\right) \tag{3}$$

where $x_i(t)$, $y_i(t)$, $z_i(t)$ are the Cartesian coordinates with the new origin, $r_i(t)$, $\theta_i(t)$, $\phi_i(t)$ are the coordinates for the i-th vertex of the t-th frame in the spherical coordinate system, and *sign* is the sign function, which is defined as

$$sign(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ -1 & \text{if } x < 0. \end{cases} \tag{4}$$

Then, the histograms of spherical coordinates are calculated. The feature vectors for a frame include three histograms for r, θ, and ϕ, respectively.

With the feature vectors, a distance is defined in Eq. (5), called a *frame distance* in this paper. The frame distance is the base of our algorithms.

$$d_f(t1, t2) = \sqrt{d_f^2(r, t1, t2) + d_f^2(\theta, t1, t2) + d_f^2(\phi, t1, t2)} \tag{5}$$

where $t1$, $t2$ are the frame ID in 3D video, $d_f(t1, t2)$ is the frame distance between the $t1$-th and the $t2$-th frames, and $d_f(\sigma, t1, t2)$ is the Euclidean distance between the feature vectors defined in Eq. (6).

$$d_f(\sigma, t1, t2) = \sqrt{\sum_{j=1}^{\max(J(\sigma,t1), J(\sigma,t2))} (h_{\sigma,j}^*(t2) - h_{\sigma,j}^*(t1))^2} \tag{6}$$

where σ denotes r, θ, or ϕ, $d_f(\sigma, t1, t2)$ is the Euclidean distance between histograms in the $t1$-th frame and the $t2$-th frame for σ, $J(\sigma, t)$ denotes the bin number of histogram in the t-th frame for σ, and $h_{\sigma,j}^*(t)$ is defined as

$$h_{\sigma,j}^*(t) = \begin{cases} h_{\sigma,j}(t) & j \leq J(\sigma, t) \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

where $h_{\sigma,j}(t)$ is the j-th bin in the histogram in the t-th frame for σ.

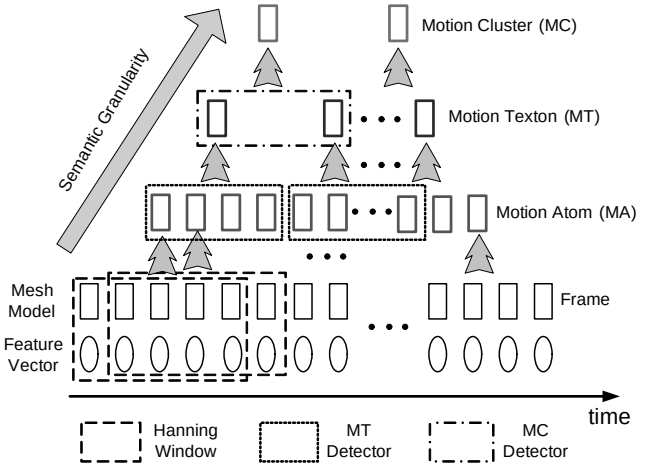


Fig. 2. Hierarchical motion structure in 3D video, a motion atom is defined as a number of successive frames with fixed length, a motion texton is a group of motion atoms, and a motion cluster is a group of motion textons

4 Motion Structure Parsing

Many human motions are cyclic such as walking and running. There is a basic motion unit which repeats several times in a sequence. More generally, such a basic motion unit will be transfered to another after several periods in a 3D video sequence such as from walking to running. Therefore, we define a basic motion unit as the term *motion texton*, which means several successive frames in 3D video which form just a complete periodic motion. And several repeated motion textons will be called a *motion cluster*, which is a group of repeated motion textons. Thus, 3D video is composed of some motion clusters, and a motion texton is repeated several times in its motion cluster. This is the motion structure of our 3D video sequences as shown in Fig. 2.

An intuitive unit to parse the motion structure is a frame. However, motion should include not only the pose of the object but also the velocity and even acceleration of motion. For example, two similar poses may have different motions with inverse orientations. Therefore, we have to consider several successive frames instead of only a frame. As shown in Fig. 2, *motion atom* is defined as some successive frames in a fixed-length window, which are our unit to parse the motion structure. Another benefit from motion atom is that some noise, which may come from the fact that the similar histograms may have different vertex distributions (i.e., histograms are global descriptions of the content), can be alleviated by considering several successive frames. The hierarchical structure is not a new idea. It is popular in text/speech/video processing [7]. Some abbreviations will be used in this paper: motion atom will be briefly called as atom or MA, motion texton as texton or MT, and motion cluster as cluster or MC.

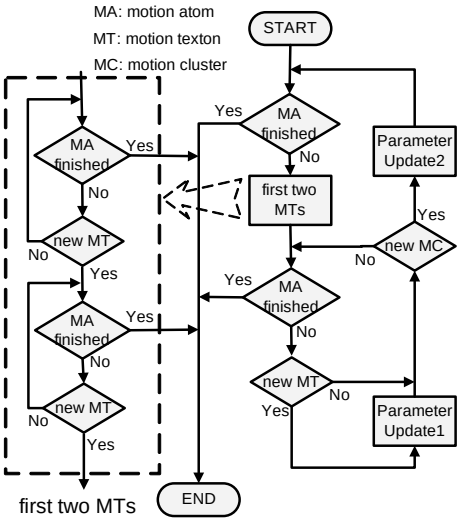


Fig. 3. Motion structure parsing procedure in 3D video, left: the detail of *first two MTs*, right: the whole procedure

To parse this motion structure, we have to detect the boundaries of motion textons and motion clusters. The main idea to detect motion textons is that the motion atom will be similar when the motion texton is repeated. And the main idea to detect motion cluster is that there should be some motion atoms which are very different from those in the previous motion cluster. Therefore, an *atom distance* is defined to measure the similarity of two motion atoms in Eq. (8).

$$d_A(t1, t2, K) = \sum_{k=-K}^{K} w(k) \cdot d_f(t1 + k, t2 + k) \tag{8}$$

where $w(k)$ is a coefficient of a window function with length of $(2K+1)$. $t1$ and $t2$ are the frame ID of the atom centers, which show the locations of motion atoms with $(2K + 1)$ frames. $d_A(t1, t2, K)$ is the atom distance between the $t1$-th and the $t2$-th atoms. In our experiment, a 5-tap Hanning window is used with the coefficients of $\{0.25, 0.5, 1.0, 0.5, 0.25\}$. From now on, we will simplify $d_A(t1, t2, K)$ as $d_A(t1, t2)$ since K is a fixed window length.

Figure 3 shows the procedure of motion structure parsing, where the dash rectangle is the detail of process of *first two MTs*. To utilize the motion atom effectively, the first texton will begin from an active motion atom which satisfies Eq. (9) because the object has no motion at the beginning in many cases.

$$d_f(t, t + 1) < d_f(t - 1, t) \quad \text{and} \quad d_f(t - 1, t) > \alpha \tag{9}$$

where α is a threshold and set as 0.04 in our experiment.

Process of *first two MTs* in Fig. 3 is to find the first two motion textons in a motion cluster by decision of *new MT*. Here suppose there are at least two motion textons in each motion cluster. The motion atoms in the two motion

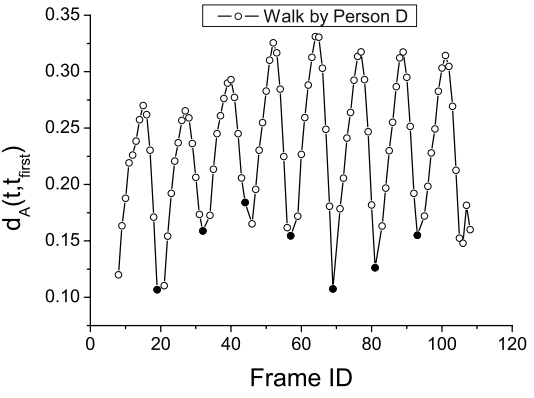


Fig. 4. Atom distance $d_A(t, t_{first})$ from the first atom in its motion texton in "Walk" sequence by Person D, the black points denote the first atom in a motion texton

textons will be used as the initial reference range $[t_{inf-C}, t_{sup-T}]$ in Eq. (11) in decision of *new MC*.

Decision of *new MT* is to decide if a new motion texton is detected. The atom distance $d_A(t, t_{first})$ between the current atom (t) and the first atom (t_{first}) in the current motion texton is calculated. Then, if $d_A(t, t_{first})$ reaches a local minimum (that means the motion atom may repeat) and the difference between the maximum and minimum in the current motion texton is large enough (since unavoidable noise may cause a local minimum), a new motion texton is detected. Figure 4 shows the atom distance $d_A(t, t_{first})$ between the first atom and current atom in "Walk" sequence by Person D, which reflects the texton is repeated. A distance in Eq. (10) is then defined as *texton distance*, which is the atom distance between the first and last atom in the texton.

$$d_T(T_i) = d_A(t_{last}, t_{first}) \tag{10}$$

where $d_T(T_i)$ is the texton distance for the i-th texton, t_{first} is the first atom in the i-th texton, and t_{last} is the last atom in the i-th texton.

Decision of *new MC* is to decide if a new motion cluster is detected. A minimal atom distance will be calculated as Eq. (11), which tries to find the most similar atom in the reference range $[t_{inf-C}, t_{sup-T}]$.

$$d_{min}(t, t_{inf-C}, t_{sup-T}) = \min_{t_{inf-C} \le t_k \le t_{sup-T}} d_A(t, t_k) \tag{11}$$

where t_{inf-C} is the first motion atom in current motion cluster, which is updated when detecting a new motion cluster in process of *parameter update2*. t_{sup-T} is the last motion atom in previous motion texton, which is updated when detecting a new motion texton in process of *parameter update1*. t_{inf-C} and t_{sup-T} are initialized in process of *first two MTs*.

Then, if two successive motion atoms have large minimal atom distances as Eq. (12), a new motion cluster is detected. We adopt two successive atoms instead

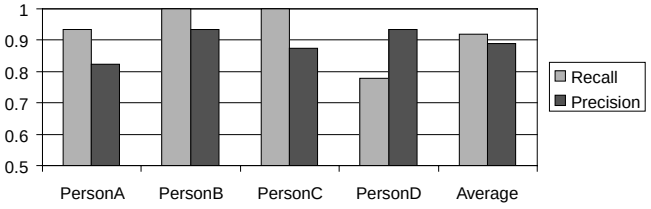


Fig. 5. Precision and recall for motion cluster detection in "BroadGym" sequences

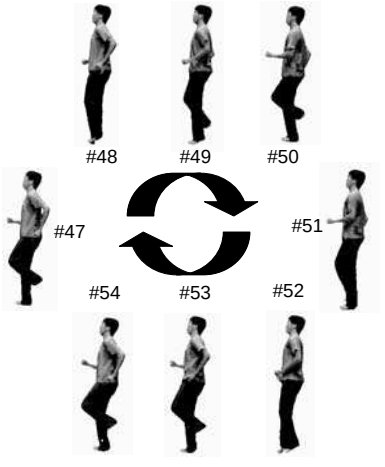


Fig. 6. Selected motion texton in "Run" by Person B

of one atom to avoid the influence of noise. High precision and recall for motion cluster detection are achieved as shown in Fig. 5.

$$d_{min}(t-1, t_{inf-C}, t_{sup-T}) > \beta \quad \text{and} \quad d_{min}(t, t_{inf-C}, t_{sup-T}) > \beta \qquad (12)$$

where β is a threshold and set as 0.07 in our experiment.

5 Motion Database

From now on, the basic unit of analysis will return to frame distance as shown in Eq. (5) since we consider the selected textons instead of the whole sequence. In Section 4, the hierarchical motion structure is parsed from the original 3D video sequences. Since the motion textons are similar in a motion cluster, we only select a representative motion texton into our motion database to reduce the redundant information. The requirement of the selected motion texton is that it should be cyclic or it can be repeated seamlessly so that the user can repeat such a motion texton many times in the edited sequence. Therefore, we select the motion texton with the minimal texton distance as shown in Eq. (13).

$$T_i^{opt} = \arg_{T_i \in C_j} \min d_T(T_i) \qquad (13)$$



Fig. 7. Interface captured from the prototype of motion editing in 3D video

where T_i and C_j are the current motion texton and motion cluster. $d_T(T_i)$ is the texton distance for the current texton, defined in Eq. (10). T_i^{opt} is a representative texton. Figure 6 shows an example of selected motion texton, where we can see the motion texton is almost self-cyclic.

The recommended texton should be able to be transited smoothly from the current texton, which should have the minimal inter-texton distance as Eq. (15).

$$d_T(T_{i1}, T_{i2}) = \min_{t_1 \in T_{i1}, t_2 \in T_{i2}} d_f(t_1, t_2) \tag{14}$$

$$T_{i2}^{opt}(T_{i1}) = \arg_{T_{i2} \in \Gamma} \min d_T(T_{i1}, T_{i2}) \tag{15}$$

where $d_T(T_{i1}, T_{i2})$ is the distance between the motion textons T_{i1} and T_{i2}. t_1 is a frame in motion texton T_{i1} and t_2 is defined similarly. $T_{i2}^{opt}(T_{i1})$ is the recommended next texton for T_{i1}. Γ is the whole motion database but T_{i1}. Two corresponding frames are marked as the transition frames between the textons T_{i1} and T_{i2}^{opt}. The optimization is to find a texton T_{i2}^{opt} that can be smooth transition from T_{i1}. To extract the key frames for the recommended texton, we adopt a similar method as [17], where the trade-off between rate and distortion is achieved. In the interface, those key frames are displayed.

6 Motion Editing

After constructing the motion database, the user can edit 3D video by selecting any motion of any object. The issue in this section is to find the transition frames

between the current texton and next texton which the user selects. Also, a simple interface prototype is implemented to demonstrate the motion editing system.

6.1 Interface Prototype

The interface is difficult to design because we have a large motion database and our 3D video sequences have four dimensions to display. In this paper, a simple interface is realized using OpenGL library. Figure 7 is captured from screen. The current texton is played frame by frame and the key frames of recommended next texton is displayed on top (due to the space limitation, only the first three key frames are displayed). At the bottom, the names of current object and motion are displayed in real time with the frame ID and frame rate. The other information is displayed in a DOS window. The user will select the motion by an configuration file and the keyboard. For example, the user can set the first motion in the configure file. Also, the user can press an "R" to select the recommended next texton or a "W" for "Walk" texton if they want. Some mouse functions are provided to change the viewpoint of object, the scale of object, and so on. The frame ID of transition frames are stored in a trace file.

6.2 Motion Transition

There are two kinds of transitions, namely transitions in motion texton (called *intra-transition*) and between motion textons (called *inter-transition*). When constructing the motion database, those selected textons are considered to have a smooth intra-transition as mentioned in Section 5. Our requirement for inter-transition is that the transition between two textons should be smooth and timely. According to his/her wishes, the user may select the recommended next texton or not. If the recommended texton is selected, the transition frames are known by Eq. (14). Otherwise, a cost function is optimized. *Smoothly* means the mesh models at transition points are as similar as possible, or the frame distance of transition frames is as small as possible; and *timely* means the transition will be as fast as possible after the user gives the commands, or the frame ID difference between the current frame and transition frame will be as small as possible. Therefore, the cost function is defined as Eq. (16).

$$cost(t0, t1, t2) = \mu \cdot \ \|t1 - t0\| + d_f(t1, t2) \tag{16}$$

$$\{t1, t2\}^{opt} = \arg_{t1 \in T_i, t2 \in T_j} \min cost(t0, t1, t2) \tag{17}$$

where μ is a weight to balance the two requirements (empirically set as 0.001 in our experiments), $t0$ is the frame ID when the user gives a command, $t1$ is the transition frame ID in the current motion texton, $t2$ is the transition frame ID in the next motion texton which is selected by the user, T_i is the current motion texton, T_j is the next motion texton selected by the user, $d_f(t1, t2)$ is calculated by Eq. (5). The optimized transition frames $\{t1, t2\}^{opt}$ will depend on the current frame $t0$ too, that is to say, different current frames may have different transition frames even if the user gives the same commands.

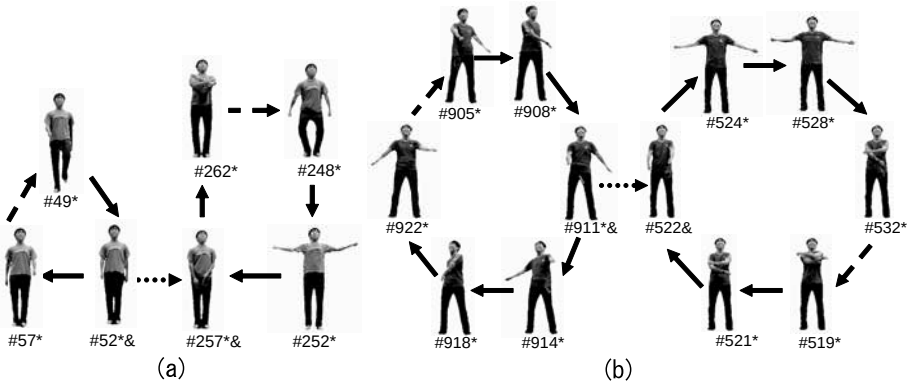


Fig. 8. Transitions to next motion texton, only key frames or/and transition frames are displayed; the dash arrow denotes the intra-transition, the round dot arrow denotes the inter-transition; the asterisk (*) beside the frame number means the key frame and the symbol (&) means the transition frame; (a) recommended texton by Person B; (b) non-recommended texton by Person D

Figure 8 (a) shows the experimental result where the user selected the recommended texton and Fig. 8 (b) shows the result where the user selected a non-recommended texton. Intra-transition is shown by the dash arrow and inter-transition is shown by round dot arrow in Fig. 8. The experiments demonstrate the effectiveness of our system.

7 Conclusions and Future Work

In this paper, we have demonstrated an efficient framework of motion editing to re-use 3D video, where the user can select any motion of any object in our motion database. For this purpose, we have proposed a method to parse the hierarchical structure in 3D video to construct the motion database. Since the user edited 3D video on motion level instead of frame level, it was easy for the user to synthesize a 3D video sequence. Motion transition was optimized by a cost function to generate a smooth and timely transition. Some other information such as a recommended texton is also provided in the interface. Although the feature vectors are the base of our system, our algorithms are rather flexible. They can easily be transfered to other feature vectors such as [18] and even other media such as 2D video if only the frame distance in Eq. (5) is well defined.

Motion editing is a powerful tool to re-use 3D video. A lot of improvements can be done in the near future. The current interface requires the user to remember the commands to change the motion or object. So a more friendly interface is preferable such as displaying the contents of motion database by key frames. On the other hand, more constraints and functions are useful in some applications. For example, the object is expected to walk at some time. In addition, motion blending between the transitions with large distances will be helpful to edit a smooth 3D video sequence as Kovar *et al.* [14] did. Another future research issue

is the management of motion database. In a large motion database, it is better to classify it into subsets by the motion genre or other criteria.

Acknowledgments

This work is supported by the Ministry of Education, Culture, Sports, Science and Technology, Japan within the research project "Development of fundamental software technologies for digital archives". The generation studio is provided by NHK, Japan. And the volunteers are greatly appreciated to generate 3D video.

References

1. Kanade, T., Rander, P., Narayanan, P.: Virtualized reality: constructing virtual worlds from real scenes. In: IEEE Multimedia. Vol. 4, No. 1, (1997)34–47
2. Tomiyama, K., Orihara, Y., et al.: Algorithm for dynamic 3D object generation from multi-viewpoint images. In: Proc. of SPIE. Vol. 5599, (2004)153–161
3. Wurmlin, S., Lamboray, E., Staadt, O. G., Gross, M. H.: 3D video recorder. In: Proc. of Pacific Graphics'02. (2002)325–334
4. Carranza, J., Theobalt, C., Magnor, M. A., Seidel, H. P.: Free–Viewpoint Video of Human Actors. In: SIGGRAPH 03. Vol. 22, No. 3, (2003)569–577
5. Hua, X., Lu, L., Zhang, H. J.: AVE-Automated Home Video Editing. In: Proc. of ACM Multimedia 03. (2003)490–497
6. Xu, J., Yamasaki, T., Aizawa, K.: Histogram-based Temporal Segmentation of 3D Video Using Spherical Coordinate System. In: IPSJ Trans. on Computer Vision and Image Media. Vol. 47, No. SIG 10, (2006, in Japanese)208–217.
7. Xu, G., Ma, Y. F., Zhang, H. J., Yang, S. Q.: An HMM–Based Framework for Video Semantic Analysis. In: IEEE Trans. on CSVT. Vol. 15, No. 11, (2005)1422–1433
8. Starck, J., Miller, G., Hilton, A.: Video-Based Character Animation. In: ACM SCA'05. (2005)49–58
9. Xu, J., Yamasaki, T., Aizawa, K.: Motion Editing in 3D Video Database. In: 3DPVT'06. (2006)
10. Christel, M., Winkler, D., Taylor, R., Smith, M.: Evolving Video Skims into Useful Multimedia Abstractions. In: CHI'98. (1998)171–178
11. Girgensohn, A., Boreczky, J., et al.: A Semi-Automatic Approach to Home Video Editing. In: ACM UIST'00. (2000)81–89
12. Boulic, R., Callennec, B. L., Herren, M., Bay, H.: Experimenting Prioritized IK for Motion Editing. In: EUROGRAPHICS 03. (2003)
13. Chao, S. P., Chiu, C. Y., Chao, J. H., Yang, S. N., Lin, T. K.: Motion Retrieval and Its application to Motion Synthesis. In: ICDCS'04. (2004)254–259
14. Kovar, L., Gleicher, M., et al.: Motion Graphs. In: SIGGRAPH 02. (2002)473–482
15. Li, Y., Wang, T., Shum, H. Y.: Motion Texture: A Two-Level Statistical Model for Character Motion Synthesis. In: SIGGRAPH 02. (2002)465–472
16. Geng, W., Yu, G.: Reuse of Motion Capture Data in Animation: A Review. In: ICCSA'03. (2003)620–629
17. Xu, J., Yamasaki, T., Aizawa, K.: Key Frame Extraction in 3D Video by Rate-Distortion Optimization. In: IEEE ICME'06. (2006)1–4
18. Xu, J., Yamasaki, T., Aizawa, K.: 3D Video Segmentation Using Point Distance Histograms. In: IEEE ICIP'05. (2005)I-701–I-704

Tamper Proofing 3D Motion Data Streams

Parag Agarwal and Balakrishnan Prabhakaran

Department of Computer Science, University of Texas at Dallas
MS EC 31, P O Box 830688, Richardson, TX 75083
parag.agarwal@student.utdallas.edu, praba@utdallas.edu

Abstract. This paper presents a fragile watermarking technique to tamper proof (Mocap) motion capture data. The technique visualizes 3D Mocap data as a series of clusters of points. Watermarks are embedded using clusters of points, where a bit is encoded in each cluster. The four point encoding mechanism uses a combination of one point encoding and three point encoding schemes. Using these schemes it is possible to distinguish between affine transformations, noise addition and reverse ordering attacks. The bits are encoded and decoded in this scheme using an extension of quantization index modulation. It has been shown that distortions are reduced to achieve imperceptibility of the watermark. The bit encoding schemes give the flexibility to achieve better accuracy in tamper detection. In addition, the paper suggests a probabilistic model, which is a function of the watermark size. Using this model, it has been proved that larger watermark sizes achieve higher accuracy in tamper detection.

Keywords: Tamper proofing, encoding, decoding, motion, data, watermarking.

1 Introduction

The advent of Motion Capture systems [10] has brought in applications like animation (games, films & TV, education), and life sciences (biomechanical research, gait analysis, rehabilitation, posture, balance and motion control, sports performance). The above applications deal with motion analysis or reusability, and can benefit from having a large repository of 3D human motions. In cases, where data is tampered, its integrity is lost, and we incur losses in terms of accuracy, effort, time and money. Tampering can be avoided by a data authenticating mechanism, such as fragile watermarking. Fragile watermarking can be achieved by embedding a watermark inside a target data. Tampering is recognized whenever during an extraction process if sub-part of the embedded watermarks is found corrupt. A fragile watermarking technique for motion data poses the following challenges:

- **Distortions in meaning of data:** Addition of watermarks distorts the original data. This changes the meaning of the data set. The visibility of distortions due to change in the meaning of motion data will fail the imperceptibility criteria of the watermarking scheme.
- **Accuracy of detection:** Data set can be attacked using motion editing operations, such as noise addition, reordering, and uniform affine transformations. As a result, the meaning of the data set at can change different locations. The watermarking methodology should be able identify the attack, and detect the change accurately.

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 731 – 740, 2007.
© Springer-Verlag Berlin Heidelberg 2007

Accuracy can be achieved by replicating the watermark at different locations in the data set. However, replication can induce more distortions resulting in loss of meaning of the data. Distortions will discourage the replication of the watermark, and eventually impact the accuracy of the technique. In addition, the fragile watermark technique needs to be storage efficient. Storage efficiency can be achieved by reducing the information required to verify the presence of watermark.

This can be achieved by blind watermarking mechanism, which uses the watermark and key to embed and extract the watermark. To the best of our knowledge, there is not a single technique that can solve all the problems for 3D motion data.

1.1 Related Work

Fragile watermarking techniques [3, 7] can be described as spatial and transform domain. Transform domain operates on the frequency components of the subset of the data, as compared to the spatial techniques that operate on the original data. Several authentication based approaches using watermarks have been proposed for images [3, 5, 7], audio [2], video [8] and 3D models [4, 6]. However, none of the approached are generalized enough to be applied to motion data. To the best of our knowledge there is no work done in this regards for fragile watermarking motion data streams. A non-watermarking method [9], explains a scheme of extracting reference code, which is used to detect tampering. However, this technique requires extra storage, which makes the scheme not scalable. Therefore, it is eminent that we need a novel scheme to tamper proof motion data.

1.2 Proposed Approach and Contributions

The paper proposes a 'spatial' mechanism to blind (fragile) watermark 3D time-series motion data by visualizing time series data as a cluster of points. Each cluster has four points, and encodes a single bit. The encoding/decoding mechanism can detect and localize any change to the data set. The contributions of the paper are listed as follows:

- **Reduction in distortions and increase in accuracy of tamper detection:** The technique reduces distortion by adaptively encoding, and imperceptibility of the watermark. It has been shown that the accuracy of detection can be designed as a probabilistic model, which is a function of the watermark size. Based on this scheme and the bit encoding scheme, it is shown to detect affine transformations, noise additions, and reordering with a high 99% of accuracy.
- **Extensions to a bit encoding scheme:** The method uses a 4-Point encoding scheme to embed a bit in the four points per cluster. This is achieved by a combination of 1-Point and 3-Point encoding schemes, which are extensions of quantization index modulation [2]. The usage of this encoding enables the detection of affine transformations, noise addition and reverse ordering attacks.

2 Scheme Design

Human body comprises of 19 joints (see Fig 1), and their motion can be captured using motion capture technology. Fig 2 shows an example motion of the hand joint

represented by positional data (x, y, and z) that can be represented in logical time. This information varies in *logical time* (frames or samples ordered sequentially). The varying joint positional information represents the 3D motion data streams. We can represent this 3D motion data as a time series data, which can be given in a matrix (see Fig 3). Mathematically this matrix (see Fig 3) can be defined as $(D_{m \times 3}) = [D_i]^T$, $1 \leq i \leq m$, where sample data-set $D_i = <X, Y, Z>$ and m – number of samples, satisfying the following properties 1) $Xi_p \rightarrow Xi_q$ ($\rightarrow$ 'happens before'), ($p < q$), and 2) Xi_p is correlated to Xi_q.

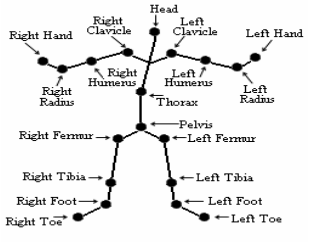


Fig. 1. Human Body Joints

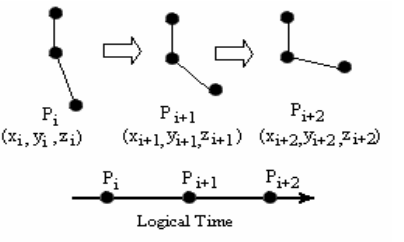


Fig. 2. Motion of hand joint

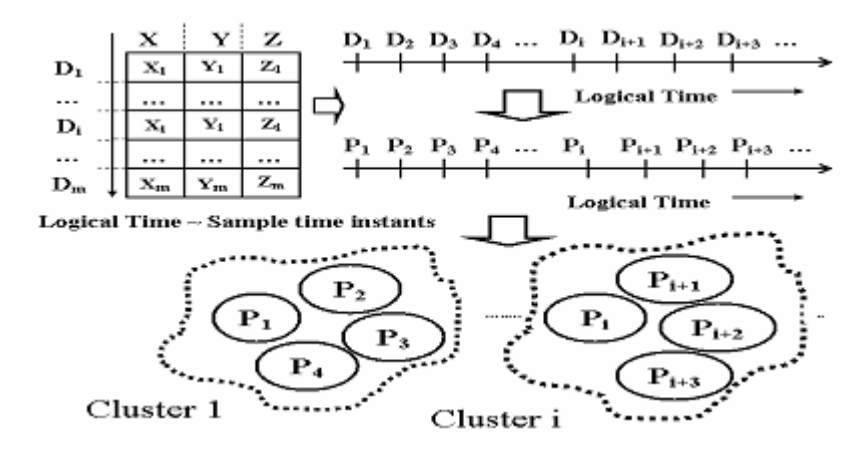


Fig. 3. Visualization of 3D motion data as point clusters

We propose to watermark each joint separately, since alteration to the data-set can be tracked on a per joint basis. Encoding inside each joint is done by visualizing it as clusters of 4 points. It can be observed that D_i can be represented as point (P_i) in 3D space (see Fig 4). Since D_i (s) are already totally ordered with respect to time, we can say that points P_i and P_j $(i < j)$ are adjacent to each other. We can identify non-intersecting cluster of points whose sizes are multiples of four. Bits are embedded inside the clusters by identifying four points at a time. For each of the four points given, the same bit is encoded using two set approach, where in fist set we have one point, and the other set the remaining three points. One point and three point encoding is done using an extension of quantization index modulation [2]. The following sections describe the technique in detail.

2.1 Custer Based Encoding

Each cluster is of size 4*k* points or samples (where $k_{min} = 1$), and can encode '*k*' bits. For simplicity we assume cluster size = 4. These bits are encoded in the sequence (or order) in which the points occur. For example, to encode each bit we use four points $\{P_i, P_{i+1}, P_{i+2}, P_{i+3}\}$, where first point subset $\{P_i,\}$ is taken for 1-Point encoding, and the subset is used for $\{P_{i+1}, P_{i+2}, P_{i+3}\}$ 3-Point encoding. We can visualize this situation in Fig 4, where points $\{P_1, P_2, P_3, P_4\}$, represented as vectors are shown. The following explanation is used to abstract the idea of encoding for 3-Point and 1-Point.

3-Point Encoding: It can be observed that uniform affine transformation preserves *proportions* of lines. The lines can be described as scalar quantities q_1 and q_2, and the ratio (q_1/q_2) must be invariant to affine transformations. The scalar q_1 and q_2 can be realized using three points represented as two vectors, as shown in Fig 5. When we consider three points (P_2, P_3 and P_4), we have can have two vectors such that one point is common among these vectors. The magnitudes of these vectors give us two scalars (*Euclidian distances between the points*) whose ratio stays invariant to affine transformations.

$$\text{B}(q_1, q_2) = bit, \text{ where } bit = \{0, 1\}. \tag{1}$$

In order to encode the bit information inside a data set, we use the function 'B (q_1, q_2)' whose inputs are the two scalar quantities. A bit can be encoded by observing the output of the equation (1). In case, a required bit condition is not met, as shown in Fig 5), where the expected bit = 1. To handle this case, we substitute the point P_3 by another point to encode the expected bit, resulting in change in scalar quantity $|P_2P_3|$.

Handling Pathological Cases: There can be cases where points in the data set have same values i.e. points P_i and P_j belong to the same cluster and have equal values. For such cases, we have to choose among these points for encoding, which may result in contention. This situation can be avoided either by excluding these points from the encoding process or by perturbing them to achieve an encoding.

A single point P_1 in 3D space can be visualized as a vector (see Fig 4) which results in only one scalar – magnitude (in Fig 4, $q = |P_1|$), which is variant to affine transformation.

1-Point Encoding: In this encoding scheme a bit is encoded inside a single point by subjecting values *x*, *y* and *z* to be perturbed in order to encode '1' or '0'.

$$\text{Bit}(q) = b, b = \{0, 1\}. \tag{2}$$

A bit can be encoded by observing the output of the equation (2). In case a required bit condition is not met, as shown in Fig 5, where the expected bit = 1. To handle such a case, we substitute the point P_1 by another point to encode the expected bit, resulting in change in scalar quantity $|P_1|$.

The functions represented in equations (1 and 2) can be implemented using a bit encoding scheme, which uses quantization index modulation (QIM) [2]. QIM has also

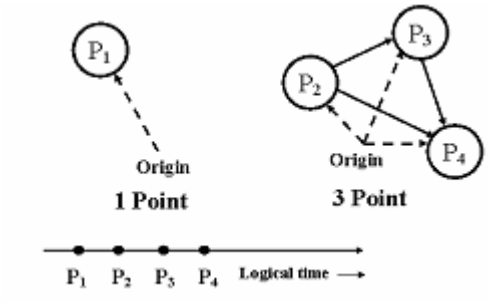


Fig. 4. Vector representations of points

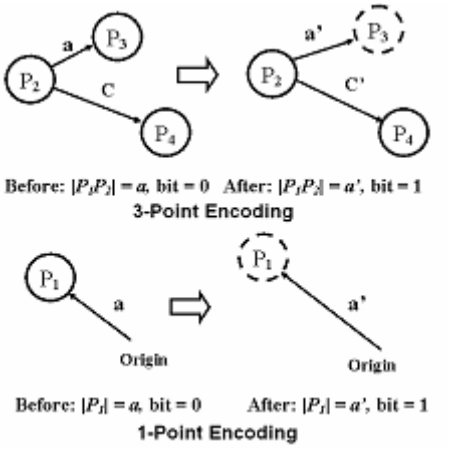


Fig. 5. Encoding Strategies

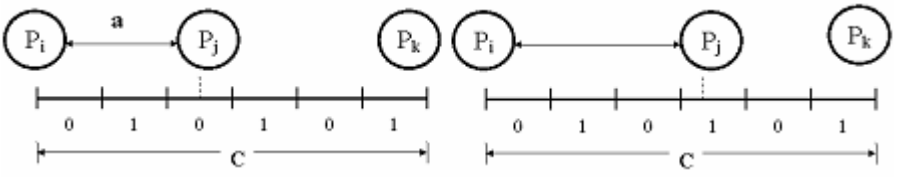


Fig. 6. Logical representation of bit encoding applied to points P_i, P_j, and P_k

been customized to watermark 3D Models for tamper proofing [7, and 9]. The customization of QIM in our case is explained as follows:

Bit Encoding Scheme. As shown in Fig 6, we take two points (P_i and P_k), the Euclidian distance '*C*' between them stays invariant. However, the distance '*a*' between P_i and P_j is variant, and changes due to encoding as explained below.

The scalar '*C*' can be divided into '*p*' number of equal intervals, where $p_{min} = 2$. The interval set of 0(s) is identified by S_0 and 1(s) by S_1, (S_i: $i \in \{0, 1\}$). Let *Pos* (P_j): position of P_j on side '*C*'. The bit '*i*' can be determined based on the position of P_j on side '*C*' using the following rules:

- *Pos* (P_j)$\in$ S_i: No modifications required
- *Pos* (P_j) $\notin$ S_i: *Pos* (P_j) has to be shifted to a *Pos* (P_j') so that *Pos* (P_j') $\in$ S_i

As observed in Fig 8 (a&b), which is a case, where the bit to encode is '1', but P_j lies on an interval where the corresponding bit is '0'. As a consequence, we need to shift the position of P_j, which results in change of the scalar $|P_i P_j|$ from (*a* to *a*').

The above explanation assumes scalars as length of vectors represented in Fig 5, and can be customized for 1-Point and 3-Point encoding as shown below:

Customization for 1-Point Encoding: We can visualize P_i as the '*origin*' in Fig 6, and P_k as an assumed point for which scalar '*C*' is fixed. Point P_1 is equivalent to

point P_j, and its scalar representation is '*a*'. The function *Bit* (*q*) is implemented by assuming ($q = a$) and the bit information is determined by *Pos* (P_j).

Customization for 3-Point Encoding: We can visualize P_i as the P_2 in Fig 6, and P_k as P_4. The function *Bit* (q_1, q_2) is implemented by assuming have '$q_1 = C$' and '$q_2 = a$', and Pos (P_j) determines the output of the function.

In both the cases, change in P_j (P_1 for 1-Point and P_3 for 3-Point, see Fig. 5) is determined by the function Pos (P_j). For 3-Point technique, we observe that the ratio $q_1 = |P_i P_j|$, and $q_2 = |P_i P_j|$, which implies that once the bit is encoded it is invariant to affine transformation, since (q_1/q_2) is invariant. The intervals on the scalar '*C*' have significance in making the technique sensitive to tampering, and imperceptibility of the watermark.

The following ideas are important in customizing the scheme to data sets, in order to control distortions and accuracy of tamper detection.

Controlling Distortions. Smaller interval size implies that P_j has to be displaced by a smaller amount during the bit encoding process. This will reduce the distortion during embedding process, which would make the watermark more imperceptible. During encoding, we need to choose an interval size, and number of such intervals to define the scale along which a single point is encoded. The size of the scale is given by the equation (3).

$$Scale = Interval\ Size * Number\ of\ intervals. \tag{3}$$

For 3-Point encoding the scale is determined by an invariant side $|P_iP_k|$, and the interval size can be determined from equation (3), by dividing the scale by the number of intervals. In case of 1-Point encoding, we derive the scale based on the choice of interval size and number of intervals, and this choice is explained as follows:

Interval Size Selection: In cases, where the interval size is greater than scalar quantities related to points, the points would be in the first interval only, and would have a bit '0' (see Fig 6). Since bits can be either '0' or '1', in cases '1' is required to be encoded, distortion will become more probable. Therefore, in order to avoid such a situation, the size (see equation (4)) should be less than or equal to the minimum size of a scalar quantity of all the points being encoded.

$$Interval\ Size \quad Min\ (|Qi|),\ Qi \in\ Data\ Set. \tag{4}$$

Number of Intervals Selection: In cases, where scalar quantities related to points are greater than the scale, the bit related to them is always the same. Since bits can be either '0' or '1', in cases '1' is required to be encoded, distortion will become more probable. Therefore, in order to avoid such a situation, the size of the scale should be greater than the largest size of the scalar quantities. To guarantee the same, the number of intervals can be determined by equation (5).

$$\text{Number of intervals} > \frac{Max(|\ Qi\ |)}{IntervalSize},\ Qi \in\ Data\ Set. \tag{5}$$

Since we are encoding, the maximum or minimum scalar quantities, as given in equations (4) and (5) might change. As a result, decoding might give us different bits, resulting in erroneous condition. To avoid this situation, for equation (5), number of

intervals is infinitely large, implying that the scale is be infinity (a very large number for practical usage). The interval size in equation (4) can be based on a point, where encoding is not enforced. To guarantee that any other point during encoding does not substitute this point, we always increment the scalar quantities for encoding. Also, if there are several similar points that are fit for interval size criteria, we take only the first occurrence. The choice of this point has the added advantage of tamper detection. Since in cases, where this point is perturbed, the interval size is changed, resulting in flipping of encoded bits for other points. This fact will act as a discouraging factor for the adversary to change this point. The above factors will help tailor the encoding scheme to adapt to the data set given, and would be beneficial in reducing the distortions.

Accuracy in Tamper Detection. Any slight displacement to point P_j (see Fig 6) to another interval might toggle the bit encoded. In cases where interval size is large, the probability is less since a larger displacement is required to change the bit. This could result in false negatives, and can be avoided by reducing the size of intervals. Therefore, smaller interval lengths can test any changes to the bit encoded in 1-Point or 3-Point scheme. However, in cases where intervals are smaller, any change resulting from tampering may cause the point P_j to be shifted into an interval which has the same bit. As a result, we cannot detect tampering which leads to false negatives. Such effects are more probable if the interval size is reduced, since the probability of presence in '*n*' intervals is '*1/n*' which decreases as '*n*' increases.

Watermark Size Based Probabilistic Model: In cases where all the bits do not change, a watermark of size '*WSize*' can be detected. The probability of detection can be expressed as 'p^{WSize}', where '*p*' is the probability that bit has not been flipped. The probability of failure to detect a watermark (P_f) can be expressed as equation (6).

$$P_f = 1 - p^{WSize} \tag{6}$$

Since ($p < 1$), from equation (6), it can be inferred that the probability of failure is least when watermark size is equal to '1'. In addition, we also observed that by increasing the size of the watermark size, we increase the likelihood to detect a watermark. This is a positive sign for tamper detection, since loss of watermark implies tampering, and this can be concluded with a high probability. Once it is confirmed the watermark has been tampered, we can localize the search to the bit that has changed. As shown above, the change in bit information depends on the false negatives during an attack, since the shift in point location could result in the same bit. It can be concluded from the above discussion that although the false negatives might be present due the small sized intervals, we can increase our chances of tamper detection by choosing a watermark of sufficient size. We can also conclude that larger the number of scalar quantities used to encode a bit, more accurate is the detection, as\number of scalar quantity is equal to watermark size.

3 Decoding and Verification

The decoding process is similar to encoding process. Verification is done by identification change in the bit information. A change in the bit information (1-Point or 3-Point encoding) is reflected as a compliment of the bit. This helps in identifying the location of the attack. In order to identify the attacks, the following rules can be followed:

Affine attack rule: An affine attack is identified by a change in the 1-Point bit, and no change in the 3-Point bit. This is because the 3-Point encoding is robust to affine transformation, as compared to 1-Point encoding.

Noise addition rule: These attacks can be identified by change in both 1-Point and 3-Point encoding scheme.

Reversing order attack rule: In this case the order of occurrence of points is reversed in logical time. This can be identified by detecting information in the reverse time. In addition, it will change the bit information for 1-Point and 3-Point encoding as well.

Combined attack rule: The above mentioned attacks can be launched at the same time, resulting in all indications.

4 Experiments and Results

The watermarking scheme has been implemented in Matlab 7.0.4, and applied to data samples that were collected from University of Texas at Dallas - Motion Capture lab (Vicon [10]). The experiments were done on a motion matrix, which is a dance sequence with (4286 frames captured at 120 frames/sec for 19 joints = 81434 points or samples). The model is justified for performance analysis, since it consists of joints moving in varying directions at different time intervals, thus giving diversity in motion data. Other motion data such as karate actions, exercise, and walk are used to analyze accuracy and distortions. The experiments in this section increase the interval size or increase the number of intervals. Both operations imply the same purpose. The performance is measured according to the following metrics:

Signal to Noise Ratio (SNR) can measure the distortion produced. Higher the distortion less imperceptible is the watermark. SNR for 3D motion data is calculated in equation (7)

$$\text{SNR}\,(M, M') = 20\,Log_{10}\frac{RMS(M)}{RMS(M - M')}\ \text{(RMS ~ root mean square)} \tag{7}$$

Detection Rate (DR) can be a measure of the accuracy with which tampering is detected. It is defined as the ratio of number of error detected to the total number of possible errors. DR is primarily equivalent to the probability of failure (P_f), as given in equation (6), as it measures the failure to detect the watermark.

4.1 Performance Analysis

The following subsections give a performance analysis of the scheme for different parameters of the scheme.

Distortion Analysis. It can be observed from Table 1 that for different motion types we have distortion > 100 dB. In order to analyze distortion for 1-Point and 3-Point encoding, we analyze them separately as follows:

1-Point Encoding: The data being subject to watermark varies in the range [1 900]. An attempt to encode a watermark inside it sees a distortion maximized (see Sub-section 2.1) when interval size is '>' 900. This can be observed in Fig 7, where the

interval size = 1000 has maximum distortion. By increasing the interval size the distortion decreases.

3-Point Encoding: As observed in Sub-section 2.1 reduction in interval length results in reduction in distortion (see Fig 7), and it becomes consistent for the range (74 to 75 dB), after the number of intervals is increased '>' 5.

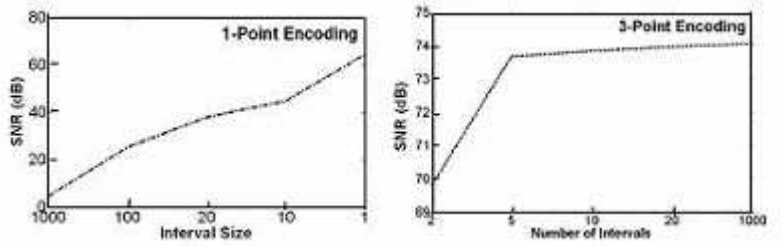


Fig. 7. Impact of interval Size on distortions due to 1-Point & 3-Point encoding

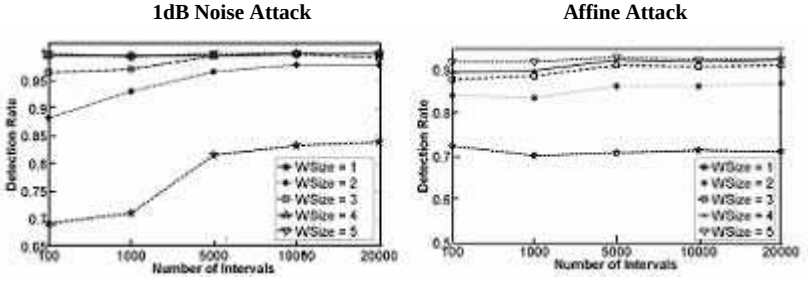


Fig. 8. Detection rate Vs Number of intervals for varying watermark size (WSize)

Table 1. Analysis of Sample Motion files Encoding, Watermark Size = 5

Motion Type	# Samples	SNR (dB)	1dB Noise (DR)
Dance	19000	86.002	0.99
Walk	17594	112.1351	0.99352
Karate 1	10928	103.3759	0.99631
Karate 2	11039	112.2567	0.99828
Exercise	19000	104.4280	1

Accuracy Analysis for Encoding. Affine transformation and noise addition attacks are uniformly carried out on the data set. It can be observed from Table 1 that for different motion types, the scheme gives high detection rate. This Sub-section also proves the claims presented in Sub-section 2.1, which show that larger sized watermark help increase the accuracy of tamper detection. Also detection ability increases by reduction in interval size, which can be done by increasing the number of intervals. Fig 8 show the results for tampering based on noise, and affine transformation. The general trend in both the graphs shows us that increase in

watermark size increases the accuracy (high value of detection rate) of tamper detection. Reverse ordering attacks were detected with 100 % accuracy.

5 Conclusion

The paper suggests a tamper proofing methodology for 3D motion data by visualizing 3D data as clusters of 3D points. Watermarks are encoded using extensions of quantization index modulation by applying 1-Point and 3-Point encoding per cluster. Using this scheme it is possible to detect affine, noise addition and reverse ordering attacks. The technique achieves imperceptibility of watermarks with reduction in distortions (SNR > 70 dB). A probabilistic model for detection of watermark shows that larger sized watermarks achieves accuracy > 90 %. Also, the encoding scheme parameters can be varied to improve the accuracy of tamper detection.

Acknowledgements. The work is supported in part by US Army Research Office grant 48645-MA and NSF under Grant No. 0237954 for the project CAREER: Animation Databases.

References

1. B. Chen and G. W. Wornell, “Quantization index modulation: A class of provably good methods for digital watermarking and information embedding," IEEE Trans. on Information Theory, vol 47, pp. 1423-1443, May 2001
2. C. S. Lu, H. Y. Mark Liao, and L. H. Chen, "Multipurpose Audio Watermarking", Proc. 15th Int. Conf. on Pattern Recognition, Barcelona, Spain, Vol. III, pp. 286-289, 2000
3. E.T. Lin and E.J. Delp, "A review of fragile image watermarks," in Proc. of ACM Multimedia & Security Workshop, Orlando, 1999, pp. 25--29.
4. F. Cayre, O. Devillers, F. Schmitt and H. Maître, Watermarking 3D Triangle Meshes for Authentication and Integrity, INRIA Research Report RR-5223, Jun. 2004
5. J. Fridrich, M. Goljan, and A. C. Baldoza, "New fragile authentication watermark for images," in Proc. IEEE Int. Conf. Image Processing, Vancouver, BC, Canada, Sept. 10--13, 2000.
6. H.T. Wu and Y.M. Cheung, A Fragile Watermarking Approach to 3D Meshes Authentication, Proceedings of the 7th Workshop on Multimedia & Security (ACM'05), pp. 117-123, 2005.
7. Ingemar Cox, Matthew Miller, Jeffrey Bloom, Mathew Miller, Digital Watermarking: Principles & Practice (The Morgan Kaufmann Series in Multimedia and Information Systems)
8. Minghua Chen; Yun He; Lagendijk, R.L., "A fragile watermark error detection scheme for wireless video communications," Multimedia, IEEE Transactions on, vol.7, no.2pp. 201-211, April 2005
9. P. Agarwal, K. Adi, B. Prabhakaran, “SVD-based Tamper Proofing of Multi-attribute Motion Data”, Proc. of The 12th International conference on Distributed Multimedia Systems (DMS) ‘06
10. Vicon, http://www.vicon.com

A Uniform Way to Handle Any Slide-Based Presentation: The Universal Presentation Controller

Georg Turban and Max Mühlhäuser

Darmstadt University of Technology
{turban, max}@informatik.tu-darmstadt.de

Abstract. We present a sophisticated approach for handling and processing presentations and multimedia content in the classroom. The main contribution of our work is the way we technically control and display our models for multimedia based presentations: In contrast to existing approaches we avoid converting the file-based representation to a home-brew format which seems to be the easiest way for the processing and appliance of own features. Instead, we present the benefit of our layered solution that creates a model-based representation of any popular slide-based presentation format like PowerPoint and PDF and uses the original presentation systems that run in parallel to our own application, the universal presentation controller. Therefore, we can keep all hot features like integrated audios or videos, animations and slide transitions, notes and even native inking of the original presentation systems, but are also able to add our own extensions in general. We can communicate with other applications and offer them access to our model and core functionality. The models can be modified and extended online, which for example allows the integration of snapshots taken from a webcam.

Keywords: multimedia representation, processing, compatibility, framework.

1 Introduction

Nowadays, lecturers often use multimedia-based presentations at universities or high schools to mediate content. For authoring, presentation and distribution of them, they often use several software tools. Working with different tools requires most of the time the conversion of the underlying file-based representation of such presentations.

To give you an example, we present the following simplified[1] but still typical workflow: authoring → categorization and archiving by the use of a learning management system → conversion and integration into an ink-aware presentation-system → augmented presentation → storage and distribution. We are especially interested in the chain link that deals with the conversion and integration of presentations. Such a conversion step is often performed by systems that rely on their own data and file format, which is usually restricted but much easier to handle. Another disadvantage is that the same content exists in two different formats resp. files, so that modifications cause the overhead of synchronization. In addition, it's

[1] We completely skipped stages like knowledge production and transfer.

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 741 – 750, 2007.
© Springer-Verlag Berlin Heidelberg 2007

likely that converters produce a very restricted result. Due to complexity of presentation formats, features like animation in PowerPoint-slides [1] are often gone.

In our opinion, this kind of conversion is a final step that doesn't sufficiently meet today's requirements and motivated us to develop a different approach. To obtain our design-goals we identified and analyzed requirements of categories like usability and interoperability under the view of users of different presentation systems who use such systems either isolated or in combination with more complex e-learning processing systems.

We present those major design-goals in the following chapter, while the remainder of this paper is organized as follows: Since chapter 3 about the implementation describes how we successfully transformed our design-issues into a sample realization, selected scenarios for the usage of our application will be presented in chapter 4 that also deals with the evaluation of our contributions. We complete our publication by a chapter that highlights our research results and contributions and the last chapter that presents related research topics for the future.

2 Design

E-learning systems are more and more used during lectures or even conferences: In such scenarios, the presentation system is often used in combination with recording subsystems that capture slides and videos of the speaker. But common systems like Microsoft's Windows Journal [2], UW Classroom Presenter [3], Lectern II [4] or methods like Authoring on the Fly [5] require a dedicated preprocessing step to convert content into their native, internal representation of a slide-set. We discuss two of the most popular conversion-approaches that are based on virtual printer redirection and dedicated tools for conversion, their limitations and impact on our main design goals and concept in the following sections.

For the presentation of the first approach, we refer to Adobe Systems professional version of Acrobat [6] that – in contrast to the freely available viewer – consists of a virtual printer driver that can be used to convert every print-inquiry into a file-dump in the portable document format (PDF) instead of a real paper-based print. E-learning and ink-aware systems like Lectern II or Microsoft's Windows Journal use the same technique to create file-based representations of presentations in their desired and most often home-brew format. E.g. in case of Lectern II a modified enhanced-meta-file-printer is used. The benefits are obvious; the desired preprocessing step can be easily performed by non-experts and covers all printable contents that would have been printed on different sheets of paper. The disadvantages based on conceptual ignorance of the original file-format are similar to that of printed versions; like on a printed sheet of paper, many features can't be taken over and are lost – think about all dynamic contents like videos, animations or sound at all.

Another approach attempts to pay more attention to the individuality of the original formats: Dedicated conversion tools – either stand-alone applications or add-ins for parent host-applications – have been implemented to support conversion of selected formats and therefore are possibly able to reach a higher degree of functional support. Certainly, such conversion tools support only the formats respectively applications for which they have been implemented. In general the most common representation of

slide-based presentations relies on a set of images that is no longer compatible with the original authoring software. In case of Classroom Presenter the result is a file in the so-called csd-format (stands for conferencing slide deck) that looses all animations and notes of the original PowerPoint-presentation.

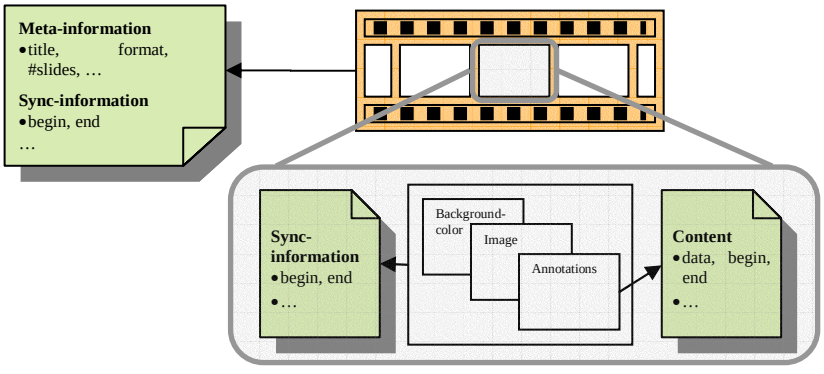


Fig. 1. Presentation and slide-model

Especially at conferences the desired preprocessing step concerning slides is not the only disadvantage; we observed that speakers feel more comfortable with a presentation environment they are used to. Nowadays, the different subsystems for presenting and controlling of slide-sets are muddled and demand training of those users. That's why a clearer conceptual separation is another design-goal for us.

We expect profits like reusability and single training for users by a decoupled architecture and separation for the controlling and presentation subsystems. In our opinion this architecture should also support multiple-monitor systems. The underlying technical constraints are widely fulfilled, since actual hardware like notebooks or graphic cards are aware of at least two independent output-devices. Based on outputs containing different signals, already existing approaches use this possibility to display an enhanced version of the currently presented slide that can be augmented by digital ink and several tools, but also allows navigation in the whole slide-set. Here, the main idea is to hide those graphical interfaces from the audience.

Because of the presented disadvantages and given limitations we feel that existing approaches are not satisfying and summarize our main design-goals in the following list, before we present the details about the implementation that follows these goals:

(a) Minimize loss of functionality during the conversion to increase compatibility with native (re-)presentation
(b) Higher interoperability with other systems, e.g., used to synchronize and coordinate cooperative processing
(c) Support multiple-monitor systems besides usage via a single display
(d) Intuitive controlling of slide-sets and navigation in them

3 Realization

According to our overall design goals, we developed the following solution: In contrast to common approaches, we avoid a dedicated preprocessing step for the conversion of the content; instead we keep the presentation in its source-format all the time. Referring to item (a) – alphabetic references point to chapter 2 – we reach a much higher degree of available functionality, because for each supported presentation-format, we use the corresponding and original presentation software.

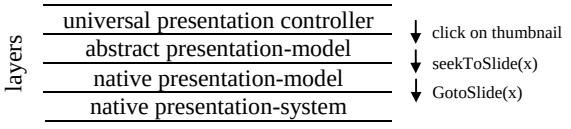


Fig. 2. Vertical view of layered architecture

We decided to wrap this native presentation layer that communicates with the different presentation or authoring systems by a layer that is able to uniformly interact, communicate and exchange content with other subsystems. For this uniform handling we had to develop a representation for all kind of presentation-formats we want to support (cf. figure 1). For clarity, we highlight our core steps of realization: We avoid converting content to a new destination format, use the corresponding native presentation-system for handling of each format (e.g. for PPT-files we use Microsoft PowerPoint) and provide a "wrapping layer" for all formats that uniformly interacts through common interfaces and protocols with any other subsystem and enables us to implement core features of our presentation controller only once.

Our architecture contains four layers. We decided to present them in separate subsections and in top-down order, which mainly reflects our procedure of design and implementation. For ease of understanding, we give you the following simple example that dives through all layers displayed in figure 2: The user currently focuses the thumbnail of a slide he wants to present. By clicking this thumbnail, the top-most layer calls the method *seekToSlide(x)* in the layer abstract presentation model. The presentation model itself is just a calling convention and its implementation is hidden in the native layer below which knows how to perform the requested operation in the native presentation system. In our example, the initial click of the user will finally call the method *GotoSlide(x)* in PowerPoint's API.

3.1 Universal Presentation Controller (UPC)

The common interfaces that assure high interoperability with other applications also meet and simplify our presented design goal (d). Related processing systems had to implement their specific presentation-controller over and over, because of their internal presentation-format. In contrast, our application provides a user-interface that can be used in combination with every format and many other processing systems.

Figure 3 shows a snapshot of UPC: The toolbar offers functions to load a presentation or to toggle communication with other systems. The information shown

below the slides are selected information of our presentation- and slide-models like the slide-id, the amount of animation-steps per slide and the comments of the presented slides. We redesigned the part of a user-interface we presented in a different context [7] and improved our handling of preview and presentation via thumbnails and controls to be more intuitive. Instead of separate preview and control-panels we decided to imitate a common filmstrip that has following two functions: initially the center image contains the currently presented and highlighted slide, while images to the left may have been just presented before or images to the right show upcoming slides and allow a smoother transition to them. The second functionality can be activated by the scrollbar below the images that allows scrolling through (and therefore previewing of) the whole slide-set without change of the currently presented and still highlighted slide. To seek to one of the previewed slides, users can easily click on the thumbnail and present the corresponding slide. If necessary – mainly for slides that contain dynamic content and several animation steps – an instantly created copy of the content that is shown to the auditorium can be displayed in addition to the thumbnails on the screen of the speaker; see figure 4 that shows a speaker using UPC.

3.2 Abstract Presentation-Model

The model of a presentation (see sample 1) serves as a uniform online-representation of any original presentation format and holds references to the models of the

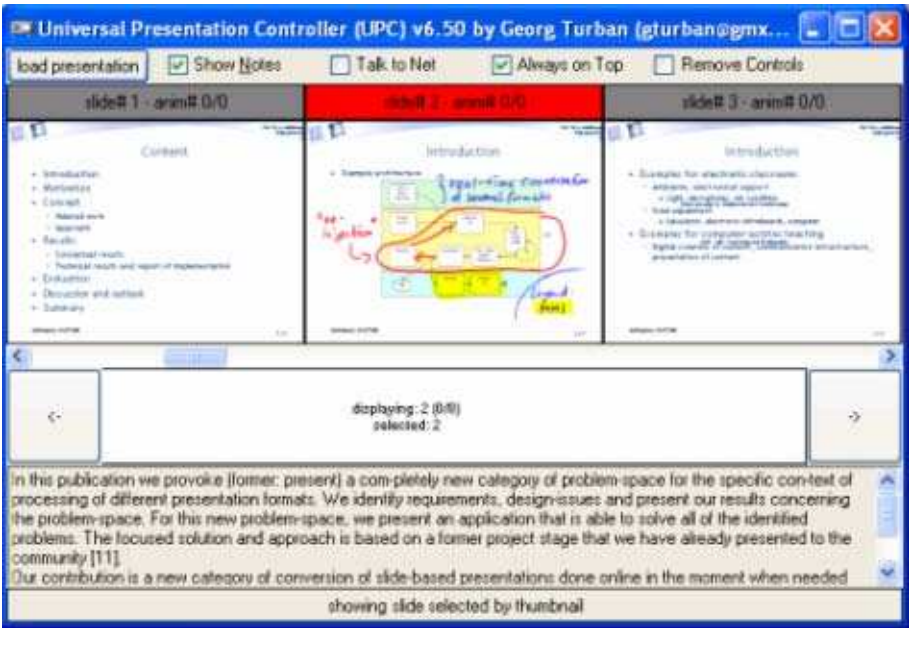


Fig. 3. A snapshot of UPC's user-interface. A PowerPoint-presentation has been loaded and is currently presented. We placed out toolbar on top of the interface, while a status-strip resides at the bottom. The remaining interface contains the following three panels: a filmstrip view that consists of slide-thumbnails, selected information of our presentation-model and slide-notes.

corresponding slides (see line 7). We developed an abstract model to provide uniform access to content and (!) features like navigation or synchronization that will be implemented in any derived model (refer to figure 1 and 2). Our abstract model – and therefore all derived implementations – contains descriptive information about the corresponding presentation like the underlying (original) format, the title and total amount of slides. Since PowerPoint animations are widely used by speakers to develop slides during their talk, we designed slide-models that can represent animations. Such models contain information like the title, a graphical representation (e.g. for thumbnails or previews), the amount of animation-steps and notes of slides.

Again, it is very important that we use this model as a representation for individual, native interaction with the original software (presentation- or authoring-systems), but global, uniform communication – to meet design goal (b) – with processing-systems like the ink-aware ones that we presented in [7] and [8].

3.3 Native Presentation-Model

This layer includes the fully functional derivations for each supported presentation format of our well-designed, but still abstract presentation-model and acts like a mediator between the abstraction layer and the native presentation systems. It essentially contains the native bindings to the original presentation-systems. The layering between one abstract and many native presentation models opens our architecture to be easily extended by pluggable components that contribute their support for other presentation-formats.

The abstract class *PresentationModel* has been simplified for illustration purpose and already contains some bundled portions of interfaces (descriptive, loadable and displayable). The interfaces listed in table 1 can be implemented in this layer to extend the representation and functionality of the core presentation model.

For example, the *extensible interface* forces the developer to implement a method *insertSlideAtCurrentPosition* that has to create a slide-model that will be added to the whole set of slides. Our implementation for PowerPoint is quite powerful: Instead of a

Sample 1. Model description for the presentation and slide:

```
01 public class PresentationModel
02 {
03   public string title;
04   public int firstSlideId, lastSlideId;
05   public int currentSelectedSlideId;          //previewed slide
06   public int currentDisplayedSlideId;         //projected slide
07   public SlideModel [] sm;
08   public virtual void seekToSlideAndAnimationStep(int sid, int aid);
09 }
```

```
01 public class SlideModel
02 {
03   public string title;
04   public int firstAnimationId, lastAnimationId;
05   public int currentAnimationId;
06   public System.Drawing.Image img;            // e.g. for thumbnails
07   public String [] Notes;                     // notes of the slide
08 }
```

simple, blank slide we obtain a slide from PowerPoint's slide-master that contains our logo, header and footer plus the correct slide-number, because all following slide-models will be updated automatically. The implementation of latter interface is absolutely essential for the Webcam-model which usually starts with zero slides from scratch; the Webcam-implementation has to call the method *insertSlideAtCurrentPosition* after each snapshot.

Table 1. Interfaces that define the core presentation model and enable its extension

Application / Extension	Descriptive	Loadable	Displayable	Navigation	Notes	Dynamic	Extensible	Storable
PowerPoint	✔	✔	✔	✔	✔	⛔	✔	✔
Image sets	✔	✔	✔	✔	⛔	✔	⛔	✔
Portable documents	✔	✔	✔	✔	⛔	⛔	⛔	✔
WebCam	✔	✔	✔	✔	⛔	✔	✔	✔

3.4 Native Presentation-System

This layer is optional for our own format, because the implementation can be included in the layer above or reside in the same application domain. But for other formats, this layer contains the referenced and corresponding (third-party) presentation-software that kindly gives us access to its controlling and presentation functionality.

In case of PowerPoint we need to use the professional version, because the freely available viewer does not contain the API and has other limitations. Referring to

Sample 2. PowerPoint specific realization of method "seekToSlideAndAnimationStep":

```
01 public override void seekToSlideAndAnimationStep(int sid, int aid)
02 {
03   GGT.Communication.Communicate();
04   SlideModel sm= getSlideModel(currentDisplayedSlideId);
05
06   if(currentDisplayedSlideId == sid)
07   {
08     switch(aid-sm.currentAnimationId)
09     {
10       case -1: pptPres.SlideShowWindow.View.Previous(); break;
11       case 0: /* no operation */ break;
12       case +1: pptPres.SlideShowWindow.View.Next(); break;
13     }
14     sm.currentAnimationId = aid;
15   }
16   else
17   {
18     sm.currentAnimationId = sm.firstAnimationId;
19     pptPres.SlideShowWindow.View.GotoSlide(sid+1,
         Microsoft.Office.Core.MsoTriState.msoTrue);
20     currentDisplayedSlideId = sid;
21   }
22 }
```

sample 2, we use PowerPoint's API at lines 10, 12 and 19 to navigate to the target position in the whole slide-set. If the evaluation in line 6 becomes true, the current slide remains, but a different animation-step has to be shown. The following call of block 7-15 forwards the task of changing the animation step within the current slide to PowerPoint-APIs commands *Previous* and *Next* in lines 10 and 12. These are basically the same calls that are performed when a human manually presses the left or right cursor key. If the slide has to be changed, block 17-21 will be executed. The call of the method *GotoSlide* in line 19 forces PowerPoint to seek to a specific slide, while the remaining instructions in lines 18 and 20 are necessary to keep our presentation model in sync with the native one.

Our solution for PDF-files is almost the same; we also use the professional version of Acrobat that contains the SDK and provides similar methods. The next application we are looking forward to support is Impress and its presentation by using the UNO Development Kit [10].

4 Evaluation

UPC is highly interoperable and can be used in several isolated scenarios or in combination with many other processing systems.

For evaluation purpose we present the following list of selected use-cases:

(i) isolated and independent usage, either on single or multiple-monitor systems
(ii) usage as a converter, transforming one presentation format into another
(iii) combined usage of multiple presentation-models and real-time integration
(iv) combined usage with (foreign) processing systems

We developed an application that can be used to present presentations in different formats via a single user-interface, while its global configuration is still easy: Settings like the destination screen for presentations can be set independently of the format, so that all presentations (PPT, PDF, etc) appear on this screen (i). Because we use the native presentation system to control and present the corresponding format, e.g. PowerPoint-presentations can be presented including animations or even fully dynamic and continuous content like sound or videos with no restrictions.

It is very important to notice that our solution implicitly covers all conversion-approaches of related work we presented in chapter two (ii), though not focused here.

To demonstrate our far-reaching capabilities in the third case, we composed a scenario we named the virtual overhead-projector: The lecturer uses digital instead of overhead-slides. While this is our first model, e.g., to represent a PDF-presentation, the second one represents a webcam that still offers the functionality of a real overhead-projector but is able to display even three-dimensional objects, while captures are integrated into the targeting first model that enables uniform digital annotation.

Regarding the combined usage with processing systems for augmentation and/or recording like InkPresenter or DLH-Presenter, we already presented results of several semesters' experiences in [7] and [8]. But also approaches that appear controversial to case (iv) can benefit from our communication infrastructure. For (pure) screen recording systems we deliver valuable synchronization information that are useful navigational indices and minimize post-processing efforts like discussed in [9].

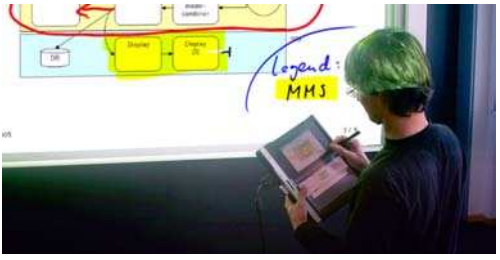


Fig. 4. UPC in the classroom. The lecturer turned around to show the convertible he uses in portrait mode. While he placed UPC in the lower portion of the display, the upper one continuously mirrors the content that is projected by PowerPoint to the students and forwards all stylus-inputs to PowerPoint, which enables him to use its native ink-mechanism.

A prior version of UPC was widely used by lecturers in a total of 14 lectures (scheduled weekly) and two block courses of several days duration each. In addition, a considerable number of national and international conferences and fairs were successfully supported by the controller. In this summer we created evaluation forms for all lecturers and received nine forms. “Which presentation formats should be available by the presentation controller besides PowerPoint?” was a question that was answered as follows: PDF (47%), Impress (27%), Image sets (13%), others (13%) and “no more” (0%). We decided not to put PDF that is already supported into the question, because we wanted to determine if this format is really requested. Nearly half of all lecturers requested PDF. The amount of requests for Impress was higher than for image based presentations which is interesting, because in a former evaluation Impress was only requested by two persons.

Requirements that have been reported to the prior version of UPC were integrated into the current version presented in figure 3 and 4. Some users reported that they feel irritated about a scrollable panel that contains the whole slide-set and two more panels representing the currently presented and upcoming slide. First, we expected that we “clearly” indicated these modes by titled borders and that their separation was the best choice, but current observations regarding the newly ordered film-strip-design, which still includes both modes, shows that this solution is definitely much easier to use. The correspondence to a common camera containing a filmstrip, where the image that is currently in front of an optic is projected, seems to be more intuitive.

5 Summary of Contributions

We presented requirements in the context of integration and presentation of individual content that we identified by observations, requirements analysis and discussions with users. Based on those discoveries we built an application that deals with given limitations and wishes. Since it is widely used, it was possible to refine it heavily in a couple of iterations and to meet user’s requirements very well.

Our application can be installed straightforwardly and requires no configuration to work with PowerPoint and Acrobat Professional. It is robust and easy to use – usually an introductory instruction of less than 5 minutes is sufficient. Hence, our users can

use a single application to present their content in the following formats: Microsoft PowerPoint, Adobe PDF, image-sets, home-brew-formats and experimentally static and dynamic captures from cams.

Introducing a clear description and representation for presentations enabled us to concentrate on the fundamental technical interactions with native presentation systems; we developed global infrastructure only once for all formats and therefore minimized complexity besides costs of our universal solution. We presented detailed results of our conceptual contributions by a mature application that is aware of a new stage in the processing chain that has been unfortunately disregarded by related work. Our contribution is able to change the traditional awareness and workflow of multimedia based presentations, because of its uniformity and online capability.

6 Outlook

Based on our concept and solution, further research may focus on following topics: Multiple instances of same and different presentation models and cross-operations, individual processing capabilities including annotations and their specific storage, the integration and recording of continuous media from different sources and altogether the impact on the extension of the presentation- and slide-models.

References

1. Microsoft Corporation, “Microsoft PowerPoint,” http://www.microsoft.com, last visited October 8th, 2006.
2. Microsoft Corporation, “Microsoft Windows Journal,” http://www.microsoft.com, last visited October 8th, 2006.
3. University of Washington, “UW Classroom Presenter,” http://www.cs.washington.edu/education/dl/presenter/, October 8th, 2006.
4. N. Joukov, T. Chiueh, “Lectern II: A multimedia lecture capturing and editing system,” In Proceedings of the International Conference on Multimedia and Expo, Baltimore, Maryland, Volume 2, pp. 681-684, July 2003.
5. W. Hürst, R. Mueller, and T. Ottmann, “The AOF Method for Production, Use, and Management of Instructional Media,” In Proceedings of the International Conference on Computer in Education, Melbourne, Australia, Common Ground Publishing, 2004.
6. Adobe Systems Incorporated, “Adobe Acrobat Professional,” http://www.adobe.com/products/acrobat/index.html, last visited October 8th, 2006.
7. G. Turban, M. Mühlhäuser, “A category based concept for rapid development of ink-aware systems for computer-assisted education,” In Proceedings of the 7th IEEE International Symposium on Multimedia, Irvine, California, USA, pp. 449-457, 2005.
8. G. Turban, G. Rößling, and C. Trompler, “Bridging media breaks in presentations,” In Proc. of the 10th annual SIGCSE conference on Innovation and Technology in Computer Science Education, Caparica, Portugal, ACM Press, New York, USA, p. 377, 2005.
9. P. Ziewer, “Navigational Indices and Full Text Search by Automated Analyses of Screen Recorded Data,” In Proceedings of E-Learn 2004, Washington, DC, USA, 2004.
10. OpenOffice.org, “UNO Development Kit (UDK) project,” http://udk.openoffice.org/, last visited October 8th, 2006.

A Tensor Voting for Corrupted Region Inference and Text Image Segmentation

Jonghyun Park, Jaemyeong Yoo, and Gueesang Lee

Dept. of Computer Science, Chonnam National University,
300, Yongbong-dong, Buk-gu, Gwangju, Korea
jhpark@chonnam.ac.kr

Abstract. Most computer vision applications often require reliable segmentation of objects when they are mixed with corrupted text images. In the presence of noise, graffiti, streaks, shadows and cracks, this problem is particularly challenging. We propose a tensor voting framework in 3D for the analysis of candidate features. The problem has been formulated as an inference of hue and intensity layers from a noisy and possibly sparse point set in 3D. Accurate region layers are extracted based on the smoothness of color features by generating candidate features with outlier rejection and text segmentation. The proposed method is non-iterative and consistently handles both text data and background without using any prior information on the color space.

Keywords: Tensor voting, Text Segmentation, Scene Analysis, Mean-Shift, Color Space.

1 Introduction

Text information in a natural scene is quite useful since it can convey very important meanings even though it is simple. Recently, we easily accumulate natural scene images by PDA (personal digital assistant), mobile phone, robot vision systems and equipped with digital camera or vision systems. It is natural that the demand for automatic detection and recognition of the text region on these images has been increased. Detecting a text region generally consists of various process steps; selection of color feature, segmentation method, noise filtering, text region extraction, text recognition, and so on. These issues have been mentioned through various researches. Divers approaches for common image segmentation have been investigated for a long time. Some segmentation algorithms only deal with gray scale images [1]. Other algorithms perform segmentation of color images in the RGB color space [2]. The segmentation is sensitive to illumination, so results are somewhat poor. Image segmentation in the HIS color space, proposed by C. Zhang and P. Wang, produces better results [3]. HIS space is therefore preferred in natural scenes to the RGB representation due to robustness to illumination changes.

In general, natural scenes have diverse objects and, among them, characters are important objects since they convey important meanings for image understanding. The fact has inspired many efforts on text recognition in static images, as well as

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 751 – 761, 2007.
© Springer-Verlag Berlin Heidelberg 2007

video sequences [4]. In [5], Jie Yang et al. develop a machine translation system which automatically detects and recognizes texts in natural scenes. In [6], Qixiang Ye et al. use Gaussian mixture models in HIS color space with spatial connectivity information to segment characters from a complex background. And then, prototype systems for sign translation have been developed for handheld device and for personal computers [7],[8]. However, they do not explicitly take into account the fact that characters in natural scenes can be severely corrupted text by noise. In such cases, characters may not be segmented as separate objects due to the corruption of strokes which may cause errors when used as input in optical character recognition (OCR), as mentioned in the future work in [9]. In this paper, we propose to use the tensor voting framework for detection and removal of noise. Tensor voting was proposed by Medioni et al. in [10], and has been applied to diverse fields such as the inference of object boundaries [11]: as a consequence, its use can explain the presence of noise based on surface saliency in the image feature space. This noise can be then removed by a densification method in 3D. The improved image is then segmented by clustering characters. Clustering requires parameters such as the number or centroid of modes [12] which are generally not known a priori. We use mean shift-based density estimation for automatic mode detection and use these modes as seed values for K-means clustering of characters. Characters are finally segmented as respective objects.

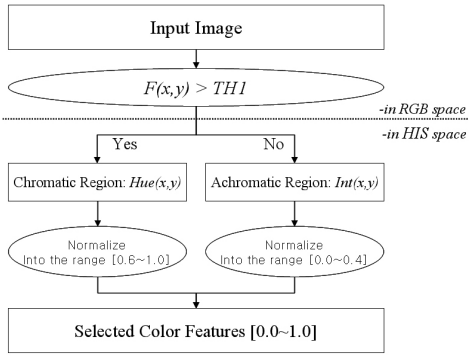


Fig. 1. Flow of color component selection in HIS space, where *Hue*(*x, y*) and *Int*(*x, y*) indicate the hue and intensity components respectively

2 Generating Candidate Color Features

This section details a decision function for classifying a pixel as chromatic or achromatic so that the appropriate feature is used in segmentation. In [13], S. Sural used the saturation value to determine the relative dominance of hue and intensity. Thresholding on saturation, however, is not illumination invariant. When a chromatic region is illuminated brightly, the saturation value is likely to be low compared to the same chromatic region with lower illumination. The low saturation incorrectly indicates an achromatic region. We propose an alternative decision function in RGB space that is independent of illumination. Instead of thresholding on saturation, we derive a chromaticity measure based on the sum of differences of r (red), g (green), and b (blue) components at each pixel (x, y).

$$F(x,y) = \frac{|r(x,y) - g(x,y)| + |g(x,y) - b(x,y)| + |r(x,y) - g(x,y)|}{3} \quad (1)$$

From our experimental observation, the smaller the sum, the closer the related position is to the achromatic regions. A hue component is affected by both intensity and saturation components. It shows that saturation varies with illumination. We can

observe that some parts such as position of high and low saturation as perceptually achromatic regions have high saturation. Meanwhile, the sum in (1) is low in all perceptually achromatic regions as well as the position of low and high saturation and high in all chromatic regions. The level of chromaticity is proportional to $F(x, y)$ in (1). A threshold value $TH1 = 20$ is used (determined heuristically) to classify a pixel with RGB components in the range of $[0 \sim 255]$. Values below $TH1$ are classified as being achromatic and analyzed using the intensity component ($Int(x, y)$) in HIS space. The remaining pixels are chromatic, and analyzed using the hue ($Hue(x, y)$). In the chromaticity labeled image, hue components are still values normalized from angles, which we take into account later. The values near 0.6 and 1.0 are clustered as one mode due to the cyclic property of hue component. In addition, leaving a gap between two feature ranges prevents that achromatic and chromatic regions are overlapped during clustering. The final values of a chromaticity labeled image are distributed in the range of $[0.0 \sim 1.0]$. The values corresponding to one image are applied to the tensor voting framework in 3D.

3 Tensor Voting in 3D for Image Analysis

3.1 Review of Tensor Voting

A voting process for feature inference from corrupted data, sparse and noisy data was introduced by Guy and Medioni, and formalized into a unified tensor framework [10],[14],[15]. Tensor voting is a local method to aggregate and propagate information. All sites aggregate the received votes to produce a local estimate of structure, such as curves or surfaces. A local marching process can then extract the most salient structures. Each pixel in an image may belong to some perceptual structure such as a corner, curve, or surface. To capture the perceptual structure of input sites, tokens are defined and used. The tokens are represented by a second order symmetric non-negative definite tensor encoding perceptual saliency. The tensor can indicate its preferred tangent, normal orientation as well as saliency corresponding to its perceptual structures and be visualized as an ellipse in 2D and an ellipsoid in 3D. Such information is collected by a communication between input sites: tensor voting. Input tokens encoded as tensors cast votes computed through a voting field (2) to their neighborhood. The voting field explains how the tokens relate their information, such as orientation and magnitude, to their neighborhood to ensure smooth continuation. All voting fields are based on the fundamental 2D stick voting, the *saliency decay function* of which is :

$$DF(s,k,\sigma) = EXP\left(-\left[\left(s^2 + ck^2\right)/\sigma^2\right]\right) \tag{2}$$

where $s = l\theta/\sin\theta$, $k = 2\sin\theta/l$.

The parameter s is the arc length, k is the curvature, c is a constant, which controls the decay with high curvature, and σ is the scale of voting field controlling the size of the voting neighborhood and the strength of votes. The orientation of the stick vote is normal to the smoothest circular path connecting the voter and receiver. All tokens accumulate votes from the neighborhood and their collected information is

computed as a covariance matrix S by the second order tensor sums (where $[v_x, v_y]$ is a vector vote generated by the neighbor pixel for center pixel.):

$$S = \begin{bmatrix} \sum v_x^2 & \sum v_x v_y \\ \sum v_y v_x & \sum v_y^2 \end{bmatrix} \tag{3}$$

While (3) is the conventional notation for the analysis of tensor voting. Given its eigensystem, consisting of two eigenvalues (λ_1, λ_2) and two eigenvectors $(\hat{e}_1, \hat{e}_2)$, the matrix S can be rewritten as:

$$S = (\lambda_1 - \lambda_2)\hat{e}_1\hat{e}_1^T + \lambda_2(\hat{e}_1\hat{e}_1^T + \hat{e}_2\hat{e}_2^T) \tag{4}$$

where $\hat{e}_1\hat{e}_1^T$ and $\hat{e}_1\hat{e}_1^T + \hat{e}_2\hat{e}_2^T$ indicate a stick and ball tensor in 2D, with respective saliency $\lambda_1 - \lambda_2$ and λ_2. Examining the eigensystem, we can infer the most likely perceptual structure of the token as either a surface, a curve, or a corner. In our case, input tokens are first encoded as 3D ball tensors in a 3D space (*x, y, value of position*). These initial tensors communicate with each other to understand the most preferred orientation information at each position. Votes are accumulated at all positions by tensor addition based on the voting field. The result of one position is given in matrix form by:

$$S_{3D} = \begin{bmatrix} \hat{e}_1 & \hat{e}_2 & \hat{e}_3 \end{bmatrix} \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix} \begin{bmatrix} \hat{e}_1^T \\ \hat{e}_1^T \\ \hat{e}_1^T \end{bmatrix} \tag{5}$$

Or equivalently :

$$S_{3D} = (\lambda_1 - \lambda_2)\hat{e}_1\hat{e}_1^T + (\lambda_2 - \lambda_3)(\hat{e}_1\hat{e}_1^T + \hat{e}_2\hat{e}_2^T) + \lambda_3(\hat{e}_1\hat{e}_1^T + \hat{e}_2\hat{e}_2^T + \hat{e}_3\hat{e}_3^T) \tag{6}$$

where $\hat{e}_1\hat{e}_1^T$ is a 3D stick tensor, $\hat{e}_1\hat{e}_1^T + \hat{e}_2\hat{e}_2^T$ is a 3D plate tensor, and $\hat{e}_1\hat{e}_1^T + \hat{e}_2\hat{e}_2^T + \hat{e}_3\hat{e}_3^T$ is a 3D ball tensor. For surface inference, surface saliency is then given by $\lambda_1 - \lambda_2$, with normal estimated as $\hat{e}_1$. Moreover, curves and junctions are inferred from the curve and junction saliency given by $\lambda_2 - \lambda_3$ and λ_3.

In general, text image normally is appeared as regions of homogeneous color. However, the text image may also be noisy, as the physical surface of the sign degrades due to corrosion, graffiti, intentional or unintentional defacing, etc. these noises are more inhomogeneous, so that the noise regions are comprised of severely different values. Even though the noise regions appear with similar values, their regions size is small than text or background. In the tensor voting framework, one image can be represented with $[x, y, H(x, y)]$. Here *x* and *y* indicate the positions in the image and $H(x, y)$ is the values corresponding to respective positions in a chromaticity labeled image, which is obtained in the previous step.

3.2 Extraction of Feature Layers Using Densification in 3D

3.2.1 Selection of Candidate from Surface Saliency

From given candidate color features, each color data is encoded into a 3D ball tensor. Then each token casts votes by using the 3D ball voting field. By the processing, the voting between tokens that lie on a smoother surface of layer derives stronger

support in the 3D space of both pixel coordinates and pixel densities. For each position (x, y) of pixel in 3D space, the candidate feature with highest surface saliency value of $(\lambda_2 - \lambda_3)$ is preserved, but others are declined as noises

3.2.2 Outlier Rejection of Corrupted Region

Among the most salient candidate feature at each pixel, corrupted regions can be incomplete pixel having received very little support and we would like to reject the tokens. Generally, tokens within the voting field have homogeneity between neighbor tokens. Thus, we reject all tokens that have received very little support by the tensor voting processing. For outlier rejection, all deficient tokens are rejected by surface saliency less than 20 % of the average saliency of the total set.

3.2.3 Densification for Finding the Best Feature

We here describe densification method for finding the best features. Because the previous step generated isolated regions at rejected regions, we have isolated regions such as pixel where no color value is available. Therefore, now that the most likely type of feature at each token has been estimated, we want to compute the densification structures in 3D that can be inferred from the neighbor token. This can be achieved by casting votes to all locations into voting field. Each pixel (x, y) has all the discrete candidate points $v_i(x, y)$ which are represented between the minimum and maximum density values in the set, within a neighborhood of the (x, y) point. The tensor voting framework accumulate votes at each candidate position (x, y, v_i). We can compute value of surface saliency after voting. The candidate token by surface saliency $(\lambda_2 - \lambda_3)$ with optimal value is maintained and then its (x, y, v_i) positions represent the most likely color value at (x, y). Finally, at every (x, y) pixel location, a dense color value field is shaped.

4 Mean Shift-Based Mode Detection and Clustering Algorithm

In this section, we briefly review the original mean shift-based density estimation show how mode of clusters is detected by density gradient estimation function [16].

4.1 Density Gradient Estimation

The image is interpreted as n data points in a d-dimensional space where n is the number of pixels. The values of improved image are distributed in the range $[0.0 \sim 1.0]$ and used directly, giving a 1-dimensional feature. The initial values for distinct characters coincide with the modes of the data. Mean shift-based density gradient estimation with sampling data finds the local maximum of the probability densities [16]. Let $\{\mathbf{X}_i\}$, $i = 1, \cdots, n$ be the set of n data points in a d-dimensional Euclidean. The multivariate kernel density estimate obtained with kernel $K(\mathbf{x})$ and window radius for bandwidth h, computed at point x is defined as:

$$\hat{f}_K(\mathbf{x}) = \frac{1}{nh^d} \sum_{i=1}^{n} K\left(\frac{\mathbf{x} - \mathbf{X}_i}{h}\right). \tag{7}$$

Here, we are interested only in a class of radically symmetric kernels satisfying $K(\mathbf{x}) = c_{K,d}k(\|\mathbf{x}\|^2)$, in which case it suffices to define the function $k(x)$ called the profile of the kernel, only for $x \geq 0$ and $c_{K,d}$ is the normalized constant which makes $K(\mathbf{x})$ integrate to one. The differentiation of the kernel allows one to define the estimate of the density gradient as the gradient of the kernel density estimate:

$$\nabla \hat{f}_K(\mathbf{x}) = \frac{1}{nh^d}\sum_{i=1}^{n}\nabla K(\frac{\mathbf{x}-\mathbf{X}_i}{h}) = \frac{2c_{K,d}}{nh^{d+2}}\sum_{i=1}^{n}(\mathbf{x}-\mathbf{X}_i)k'(\|\frac{\mathbf{x}-\mathbf{X}_i}{h}\|^2). \quad (8)$$

We define the derivative of the kernel profile as a new function $g(x) = -k'(x)$, and assume that this exists for all $x \geq 0$, except for a finite set of points. Now, if we use a function for profile, the kernel is defined as $G(\mathbf{x}) = c_{G,d}g(\|\mathbf{x}\|^2)$, where $c_{G,d}$ is the corresponding normalization constant. In this case, the kernel $K(\mathbf{x})$ is called the shadow of kernel $G(\mathbf{x})$. If we use a function $g(x)$ in formula (8), then the gradient of the density estimator is written by

$$\nabla \hat{f}_K(\mathbf{x}) = \frac{2c_{K,d}}{nh^{d+2}}\sum_{i=1}^{n} g\left(\|\frac{\mathbf{x}-\mathbf{X}_i}{h}\|^2\right)\left(\frac{\sum_{i=1}^{n} g(\|\frac{\mathbf{x}-\mathbf{X}_i}{h}\|^2)\mathbf{X}_i}{\sum_{i=1}^{n} g(\|\frac{\mathbf{x}-\mathbf{X}_i}{h}\|^2)} - \mathbf{x}\right) \quad (9)$$

Here, this is given as the product of two terms having special meaning. The first term in the expression (9) is proportional to the density estimate at $\mathbf{x}$ computed with the kernel $G(\mathbf{x})$

$$\hat{f}_G(\mathbf{x}) = \frac{1}{nh^d}\sum_{i=1}^{n} G(\frac{\mathbf{x}-\mathbf{X}_i}{h}) = \frac{c_{G,d}}{nh^d}\sum_{i=1}^{n} g(\|\frac{\mathbf{x}-\mathbf{X}_i}{h}\|^2),$$

and the second term is defined as the mean shift vector

$$\mathbf{m}_G(\mathbf{x}) = \left[\left\{\sum_{i=1}^{n} g\left(\left\|\frac{\mathrm{x}-\mathrm{X}_i}{h}\right\|^2\right)\mathrm{X}_i\right\} \Big/ \left\{\sum_{i=1}^{n} g\left(\left\|\frac{\mathrm{x}-\mathrm{X}_i}{h}\right\|^2\right)\right\}\right] - \mathrm{x}. \quad (10)$$

This vector is the difference between the weight mean using the kernel $G(\mathbf{x})$ for weights and the center of the kernel. Then, we can rewrite the expression (9) as

$$\nabla \hat{f}_K(\mathbf{x}) = \frac{2c_{K,d}}{h^2 c_{G,d}}\hat{f}_G(\mathbf{x})\mathbf{m}_G(\mathbf{x}),$$

which yield,

$$\mathbf{m}_G(\mathbf{x}) = \frac{1}{2}h^2 c\frac{\nabla \hat{f}_K(\mathbf{x})}{\hat{f}_G(\mathbf{x})}. \quad (11)$$

The expression (11) shows the mean shift vector being proportional to the gradient of the density estimate at the point it is computed. As the vector points in the direction of maximum increase in density, it can define a path leading to a local density maximum which becomes a mode of density. It also exhibits a desirable adaptive behavior, with the mean shift step being large for low-density regions and decreases as a point $\mathbf{x}$ approaches a mode. Each data point thus becomes associated to a point of convergence, which represents a local mode of the density in the *d*-dimensional space.

4.2 Mean Shift-Based Model Detection

Input dates us denote by $\{\mathbf{y}_1, \mathbf{y}_2, \cdots\}$ the sequence of successive locations of kernel $G(\mathbf{x})$, where these points are computed by the following formula

$$\mathbf{y}_j = \frac{M}{N} \ , \ here, M = \sum_{i=1}^{n} g\left(\| \frac{\mathbf{x} - \mathbf{X}_i}{h} \|^2 \right) \mathbf{X}_i \ \ and \ \ N = \sum_{i=1}^{n} g\left(\| \frac{\mathbf{x} - \mathbf{X}_i}{h} \|^2 \right), \ \ j = 1, 2, \cdots . \qquad (12)$$

This is the weighted mean at $\mathbf{y}_j$ computed with kernel $G(\mathbf{x})$ and $\mathbf{y}_1$ is the center of the initial position of the kernel, $\mathbf{x}$. The corresponding sequence of density estimates computed with shadow kernel $K(\mathbf{x})$ is given by $\hat{f}_K(j) = \hat{f}_K(\mathbf{y}_j), \quad j = 1, 2, \cdots$.

Here, if the kernel has a convex and monotonically decreasing profile, two sequences $\{\mathbf{y}_1, \mathbf{y}_2, \cdots\}$ and $\{\hat{f}_K(1), \hat{f}_K(2), \cdots\}$ converge and $\{\hat{f}_K(1), \hat{f}_K(2), \cdots\}$ is monotonically increasing. After that, let us denote by $\mathbf{y}_c$ and $\hat{f}_K^c$ the convergence points of their sequences respectively. Here, we can get two kinds of implications from the convergence result. First, the magnitude of the mean shift vector converges to zero. In fact, the j-th mean shift vector is given as $\mathbf{m}_G(\mathbf{y}_j) = \mathbf{y}_{j+1} - \mathbf{y}_j$, and this is equal to zero at the limit point, $\mathbf{y}_c$. In other words, the gradient of the density estimate computed at $\mathbf{y}_c$ is zero. That is, $\nabla \hat{f}_K(\mathbf{y}_c) = 0$. Hence, $\mathbf{y}_c$ is a stationary point of density estimate, $\hat{f}_K(\mathbf{x})$. Second, since $\{\hat{f}_K(1), \hat{f}_K(2), \cdots\}$ is monotonically increasing, the trajectories of mean shift iterations are attracted by local maximum if they are unique stationary points. That is, once $\mathbf{y}_j$ gets sufficiently close to a mode of density estimate, it converges to mode. The theoretical results obtained from the above implications suggest a practical algorithm for mode detection:

Step1: Run the mean shift procedure to find the stationary points of density estimates.
Step2: Prune these points by retaining only the local maximum.

This algorithm automatically determines the number and location of modes of estimated density function. We shall use the detected mode or cluster centers from the mean shift procedure to be manifestations of underlying components of the clustering algorithm for our image segmentation task.

4.3 K-Means Algorithm for Text Region Segmentation

The number and centroid of modes selected in the subsection 4.2 are used as seed values in K-means clustering. K-means clustering is then applied to the values in the improved image to segment the character [2]. In our case, we should perform two different K-means clustering algorithm because intensity values are linear and hue values are characterized with the cyclic property. First, intensity values and their seed values fall in the range $[0.0 \sim 0.4]$ as normalized in chromaticity labeled image as well as the improved image. Intensity values compute Euclidean distance between itself value and seed values to find the closet seed value without considering the seed

values in the range $[0.6 \sim 1.0]$. The second K-means clustering algorithm should be used for hue values normalized into the range $[0.6 \sim 1.0]$ so that the algorithm can account for the cyclic property. In that case, the values of every pixel find the closest one among seed values in the range $[0.6 \sim 1.0]$ based on the approach in [4]. Chi Zhang et al. in [4] show that values near the minimum (0.6) and maximum (1.0) are clustered as one mode. Two K-means clustering passes are therefore performed while maintaining both the linear property of intensity values in the range $[0.0 \sim 0.4]$ and the cyclic property of hue values in the range $[0.6 \sim 1.0]$.

5 Experimental Results

To assess the performance of the proposed segmentation algorithm, we have conducted the experiment using data obtained from natural scene image, which are corrupted by noise. In our experiment, text regions are manually detected and the selected regions are segmented using our method. Fig. 2 shows our experimental results. The first and third image contains nonlinear red components which can typically cause problems when using the hue component for image segmentation. The results show that our approach is considering the nonlinear parts in hue component as well as removing noise. And then, in Fig. 3, we show a comparison of our approach to three other segmentation approaches (EDISON [16], by median filter, and GMM [17]) in respect of error rates. Fig. 3-(a) illustrates image data extracted from original natural scenes and fig. 3-(a) shows results segmented in manually labeled ground truth images. Compared to the results segmented by the proposed method in fig. 3-(b), we indicate errors as both FP and ND in fig. 3-(b). FP (false positive) indicates background pixels classified as character pixels in a segmented image and ND (no detection) indicates character pixels classified as background pixels or noise values in a segmented image. To show the error rate (ER) as one numerical value, we also

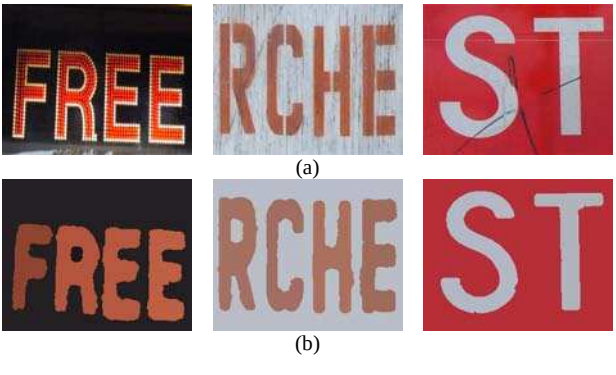


Fig. 2. Experimental results : (a) corrupted images, (b) segmented images

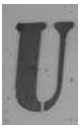
calculate the similarity between results segmented from ground truth images and original noisy images by:

$$Similarity = Result_ni \cap Result_gt / Result_ni \cup Result_gt \quad (13)$$

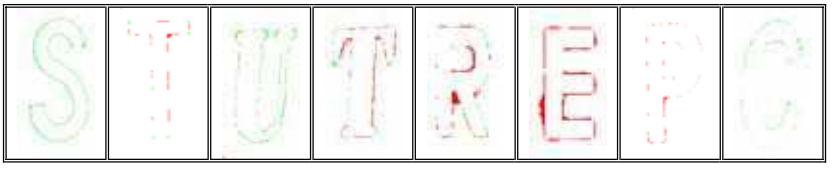
$$ER(\%) = [1 - Similarity] \times 100 \quad (14)$$

where *Result_ni : a character result segmented from an original noisy image,*
Result_gt : a character result segmented from the ground truth image.

Similarity in (13) measures the ratio of pixels with the same assignment in the ground truth and the results by our approach. Exact matching and no-matching have an ER of 0% and 100% respectively. Table 1 shows the statistical data of ER. Experimental result by our approach is the closest to 0% in table 1. Our approach has performed better segmentation, potentially improving accuracy and reducing computational complexity of OCR algorithms. Experimentally, this approach provides a superior segmentation through reducing the noise remarkably from a corrupted color text images.

Input Images								
Ground Truth Images								
num.	Img1	Img2	Img3	Img4	Img5	Img6	Img7	Img8
size	141x256	119x256	143x256	127x256	71x256	68x256	145x256	131x256

(a) partial image data and ground truth to assess the performance



(b) the errors of our approach: FP (red) and ND (green)

Fig. 3. Performance comparison of our approach to other segmentation methods

Table 1. Performance comparision of four approaches with error rates(ER)

	Proposed method(%)	**EDISON(%)**	**Median(%) (5 x 5)**	**GMM(%)**
Img 1	**2.263**	5.128	4.267	5.112
Img 2	**0.653**	4.983	3.557	3.435
Img 3	**1.885**	3.286	5.294	4.125
Img 4	**2.300**	2.424	4.239	5.238
Img 5	**2.760**	4.671	3.563	5.234
Img 6	**4.906**	3.401	3.785	7.239
Img 7	**0.533**	1.456	1.448	4.234
Img 8	**0.951**	1.342	1.964	3.442

6 Conclusion

In our experiment, we have proposed a text image segmentation using tensor voting framework in 3D for corrupted text image by noise. The proposed method is a new method to automatically restore corrupted text images. Color features in the given image are defined with the corresponding hue and intensity component. Next, tensor voting framework is used for image analysis. Tensor voting analysis can detect the presence of noise such as crack or scrawl in a given image. Densification then generates the most proper values to replace the noise values which are present on texts. The improved image is used with a density estimation to find proper modes so such that K-means clustering algorithm can generate automatic seed values and perform text segmentation. Unlike other existent text segmentation methods, our approach can remove different kinds of noise well and segment a character as a single object. We have demonstrated very encouraging results on natural scenes using our method, and compared to existing methods. The result can contribute to improving text recognition rate as well as reducing the complexity of final step in OCR text recognition. This approach can then be extended to handle text recognition in natural scenes.

References

1. N.R. Pal, S. K. Pal, A review on image segmentation techniques, *Pattern Recognition*, vol. 26, No. 9, (1993)1277-1294.
2. A.Moghaddamzadeh, N.Bourbakis, A fuzzy region growing approach for segmentation of color images, *Pattern Recognition*, vol. 30, no. 6, (1997) 867- 881.
3. C. Zhang, P.Wang, A new method of color image segmentation based on intensity and hue clustering, *IEEE International Conference on Pattern Recognition*, vol. 3, (2000) 3617-3621.
4. K. Jain, B. Yu, Automatic Text location in images and video frames, *Pattern Recognition*, vol. 31, (1998) 2055-2076.
5. J. Zhang, X. Chen, J. Yang, A. Waibel, A PDA-based sign translator, *IEEE Int. Conf. on Multimodal Interfaces*, (2002) 217-222.
6. Q. Ye, W. Gao, Q. Huang, Automatic text segmentation from complex background, *IEEE Int. Conf. on Image Processing*, vol. 5, (2004) 2905-2908.
7. C. Li, X. Ding, Y. Wu, Automatic text location in natural scene images, *International Conference on Document Analysis and Recognition*, (2001) 1069-1073.
8. K. Wang, J. A. Kangas, Character location in scene images from digital camera, *Pattern Recognition*, vol. 36, (2003) 2287-2299.
9. S. M. Lucas, A. Panaretos, L. Sosa, A.Tang, S. Wong, R. Young, ICDAR 2003 robust reading competitions, *IEEE Int. Conf. on Document Analysis and Recognition*, 682-687, 2003.
10. G. Medioni, M.S. Lee, C.K. Tang, *A Computational Framework for Segmentation and Grouping*, Elsevier, 2000.
11. W.S. Tong, C.K. Tang, P. Mordohai, G. Medioni, First order augmentation to tensor voting for boundary inference and multiscale analysis in 3D, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 26, no.5, (2004) 569-611.

12. L. Lucchese, S.K. Mitra, Unsupervised segmentation of color images based on k-means clustering in the chromaticity plane, *IEEE Workshop on Content-based Access of Image and Video Libraries*, (1999) 74-78.
13. S. Sural, G. Qian, S. Pramanik, Segmentation and Histogram Generation using The hsv Color Space for Image Retrieval, *IEEE Int. Conf. on Image Processing*, vol.2, (2002) 589-592.
14. G. Guy, G. Medioni, Inference of Surfaces, 3-D Curves, and Junctions from Sparse, Noisy 3-D Data, *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 19, no. 11, (1997) 1265-1277.
15. J. Jia, C.K. Tang, Image Repairing: Robust Image Synthesis by Adaptive ND Tensor Voting, *IEEE Computer Vision and Pattern Recognition*, vol. 1, (2003) 643-650.
16. D. Comaniciu, P. Meer, Mean Shift: A Robust Approach Towards Feature Space Analysis, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, (2001) 1-18.
17. C.E. Rasmussan, The Infinite Gaussian Mixture Model, *Advances in Neural Information Processing Systems 12*, (2000) 554-560.

A Novel Coarse-to-Fine Adaptation Segmentation Approach for Cellular Image Analysis

Kai Zhang[1], Hongkai Xiong[1], Lei Yang[1], and Xiaobo Zhou[2]

[1] Institute of Image Communication and Information Processing
Shanghai Jiao Tong University
Dongchuan Road 800, Shanghai 200240, P.R. China
{zhangkai0619, xionghongkai, wangyaya}@sjtu.edu.cn
[2] Center for Bioinformatics, Harvard Center for Neurodegeneration and Repair,
Harvard Medical School
Functional and Molecular Imaging Center, Department of Radiology
Brigham & Woman's Hospital
Boston, MA 02215, USA
zhou@crystal.harvard.edu

Abstract. Cellular image content analysis is one of the most important aspects of the cellular research and often requires collecting a great amount of statistical information and phenomena. Automated segmentation of time-lapse images gradually becomes the key problem in cellular image analysis. To address fuzzy, irregular, and ruffling cell boundaries in time-lapse cellular images, this paper introduces a hierarchical coarse-to-fine approach which is composed of iteration-dependent adaptation procedures with high-level interpretation: initial segmentation, adaptive processing, and refined segmentation. The iteration-dependent adaptation lies in that the adaptive processing and the refined segmentation be deliberately designed without a fixed order and a uniform associated iteration number, to connect coarse segmentation and refined segmentation for locally progressive approximation. The initial segmentation could avoid over-segmentation from watershed transform and converge to some features using *a priori* information. Experimental results on cellular images with spurious branches, arbitrary gaps, low contrast boundaries and low signal-to-noise ratio, show that the proposed approach provides a close matching to the manual cognition and overcomes several common drawbacks from other existing methods applied on cell migration. The procedure configuration of the proposed approach has a certain potential to serve as a biomedical image content analysis tool.

Keywords: Image segmentation, content analysis, coarse-to-fine, iteration-dependent adaptation

1 Introduction

The cellular image content analysis is regarded as an important field of an investigation in disease mechanisms and signaling pathways at the cell and molecular biology levels. The typical scenario is that high resolution images of cancer cells be

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 762 – 771, 2007.
© Springer-Verlag Berlin Heidelberg 2007

used to determine the progression of cancer cell migration, aiming to indicate the invasion of cancer cells and cancer metastases [1]. With the increasing popularity of the automated fluorescence microscopy for the acquisition of time-lapse cellular images, large amounts of image datasets induce the traditional manual content analysis methods not to be feasible to operate the datasets. Thus, the image content analysis in time-lapse bioimaging urges highly automatic and fully adaptive representation methods mapping to the psychophysical and physiological characteristics. As an important geometric feature of shape representation, there exist a lot of image content analysis methods based on edge detection, wherein lie two major approaches for edge-based image segmentation: the watershed algorithm from mathematical morphology [2] and the minimization of certain energy function [3].

The watershed approaches are dependent on an immersion process analogy with edge evidence derived from the morphological gradient. Despite the underlined advantages in the proper operation of gaps and the orientation of the boundaries [4], the watershed algorithm is unacceptable for the dedicated cellular image content analysis because of its drawbacks with regard to a sensitivity to noise and a poor detection of thin structures and areas with low contrast boundaries. Unlike the watershed transform, snake-based methods behave as curves moving under the influence of internal forces from the curve itself and external forces from the image data [5]. However, the performance of snake-based method not only is highly restricted by the start position, but also has difficulties in tracing the boundary cavities.

Several automated approaches using morphological methods have already been proposed. In [6], a snake-based method has been introduced to extract axons boundaries. Another snake-based method has been proposed to analyze muscle fiber images [7]. However, none of these methods can provide us with satisfactory results when automatically analyzing microscopy images of cellular study. It is noted that time-lapse cellular images derived from automated fluorescence microscopy are in common with spurious branches, arbitrary gaps, low contrast boundaries and low signal-to-noise ratio. Compared to the conventional image segmentation, the underlined motivation makes urgent an appropriate segmentation approach for time-lapse cellular images that should not only behave with full automatism and reliability, but also be capable of dealing with low SNR images, especially addressing fuzzy, irregular, and ruffling cell boundaries. In this paper, we define a hierarchical analytic approach exploiting high-level interpretation which can be divided into three stages: attain a coarse boundary, refine an accurate boundary, and adjust certain iteration-dependent adaptive processing for locally progressive approximation.

2 Proposed Approach

As we have mentioned, our approach is composed of three steps: initial segmentation for shaping an initial boundary, adaptive processing for reducing the influence from image acquisition, and refined segmentation for attaining a closer boundary with the second degree continuation.

2.1 Initial Segmentation

We begin with defining a two-dimensional gray-scale image **I**. Assuming that an arbitrary pixel $\forall P \in \mathbf{I}$ has a gray level $G_P \in [0, N]$, where N stands for value of the highest gray level of **I**.

Pre-processing: After smoothing the original image with the help of a 3×3 window, we calculate the gradient image, which is the input of watershed algorithm. Let N_P denotes the neighborhood of pixel P, and the operation can be expressed as:

$$Grad_P = A * \sqrt{\sum_{P' \in N_P} (G_{P'} - G_P)^2} \tag{1}$$

where A is a constant.

Watersheds algorithm [4] is used to form a cell pixel set **W** and mark the boundary. Because the simple computation of image's watersheds mostly results in an over-segmentation, we introduce a threshold ($Grad_T$) here. Only when the input gradient image pixel's gray level $Grad_P$ is higher than $Grad_T$, we consider these pixels useful for computation. After this operation, pixel set $\mathbf{W} \subseteq \mathbf{I}$ is introduced for denoting cell pixels.

Post-processing: We can obtain certain high-level information from the cell biologist before we process these images. For instance, the size of cells appearing in the image is an important feature which can be used to distinguish the relevant parts of the study from the irrelevant parts. With this information, we can define a threshold for the quantity of pixels enclosed by the marked pixels. Thus, any cluster smaller than the threshold can be considered irrelevant as an environmental noise. Then we use a "filter" to detect and clear them. The "filter" with different surfaces is used for certain times, to remove all unwanted parts. In the following, we employ a dilation algorithm to fill in the "gap" inside the cell. A new set **W** is formed after dilation.

Then, we consider a set $\mathbf{B} \subset \mathbf{W}$ including the pixel which has more than one unmarked pixel and more than one marked pixel in its neighborhood as the boundary of cell. This property of set **B** can be described as:

$\mathbf{N}_{P_b} \cap (\mathbf{I} - \mathbf{W}) \neq \varnothing$, $\forall P_b \in \mathbf{B}$ and $\mathbf{N}_{P_w} \cap (\mathbf{I} - \mathbf{W}) = \varnothing$, $\forall P_w \in (\mathbf{W} - \mathbf{B})$, respectively.

After the post-processing, we get a smaller area **W** that contains cell and an initial boundary **B**. The size of the area is lager than we expected, and we have not obtained a clear boundary during this step. We need a further process to refine the boundary.

2.2 Adaptive Processing

To obtain a more accurate boundary, additional specific methods should be introduced. Those methods would be deliberately designed to connect coarse segmentation and refined segmentation, accompanied with anisotropic operation constraints from the adaptive processing and the refined segmentation presented in Section 2.3 dependent on the requisite validation.

Adaptive erosion with gradient information: A kind of adaptive erosion algorithm is adopted to deal with the accurate segmentation no matter what kind of boundary

the cell has. This algorithm starts from an initial boundary pixel set **B** given by Section 2.1. All boundary pixels $\forall P_b \in \mathbf{B}$ will move towards the inside part of the cell by certain distance, but no longer than a threshold denoted by D_M . D_M is decided by dilation algorithm stated in Section 2.1. The moving direction should be from set $\mathbf{I}-\mathbf{W}$ (outside of cell boundary) to set **W** (inside of cell boundary). A 3×3 operator (illustrated in Fig. 1) is used to decide the moving direction, in which pixels will move to the point with the highest gradient. The movements of pixel will generate a "trace" denoted by set T_{P_b} . We expand the width of trace to 3 pixels denoted by $\mathbf{T}'_{P_b}$ and convert the status of pixels covered by the trace from "inside pixel" to "outside pixel". This operation can be expressed by $\mathbf{W} = \mathbf{W} - \mathbf{T}'_{P_b}$, $\forall P_b \in \mathbf{B}$. After all boundary pixels have been moved, a new boundary pixel set **B** and a smaller **W** can be detected. Fig. 2 provides a typical example of the proposed algorithm.

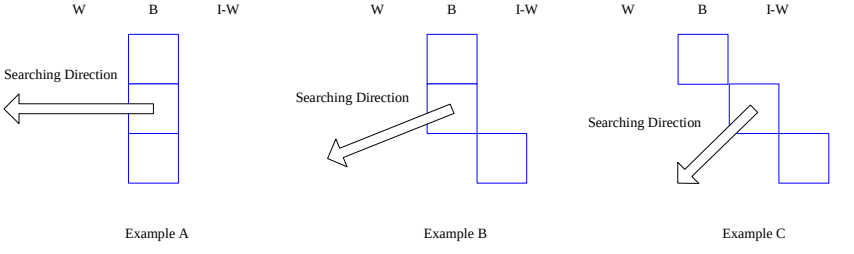


Fig. 1. Example of a searching direction operator

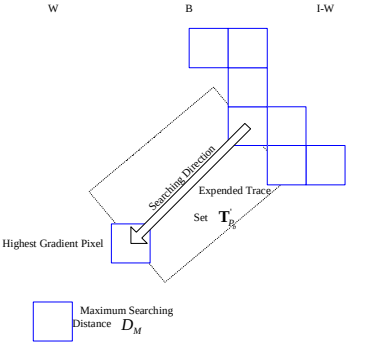


Fig. 2. Example of adaptive erosion algorithm with gradient information

Adaptive erosion of artificial part: Certain artificial noise and interference cannot be avoided during the image acquisition. Taking our data set as an example, the images contain some white parts which can be considered as noise and interference introduced by microscopy. As these parts have stronger boundaries than the real cell boundaries, they have destructive influence on the accuracy of segmentation. Still, we

can make good use of it and get a closer boundary set **B** . The erosion algorithm we have applied can be defined as follows:

For $\forall P_b \in \mathbf{B}$, remove P_b from set **W** if $\exists P' \in \mathbf{N}_{P_b}$ and $G_{P'} >$ threshold G_w .

After checking all the boundary pixels, a new and better boundary set **B** can be detected. To repeat this algorithm for R_W times till all the white parts could be removed.

2.3 Refined Segmentation

Refined segmentation is dedicated to getting a closer boundary while preserving detail information of the cell boundary. It is noted that the refined segmentation and the adaptive processing within the proposed hierarchical approach do not necessarily operate with a fixed order and the implicative number of the associated iterations can be chosen in terms of the requisite performance and the application requirements. Therefore, the proposed approach is presented to operate with an iteration-dependent adaptation.

Pre-processing: Because there are some spiky parts in the boundary, we use B-spline smooth to remove them in order to obtain a smooth boundary pixel set **B** for subsequent processing.

Greedy snake algorithm: An algorithm should be designed for detecting seed points of snake algorithm. These snake points should be stored in order. For example, if the image exists two cells, we use two arrays $\mathbf{A}_1$ and $\mathbf{A}_2$ to store the seed points. In each array, the seed point P_n's neighbor unit (P_{n+1} and P_{n-1}) should be its closest connected neighbor seed points in the boundary. Then, the greedy snake algorithm is used to get a more accurate boundary. Energy in this active contour model is represented by four energy terms E_{con} , E_{cuv} , E_{img} , and $E_{penalty}$. E_{con} and E_{cuv} is responsible for maintaining continuity between points by controlling segment length and vector curvature described by (2) and (3):

For $\forall P_n^{'} \in \mathbf{N}_{P_n}$:

$$E_{con} = \left| P_{n+1} - P_n^{'} \right| - \frac{1}{m} \sum_{P_x \in A_x} \left| P_{x+1} - P_x \right| \tag{2}$$

$$E_{cur} = \left| P_{n+1} + P_{n-1} - 2P_n^{'} \right| \tag{3}$$

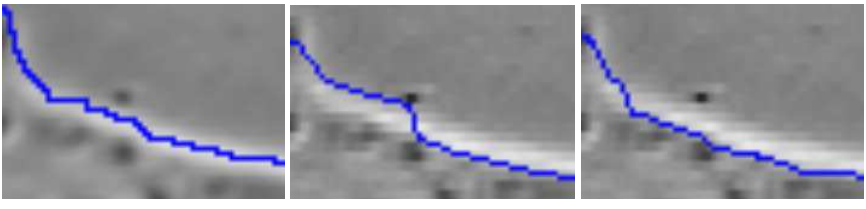
where m is the number of points in array $\mathbf{A}_x$.

Image energy E_{img} is represented as the magnitude of the gradient described by (4):

$$E_{img} = -Grad_{P_n^{'}} \tag{4}$$

The last energy term $E_{penalty}$ represents a special external constraint. Fig, 3(a) is the initial boundary given by Section 2.2 which has peak interference near the boundary.

Fig. 3(b) shows the processing result without the penalty term. The active contour moved to the outside noise point because the outside peak interference possesses stronger attractive force than the real cell boundary. In order to avoid this very common situation in cell image processing, we introduce $E_{penalty}$ into the ordinary snake model. Any movement to the outside of set $\mathbf{W}$ would be "punished" but not forbidden, then better performance (Fig. 3(c)) can be achieved as what are shown in the pictures.



(a) (b) (c)

Fig. 3. Example of boundary with nearby peak interference

The total energy of $\forall P_n^{'} \in \mathbf{N}_{P_n}$ is described by (5):

$$E_{P_n^{'}} = E_{con} + E_{cuv} - \beta E_{img} + E_{penalty} \tag{5}$$

β is a constant usually taken as 1.2-1.8.

where $E_{penalty} = \gamma \left| E_{con} + E_{cuv} - \beta E_{img} \right|$ and γ is the penalty function which can be described by (6):

$$\gamma = \begin{cases} -\dfrac{C}{D} & P_n^{'} \in \mathbf{W} - \mathbf{B} \\ 0 & P_n^{'} \in \mathbf{B} \\ CD^2 & P_n^{'} \in \mathbf{I} - \mathbf{W} \end{cases} \tag{6}$$

where C is a constant and D is the distance between $P_n^{'}$ and set $\mathbf{B}$. Then the algorithm can be represented as: For all $P_n \in \mathbf{A}_x$ and $P_n^{'} \in \mathbf{N}_{P_n}$ which has the minimum energy in set $\mathbf{N}_{P_n}$, if $E_{P_n} > E_{P_n^{'}}$ then replace P_n with $P_n^{'}$. It is inferred that we are able to use $\mathbf{A}_1, \mathbf{A}_2 ..... \mathbf{A}_X$ to generate new boundary pixel set $\mathbf{B}$ and cell pixel set $\mathbf{W}$.

3 Experimental Results

The proposed approach has been validated by applying it to a representative image of cell migration. Fig. 4 includes two 3T3 cells which were cultured in DME (Dulbecco-Vogt's modified Eagle's medium) with 5% donor bovine serum in the presence of penicillin and streptomycin. These two cells contain all types of boundaries in target images, namely, contrast, spiky, fuzzy and ruffly. Thus, we can test our approach on each type in an image.

As shown in Fig. 5, we can see that the result of watershed transform will not be satisfactory enough for biomedical images content analysis. Because of the environmental noise introduced during image acquisition, a simple application of watershed transform yields to results greatly influenced by superfluous noise and some areas of over-segmentation. After employing post-processing of watersheds, we get the result that is presented by Fig. 6. As to the parameter $Grad_T$, we set the value to 4. Notice that the cell pixel set **W** we obtain here should cover all parts of the interested cell for further refine.

Fig. 7 and Fig. 8 show the results after the adaptive processing. In our experiments, we select $D_M = 7$ and $R_W = 6$. A spiky but more accurate boundary set **B** is generated after Section 2.2.



Fig. 4. Original image of two 3T3 cell

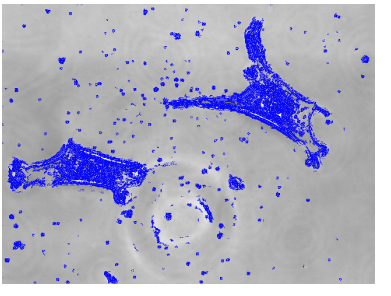


Fig. 5. Image segmented by watershed transform

With snake algorithm companied by additional pre or post-processing procedure, we obtain a cell boundary with second degree continuation which facilitates the following researches. For the final result, with certain amount of experiments on the parameters in the energy function applied in the snake algorithm, we have concluded the best choice of these parameters which yield to the best segmentation results. Our final segmentation result is presented in Fig. 10. The process of refine segmentation can be iterated for a couple of times if necessary for more accurate result. Fig. 11 presents the manual segmentation result. We evaluate our approach by comparing the final segmentation result with manually segmented image. Obviously, the two results are very close except some minor differences. Quantitative result is given in Table 1 by measuring the percentage of the overlapping area of automated and manual segmentation. We can clearly see that the result is ameliorated after every step of our approach.

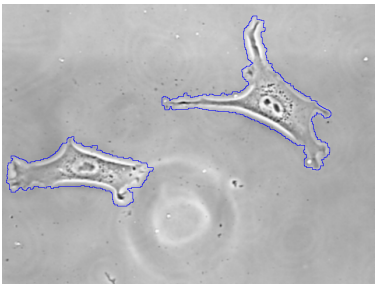


Fig. 6. Result after initial segmentation

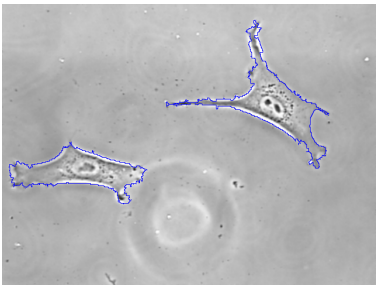


Fig. 7. Result after adaptive erosion

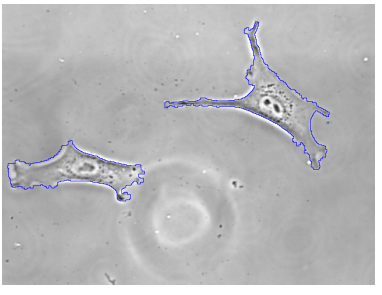


Fig. 8. Result after white erosion



Fig. 9. Initial position of seed points



Fig. 10. Final segmentation result



Fig. 11. Manual segmentation result

Table 1. Similarity between automated and manual segmentation (presented in the form of overlapping area percentage)

Initial segmentation		Adaptive processing		Refine segmentation	
Fig. 5	Fig. 6	Fig. 7	Fig. 8	Results after first iteration	Results after second iteration(Fig. 10)
72.87%	80.36%	89.90%	92.81%	94.88%	95.05%

4 Conclusion

In our paper, we have introduced a coarse-to-fine segmentation approach with an iteration-dependent adaptation for the extraction of cell boundaries from gray-value images with low contrast edges and greatly influenced by environmental noise and interference. Our approach is composed of three steps, a coarse segmentation using watershed transform with pre-process and post-process, a refined segmentation using B-spline curve smoothing and greedy snake model and an adaptive processing method connect them together. The last two steps are considered iteration-dependent, which means that the respective iteration times and parameters are both demand-adaptive.

From the illustrations above, we have shown that the robustness of our approach against the environmental noise and interference and ability of extract low contrast edges. Comparison between classical approaches, such as snakes and watershed, and our approach shows that with high-level interpretation explored and utilized, our approach yields much better results in image segmentation especially for cell images. We can believe in the potential of our approach becoming one of the basic tools in cell image content analysis. Furthermore, we also expect new improvements of our approach and its application in other domains.

Acknowledgments. The authors would like to thank the fruitful discussions with Dr. Xiaobo Zhou and his biology collaborators in the Department of Cell Biology at Harvard Medical School, and thank them for providing the cancer cells migration data set.

References

1. Annie C. Mathew, Talitha T. Rajah, Gina M. Hurt, S. M. Abbas Abidi, John J. Dmythryk and J. Thomas Pento. Influence of antiestrogens of the migration of breast cancer cells using an in vitro wound model. *Clinical and Experimental Metastasis*, 15(4), 1997
2. J. B. T. M. Roerdink and A. Meijster: “The watershed transform: Definitions, algorithms and parallelization strategies,” Fundamental Information, 41: 187-228, 2000
3. Donna J.Williams and Mubarak Shah: “A fast algorithm for active contours”, Computer Vision, 1990. Proceedings, Third International Conference on 4-7 Dec. 1990:592-595
4. Luc Vincent and Pierre Soille: “Watersheds in digital spaces: an efficient algorithm based on immersion simulations”, I *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(6):583-598, June 1991
5. Foued Derraz, Mohamed Beladgham and M’hamed Khelif: “Application of active contour models in medical image segmentation”, Proceedings of the international conference on information technology: Coding and Computing (ITCC’04), 2004, 2: 675-681
6. Y. L. Fok, J. C. K. Chan, and R. T. Chin, "Automated analysis of nerve-cell images using active contour models," *IEEE Transactions on Medical Imaging*, 15:353-368, 1996
7. Klemencic, S. Kovacic, and F. Pernus, "Automated segmentation of muscle fiber images using active contour models," *Cytometry*, 32: 317-326, 1998
8. Jaesang Park and James M. Keller. Snakes on the watershed. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(10):1201-1205, October 2001

9. Hieu Tat Nguyen, Marcel Worring, and Rein van den Boomgaard. Watersnakes: Energy-driven watershed segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(3):330-342, March 2003
10. Olivier Cuisenaire. Locally adaptable mathematical morphology. IEEE International Conference on Image Processing, 2005. ICIP 2005. 2: 125-128, 11-14 September 2005
11. Farhang Sahba, Hamid R Tizhoosh and Magdy M Salama. A coarse-to-fine approach to prostate boundary segmentation in ultrasound images. BioMedical Engineering Online, 11 October 2005

Vehicle Classification from Traffic Surveillance Videos at a Finer Granularity

Xin Chen and Chengcui Zhang

Department of Computer and Information Sciences
University of Alabama at Birmingham
Birmingham, AL 35294, USA
{chenxin, zhang}@cis.uab.edu

Abstract. This paper explores the computer vision based vehicle classification problem at a fine granularity. A framework is presented which incorporates various aspects of an Intelligent Transportation System towards vehicle classification. Given a traffic video sequence, the proposed framework first segments individual vehicles. Then vehicle segments are processed so that all vehicles are along the same direction and measured at the same scale. A filtering algorithm is applied to smooth the vehicle segment image. After these three steps of preprocessing, an ICA based algorithms is implemented to identify the features of each vehicle type. One-class SVM is used to categorize each vehicle into a certain class. Experimental results show the effectiveness of the framework.

Keywords: ICA, vehicle classification, Intelligent Transportation Systems.

1 Introduction

Due to its great practical importance, Intelligent Transportation Systems has been an active research area for years. Vehicle classification is one of the key tasks in an Intelligent Transportation System. Typically, acoustic or seismic sensors are used for such a purpose [1][8][13][14]. However, for road traffic analysis, the most available sources are traffic surveillance videos taken by fixed cameras. Since only the visual information can be reliably extracted and verified for such videos, computer vision based methods from the area of multimedia are required for video content analysis.

In order to identify vehicles, video object tracking needs to be performed before we can analyze each individual vehicle. There are a large amount of literatures on vehicle tracking based incident detection for traffic surveillance system. However, there has been relatively little work done in the field of vehicle classification. This is because it is an inherently hard problem. Some vehicle detection and tracking works even depend on classification techniques. [16] proposes a vehicle detection method with one of its step being classification i.e. a two class classification of vehicles and non-vehicles. A method called “Adaboost” is used for such a purpose.

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 772 – 781, 2007.
© Springer-Verlag Berlin Heidelberg 2007

In [15], an object tracking and classification method is proposed. Three categories of objects are differentiated – human, automobiles and background. For classification of human and automobiles, a concept called "dispersedness" is used based on the priori that human has smaller yet more complex shape than that of a vehicle. This is among one of the earliest works that address object classification from video. Most of the current work is purely dimension based (such as height and length of a vehicle) or shape based. In [7], a parameterized model is proposed to describe vehicles, in which vertices and topological structure are taken as the key features. One requirement of this method is that the image quality has to be sufficiently good to have the topological structures of vehicles exposed. However, this cannot be always satisfied in a real traffic surveillance system. Gupte et al. [3] propose a system for vehicle detection and classification. The tracked vehicles are classified into two categories: cars and non-cars. The classification is based on dimensions and is implemented at a coarse granularity. Its basic idea is to compute the length and the height of a vehicle, according to which a vehicle is classified as a car or a non-car. In order to classify vehicles at a finer granularity, we need a more sophisticated method that can detect the invariable characteristics for each vehicle type. In [6], the virtual loop assignment and direction-based estimation methods are used to identify vehicle types. Each vehicle type is represented by a 1-D signature chart. In their experiment, vehicles are classified into four categories: 7-seat van, fire engine, sedan and motor cycle. With this method, as mentioned in the paper, only a rough estimation of vehicle types based on vehicle length is possible. It cannot distinguish vehicles whose lengths are in approximately the same range, e.g. truck and bus. Another problem of this method is that only those vehicles traversing across virtual loops along the road direction can be detected. Therefore, we still need to further explore a method that can unveil the real, invariant characteristics of a type of vehicle. In this paper, we design an algorithm for vehicle classification at a finer granularity.

Principal Component Analysis (PCA) is a well-known algorithm in the field of object recognition. It used in the computer vision problem of human face recognition. The similarity between face detection and vehicle detection is that both analyze a 2-D image and try to find out the feature of the image content.

Independent Component Analysis (ICA) is another subspace method that has been applied to face recognition. Many works compare between ICA and PCA and show the advantages of ICA [2][4][5]. In [2], the authors applied both methods in analyzing the Coil-20 database. In [4], the authors demonstrate that ICA outperforms PCA for object recognition under varying illumination. [5] compares the effectiveness of both methods in object recognition. Since the traffic videos are taken during different time periods of the day, it is preferably that the algorithm is robust to varying illumination conditions. In this paper, we propose an ICA based vehicle classification platform.

By analyzing vehicle images with ICA, a set of features are extracted for each vehicle. These features represent the innate characteristics of the vehicle type and are fed into the classification module -- One-Class Support Vector Machine [9]. The representative features of vehicles in each vehicle type are used as training data. We build one classifier for each vehicle type which distinguishes that vehicle type from the others. In the testing phase, each set of test vehicles is tested against the classifier of each vehicle type. A test vehicle is then classified into one of the vehicle types according to the highest score it receives from each classifier.

In our experiments, we use grayscale traffic surveillance videos. It is desired that the classification is robust to the varying intensities of vehicles. For example, black and white passenger cars are expected to be classified into the same class. However, their different intensities may affect the classification result. Therefore, a filter in the preprocessing step is necessary to alleviate such problems. In this paper, a texture analysis tool is used for this purpose.

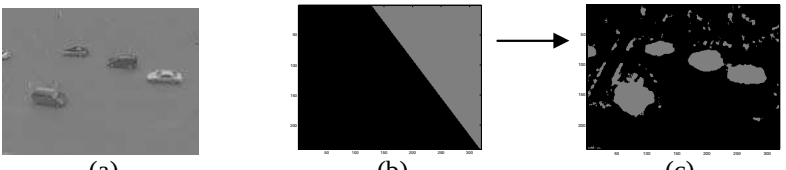
We propose an integrated system that can automatically track and categorize vehicles within a traffic surveillance video sequence. The system first tracks and segments vehicles from raw surveillance videos. Then the tracked vehicles and their features are normalized. In the final step vehicles are classified, which can provide more detailed and useful information to traffic administration. The vehicle tracking and normalization phases are based on Zhang et al.'s work [11]. In this study, improvement in the classification result by using ICA and one-class SVM is demonstrated by the experimental results at the end of this paper.

The detailed design and implementations are illustrated in the following order: Section 2 briefly introduces preprocessing module -- vehicle segmentation, adjustment and filtering. Section 3 discusses the technical details of the algorithm. Section 4 presents the system overview and the experimental results. Section 5 concludes the paper.

2 Preprocessing

2.1 Vehicle Tracking and Segmentation

For vehicle tracking and segmentation, an unsupervised video segmentation method called the Simultaneous Partition and Class Parameter Estimation (SPCPE) algorithm, coupled with a background learning algorithm, is applied to identify the vehicle objects in video sequences [10]. Figure 1 shows an example of vehicle segmentation from the initial random partition (Figure 1(a)) to the final segmentation result (Figure 1(c)).



(a) (b) (c)

Fig. 1. An example of vehicle segmentation. (a) Original frame with background removed; (b) Initial random partition; (c) Final segmentation result.

The algorithm in [10] also has the ability to track moving vehicle objects (segments) within successive video frames. By distinguishing the static objects from mobile objects in the frame, tracking information can be used to determine the trails of vehicle objects.

2.2 Vehicle Image Adjustment and Filtering

For normalization purposes, a transformation model is needed to rotate the subject cars to the same orientation and scale them to the same level. For vehicles driving toward the same direction, their rotation angles are the same. The scaling factor is determined by the shooting distance between the camera and the vehicle object. Once the rotation angle θ and the scaling factor s are available, the transformation model can be built. To preserve the co-linearity (i.e., all points lying on a line initially should still lie on a line after transformation) and the ratios of distances within the image, we use the affine transformation as our transformation function to rotate and scale vehicle objects to comparable conditions. The affine transformation is defined as follows:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = s \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x - x_0 \\ y - y_0 \end{bmatrix} \tag{1}$$

where θ is the rotation angle and s is the scaling factor. After applying the affine transformation, we make all subject vehicles in consistent orientation and at the same scale level. This module is implemented based on Zhang et al.'s work in [11].

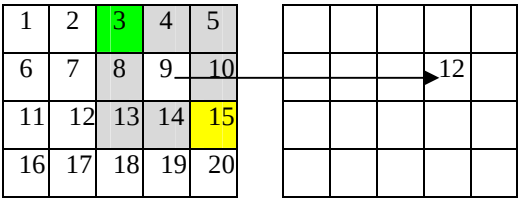


Fig. 2. Local range filtering

Although the vehicle images are transformed to grayscale images, there is still a difference with respect to intensities between bright colored (e.g. white) and dark colored (e.g. red or black) vehicles. The change of lighting conditions during the day can also cause the variations in image intensities. As mentioned in Section 1, ICA is comparatively robust in dealing with varying illuminations. Furthermore, in order to alleviate the effect of varying intensities, a filtering technique is used in the proposed framework. It calculates the local range of an image and tries to smooth out pixels within the same neighborhood. Suppose we use a 3 by 3 neighborhood window. The above figure shows the mechanism of this texture based filter.

The shaded area is the neighborhood of the pixel whose intensity value is 9. After filtering, its intensity in the corresponding position is 12 which is the difference of the maximum intensity (15) and the minimum intensity (3) of its neighborhood pixels.

Figure 3(a) is an example of a vehicle image. Figure 3(b) is the filtered image. After filtering, the outline of the vehicle is evident. In a neighborhood area, if the intensity difference is small, the whole area is smoothed to a dark patch. Otherwise, the area is highlighted such as the skeletons of vehicles. Therefore, the original color of the vehicle will not matter that much (as before); only its outline information is kept. Thus, the influence of the vehicle's original color is alleviated.



(a) Original Vehicle Image



(b) Filtered Vehicle Image

Fig. 3. An example of filtered image

3 Classification

3.1 Obtain Vehicle Samples

After vehicle segmentation, the bounding boxes of vehicle segments are extracted. One factor we need to take into consideration is that the sizes of bounding boxes are different due to different vehicle sizes. This factor can affect the result of the next step – Independent Component Analysis. Therefore, we set a uniform bounding box whose size is the biggest bounding box among all samples. For those whose bounding boxes are smaller, we pad them with the mean values of their background pixels surrounding the vehicle segments. In this way, we obtain a set of training samples for each type.

Each vehicle sample is actually a 2-D image $x_i \in \Re^{m \times n}$. It can be represented as an *m* by *n* vector with *m* being the image height and *n* being the image width. We then read x_i in column-wise order, one pixel at a time, and restructure it as $x_i^{'} \in \Re^{1 \times mn}$. With *k* being the number of samples in the training set, we can have a matrix of *k* columns $X^{'} = [x_1^{'}, x_2^{'}, ..., x_k^{'}]$. The length of each column is *mn*. The mean vector ω is calculated as follows:

$$\omega = \frac{1}{k} \sum_{i=1}^{k} x_i^{'} \tag{2}$$

Since ω is also a $1 \times mn$ vector, we can restore it into an *m* by *n* matrix and output it as an image. The mean "passenger car" constructed this way is shown in Figure 4. By deducting the mean vector from each vehicle image vector $x_i^{'}$, $X^{'}$ becomes a zero mean matrix, which is the random dataset we will analyze later.

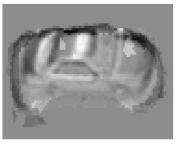


Fig. 4. The mean image of passenger car samples

3.2 Independent Component Analysis

The Independent Component Analysis (ICA) is a statistical method for revealing the underlying factors of a set of data, which are mutually independent. ICA views

data as a linear mixture of sources i.e. independent components. There is little knowledge of the sources and how they are mixed. The only information we have is the observed random dataset. In order to separate the independent sources, ICA seeks an un-mixing matrix for linearly transforming to coordinates in which data are maximally statistically independent. ICA is often compared with a well known method – Principle Component Analysis (PCA) which is used to find the orthogonal bases of dataset. With PCA, data are decorrelated by being projected onto these bases. Although both ICA and PCA explore subspaces to decorrlate data, the purpose of ICA is theoretically loftier than that of PCA since ICA tries to find an un-mixing matrix such that sources are not only decorrelated but also statistically independent. Some research results have shown the advantage of ICA over PCA [2][4][5].

In ICA model, the random dataset is denoted as:

$$X' = AS \tag{3}$$

where X' contains k observed data points $[x_1, x_2, \ldots, x_k]$. In our case, x_i is a vehicle image represented by a vector. k is the number of training samples in the training set. A is the mixing matrix and S is the matrix containing the independent components that are mixed by A to represent the observed dataset X'. All we observe is the random dataset X'. A and S must be estimated according to X'. In our experiment, a fixed point version of this algorithm – FastICA [12] is used. Our assumption is that the independent components have nongaussian distributions. After estimating A, its inverse W can be computed and the independent components S is obtained by the following equation:

$$S = WX' \tag{4}$$

The length of each independent component is mn. Similarly to how we construct the mean image, we can reconstruct this vector into a 2-D image. For vehicle classification, the independent components in S are used as the bases for a low-dimensional representation. For each sample vehicle image in the training set, the following equation is used to compute its weight vector consisting of the weight of each independent component in representing that vehicle image.

$$\beta = S^T X' \tag{5}$$

The rows of β are weight vectors of vehicle images in the training set. These weight vectors are normalized to the scale of [0, 1] to avoid bias.

3.3 One-Class Support Vector Machine

One-Class classification is a kind of supervised learning mechanism. It tries to assess whether a test point is likely to belong to the distribution underlying the training data. In our case, a training set is composed of a set of vehicles of the same type. One-Class SVM has so far been studied in the context of SVMs. The objective is to create a binary-valued function that is positive in those regions of input space where the data predominantly lies and negative elsewhere.

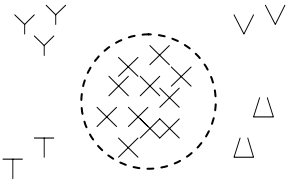


Fig. 5. One-Class classification

The idea is to model the dense region as a "ball". Vehicles that belong to the class are inside the "ball" and the others are outside. This is shown in Figure 5 with the crosses representing the data that belongs to the positive class. If the origin of the "ball" is $\vec{\alpha}$ and the radius is *r*, a point $\vec{p}_i$ is inside the "ball" iff $\left\|\overrightarrow{p_i} - \vec{\alpha}\right\| \le r$. In our case, a point is the weight vector that represents the features of a vehicle. This "ball" is actually a hyper-sphere. The goal is to keep this hyper-sphere as "pure" as possible and include as many vehicles that belong to this class as possible. Details can be found in Schölkopf's One-Class SVM [11].

The process of classifying a new (unknown) vehicle x_{new} to one of the classes (known vehicles) proceeds in three steps:

1. Train a set of One-class SVM classifiers with the weight vectors of the sample images in the training sets. A classifier is generated for each vehicle type.
2. Reshape x_{new} into x'_{new} and obtain $\sigma_{new} = x'_{new} - \omega$. Transform σ_{new} with the independent components of the training set and obtain the feature vector β_{new} (weight vector) by Equation 5.

Test β_{new} against each classifier generated in the first step and obtain a set of scores which indicates the possibility of x_{new} belonging to each vehicle type. Finally, x_{new} will be classified into the vehicle type from which it receives the highest score.

In our experiment, there are three training sets, one for each type of vehicles: passenger car, pick-up and van. Each type of vehicles is represented by a set of weight vectors and trained by One-class SVM. Then we use the trained One-class SVM classifiers to classify new vehicles.

4 Experimental Results

From vehicle tracking and segmentation to vehicle classification, we now have an integrated system that can automatically track and classify vehicles in traffic surveillance videos. A real-life traffic video sequence with 67635 frames is used to

analyze the performance of the proposed vehicle classification algorithm. The video sequence is obtained from a high way survelliance camera.

By vehicle segmentation and tracking, all distinct vehicle segments are extracted and form a sample pool. By "distinct", we mean each vehicle segment in the sample pool corresponds to a real distinct vehicle in reality. For repetitive appearances of a vehicle object across multiple frames, only one instance (segment) of that vehicle is chosen for training or testing. The preprocessing step is time consuming and is performed offline. ICA Analysis step requires some mannul work i.e. selecting the training samples and therefore is also executed offline. The classification step can work in real time.

In our experiment, three sets of training samples are formed for three categories of vehicles. They are "passenger cars (PC)", "pickup trucks (PK)" and "vans and SUVs (VAN)". In each training set, there are 50 vehicles. It is worth mentioning that, the system can be easily extended to detect more categories of vehicles. The only modification for this is to gather samples for each category of vehicles.

Table 1. The Test Result with ICA

ICA-SVM		**Test 1**	**Test 2**	**Test 3**
PC	***Recall***	74%	66%	64%
	Precision	84.60%	82%	78%
PK	***Recall***	68%	56%	70%
	Precision	72.3%	72%	83.3%
VAN	***Recall***	64%	58%	74%
	Precision	74.70%	68%	79%

We have three sets of test samples with each containing 150 vehicles randomly chosen from the sample pool. Table 1 shows the precision and recall values of the proposed ICA-based algorithm and the test result of using the PCA-based algorithm is presented in Table 2.

Table 2. The Test Result with PCA

PCA-SVM		**Test 1**	**Test 2**	**Test 3**
PC	***Recall***	40%	66%	54%
	Precision	57.3%	71.3%	62.7%
PK	***Recall***	54%	56%	52%
	Precision	62%	62.7%	57.3%
VAN	***Recall***	64%	54%	66%
	Precision	73.3%	64%	69.3%

From the above two tables we can see that ICA performs better than PCA. It is worth mentioning that the precision of ICA-based algorithm is much higher than that of PCA. This is because ICA can better identify negative samples in the testing data

set. The system proposed in this paper incorporates video segmentation, vehicle tracking, and vehicle classification into one single integrated process. Especially, the classification is designed to find the invariant features of vehicles so as to categorize them at a fine granularity.

5 Conclusion

In this paper, a vehicle classification framework is proposed which incorporates several stages of work. First, traffic video sequence is processed to extract vehicle segments, which provides a means for vehicle tracking and classification. Secondly, vehicle segments are normalized so that all vehicles are along the same direction and uniformly scaled. A texture analysis technique is then used to filter the vehicle images. The final stage is classification, in which an ICA-based algorithm is applied. We choose ICA because of its ability to find inner characteristics of a group of data. The ICA based algorithm is compared with a well-known subspace analysis technique – PCA. Experimental results show that given a sufficient amount of sample data our system can effectively categorize vehicles at a fine granularity.

Acknowledgement

The work of Chengcui Zhang was supported in part by SBE-0245090 and the UAB ADVANCE program of the Office for the Advancement of Women in Science and Engineering.

References

1. Marco, F., Yu, H.H.: Vehicle Classification in Distributed Sensor Networks. Journal of Parallel and Distributed Computing. V64:7, (2004).
2. Sahambi, H. S., Khorasani, K. A: Neural-Network Appearance-Based 3-D Object Recognition Using Independent Component Analysis. IEEE Trans. on Neural Networks. vol. 14, no. 1, (2003), pp. 138-149.
3. Gupte, S., Masoud, O., Martin, R. F. K., Papanikolopoulos, N. P.: Detection and Classification of Vehicles. IEEE Trans. on Intelligent Transportation Systems. v3:1, (2002), pp. 37-47.
4. Fortuna, J., Schuurman, D., Capson, D.: A Comparison of PCA and ICA for Object Recognition under Varying Illumination. Proc. of 16th International Conference on Pattern Recognition. vol. 3, (2002), pp 11-15.
5. Sezer, O. G., Ercil, A., Keskinoz, M.: Subspace Based Object Recognition Using Support Vector Machines. Proc. of European Signal Processing Conference (EUSIPCO), (2005).
6. Lai, A. H. S., Yang, N. H. C.: Vehicle-Type Identification through Automated Virtual Loop Assignment and Block-Based Direction-Biased Motion Estimation. IEEE Trans. on Intelligent Transportation Systems, v1:2, (2000), pp. 86-97.
7. Wu, W., Zhang, Q., Wang, M.: A Method of Vehicle Classification Using Models and Neural Networks. Proc. of IEEE 53rd Vehicular Technology Conference. Vol. 4, (2001), Rhodes, Greece, pp. 3022-3026.

8. Harlow, C., Peng, S.: Automatic Vehicle Classification System with Range Sensors. Transportation Research Part C: Emerging Technologies. Vol. 9, No. 4, (2001), pp. 231-247.
9. Schölkopf, B., Platt, J. C. et al.: Estimating the Support of a High-dimensional Distribution. Microsoft Research Corporation Technical Report MSR-TR-99-87, (1999).
10. Chen, S.-C., Shyu, M.-L, Sista, S., Zhang, C.: Learning-Based Spatio-Temporal Vehicle Tracking and Indexing for Transportation Multimedia Database Systems. IEEE Trans. on Intelligent Transportation Systems, v4:3, (2003), pp. 154-167.
11. Zhang, C., Chen, X., Chen, W.-B.: A PCA-based Vehicle Classification Framework. Proc. of IEEE International Workshop on Multimedia Databases and Data Management, in conjunction with IEEE International Conference on Data Engineering (ICDE 2006), Atlanta, Georgia, USA.
12. Hyvärinen, A., Oja, E.: A Fast Fixed-point Algorithm for Independent Component Analysis. Neural Computation, v9:7, (1997), pp. 1483-1492.
13. Abdelbaki, H.M., Hussain, K., Gelenbe, E.: A Laser Intensity Image Based Automatic Vehicle Classification. Proc. of IEEE Intelligent Transportation Systems, (2001), Oakland, CA, U.S.A, pp. 460-465.
14. Nooralahiyan, A.Y., Kirby, H.R., Mckeown, D.: Vehicle Classification by Acoustic Signature. Mathematical and Computer Modeling, vol. 27, No. 9, (1998), pp. 205-214.
15. Lipton, A.J., Fujiyoshi, H., Patil, R.S.: "Moving Target Classification and Tracking from Real-time Video", Proc. of Fourth IEEE Workshop on Applications of Computer Vision, Princeton, NJ, USA, pp.8-14, Oct 19-21, 1998.
16. Khammari, A., Nashashibi, F., Abramson, Y., Laurgeau, C.: "Vehicle Detection Combining Gradietn Analysis and AdaBoost Classification", Proc. of 8th International Conference on Intelligent Transportation Systems, pp. 66-71, Vienna, Austria, Sept. 13-15, 2005.

A Fuzzy Segmentation of Salient Region of Interest in Low Depth of Field Image

KeDai Zhang[1], HanQing Lu[1], ZhenYu Wang[1], Qi Zhao[2], and MiYi Duan[2]

[1] National Laboratory of Pattern Recognition Institute of Automation, CAS
[2] Beijing Graphics Institute, Beijing, China
kdzhang@nlpr.ia.ac.cn

Abstract. Unsupervised segmenting region of interest in images is very useful in content-based application such as image indexing for content-based retrieval and target recognition. The proposed method applies fuzzy theory to separate the salient region of interest from background in low depth of field (DOF) images automatically. First the image is divided into regions based on mean shift method and the regions are characterized by color features and wavelet modulus maxima edge point densities. And then the regions are described as fuzzy sets by fuzzification. The salient region interest and background are separated by defuzzification on fuzzy sets finally. The segmentation method is full automatic and without empirical parameters.

Keywords: Image segmentation, Fuzzy theory, Mean shift, Wavelet modulus maxima.

1 Introduction

Image segmentation is the first and important phase in analyzing and understanding the content of an image. However, it is difficult to isolate the meaningful region from the scene without a priori knowledge. Designing a general segmentation algorithm for all images is nearly impossible at the present time.

Low depth of field (DOF) is an important technique widely used by professional photographers. The sharpness in the image of objects in front of and behind the focused distance falls off gradually. Within a certain range of object distances this sharpness loss is still comparatively unnoticeable. This range is the depth of field (DOF) [1]. Low DOF is one of the main techniques used by professionals to simplify their photographs and focus attention on the intended subject of the picture. It can eliminate a distracting background by throwing it out of focus. In low DOF image, the interested object is sharply focused, whereas background objects are blurred to out-of-focus. The observer's attention can be easily concentrated on the focused region of the pictures. In this paper, we aim at the segmentation of salient interested region in image with low DOF. This research can be applied to many content-based applications such as content-based image retrieval and target recognition.

In our method, we use mean shift method to divide the image into regions firstly, which are characterized by color features and wavelet modulus maxima edge point densities. And then the regions are described as fuzzy sets by fuzzification. At last,

T.-J. Cham et al. (Eds.): MMM 2007, LNCS 4351, Part I, pp. 782 – 791, 2007.
© Springer-Verlag Berlin Heidelberg 2007

the salient region interest and background are separated by defuzzification on fuzzy sets. We apply fuzzy theory to the segmentation procedure due to the following factors:

- The uncertainty in the quantization of the color components and the human perception of colors,
- The imprecision of color region segmentation,
- The inaccuracy of wavelet modulus maxima edge location,
- In the low DOF image, a few background things nearby the object of interest are focused too.

The outline of this paper is as follows: In Section 2, we review the related works. Our approach is described in detail in Section 3. In Section 4, the experimental results on real-world image data are presented. Finally, the paper is concluded in Section 5.

2 Related Work

There were many works taking advantages of DOF in the field of computer vision such as depth from focus and shape from focus (image focus and defocus analysis) [2][3][4]. The level of focus or defocus of the objects was taken accounted in determining distance in this application.

In segmentation applications, Tsai et al. [7] proposed an edge-based segmentation approach. The approach detects the focused objects relying on the measurement of defocus for object edges in an image. The amount of defocus at each edge pixel is evaluated using the moment-preserving principle, and the regions of focused objects can be bounded by closed boundaries followed by the edge-linking procedure. The approach has demonstrated high accuracy for objects with clear boundary edges. However, it is not suitable for segmenting natural objects with blunt edges because of the dependence of an edge-linking procedure to find the closed boundaries in the algorithm.

J.Z. Wang and Jia Li [5] designed a multiscale segmentation algorithm based on high frequency wavelet coefficients and statistics feature. The focused object regions have more high value wavelet coefficients in the high frequency bands of the transform. The algorithm first classifies the large image blocks into interest object and background according to high frequency wavelet coefficients. Then the crude classification result is adjusted using statistic context information through a multiscale approach. Their algorithm may be robust because it does not rely on the colors or the boundary edges. But if the interest object is highly smooth or the initial classification is incorrect, their algorithm may fail.

Two of the above both aim at gray level image. They don't use the color information of the image. We have proposed an automatic algorithm [6] combining the sharp edges of the focused object and color regions information together to separate the focused object from the blurred background. The experimental results of that algorithm are good, but some empirical parameters are required in it. Furthermore, if the inner of the focused object highly smooth or the object having blur boundary, some regions of the object may be segmented as background.

3 The Proposed Method

We take use of the characteristic of interested object sharply focused and background object blurred in low DOF image to isolate the salient region of interest from the background.

As shown in Fig.1, the segmentation algorithm is comprised of five steps:

1. Segmenting color regions based on mean shift method,
2. Detecting wavelet modulus maxima edges,
3. Calculating the wavelet modulus maxima edge point density for each region, which is together with average color representing the region,
4. Fuzzification: construct the fuzzy sets of salient region of interest and background, and calculate fuzzy membership degrees of them for each region.
5. Defuzzification: classifying the regions into salient region interest and background according to fuzzy membership degree.

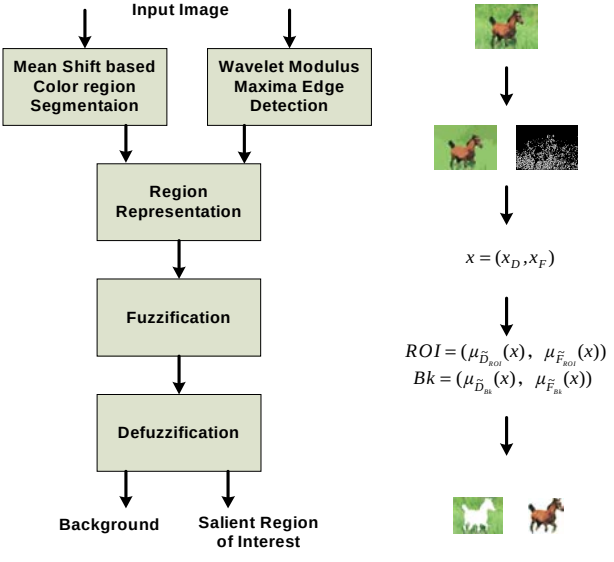


Fig. 1. The Framework of the Segmentation

3.1 Mean Shift Based Color Region Segmentation

In this paper we use mean-shift clustering [8, 9, 10] in the color feature space for segmentation. It was early proposed by Fukunaga et al. [8], and introduced comprehensively into computer vision field for clustering, objecting tracking, and segmentation by D. Comaniciu and P. Meer [9, 10, 11]. The idea is to assign each point in the feature space to the nearest local mode it converges to.

Let $\{x_i, i = 1, 2, ...n\}$ be the set of points in the d-dimensional feature space, each representing the feature vector of one pixel. The multivariate kernel density estimate with kernel $K(\cdot)$ on point x is

$$\hat{f}(x)=\frac{1}{nh^d}\sum_{i=1}^{n}K\left(\frac{x-x_i}{h}\right), \tag{1}$$

Where the bandwidth h can either be fixed, or chosen according to the proposed variable bandwidth techniques [9, 10]. The optimum kernel in a mean integrated square sense is the Epanechnikov kernel [11] $K_E(x)=\frac{1}{2}C_d^{-1}(d+2)(1-x^Tx)(\text{if} x^Tx<<1)$, where C_d is the volume of the unit d-dimensional sphere. The corresponding density gradient can be estimated as

$$\hat{\nabla}f(x)=\frac{n_x}{n(h^dC_d)}\frac{d+2}{h^2}\left(\frac{1}{n_x}\sum_{x_i\in S_h(x)}[x_i-x]\right), \tag{2}$$

where n_x is the number of points falling in the local window $S_h(x)$ around *x*. The last term in the right side of (2) is called sample mean shift $M_h(x)$. It means that the estimate of the normalized gradient can be obtained by computing sample mean shift in a uniform kernel on *x*. If we continue the iteration as $y_{k+1}=\frac{1}{n_k}\sum_{x_i\in S(y_k)}x_i$ [9], then y_k will converge to the nearest local mode under properly chosen bandwidth. Proof of convergence can be found in [11]. Clustering through mean shift has such advantages: (1) By counting all the local modes of convergence, the number of classes can be determined automatically. (2) By processing in the joint spatial-range domain [10], mean-shift greatly reduces the neighbor searching price and proves to be desirably fast. (3) Mean shift can find classes with arbitrary shapes regardless of the variance difference on each dimension.

In mean shift segmentation procedure, the bandwidth selection principle for our approach is the number of region not large.

3.2 Wavelet Modulus Maxima Edges Detection

Mallat and coworkers [12] have proposed the use of wavelet transform modulus maxima for characterization of regularity of signals. By examining the wavelet modulus maxima (WMM), they could measure the local Lipschitz exponents of a signal. This turned out to be an effective way to locate the singularity or edges of an image. As shown in Fig.2, the points having local maximum of wavelet transform modulus are corresponding to the abrupt change locations in the image, and the gradual changes are set to zero. Therefore, the wavelet modulus maxima points can locate the sharply edges and the inner texture details of the focused object in the low DOF image.

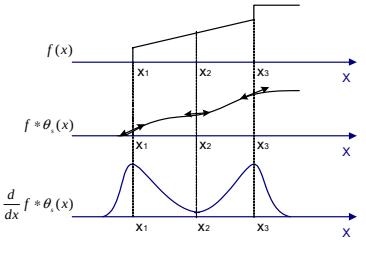


Fig. 2. The characterization of wavelet modulus maxima

3.3 Region Representation

After the image has been divided into some color regions and the wavelet modulus maxima edges have been detected, we calculate the WWM edge point densities for each color region.

Suppose a low DOF image is divided into *N* color regions denoted by $\{R_i\}_{i=1...N}$ and their centers by $\{C_i\}_{i=1...N}$. Let $\{W_k\}_{k=1...P}$ be the WMM points translated from the low DOF image. For each WMM edge point W_k, compute the Euclidean distances d_i^k from W_k to each color region center C_i, where k=1…*P*, i=1…*N*. The point W_k is attached to the region R_i, where $d_l^k = \min_{0<i\leq N}(d_i^k)$, $0<l$ *N*. And then the WMM densities for regions can be calculated.

In the low DOF image, a few background things nearby the object of interest are focused too, and the inner of the focused object may be highly smooth or the object having blur boundary. So the region having relative high WMM density may not be the interested region. On the other hand, the region having low WMM density not always is background too. Therefore, we use average color and WMM density together to represent the region in our method. The denotation of the region is $x = (x_D, x_F)$, x_D and x_F representing the attributes of WMM densities and average color respectively.

3.4 Fuzzification

For the uncertainties mentioned in the introduction, we try to applying fuzzy theory to the segmentation procedure to improve the accuracy of salient interested region extraction.

Since the introduction by Zadeh [13], the importance of fuzzy sets for the analysis of complex natural systems has been established in various application domains. As enounced in the description of the taxonomic categories, the fuzzy paradigm-based techniques are constructed according to a fuzzy model of the objects of the universe. In the case of salient interested region and background in digital images being mappings of natural scenes, utilizing fuzzy set to characterize them may obtain the better performance.

Fuzzy sets are characterized by a membership distribution $\mu_{\tilde{A}_x} \in [0,1]$, to denote the attachment or compatibility of *x* to the fuzzy set A. Fuzzy numbers are the normal convex fuzzy sets on a real line, and have been investigated in detail by Dubois and Prade [14].

In our method, assume all of the color regions in image are the universe of discourse *U*, and the salient regions of interest and the background are two fuzzy subsets on *U*. Let *ROI* and *Bk* be the fuzzy subset denotations of salient region of interest and the background respectively, and they have two attributes: WWM point density and average color feature. Their denotations are:

$$\tilde{R}_{ROI} = (\tilde{D}_{ROI}, \tilde{F}_{ROI}) \tag{3}$$

$$\tilde{R}_{Bk} = (\tilde{D}_{Bk}, \tilde{F}_{Bk}) \tag{4}$$

We regard $\tilde{R}_{ROI}$ and $\tilde{R}_{Bk}$ as normal fuzzy set, and the attribute descriptions $\tilde{D}_{ROI}$, $\tilde{F}_{ROI}$, $\tilde{D}_{Bk}$ and $\tilde{F}_{Bk}$ obey normal distribution consequently. We construct their membership functions in the following.

We can not say which region's WMM point density is high or low easily. Because "high" is not a clear-cut concept, but a fuzzy one. So we use fuzzy membership degree to describe WMM point density. We have assumed that the salient region of interest has very high WMM point density, and the background has very low one. As the consequence, the higher grade of interest, the larger degree of membership $\mu_{\tilde{D}_{ROI}}$. Similarly, for the attribute $\tilde{D}_{Bk}$, the higher grade of disinterest, the larger degree of membership $\mu_{\tilde{D}_{Bk}}$. The corresponding membership functions are:

$$\mu_{\tilde{D}_{ROI}}(x)=\begin{cases} e^{-\left(\frac{x-\bar{d}_{ROI}}{\sigma_{ROI\,D}}\right)^2}, 0<x<\bar{d}_{ROI} \\ 1 \qquad\quad , x\geq \bar{d}_{ROI} \end{cases} \tag{5}$$

$$\mu_{\tilde{D}_{Bk}}(x)=\begin{cases} e^{-\left(\frac{x}{\sigma_{Bk}}\right)^2}, x>0 \\ 1 \qquad , x=0 \end{cases} \tag{6}$$

Where $\bar{d}_{ROI}$ and σ_{ROI_D} is the mean and variance of the WMM densities of all of the regions in image. For the convenience, we set $\sigma_{ROI_D}=\sigma_{Bk_D}$. We consider that if the WMM density is larger than its mean of all the regions' in image, then it is very high. If the WMM density equals zero, it is very low. Equation (5) is the fuzzy membership function of salient interested region, and (6) is background.

The membership functions of $\tilde{F}_{ROI}$ and $\tilde{F}_{Bk}$ are:

$$\mu_{\tilde{F}_{ROI}}(x)=\begin{cases} e^{-\frac{d^2(x-\bar{f}_{ROI})}{\sigma_{ROI_f}^2}}, & |x-\bar{f}_{ROI}|\leq\sigma_{ROI_f} \\ 0 \quad , & |x-\bar{f}_{ROI}|>\sigma_{ROI_f} \end{cases} \tag{7}$$

$$\mu_{\tilde{F}_{Bk}}(x)=\begin{cases} e^{-\frac{d^2(x-\bar{f}_{Bk})}{\sigma_{Bk_f}^2}}, & |x-\bar{f}_{Bk}|\leq\sigma_{Bk_f} \\ 0 \quad , & |x-\bar{f}_{Bk}|>\sigma_{Bk_f} \end{cases} \tag{8}$$

Where $\bar{f}_{ROI}$ is the average color feature of the region whose WMM densities are larger than $\bar{d}_{ROI}$, and $\bar{f}_{Bk}$ is the average color feature of the region which WMM density equals 0. σ_{ROI_f} and σ_{Bk_f} are the color variances of these two region groups.

Let $x=(x_D,x_F)$ be any region in the universe of discourse *U*. Calculate the fuzzy membership values $\mu_{\tilde{R}_{ROI}}$ and $\mu_{\tilde{R}_{Bk}}$, which characterize the degrees of x belonging to

fuzzy sets *ROI* and *Bk*. Then the fuzzification is accomplished. $\mu_{\tilde{R}_{ROI}}$ and $\mu_{\tilde{R}_{Bk}}$ can be calculated by:

$$\mu_{\tilde{R}_{ROI}}(x) = \tfrac{1}{2}(\mu_{\tilde{D}_{ROI}}(x) + \mu_{\tilde{F}_{ROI}}(x)) \tag{9}$$

$$\mu_{\tilde{R}_{Bk}}(x) = \tfrac{1}{2}(\mu_{\tilde{D}_{Bk}}(x) + \mu_{\tilde{F}_{Bk}}(x)) \tag{10}$$

3.5 Defuzzification

In defuzzification phase, all the elements in universe of discourse *U* are attached to fuzzy set *ROI* and *Bk* according to their membership degrees. The fuzzy membership value $\mu_{\tilde{R}_{ROI}}$ and $\mu_{\tilde{R}_{Bk}}$ illustrate the interest and disinterest degree of the region respectively. On the principle of maximum membership degree, we classify the region x to two fuzzy sets on the following rule:

$$If\ \mu_{\tilde{R}_{ROI}}(x) > \mu_{\tilde{R}_{Bk}}(x),\ then\ x \in ROI,\ else\ x \in Bk. \tag{11}$$

Consequently, the regions in image can be classified to salient region of interest and background correspondently.

4 Experimental Results

The segmentation algorithm has been implemented in C++ on a single-CPU 2.8GHz with the Windows operating system. We test the algorithm on more than 200 low DOF images selected from COREL image collection. For 10 images of these testing images, we segmented the interest object areas by hand as the ground truth, and calculate the error percentage of this paper's method. In general, total error is less than 10 percent, and an image of 384×256 pixels can be processed within eight seconds. In order to validating the efficiency of fuzzy method on segmentation, we compare the error percentage of this paper's method with the approach we proposed in [6] on the same testing images. The segmentation results and the comparison between two methods are given in Table 1 and Fig 3.

Table 1. Segmentation results comparison

Image Id	Image Pixel Count	ROI Pixel Count of Ground Truth	Error Percentage	
			The method in this paper	The method in paper [6]
1	98304	89044	5.16%	6.43%
2	98304	77194	6.43%	4.05%
3	98304	85571	11.9%	23.5%
4	98304	85588	16.3%	35.4%
5	98304	76851	5.74%	2.49%
6	98304	89123	4.95%	28.9%
7	98304	83361	9.19%	42.8%
8	98304	46254	1.44%	30.3%
9	98304	54101	9.64%	46.1%
10	98304	82888	2.78%	25.2%
Average Error Percentage			7.31%	24.5%

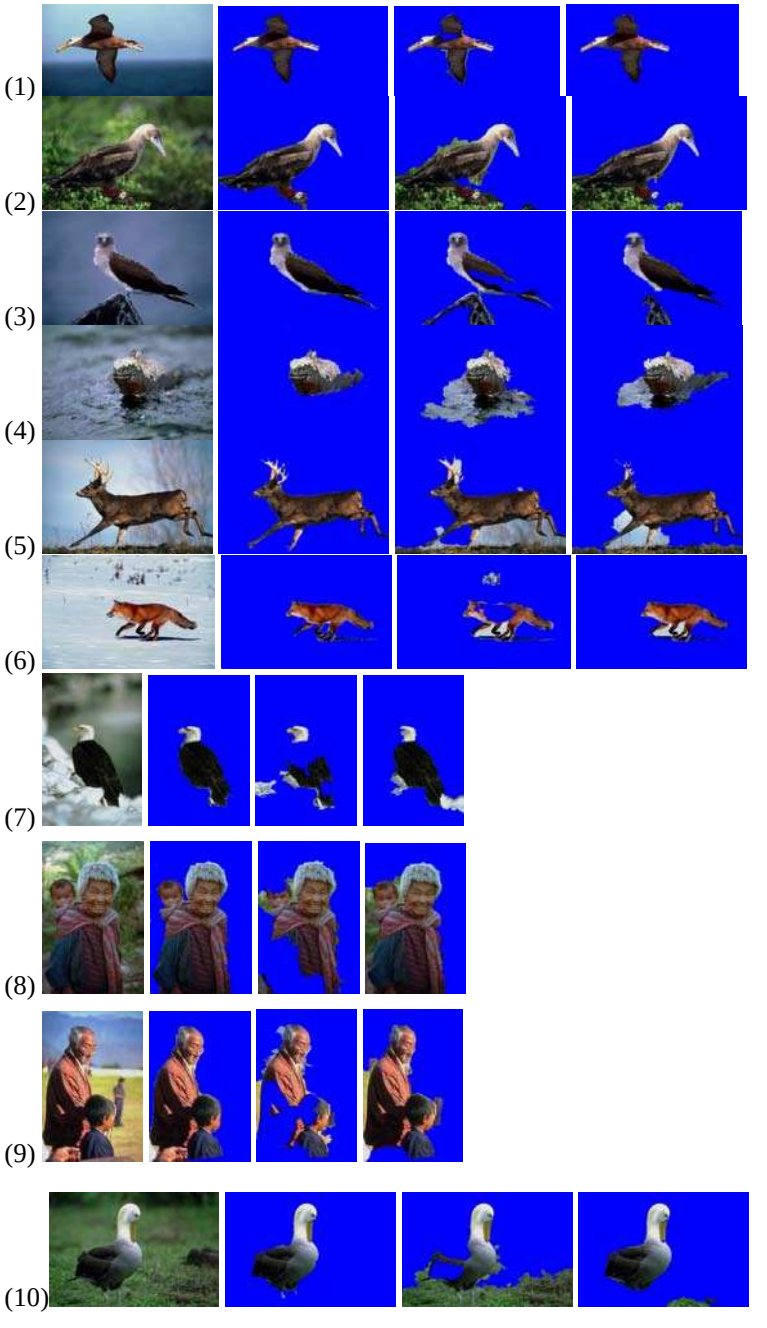


Fig. 3. The segmentation results of 10 testing images. The numbers correspond to the image ID in Table 1. The first column images are the original, the second are the ground truth, the third are results with algorithm in paper [6], and the forth are results with algorithm proposed in this paper.

For typical low DOF image, such as images in Fig. 3, our proposed methods in this paper and in [6] have relative accurate segmentation results. For images having smooth inside such as images (1), (3), (6), (7), (8) and (9), this paper's method can remove the false segmented background regions inside the interest objects. For images (2) and (10) example, a few background things nearby the object of interest are focused too. In this paper's method results, these background things are classified to the right categories.

5 Conclusion

In this paper, we demonstrated a novel fuzzy segmentation algorithm for low DOF images. The proposed method applies fuzzy theory to separate the salient region of interest from background in low depth of field (DOF) images automatically. First the image is divided into regions based on mean shift method and the regions are characterized by color features and wavelet modulus maxima edge point densities. And then the regions are described as fuzzy sets by fuzzification. The salient region interest and background are separated by defuzzification on fuzzy sets finally. The most important features of this segmentation method are full automatic and without empirical parameters. A number of COREL images have been tested on the algorithm, good segmentation results having been received. The experimental results show that fuzzy method is efficient to solve some uncertainty problems in image segmentation.

References

1. A. Adams, The Camera. Boston: New York Graphic Soc., 1980.
2. S. K. Nayar and Y. Nakagawa, "Shape from Focus: An Effective Approach for Rough Surfaces", IEEE Trans. on PAMI, 16 (8): 824-831, Aug. 1994.
3. S. Chaudhuri and A. Rajagopalan. "Depth from defocus: a real aperture imaging approach". Springer Verlag, 1999.
4. Y. F. Liu, "A Unified Approach to Image Focus and Defocus Analysis", Ph.D. Thesis, Dept. of Electrical Engg, SUNY at Stony Brook, 1998.
5. James Z. Wang, Jia Li, Robert M. Gray and Gio Wiederhold, "Unsupervised Multiresolution Segmentation for Images with Low Depth of Field", IEEE Trans. on PAMI, vol. 23, no. 1, pp. 85-90, 2001.
6. KeDai Zhang, HanQing Lu, Automatic Salient Regions Extraction Based on Edge and Region Integration, IEEE International Symposium on Industrial Electronics, Canada, 2006.
7. D. M. Tsai and H. J. Wang, 1998, "Segmenting Focused Objects in Complex Visual Images", Pattern Recognition Letters, vol. 19, pp. 929-940.
8. K. Fukunaga and L. D. Hostetler, "The estimation of the gradient of a density function, with applications in pattern recognition", IEEE Trans. Info. Theory, vol. 21, pp. 32-40, 1975.
9. Dorin Comaniciu," An Algrithm for Data Driven Bandwidth Selection," IEEE Trans. Patt. Anal. Mach. Intell, vol.25. No. 2. February 2003.

10. Dorin Comaniciu and Peter Meer, ”Distribution Free Decomposition of Multivariate Data”, Pattern Analysis Application 1999 2:22-30.
11. Dorin Comaniciu, Peter Meer, ”Mean shift Analysis and Applications” IEEE Int’l Conf. Comp. Vis. Kerkyra, Greece,1197-1203, 1999
12. S. Mallat, and W. L. Hwang, "Singularity detection and processing with wavelet", IEEE Trans. Info. Theory, vol. 38, pp. 617-643, 1992.
13. L. Zadeh, “Fuzzy sets,” Information and Control, vol. 8, pp. 338–353, 1965.
14. D. Dubois and H. Prade, Fuzzy Sets and Systems: Theory and Applications. New York: Academic, 1980.

Author Index

Abbott, A. Lynn I-309
Agarwal, Parag I-731
Ahmed, Fawad II-51
Ahn, Seongjin II-576
Aizawa, Kiyoharu I-719
Alexandrov, Vassil II-713
Ali, Mortuza I-227
Altmanninger, Kerstin II-730
Anthes, Christoph II-722
Ardizzone, E. I-462
Ashraf, Golam II-226
Atrey, Pradeep K. II-155, II-343
Azmi, Reza I-670

Ba, Shan II-73, II-304
Bang, Kee Chun II-528
Benmokhtar, Rachid I-517, II-196
Bergstraesser, Sonja I-34
Bhattacharya, Prabir I-440
Bhattarai, Bibek I-364
Boll, Susanne I-332
Boucher, Alain I-418
Bouguet, JeanYves I-625
Bressler, Helmut II-722
Bu, Jiajun I-557
Byun, Jeongyong II-662

Carrive, Jean I-680
Cha, Guang-Ho I-344
Chan, Chi-Shiang II-461
Chang, Chin-Chen I-289, II-461
Chen, C. II-383
Chen, Cheng I-116
Chen, Chun I-557
Chen, Chunxiao I-475
Chen, Jun-Cheng I-606
Chen, Xin I-772
Chen, Yinpeng II-393
Cheng, Corey I-545
Cheng, Wen-Huang I-606
Chia, Liang-Tien II-238
Cho, Dae-Jea II-627
Cho, Seong Chul II-702
Cho, Seong Ik II-544
Cho, Wan-Hyun I-86
Choh, Ikuro I-147
Choi, Junkyun II-576
Choi, Kyungho II-544
Choi, Sang-Bang II-104
Chong, Jong-Wha II-41
Chua, Tat-Seng I-13, I-247
Chun, Junchul I-700
Chung, Tai-Myoung II-552
Chung, Wonsik II-502

Dai, Li-Rong I-649
Dai, Qionghai I-450
De Bosschere, Koen II-11
De Bruyne, Sarah I-1
De Neve, Wesley I-1
De Schrijver, Davy I-1
de With, Peter H.N. II-279, II-405
De Wolf, Koen I-1
Desai, Bipin C. I-440
Ding, Guiguang I-237
Dou, Yong II-373
Duan, Lingyu II-63
Duan, MiYi I-782
Dulong, Carole II-83
Durak, Nurcan I-376

Eeckhout, Lieven II-11
El Saddik, Abdulmotaleb II-155

Fan, Wei I-625
Fang, Li II-323
Farin, Dirk II-279
Feng, Songhe I-198
Foo, Jun Jie II-433
Fujii, Toshiya I-147

Gao, Yi II-415, II-423
Gatani, L. I-462
George, Roy I-376
Guo, Hui I-485, II-135
Guo, Pan-Hong II-135

Ha, Tae-Jun II-599, II-606
Hacid, Hakim I-586

Haffegee, Adrian II-713
Hahn, Minsoo I-138
Hamers, Juan II-11
Han, Jungong II-279, II-405
Han, Peng II-635
Han, Tae-Man II-643
Han, Young-Ju II-552
Han, Zhen I-568
Hattori, Shun I-690
He, Shumian I-176
Hirashima, Tsukasa I-126
Hong, Hyunki II-289
Hong, Youn-Sik I-217
Hou, Zujun I-106
Hsiao, Ming-Ho II-175
Hu, Rui II-415, II-423
Hu, Ruimin I-568
Hu, Wei I-625, II-83
Hu, Yang I-96
Hua, Xian-Sheng I-649, II-94
Huang, Zhiyong I-54, I-106
Huang, Zhuan Qing I-166
Huet, Benoit I-517, II-196
Hwang, Iksoon II-560
Hwang, Sun-myung II-694
Hwang, Yongho II-289

Jain, Ramesh I-13
James, Jodi II-393
Jammalamadaka, Nataraj II-206
Jatowt, Adam I-535
Javadtalab, Abbas I-670
Jawahar, C.V. II-206
Jee, Jeong Gyu II-568
Jeong, Dong-Hoon II-606
Jiang, Zhuhan I-166
Jiao, Licheng II-415, II-423
Jiten, Joakim I-616
Jung, Jin-Ha II-104

Kang, Seo-Il II-487
Kankanhalli, Mohan S. II-145, II-155, II-343
Kazuki, Joe I-397
Kim, Beomjoon II-560
Kim, Byung-Guk II-606
Kim, Dong-Won II-643
Kim, Eun Yi II-471
Kim, Hee-Sun II-627
Kim, Hyuncheol II-576
Kim, Jeong-Sig II-363
Kim, Ji-Hong I-217
Kim, Jin Up II-702
Kim, Jonghyung II-599
Kim, Junhwan II-289
Kim, Keon-Woo II-510
Kim, Sang-Ha II-643
Kim, Soo Hong II-528
Kim, Sun-Worl I-86
Kim, Tae Ho II-613
Kim, Tai-hoon II-694
Kim, Yong-Hyun I-217
Kim, Yongkwon II-544
Kim, Young Yong II-536
Ko, Eung Nam II-528
Ko, Il Seok II-671, II-678, II-686
Koh, Soo Ngee I-576
Kori, Hiroshi I-690
Ku, William II-145
Kwon, Ohryun I-700

La Cascia, M. I-462
Lai, Shang-Hong I-188
Lai, Wei I-649, II-94
Landertshamer, Roland II-722
Lao, Weilun II-405
Le, Thi-Lan I-418
Lee, Chin-Hui I-247
Lee, Chulhee II-314
Lee, Chungwon II-599
Lee, Deok Gyu II-487, II-528
Lee, Gi-Sung II-584
Lee, Gueesang I-751
Lee, Im-Yeong II-487
Lee, Jaeseon I-138
Lee, Jeomho II-620
Lee, Jin-Heung II-510
Lee, Jong-Hyouk II-552
Lee, Junseak I-44
Lee, Keun Wang II-568, II-592
Lee, Keun-Young II-363
Lee, Kisung II-544
Lee, Kwang Hyoung II-568, II-592
Lee, Kwon II-314
Lee, Myung-Eun I-86
Lee, Sang-Mook I-309
Lee, Seong-Whan I-44
Lee, Seungjae II-599
Lee, Soobeom II-613
Lee, Sooil II-613, II-620

Lee, Suh-Yin II-175
Lee, SuKyoung II-502
Leow, Wee Kheng I-74
Leung, Maylor K.H. II-323
Li, Bao-Feng II-373
Li, Bing I-198
Li, Chuanjun I-386
Li, Chunlin II-494
Li, Haojie II-73
Li, Houqiang I-116, II-32, II-269
Li, Jiang I-485
Li, Jianguo II-83
Li, Jianzhong II-216
Li, Jintao II-304
Li, Junwei II-296
Li, Layuan II-494
Li, Mingjing I-96, I-259, II-269
Li, Te I-576
Li, Z.G. II-383
Li, Zhao I-495
Li, Zhiwei I-96, I-259
Liang, Ke II-23
Liao, Wen-Hung II-453
Lim, Jiyoung II-479
Lim, Joo-Hwee II-145
Lim, Tea-Hun II-510
Lin, Chih-Yang I-289
Lin, Shouxun II-73
Liu, Fang I-299
Liu, Lu II-94
Liu, Qingshan II-63
Liu, Qiong I-568
Liu, Qiwei II-32
Liu, Xiaolong II-296
Liu, Yanan I-527
Liu, Yanli I-299
Liu, Yi I-557
Liu, Ying II-185
Lo, Kwok-Tung I-485
Lo Re, G. I-462
Lu, Feng II-238
Lu, Guojun II-185
Lu, Hanqing I-782, II-63
Lu, Wei I-279
Lu, Yan II-32
Lukac, Rastislav II-443
Luo, Jian-Guang II-125, II-333

Ma, Wei-ying I-96
Marchenko, Yelizaveta I-13
Merialdo, Bernard I-616
Meyer, Marek I-34
Min, Kyeong-Yuk II-41
Min, So Yeon II-568
Mo, Linjian I-557
Mompó, Rafael II-1
Monteverde, Levi C. I-54
Mühlhäuser, Max I-741
Murshed, Manzur I-227, II-353

Na, Yun Ji II-686
Nam, Wonsuk I-147
Nang, Jongho I-659
Nayebi, Kambiz I-670
Nee, A.Y.C. II-258
Ngo, Chong-Wah I-269
Nie, Feiping I-24

Oh, Seunghyun II-651, II-662
Omidyeganeh, Mona I-670
Ong, S.K. II-258
Ortolani, M. I-462

Pan, Xuefeng II-304
Park, Byungyeon II-576
Park, Eunjeong II-115
Park, Jaesung II-560
Park, Je Jin II-606
Park, Jeong Doo II-536
Park, Jeong-Seon I-44
Park, Jong-Cheon II-592
Park, Jong Hyuk II-528, II-536
Park, Jong-Seung I-709
Park, Jonghyun I-751
Park, Joohyoun I-659
Park, Kyoung Shin I-138
Park, Peom I-700
Park, Se Hyun II-471
Park, Soon-Young I-86
Pattanasri, Nimit I-535
Paul, Manoranjan II-353
Peng, Nan I-106
Poli, Jean-Philippe I-680
Prabhakaran, Balakrishnan I-386, I-731
Pradhan, Gaurav N. I-386
Pudi, Vikram II-206
Pyen, Seung-Man II-41

Quan, Shan Guo II-536

Rahardja, Susanto I-576
Rahman, Md. Mahmudur I-440

Ramakrishnan, K.R. II-248
Regueras, Luisa M. II-1
Rensing, Christoph I-34
Robinson, Charles Q. I-545
Ruiz, Conrado R. Jr. I-54

Sakurai, Kouich II-694
Sandhaus, Philipp I-332
Scherp, Ansgar I-332
Shi, Shengfei II-216
Shin, Bum-Joo II-471
Shin, Heonshik II-115
Shin, Kwang-Sik II-104
Shin, Sang-Uk II-510
Singh, Rahul I-364
Singh, Vivek K. II-343
Sinha, Ranjan II-433
Siyal, M.Y. II-51
Soh, Y.C. II-383
Song, Jihyun II-576
Song, Minseok I-207
Song, Yan I-649
Sood, Varun I-440
Steinmetz, Ralf I-34
Suh, Jeong-Jun II-536
Sun, Bing II-635
Sun, Jia-Guang II-216
Sun, Lifeng I-322, II-23, II-333
Sun, Qiang II-494
Sun, Rui I-495
Sundaram, Hari II-393
Sung, Mee-Young I-217

Takata, Masami I-397
Tan, Ah-Hwee II-185
Tan, Hung-Khoon I-269
Tanaka, Katsumi I-156, I-176, I-535, I-690
Tang, Sheng II-304
Tang, Yun II-125, II-333
Tedjokusumo, Jefry I-74
Tezuka, Taro I-156, I-690
Thieme, Sabine I-332
Thonnat, Monique I-418
Thoshkahna, Balaji II-248
Tong, Xiaofeng II-83
Trisiripisal, Phichet I-309
Truong, Ba Tu I-505, I-636
Tsai, Wen-Jiin II-175
Turban, Georg I-741

Van de Walle, Rik I-1
Velusamy, Sudha II-248
Venkatesh, Svetha I-505, I-636
Verdú, María Jesús II-1
Verhoeve, Piet I-1
Volkert, Jens II-722

Wakiyama, Kouki I-126
Wang, Bin I-259
Wang, Chaokun II-216
Wang, Chia-Wei I-606
Wang, Fan I-450
Wang, Fangshi I-198, I-279
Wang, Jianmin II-216
Wang, Jinqiao II-63
Wang, Meng I-649
Wang, Ren-Hua I-649
Wang, Tao I-625, II-83
Wang, Yushun I-64
Wang, ZhenYu I-782
Watanabe, Toshinori I-428
Wöß, Wolfram II-730
Won, Jaimu II-613
Wong, Mike I-364
Wu, Chi-Jiunn I-188
Wu, Fei I-408, I-527, I-596
Wu, Feng II-32
Wu, Huafeng II-520, II-635
Wu, Ja-Ling I-606
Wu, Lei II-269
Wu, Si II-73
Wu, Weixin I-279

Xiang, Shiming I-24
Xiao, Yang I-247
Xiong, Hongkai I-762
Xiong, Yongqiang I-475
Xu, Changsheng II-63
Xu, De I-198, I-279
Xu, Jianfeng I-719
Xu, Ke I-495

Yamasaki, Toshihiko I-719
Yang, Ching-Nung II-443
Yang, Feng I-237
Yang, Lei I-762
Yang, Min II-635
Yang, Shi-Qiang I-475, II-23, II-94, II-125, II-333
Yang, Shu-Sian I-606

Yang, Yang II-135
Yang, Yi II-165
Yang, Zhi I-557
Yazici, Adnan I-376
Ye, Shengfeng I-568
Yeung, Dit-Yan I-625
Yokoyama, Takanori I-428
Yoo, Jaemyeong I-751
Yoon, Wan-Oh II-104
Yoshitaka, Atsuo I-126
You, Ilsun II-479
Yu, Bo II-635
Yu, Kun-Hsuan II-443
Yu, Nenghai II-269
Yu, Rongshan I-545
Yu, Xiaozhou II-323
Yu, Yi I-397
Yuan, M.L. II-258

Zhang, Changshui I-24
Zhang, Chengcui I-772
Zhang, Chunxia I-24
Zhang, Dengsheng II-185
Zhang, Hong I-596, II-296
Zhang, Jianning I-322
Zhang, Kai I-762
Zhang, KeDai I-782
Zhang, Li I-495
Zhang, Meng I-475, II-125
Zhang, Qian I-475
Zhang, Xiafen I-354
Zhang, Xiangrong II-423
Zhang, Yimin I-625, II-83
Zhang, Yongdong II-73, II-304
Zhao, Qi I-782
Zheng, Feng II-494
Zhong, Yuzhuo I-322, II-23, II-333
Zhou, Junyu II-226
Zhou, Qiang II-635
Zhou, Weida II-415, II-423
Zhou, Xiaobo I-116, I-762
Zhu, Zhenfu II-296
Zhuang, Yi I-408
Zhuang, Yueting I-64, I-354, II-165
Zimmermann, Birgit I-34
Zobel, Justin II-433