3D Imaging for Safety and Security

Computational Imaging and Vision

Managing Editor

MAX VIERGEVER
Utrecht University, The Netherlands

Series Editors

GUNILLA BORGEFORS, *Centre for Image Analysis, SLU, Uppsala, Sweden*
RACHID DERICHE, *INRIA, France*
THOMAS S. HUANG, *University of Illinois, Urbana, USA*
KATSUSHI IKEUCHI, *Tokyo University, Japan*
TIANZI JIANG, *Institute of Automation, CAS, Beijing*
REINHARD KLETTE, *University of Auckland, New Zealand*
ALES LEONARDIS, *ViCoS, University of Ljubljana, Slovenia*
HEINZ-OTTO PEITGEN, *CeVis, Bremen, Germany*
JOHN K. TSOTSOS, *York University, Canada*

This comprehensive book series embraces state-of-the-art expository works and advanced research monographs on any aspect of this interdisciplinary field.

Topics covered by the series fall in the following four main categories:

- Imaging Systems and Image Processing
- Computer Vision and Image Understanding
- Visualization
- Applications of Imaging Technologies

Only monographs or multi-authored books that have a distinct subject area, that is where each chapter has been invited in order to fulfill this purpose, will be considered for the series.

Volume 35

3D Imaging for Safety and Security

Edited by

Andreas Koschan
The University of Tennessee, Knoxville, TN, USA

Marc Pollefeys
University of North Carolina at Chapel Hill, NC, USA

and

Mongi Abidi
The University of Tennessee, Knoxville, TN, USA

Springer

A C.I.P. Catalogue record for this book is available from the Library of Congress.

ISBN 978-1-4020-6181-3 (HB)
ISBN 978-1-4020-6182-0 (e-book)

Published by Springer,
P.O. Box 17, 3300 AA Dordrecht, The Netherlands.

www.springer.com

Printed on acid-free paper

Contents

Contributing Authors vii

Preface xiii

PART I: BIOMETRICS 1

3D Assisted Face Recognition: A Survey 3
M. HAMOUZ, J. R. TENA, J. KITTLER, A. HILTON, AND J. ILLINGWORTH

A Survey on 3D Modeling of Human Faces for Face Recognition 25
S. HUQ, B. ABIDI, S. G. KONG, AND M. ABIDI

Automatic 3D Face Registration without Initialization 69
A. KOSCHAN, V. R. AYYAGARI, F. BOUGHORBEL, AND M. A. ABIDI

A Genetic Algorithm Based Approach for 3D Face Recognition: Using Geometric Face Modeling and Labeling 95
Y. SUN AND L. YIN

Story of Cinderella: Biometrics and Isometry-Invariant Distances 119
A. M. BRONSTEIN, M. M. BRONSTEIN, AND R. KIMMEL

Human Ear Detection from 3D Side Face Range Images 133
H. CHEN AND B. BHANU

PART II: SAFETY AND SECURITY APPLICATIONS 157

Synthetic Aperture Focusing Using Dense Camera Arrays 159
V. VAISH, G. GARG, E.-V. TALVALA, E. ANTUNEZ, B. WILBURN, M. HOROWITZ, AND M. LEVOY

Dynamic Pushbroom Stereo Vision: Dynamic Pushbroom Stereo Vision for Surveillance and Inspection 173
Z. ZHU, G. WOLBERG, AND J. R. LAYNE

3D Modeling of Indoor Environments for a Robotic Security Guard 201
P. BIBER, S. FLECK, T. DUCKETT, AND M. WAND

3D Site Modelling and Verification: Usage of 3D Laser Techniques for Verification of Plant Design for Nuclear Security Applications 225
V. SEQUEIRA, G. BOSTRÖM, AND J.G.M. GONÇALVES

Under Vehicle Inspection with 3D Imaging: Safety and Security for Check-Point and Gate-Entry Inspections 249
S. R. SUKUMAR, D. L. PAGE, A. F. KOSCHAN, AND M. A. ABIDI

Colour Plate Section 279

Index 307

Contributing Authors

Besma Abidi
Imaging, Robotics, and Intelligent Systems Laboratory
The University of Tennessee, 334 Ferris Hall
Knoxville, TN 37996, USA

Mongi A. Abidi
Imaging, Robotics, and Intelligent Systems Laboratory
The University of Tennessee, 334 Ferris Hall
Knoxville, TN 37996, USA

Emilio Antunez
Geometric Computing Group
Computer Science Department
Stanford University, Stanford, CA 94305, USA

Venkat R. Ayyagari
Imaging, Robotics, and Intelligent Systems Laboratory
The University of Tennessee, 334 Ferris Hall
Knoxville, TN 37996, USA

Bir Bhanu
Center for Research in Intelligent Systems
University of California
Riverside, CA 92521, USA

Peter Biber
Graphical-Interactive Systems
Wilhelm Schickard Institute for Computer Science
University of Tübingen
Sand 14, 72076 Tübingen, Germany

Gunnar Boström
European Commission - Joint Research Centre
TP210, I-21020 Ispra, Italy

Faysal Boughorbel
Imaging, Robotics, and Intelligent Systems Laboratory
The University of Tennessee, 334 Ferris Hall
Knoxville, TN 37996, USA

Alexander M. Bronstein
Department of Computer Science
Technion – Israel Institute of Technology
Haifa 32000, Israel

Michael M. Bronstein
Department of Computer Science
Technion – Israel Institute of Technology
Haifa 32000, Israel

Hui Chen
Center for Research in Intelligent Systems
University of California
Riverside, CA 92521, USA

Tom Duckett
AASS Research Center
Department of Technology
Örebro University
SE-70182 Örebro, Sweden

Sven Fleck
Graphical-Interactive Systems
Wilhelm Schickard Institute for Computer Science
University of Tübingen
Sand 14, 72076 Tübingen, Germany

Gaurav Garg
Computer Graphics Laboratory
Department of Electrical Engineering
Stanford University, Stanford, CA 94305, USA

João G.M. Gonçalves
European Commission - Joint Research Centre
TP210, I-21020 Ispra, Italy

Miroslav Hamouz
Centre for Vision, Speech and Signal Processing
University of Surrey
Guildford, GU2 7XH, United Kingdom

Adrian Hilton
Centre for Vision, Speech and Signal Processing
University of Surrey
Guildford, GU2 7XH, United Kingdom

Mark Horowitz
Computer Systems Laboratory
Computer Science Department
Stanford University, Stanford, CA 94305, USA

Shafik Huq
Imaging, Robotics, and Intelligent Systems Laboratory
The University of Tennessee, 209 Ferris Hall
Knoxville, TN 37996, USA

John Illingworth
Centre for Vision, Speech and Signal Processing
University of Surrey
Guildford, GU2 7XH, United Kingdom

Ron Kimmel
Department of Computer Science
Technion – Israel Institute of Technology
Haifa 32000, Israel

Josef Kittler
Centre for Vision, Speech and Signal Processing
University of Surrey
Guildford, GU2 7XH, United Kingdom

Seong G. Kong
Imaging, Robotics, and Intelligent Systems Laboratory
The University of Tennessee, 310 Ferris Hall
Knoxville, TN 37996, USA

Andreas Koschan
Imaging, Robotics, and Intelligent Systems Laboratory
The University of Tennessee, 334 Ferris Hall
Knoxville, TN 37996, USA

Jeffery R. Layne
Air Force Research Laboratory
2241 Avionics Circle, WPAFB, Ohio 45433-7318, USA

Marc Levoy
Computer Graphics Laboratory
Computer Science Department
Stanford University, Stanford, CA 94305, USA

David L. Page
Imaging, Robotics, and Intelligent Systems Laboratory
The University of Tennessee, 334 Ferris Hall
Knoxville, TN 37996, USA

Vitor Sequeira
European Commission - Joint Research Centre
TP210, I-21020 Ispra, Italy

Sreenivas R. Sukumar
Imaging, Robotics, and Intelligent Systems Laboratory
The University of Tennessee, 334 Ferris Hall
Knoxville, TN 37996, USA

Yi Sun
Computer Science Department
State University of New York at Binghamton
Binghamton, New York 13902 USA

Eino-Ville Talvala
Computer Graphics Laboratory
Department of Electrical Engineering
Stanford University, Stanford, CA 94305, USA

Jose Rafael Tena
Centre for Vision, Speech and Signal Processing
University of Surrey
Guildford, GU2 7XH, United Kingdom

Vaibhav Vaish
Computer Graphics Laboratory
Computer Science Department
Stanford University, Stanford, CA 94305, USA

Michael Wand
Graphical-Interactive Systems
Wilhelm Schickard Institute for Computer Science
University of Tübingen
Sand 14, 72076 Tübingen, Germany

Bennett Wilburn
Computer Graphics Laboratory
Department of Electrical Engineering
Stanford University, Stanford, CA 94305, USA

George Wolberg
Department of Computer Science
The City College of New York
New York, NY 10031, USA

Lijun Yin
Computer Science Department
State University of New York at Binghamton
Binghamton, New York 13902 USA

Zhigang Zhu
Department of Computer Science
The City College of New York
New York, NY 10031, USA

Preface

The past decades have seen significant improvements in 3D imaging where the related techniques and technologies have advanced to a mature state. These exciting developments have sparked increasing interest in industry and academia in the challenges and opportunities afforded by 3D sensing. As a consequence, the emerging area of safety and security related imaging incorporates these important new technologies beyond the limitations of 2D image processing.

This book is so far the first that covers the current state of the art in 3D imaging for safety and security. It reports about selected contributions given at the "Workshop on Advanced 3D Imaging for Safety and Security" held in conjunction with the International Conference on Computer Vision and Pattern Recognition CVPR 2005, June 2005, San Diego, CA. The workshop brought together pioneering academic and industrial researchers in the field of computer vision and image analysis. Special attention was given to advanced 3D imaging technologies in the context of safety and security applications. Comparative evaluation studies showing advantages of 3D imaging over traditional 2D imaging for a given computer vision or pattern recognition task were emphasized. Moreover, additional experts in the field of 3D imaging for safety and security were invited by the editors for a contribution to this book.

The book is structured into two parts, each containing five or six chapters on (1) Biometrics and (2) Safety and Security Applications. Chapter 1 introduces a survey on 3D assisted face recognition which is followed by a survey of technologies for 3D modeling of human faces in Chapter 2. Chapter 3 explains automatic 3D face registration which overcomes the traditional initialization constraint in data registration. Chapter 4 presents a

genetic algorithm based approach for 3D face recognition using geometric face modeling and labeling. Chapter 5 looks into biometrics and isometry-invariant distances starting from the story of Cinderella as an early example of 3D biometric identification and biometric frauds. Chapter 6 reports on human ear detection from 3D side face range images considering ear as a viable new class of biometrics since ears have desirable properties such as universality, uniqueness and permanence.

The second part of the book is devoted to safety and security applications introducing synthetic aperture focusing with dense camera arrays in Chapter 7. This chapter demonstrates practical applications of surveillance in addition to the theory. Chapter 8 presents a dynamic pushbroom stereo geometry model for both 3D reconstruction and moving target extraction in applications such as aerial surveillance and cargo inspection. Autonomous mobile robots play a major role in future security and surveillance tasks for large scale environments such as shopping malls, airports, hospitals and museums. The challenge of building such a model of large environments using data from the robot's own sensors: a 2D laser scanner and a panoramic camera is addressed in Chapter 9.

In Nuclear Security it is important to detect changes made in a given installation or track the progression of the construction work in a new plant. Chapter 10 describes a system accepting multi-sensory, variable scale data as input. Scalability allows for different acquisition systems and algorithms according to the size of the objects/buildings/sites to be modeled. The chapter presents examples of the use in indoor and outdoor environments. A modular robotic “sensor brick” architecture that integrates multi-sensor data into scene intelligence in 3D virtual reality environments is introduced in Chapter 11. The system is designed to aid under vehicle inspection with 3D imaging for check-point and gate-entry inspections.

Last, but not least, we wish to thank all the members of the Program Committee of the “Workshop on Advanced 3D Imaging for Safety and Security” 2005 for their valuable work, and all the authors who contributed to this book.

Andreas Koschan
Marc Pollefeys
Mongi Abidi

Knoxville and Chapel Hill, January 2007

Part I

Biometrics

Chapter 1

3D ASSISTED FACE RECOGNITION: A SURVEY

M. Hamouz, J. R. Tena, J. Kittler, A. Hilton, and J. Illingworth
Centre for Vision, Speech and Signal Processing, University of Surrey, Guildford, GU2 7XH, United Kingdom, {m.hamouz,j.tena,j.kittler,a.hilton,j.illingworth}@surrey.ac.uk

Abstract: 3D face recognition has lately been attracting ever increasing attention. In this chapter we review the full spectrum of 3D face processing technology, from sensing to recognition. The review covers 3D face modelling, 3D to 3D and 3D to 2D registration, 3D based recognition and 3D assisted 2D based recognition. The fusion of 2D and 3D modalities is also addressed. The chapter complements other reviews in the face biometrics area by focusing on the sensor technology, and by detailing the efforts in 3D face modelling and 3D assisted 2D face matching. A detailed evaluation of a typical state-of-the-art 3D face registration algorithm is discussed and conclusions drawn.

Key words: 3D sensors, 3D face models, 3D face registration, 3D face recognition

1. INTRODUCTION

Face recognition and verification have been at the top of the research agenda of the computer vision community for more than a decade. The scientific interest in this research topic has been motivated by several factors. The main attractor is the inherent challenge that the problem of face image processing and recognition poses. However, the impetus for better understanding of the issues raised by automatic face recognition is also fuelled by the immense commercial significance that robust and reliable face recognition technology would entail. Its applications are envisaged in physical and logical access control, security, man-machine interfaces and low bitrate communication.

To date, most of the research efforts, as well as commercial developments, have focused on 2D approaches. This focus on monocular imaging has partly been motivated by costs but to a certain extent also by the need to retrieve faces from existing 2D image and video databases. Last but

A. Koschan et al. (eds.), 3D Imaging for Safety and Security, 3–23.

not least, it has been inspired by the ability of human vision to recognise a face from single photographs where the 3D information about the subject is not available and therefore the 3D sensing capability of the human perception system cannot be brought to bear on the interpretation task.

The literature addressing the problem of 2D face recognition and verification is truly extensive, offering a multitude of methods for each of the major steps of the process: 1) Detection and localization of faces 2) Geometric normalization 3) Photometric normalisation 4) Feature extraction and 5) Decision making. It is beyond the scope of this chapter to do a proper justice to all the contributions made to this topic. Accordingly, we shall refer the reader to the major surveys that have recently appeared in the literature. These include the work of Zhao[48] on face representation and decision making, the face detection review of Yang et al.[42], and the overview of photometric face normalisation methods in Short et al.[36].

The literature also includes reports of major evaluation studies of 2D face recognition technology such as the Face Recognition Vendor Test[32] and face verification competition[29]. There is a general consensus that the existing 2D face recognition technology performs quite well in controlled conditions, where the subject is presented in a frontal pose under good illumination which is the same for images used for training and those acquired during the subsequent (test) operation. However, it has been observed that the performance can degrade very rapidly when the imaging conditions change. This lack of robustness renders current systems unsuitable for many applications where invariance of the imaging conditions cannot be guaranteed.

The above sensitivity of 2D face recognition solutions can be attributed to several factors. First of all, the reliance on holistic approaches of the currently favoured recognition methods, whereby the face image pixels define the input measurement space for feature extraction and decision making, makes the recognition process very sensitive to registration errors. Basically two images can be compared only when they have been transformed so that certain landmarks in the respective pair coincide. If the images are misregistered, the decision making process will tend to compare measurements that do not correspond to each other and the result of the comparison will be meaningless.

The second major contributor is the pose. If subject's pose deviates from the frontal, different parts of the face are imaged and this again destroys comparability. This problem can, to a certain degree, be mitigated by transforming the probe image into a canonical position[22]. However, this solution is riddled with difficulties. It again relies on accurate landmark detection. The true transformation is nonlinear and subject dependent and therefore unknown. Most importantly, the method cannot recover the

information lost due to lack of visibility. The alternative is to design a separate system for each pose[26]. However, this would require a huge quantity of training data, and even if that was available it could not be done without the quantisation of the view sphere which would again lead to inaccuracies. The use of statistical models[11] to capture the variability due to pose changes has also proved not fully successful.

Third, and perhaps most influential on performance, is the effect of illumination. The reflected light captured by the camera is a complex function of the surface geometry, albedo, illumination and the spectral characteristics of the sensor. Even for pose registered faces, a change in illumination dramatically affects the pixel measurements and therefore their comparability. This problem has again been tackled in a number of different ways including training over different illumination scenarios, illumination modelling, image synthesis, and photometric normalisation methods to mention just a few, but so far only with a limited success.

The above list of problems motivated a radically different approach to face recognition, which is based on 3D properties of the face. This is the only information that is invariant in face imaging and should therefore constitute a solid basis for face recognition. The aim of this chapter is to review the methods of 3D face data acquisition and modelling as well as the various ways the 3D model can be used for face recognition and verification. The review differs from previous efforts in a number of respects. First of all, it focuses on the 3D sensing technology, discussing the principles of active sensing and advantages and drawbacks of the currently available solutions. Second, it includes a review of 3D face representation methods. Finally, in contrast to Bowyer et al.[5], in addition to discussing separate 2D and 3D based recognition and the fusion of these modalities, we also address the problem of 3D assisted 2D recognition.

The chapter is organised as follows. In the next section we briefly review the various methods of sensing 3D facial biometrics. Modelling and representation of 3D facial data is discussed in Section 3. The role of 3D in landmark detection and face registration is discussed in Section 4. Section 5 expounds on the use of 3D in face recognition. The review is drawn to conclusion in Section 6.

2. 3D SENSING FOR FACIAL BIOMETRICS

Facial biometrics can utilise 3D reconstruction of faces in two contexts:

Enrolment: Offline database capture of individuals for use in training and as exemplars for verification and/or recognition.

Identification: Online face capture for identity recognition or verification.

Table *1-1* summarises important requirements for 3D face capture in each of these contexts. All applications require the simultaneous capture of 3D shape and 2D appearance with known alignment between them. A primary difference for identification is the requirement for instantaneous single shot capture in a relatively uncontrolled environment (variable illumination, subject movement etc.). Face capture for enrolment purposes can be performed in a controlled environment with the subject asked to remove glasses and remain static or perform set expressions.

To model intra and inter person variability database capture requires acquisition of multiple expressions and also at least the appearance of elements such as facial hair (many active 3D sensing systems fail to reliably capture hair). For identification purposes the subject can not be assumed to be cooperative, therefore the capture system should be able to cope with movement, variation in expression and additional elements such as glasses. Sensor accuracy and spatial resolution for both enrolment and identification will also impact on performance. Accuracy of shape measurement required to resolve facial features is expected to be in the range 1-5 *mm*.

In this section we review how current visual reconstruction approaches meet the requirements for 3D facial biometrics. Visual reconstruction techniques are reviewed in two categories: active sensing where a structured illumination pattern is projected onto the face to facilitate reconstruction; and passive sensing where reconstruction is performed directly from the facial appearance in images or video.

Table 1-1. Requirements of 3D reconstruction for facial biometrics (• firm, p=possible).

	Enrolment	Identification
Shape	•	•
Registered appearance	p	•
Single-shot capture		•
Robust to variable light		•
Variation in shape	p	
Variation in appearance	p	
Facial hair	p	
Glasses	p	•
Sequence Capture	p	p

2.1 Passive sensing

Reconstruction of shape from multiple view images and video has produced a number of Shape-from-X techniques. The principal limitation in the application of these approaches to faces is the relatively uniform appearance resulting in low-accuracy reconstruction. Potential advantages of

passive techniques include reconstruction of faces from image sequences with natural illumination, simultaneous acquisition of colour appearance and video-rate shape capture. To overcome limitations on accuracy, model-based approaches[14, 47] have been developed to constrain face reconstruction in regions of uniform appearance. Introduction of prior models to regularize the reconstruction has the potential to incorrectly reconstruct detailed features in facial shape which are uniform in appearance. The relatively low-accuracy and reliability of passive facial reconstruction approaches has resulted in active sensing being widely used for acquisition of face shape. Increases in digital camera resolution offer the potential to overcome limitations of uniform appearance of the face allowing accurate passive reconstruction.

2.2 Active sensing

A number of active sensing technologies have been developed for 3D surface measurement, which operate on the principle of projecting a structured illumination pattern into the scene to facilitate 3D reconstruction. 3D acquisition systems work on two principles: time-of-flight; and triangulation.

Time-of-flight sensors measure the time taken for the projected illumination pattern to return from the object surface. This sensor technology requires nanosecond timing to resolve surface measurements to millimetre accuracy. A number of commercial time-of-flight systems are available based on scanning a point or stripe across the scene (Riegl, SICK). Area based systems have also been developed which project a light pulse which covers the entire scene allowing single-shot capture of moving scenes (Z-cam). The requirement for high-speed timing has limited the application of time-of-flight to systems for large-scale environment measurement with centimetre accuracy.

Active systems which reconstruct depth by triangulation between projector-to-camera or camera-to-camera allow accurate reconstruction of depth at short range. Projection of a structured illumination pattern facilitates correspondence analysis and accurate feature localisation. Point and stripe patterns provide visually distinct features with accurate localisation allowing reconstruction of depth to μm accuracy for short range measurement ($<1m$). To capture an object surface the point or stripe must be scanned across the scene which prohibits the capture of moving objects. Commercial face and body shape capture systems based on stripe scans (laser, white-light or infra-red) require a capture time of approximately 5-20s (Minolta, Cyberware, Hamamatsu) during which the subject should remain stationary. This requirement restricts the sensor use to enrolment where subjects are compliant and prohibits capture of dynamic events such as expressions.

Area based triangulation systems capture the entire object surface simultaneously from one or more images. The primary problem in projection of an area pattern is the unique identification of corresponding features. Gray code projection involves a binary series of N area patterns allowing unique identification of 2^N stripe boundaries and accurate reconstruction. This approach is the basis for a number of commercial systems for both reverse engineering and measurement of the human face and body (Wicks&Wilson, ABW, K2T). Acquisition time depends on the camera frame-rate, typically N=8-10 corresponding to a capture time of approximately half a second which is sufficient for enrolment of compliant subjects but prohibits natural head movement.

To reduce the number of image frames required for shape reconstruction, boundary-coded[35] and colour[39] stripe patterns have been introduced. In Rusinkiewicz et al.[35] the temporal sequence of stripe boundaries over 4 consecutive frames was used to uniquely identify lines allowing reconstruction at each frame. Colour sinusoidal fringe[39] or stripe patterns have also been used for correspondence analysis and spatial localisation. The use of colour is adversely affected if the object surface is not white resulting in reconstruction failure. Use of visible patterns also requires additional images with uniform illumination for appearance capture. A number of systems based on stripe pattern projection[46, 39] have recently been developed for capture of dynamic face sequences.

To achieve single-shot area-based acquisition shape capture allowing for object movement, a number of technologies have been developed. Projection of a grid pattern[34] allows single-shot capture of a continuous surface based on local grid distortion. Projection of a visible grid prohibits simultaneous capture of appearance, which requires a second image to be captured. Grid projection has been developed as a commercial system for face capture (Eyetronic). Projection of a random texture pattern[37] allows camera-to-camera stereo correlation for objects with regions of uniform appearance such as faces. This approach has been utilised to achieve accurate single-shot face reconstruction in a number of commercial systems (3D-Matic, 3dMD, SurfIm). Random pattern projection is unaffected by surface appearance allowing reconstruction of hair and overlapping pattern projection for simultaneous shape acquisition from multiple views. Recently this approach has been extended to infra-red pattern projection[44] enabling simultaneous video-rate capture of shape and colour appearance for face image sequences, see Figure *1-1*.

A summary of the principal characteristics of active sensing technologies is presented in Table *1-2*. Acquisition time is critical for accurate face capture unless the subject is prevented from movement. Area based systems reconstruct shape for the entire scene from a small number of images

allowing rapid acquisition. Projection of visible patterns requires additional frames to capture texture allowing the possibility of movement and consequent mis-registration between shape and colour. Infra-red pattern projection allows simultaneous single shot capture of shape and colour eliminating registration problems. However, infra-red is absorbed by the skin resulting in a blurring of the pattern and consequent reduction in localisation accuracy.

A limitation common to active sensor technologies is their lack of robustness in environments with uncontrolled illumination such as outdoors. Background illumination can saturate the projected pattern resulting in loss of accuracy or sensor failure. This limits the use of a number of existing active sensor technologies to indoor scenes without direct sunlight.

Figure 1-1. Video-rate 3D colour shape capture [44].

Table 1-2. Active sensor performance characteristics

	Range	Accuracy	Time
Time-of-flight:			
Point/stripe	2-200*m*	<2*cm*	>1min
Area	1-20*m*	<1*cm*	1-frame
Triangulation:			
Point/stripe	0.2-1.5*m*	<0.1*mm*	5-20s
Gray-code	0.5-10*m*	<0.1*mm*	N-frames
Boundary-code[35]	0.5-10*m*	<0.5*mm*	4-frames
Colour-code[39]	0.5-1.5*m*	<0.5*mm*	2-frames
Grid[34]	0.5-1.5*m*	<1*mm*	2-frames
Random[37]	0.5-1.5*m*	<0.5*mm*	2-frames
Random IR[44]	0.5-1.5*m*	<1*mm*	1-frame

3. 3D FACE MODELS

3D acquisition systems capture raw measurements of the face shape together with associated texture image. Measurements are typically output as a point set or triangulated mesh. Raw measurement data is typically unstructured, contains errors due to sensor noise and can contain holes due to occlusion. Direct comparison with measurement data may lead to erroneous results. An intermediate structured face model could be required for recognition.

Simple models. A coarse 3D geometry combined with multiple texture maps was used in the identification system of Everingham and Zisserman[13]. A 3-D ellipsoid approximation of the person's head is used to train a set of generative parts-based constellation models which generate candidate hypotheses in the image. The detected parts are then used to align the model across a wide range of pose and appearance.

Biomechanical models. Biomechanical models which approximate the structure and musculature of the human face have been widely used in computer animation[30] for simulation of facial movement during expression and speech. Models have been developed based on 3D measurement of a persons face shape and colour appearance[24]. Simplified representations are used to model both the biomechanical properties of the skin and underlying anatomical structure. DeCarlo et al.[12] used anthropometric statistics of face shape to synthesise biomechanical models with natural variation in facial characteristics. Biomechanical model of kinematic structure and variation in face shape provide a potential basis for model-based recognition of faces from 3D data or 2D images.

Morphable models. In the approach of Blanz and Vetter[3], 3D shape and texture is estimated using a single image. The estimate is achieved by fitting

a statistical, morphable model of faces to the image. The model is learnt from textured 3D data collected with a laser-stripe scanner. A probabilistic PCA-based model is used to represent the statistical variation of shape and texture of human heads and the model of Phong[33] is used to deal with the illumination variations. The process of fitting the model to particular image data is stochastic optimization of a multi-variate cost function and does not achieve realtime performance. In order to achieve an accurate fit of facial features, fitting is first performed holistically and then eyes, mouth and nose are fitted independently. In the method of Yin and Yourst[43] shape is reconstructed by adapting a deformable elastic model to hyperresolution-enhanced input frontal and profile images. Similar approaches include the work of Grammalidis et al.[17] and Ansari and Abdel-Mottaleb[1].

4. FACE REGISTRATION

Automatic face registration remains a critical part of the system influencing heavily the overall performance. With regard to the use of 3D in registration, methods can be categorized into two separate groups.

4.1 3D to 3D registration

A technique that combines an Active Shape Model with Iterative Closest Point method is presented by Hutton et al.[20]. A dense model is built first by aligning the surfaces using a sparse set of hand-placed landmarks, then using spline warping to make a dense correspondence with a base mesh. A 3D point distribution model is then built using all mesh vertices. The technique is used to fit the model to new faces. Wang et al.[40] presented a method which aims at localization of four fiducial points in 3D. Landmark points are represented by jets of point signatures[9].

In the work of Colbry et al.[10] a hybrid Iterative Closest Point algorithm is initialized by three detected anchor points on the face. The key feature points are preselected manually and curvature-based measurements model the local shape. Points are selected so they are least influenced by the change of expression. False hypotheses are removed by discrete relaxation. Levine and Rajwade[25] propose a hierarchical strategy using support vector regression and ICP for pose normalization.

Mao et al.[28] developed a semi-automatic method that uses 5 manually identified landmarks for Thin-Plate Spline warping. Rather than taking the nearest point, correspondences are established taking the most similar point according to a similarity criterion that is a combination of distance, curvature, and a surface normal.

As published literature on the topic generally lacks detailed error analysis we selected Mao's method to undergo a thorough accuracy evaluation on a challenging set of 3D faces. The objective of the evaluation was to determine if a single generic model could be deformed to accurately represent both inter and intra-personal variations in face shape due to different identity and expression. This analysis is important for the evaluation of the robustness of deformable models for 3D face recognition. Accordingly, two different databases were used. The first database is a subset of the Face Recognition Grand Challenge Supplemental Database[31] (referred to as FRGC). It contains 33 faces representing 17 different individuals and a set of 4 landmarks (right eye corner, left eye corner, tip of the nose, and tip of the chin) per face. The second database is purpose-collected data of our own (referred to as OWN) and contains 4 sets of 10 faces, each set corresponding to 1 individual acting 10 different expressions. The same set of 4 landmarks as that in the FRGC data were manually identified for the OWN database. Both databases were collected with a high-resolution 3dMDface™ sensor.

The algorithm was used to establish dense correspondence between a generic face model (M_G) and a 3D surface (M_D). The conformation process comprises of three stages: i) global mapping, ii) local matching and iii) energy minimisation. During the global mapping stage, a set of landmarks is identified on M_G and M_D. The two sets of landmarks are brought into exact alignment using the Thin-Plate Spline interpolation technique[4], which smoothly deforms M_G minimizing the bending energy. The aligned M_G and M_D are then locally matched by finding for each vertex of M_G the most similar vertex on M_D. Similarity (S) between a vertex on M_G and a vertex on M_D is determined by a weighted sum of the distance between them (D), the angle between their normals (N), and the difference between curvature shape index (C) (see Mao et al.[28] for details):

$$S = -\alpha D - \beta(\arccos\ (N_G \cdot N_D)/\pi) - \gamma(C_G - C_D) \qquad (1)$$

where the values of all weights (α,β,γ) sum to one. Curvature was estimated using the extended quadric fitting method[15]. For the purposes of this evaluation only distance and curvature difference were explicitly used in the computation of the similarity score. The mean and standard deviation are calculated for the similarity scores of all the most similar vertices, and are used to determine a threshold. The most similar vertices whose similarity score is less than the threshold are deemed unreliable, discarded, and interpolated from the most similar vertices of their neighbours. The final set of most similar points (vertices and interpolated points) is used to guide the

energy minimisation process that conforms M_G to M_D. The energy of M_G is defined as:

$$E = E_{ext} + \varepsilon E_{\text{int}} \tag{2}$$

where the parameter ε balances the trade-off between adherence to M_D and maintaining the smoothness of M_G. The external energy attracts the vertices of M_G to their most similar points on M_D:

$$E_{ext} = \sum_{i=1}^{n} \left(\left\| \tilde{\mathbf{x}}_i - \mathbf{x}_i \right\| \right)^2 \tag{3}$$

where n is the number of vertices in M_G, $\mathbf{x}_i$ is the ith vertex and $\tilde{\mathbf{x}}_i$ is its most similar point on M_D. The internal energy constrains the deformation of M_G thus maintaining its original topology:

$$E_{\text{int}} = \sum_{i=1}^{n} \sum_{j=1}^{m} \left(\left\| \mathbf{x}_i - \mathbf{x}_j \right\| - \left\| \bar{\mathbf{x}}_i - \bar{\mathbf{x}}_j \right\| \right)^2 \tag{4}$$

where m is the number of neighbour vertices of the ith vertex, $\mathbf{x}_i$ and $\mathbf{x}_j$ are neighbouring vertices and $\bar{\mathbf{x}}_i$ and $\bar{\mathbf{x}}_j$ are their original positions in M_G. The energy function is minimized using the conjugate gradient method, ending the conformation process (see Figure *1-2*).

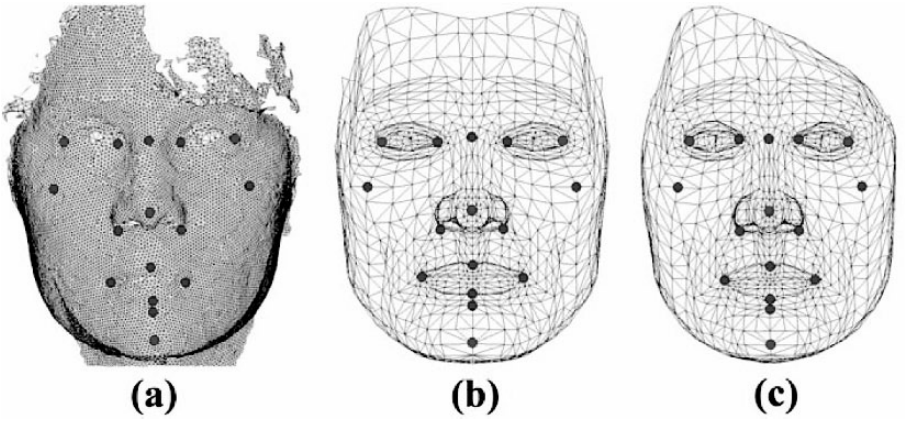


Figure 1-2. The target data (a), the generic model (b), and the conformed model (c). Ground truth landmarks are shown.

The evaluation protocol was designed to test the conformation algorithm for fitting accuracy and correspondence consistency. Fitting accuracy was evaluated by measuring the fitting error, defined at each vertex of the fitted generic model surface as the distance from the vertex to the closest point on the target 3D surface (which might be another vertex or a point lying on the surface of a polygon). Since the deformed models are in correspondence, the fitting error can be statistically evaluated across different subjects. This setting allows evaluation of the fitting accuracy for different face areas with both identity and expression variation. Consequently, to evaluate fitting accuracy, the generic model was conformed to each face in the databases. The polygons were then subdivided to increase the vertex density. The fitting error was measured for all the faces in the FRGC database and for the 4 sets of the OWN database.

The tradeoff parameter (ε) in equation 1 was set to 0.25, as suggested in Kaus et al.[21] and experimentally verified to yield the best deformation results. The weights given to distance (α) and curvature (γ) in the local matching step of the conformation process were varied in discrete steps of 0.2 for the range [0,1] following a $\gamma = 1 - \alpha$ scheme. Mean fitting error and its standard deviation at all vertices were calculated for the FRGC database and the 4 sets of the OWN database independently; global mean and standard deviation were also calculated for each subset.

To evaluate the correspondence consistency, 16 landmarks corresponding to 16 vertices of the generic model were manually identified on the faces of the FRGC database. Figure *1-2* shows the 16 landmarks used for evaluation. Four of the landmarks (external eye corners, tip of the nose, and chin) were provided with the FRGC database. In addition the manual landmark annotation was repeated by another marker to obtain a crude estimation of the interobserver variability in landmarking. The correspondence error was defined at each landmark as the distance between its location in the conformed model and its location in the target, which should be zero for perfect correspondence. The mean correspondence error and its standard deviation across the FRGC database were calculated for all 16 landmarks under the same conditions used to estimate the fitting error.

The results of evaluating the fitting accuracy of the conforming algorithm on the FRGC and OWN databases are shown in Table *1-3*. The error measurements obtained using the FRGC database represent the fitting accuracy across different individuals with slight expression changes. The results show accurate model fitting to within 2*mm* for 90% of points when local matching is guided by distance only ($\alpha = 1$). The measurements on the OWN database show fitting accuracy across drastic expression changes. Results show 85% of points within 2*mm* fitting error again when $\alpha = 1$. As the curvature is introduced the fitting error increases as expected, since the

definition of fitting error is directly related to distance. Figure *1-3* shows the distribution of the fitting error across the generic model with α = 0.8 for subject 3 of the OWN database. The error is shown to be higher in the mouth and its surroundings, an expected result since this area is deformed by expressions.

Table 1-3. Fitting error: global mean and std. deviation, % of vertices for which the error (E) was less than 1*mm* and 2*mm* are given.

α	Mean [*mm*]	StdDev [*mm*]	E<1*mm* [%]	E<2*mm* [%]
		FRGC Database		
1.0	1.045	0.698	60.89	93.24
0.8	1.059	0.707	59.59	92.89
0.4	1.151	0.727	53.14	91.76
0.0	1.530	0.838	23.81	82.04
		OWN Database Subject 1		
1.0	1.084	1.101	63.90	90.67
0.8	1.078	1.107	64.09	90.76
0.4	1.137	1.123	61.65	90.08
0.0	1.542	1.382	41.27	81.09
		OWN Database Subject 2		
1.0	1.224	0.866	49.51	84.38
0.8	1.220	0.856	49.21	84.47
0.4	1.282	0.880	47.06	84.17
0.0	1.657	0.091	33.30	71.31
		OWN Database Subject 3		
1.0	1.027	0.856	62.79	91.33
0.8	1.043	0.857	61.59	91.16
0.4	1.175	0.914	54.14	87.90
0.0	1.548	1.035	32.28	77.50
		OWN Database Subject 4		
1.0	1.183	0.959	53.73	85.42
0.8	1.198	0.965	53.17	85.18
0.4	1.317	1.005	47.08	81.61
0.0	1.726	1.171	30.83	67.81

Table *1-4* shows the correspondence error measured on the FRGC. The error for only 12 of the 16 landmarks, for which error was estimated, is presented for simplicity. It has been observed that the 16 ground truth landmarks manually localised for the FRGC data contain a considerable error, of up to 5*mm*. Therefore errors of the automatic correspondence of up to the same 5*mm* could be considered acceptable. As α is decreased, the correspondence error for each landmark appears to follow different trends, either increasing or decreasing towards a minimum and then increasing again. All 16 landmarks selected for the evaluation correspond to points of

the face where curvature shape index reaches local extrema. Therefore a combination of curvature and distance for driving local matching near the landmarks can be considered to be more effective as supported by the results obtained. All together the evaluation of fitting and correspondence error suggests that α should take different values across the regions of the face, taking smaller values near features that represent curvature local extrema.

Table 1-4. Correspondence error on the FRGC database. Mean error and standard deviation for the outer eye corner (EC), nostril (N), mouth corner (MC), nose tip (NT), chin tip (CT), nose bridge (NB), upper lip (UL), lower lip (LL) and chin dip (CD).

α	**Right**			**Left**		
	EC	N	MC	EC	N	MC
			Mean Error [*mm*]			
1.0	3.83	5.50	4.57	3.95	5.40	3.78
0.8	3.87	5.46	4.51	3.96	5.37	3.69
0.6	3.97	5.42	4.40	3.99	5.31	3.62
0.4	3.92	5.39	4.41	3.98	5.30	3.58
0.2	3.94	5.48	4.50	3.97	5.31	3.92
0.0	4.42	5.21	5.17	4.41	5.02	4.37
			Error's Standard Deviation [*mm*]			
1.0	3.22	3.81	7.28	6.40	5.66	9.79
0.8	3.02	3.75	7.60	6.75	5.54	9.52
0.6	3.07	3.93	7.00	6.68	5.01	9.18
0.4	3.17	3.45	7.89	6.99	4.82	8.93
0.2	3.01	2.53	7.47	7.06	4.33	8.23
0.0	6.02	3.52	5.80	7.90	4.73	6.52
α			**Midface**			
	NT	CT	NB	UL	LL	CD
			Mean Error [*mm*]			
1.0	5.34	4.25	4.99	6.25	4.68	2.40
0.8	5.32	4.21	4.87	6.10	4.66	2.40
0.6	5.37	4.19	4.79	5.87	4.44	2.61
0.4	5.34	4.17	4.77	5.65	4.14	3.09
0.2	5.42	4.25	4.70	5.58	4.64	3.40
0.0	6.41	4.84	5.13	5.90	5.86	3.73
			Error's Standard Deviation [*mm*]			
1.0	5.26	2.05	5.65	9.01	9.14	1.38
0.8	5.30	2.03	5.61	8.74	8.96	1.31
0.6	5.37	1.85	5.39	8.12	8.96	1.55
0.4	5.50	1.72	5.04	7.84	6.72	1.57
0.2	5.60	1.74	5.08	7.29	5.41	1.69
0.0	5.49	2.48	2.98	10.83	7.49	1.27

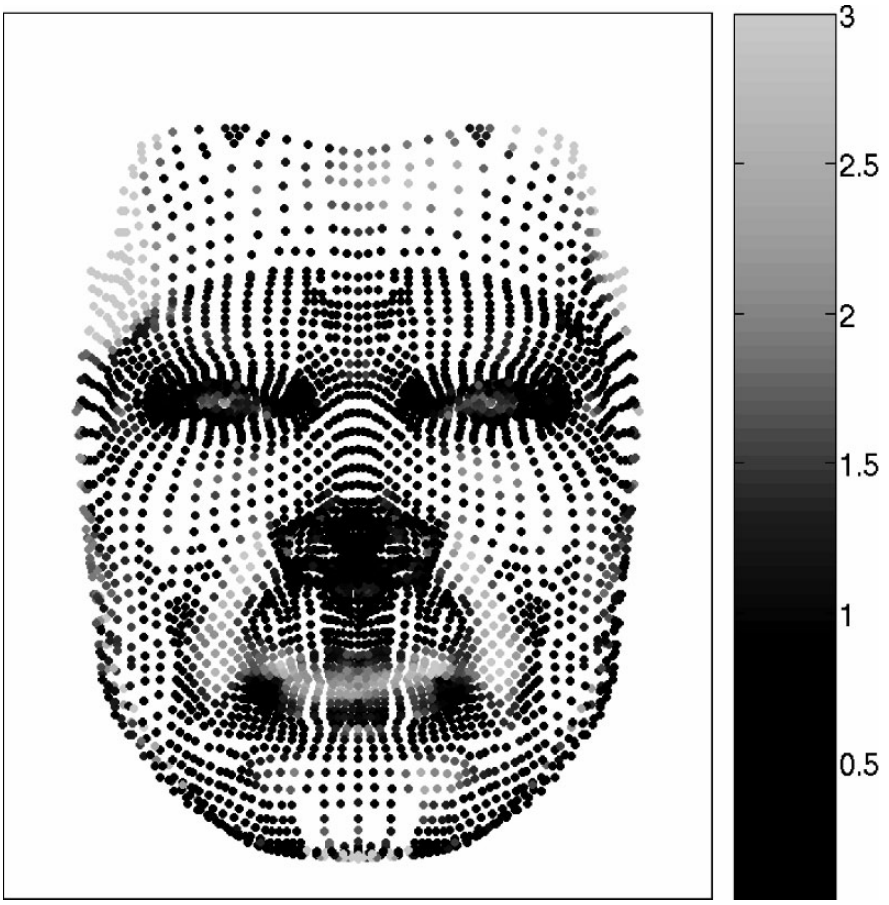


Figure 1-3. Fitting error for OWN database subject 3 with α = 0.8. Error is given in *mm* according to the grey scale.

The results above suggest that a typical state-of-the-art 3D registration algorithm representative is still far from perfect in the context of face recognition. The registration part of the recognition chain must be able to achieve high registration accuracy and be robust to inter- and intra-personal variations facilitating accurate preservation of personal traits and separation of the nonpersonal ones.

4.2 3D to 2D registration

These approaches exploit a 3D morphable model which is used to fit 2D data[3,19]. In an analysis-by-synthesis loop the algorithm iteratively creates a 3D textured mesh, which is then rendered back to the image and the

parameters of the model updated according to the differences. This approach easily separates texture and shape and is used in 3D shape assisted 2D recognition, see Section 5.2.

5. USE OF 3D IN RECOGNITION

5.1 Recognition using 3D shape only

One of the first approaches investigating the use of range images in face recognition is the work of Lee and Milios[23]. Correlation between Gaussian images of convex regions on the face is used as a similarity measure. Another approach investigating the use of 3D for face recognition is the method of Gordon[16]. Curvature-based descriptors are computed over various regions of the face and used as features. Another curvature-based method is presented by Tanaka et al.[38]. Point signatures are used in the work of Chua et al.[8]. The algorithm can deal with the human facial expressions. An example of a recent approach basing recognition on shape only is the method of Xu[41]. Heseltine et al.[18] proposed an algorithm based on a combination of a variety of facial surface representations using Fisher surface method. Other methods together with performance comparisons can be found in the survey of Bowyer et al.[5].

5.2 3D shape assisted 2D recognition

Pose and illumination were identified as major problems in 2D face recognition. Approaches trying to solve these two issues in 2D are bound to have limited performance due to the intrinsic 3D nature of the problem.

Blanz and Vetter[3] proposed an algorithm which takes a single image on the input and reconstructs 3D shape and illumination-free texture. Phong's model is used to capture the illumination variance. The model explicitly separates imaging parameters (such as head orientation and illumination) from personal parameters allowing invariant description of the identity of faces. Texture and shape parameters yielding the best fit are used as features. Several distance measures have been evaluated on the FERET and the CMU-PIE databases.

Basri and Jacobs[2] proved that a set of images of a convex Lambertian object obtained under arbitrary illumination can be accurately approximated by a 9D linear space which can be analytically characterized using surface spherical harmonics. Zhang and Samaras[45] used Blanz and Vetter's morphable model together with a spherical harmonic representation for 2D

recognition. The method is reported to perform well even when multiple illuminants are present.

Yin and Yourst[43] describe their 3D face recognition system which uses 2D data only. The algorithm exploits 3D shape reconstructed from front and profile images of the person using a dynamic mesh. A curvature-based descriptor is computed for each vertex of the mesh. Shape and texture features are then used for matching.

These approaches represent significant steps towards the solution of illumination, pose and expression problems. However there are still several open research problems like full expression invariance, accuracy of the Lambertian model with regard to the specular properties of human skin and stability of the model in presence of glasses, beards and changing hair, etc.

5.3 Recognition using 3D shape and texture

Approaches in this group attempt to exploit all available information in the decision process. In most cases, texture and shape information are fused either at the feature level or the decision level. In the approach of Lu[27] a robust similarity metric combining texture and shape features is introduced. Bronstein et al.[6], morph 2D face texture onto a canonical shape computed from the range data. Canonical shape is a surface representation, which is invariant to isometric deformations, such as those resulting from different expressions and postures of the face. This results in a special 2D image that incorporates the 3D geometry of the face in an expression-invariant manner. PCA is then used to capture variability of these signature images. A comparison of multimodal approaches fusing 2D and 3D is presented in [5,7].

6. SUMMARY AND CONCLUSIONS

The use of 3D information in face recognition has lately been attracting ever increasing levels of attention in the biometrics community. In this chapter we reviewed the full spectrum of 3D face processing technology, from sensing to recognition. The review covered 3D face modelling, 3D to 3D and 3D to 2D registration, 3D based recognition and 3D assisted 2D based recognition. The fusion of 2D and 3D modalities was also addressed. The chapter complements other reviews in the face biometrics area by focusing on the sensor technology, and by detailing the efforts in 3D modelling and 3D assisted 2D matching.

The review identified considerable scope for further improvements in 3D face biometric technology. The existing sensors are not ideally suited for 3D based person identification as they are not dynamic, and often cannot

provide reliable data from uncompliant subjects, wearing glasses, in arbitrary conditions. The main reasons for this are the inherent limitations of the sensing techniques used as well as the fact that most of the sensors have been designed for other applications. Although 3D data is not subject to changes in illuminations, it is affected by other artifacts such as changes in expression, and holes in data caused by imaging effects. As the registration error analysis experiments showed, the current algorithms still underachieve and a room for improvement exists. In consequence, the currently achievable performance of 3D face recognition systems is not greatly superior to that of 2D systems. Although some benefits may accrue directly from the fusion of 2D and 3D face biometric modalities, considerable effort will be required on all aspects of 3D face processing to reach the level of maturity required from systems to be deployed commercially.

ACKNOWLEDGMENTS

This work was supported by EPSRC Research Grant GR/S98528/01 with contributions from EU Project BIOSECURE.

REFERENCES

1. A.-N. Ansari and M. Abdel-Mottaleb. 3D face modeling using two orthogonal views and a generic face model. *In ICME 03, Proceedings of International Conference on Multimedia and Expo*, 3:289-92, 2003.
2. R. Basri and D. W. Jacobs. Lambertian Reflectance and Linear Subspaces. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(2):218-233, 2003.
3. V. Blanz and T. Vetter. Face Recognition Based on Fitting a 3D Morphable Model. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(9):1063-1074, 2003.
4. F. Bookstein. Principal warps: Thin-plate splines and the decomposition of deformations. *IEEE PAMI*, 2(6):567-585, June 1989.
5. K. W. Bowyer, K. Chang, and P. Flynn. A Survey Of Approaches To Three-Dimensional Face Recognition. *In Proc. of 17th International Conference on Pattern Recognition (ICPR'04)*, 1:358-361, 2004.
6. A. M. Bronstein, M. M. Bronstein, and R. Kimmel. Expression-Invariant 3D Face Recognition. *In Proc. 4th Int. Conf. on Audio- and Video-based Biometric Person Authentication*, pp. 62-69, 2003.
7. K. I. Chang, K. W. Bowyer, and P. J. Flynn. An Evaluation of Multimodal 2D+3D Face Biometrics. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27(4):619-624, 2005.
8. C. Chua, F. Han, and Y. Ho. 3D human face recognition using point signature. *In Proceedings of the Fourth IEEE International Conference on Automatic Face and Gesture Recognition*, pp. 233-238, 2000.
9. C. Chua and R. Jarvis. Point Signatures: A New Representation for 3D Object Recognition. *Int. J. Comput. Vision*, 25(1):63-85, 1997.

10. D. Colbry, X. Lu, A. Jain, and G. Stockman. 3D face feature extraction for recognition. *Technical Report MSU-CSE-04-39, Michigan State University, East Lansing, Michigan*, 2004.
11. T. F. Cootes, G. J. Edwards, and C. J. Taylor. Active Appearance Models. *In Proc. of European Conference of Computer Vision*, 2:484-498, 1998.
12. D. DeCarlo, D. Metaxas, and M. Stone. An Anthropometric Face Model using Variational Techniques. *In SIGGRAPH '98: Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, pages 67-74, 1998.
13. M. Everingham and A. Zisserman. Automated Visual Identification of Characters in Situation Comedies. *In Proc. of 17th International Conference on Pattern Recognition, (ICPR'04)*, pp. 983-986, 2004.
14. P. Fua. Regularized Bundle-Adjustment to Model Heads from Image Sequences without Calibration Data. *International Journal of Computer Vision*, 38(2):153--171, 2000.
15. R. Garimella and B. Swartz. Curvature estimation for unstructured triangulations of surfaces. *Technical report, Los Alamos National Laboratory*, 2002.
16. G. G. Gordon. Face recognition based on depth and curvature features. *In Proceedings CVPR '92*, pp. 808-810, 1992.
17. N. Grammalidis, N. Sarris, C. Varzokas, and M.G. Strintzis, Generation of 3-D head models from multiple images using ellipsoid approximation for the rear part. *In International Conference on Image Processing*, 1:284-287, 2000.
18. T. Heseltine, N. Pears, and J. Austin. Three-Dimensional Face Recognition Using Surface Space Combinations. *In Proc. of British Machine Vision Conference (BMVC'04)*, pages 527-536, 2004.
19. J. Huang, B. Heisele, and V. Blanz. Component-based Face Recognition with 3D Morphable Models. *In Proc. 4th Int. Conf. on Audio- and Video-based Biometric Person Authentication*, pp. 27-34, 2003.
20. T.J. Hutton, B. F. Buxton, and P. Hammond. Dense Surface Point Distribution Models of the Human Face. *In Proceedings of IEEE Workshop on Mathematical Methods in Biomedical Image Analysis, MMBIA 2001*, pp. 153-160, 2001.
21. M. R. Kaus, V. Pekar, C. Lorenz, R. Truyen, S. Lobregt, and J. Weese. Automated 3-D PDM construction from segmented images using deformable models. *IEEE Transactions on Medical Imaging*, 22(8):1005-1013, August 2003.
22. T. Kim and J. Kittler. Locally Linear Discriminant Analysis for Multimodally Distributed Classes for Face Recognition with a Single Model Image. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27(3):318-327, 2005.
23. J. C. Lee and E. Milios. Matching range images of human faces. *In Proceedings of ICCV*, pp.722-726, 1990.
24. Y. Lee, D. Terzopoulos, and K. Waters. Realistic Modeling for Facial Animation. *In Proceedings of ACM SIGGRAPH*, pages 55-62, 1995.
25. M. D. Levine and A. Rajwade. 3D View-Invariant Face Recognition Using a Hierarchical Pose-Normalization Strategy. *Technical Report http://www.cim.mcgill.ca/~levine/, Center of Intelligent Machines, McGill University*, 2004.
26. Y. Li, S. Gong, and H. Lidell. Support Vector Regression and Classification Based Multi-view Face Detection and Recognition. *In Proc. of IEEE Int. Conference on Face and Gesture Recognition*, pp. 300-305, 2000.
27. X. Lu, D. Colbry, and A. K. Jain.Three-Dimensional Model Based Face Recognition. *In Proc. of 17th International Conference on Pattern Recognition, (ICPR'04*), Volume 1:362-366, 2004.

28. Z. Mao, P. Siebert, P. Cockshott, and A. Ayoub. Constructing dense correspondences to analyze 3d facial change. *In Proc. ICPR 04*, pages 144-148, August 2004.
29. K. Messer, J. Kittler, M. Sadeghi, M. Hamouz, A. Kostin, S. Marcel, S. Bengio, F. Cardinaux, C. Sanderson, N. Poh, Y. Rodriguez, K. Kryszzcuk, J. Czyz, L. Vandendorpe, J. Ng, H. Cheung, and B. Tang. Face authentication competition on the Banca database. *In D Zhang and A K Jain, editors, Proc First International Conference on Biometric Authentication*, pages 8-15, Springer, July 2004.
30. F.I. Parke and K. Waters. *Computer Facial Animation*. A.K. Peters, 1996.
31. P. J. Phillips, P. J. Flynn, T. Scruggs, K. W. Bowyer, J. Chang, K. Hoffman, J. Marques, J. Min, and W. Worek. Overview of the face recognition grand challenge. *In IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 947-954, 2005.
32. P. J. Phillips, P. Grother, R. Micheals, D.M. Blackburn, E. Tabassi, and M. Bone. Face recognition vendor test 2002: Evaluation Report. *Technical Report IR 6965, http://www.itl.nist.gov/iad/894.03/face/face.html, National Institute of Standards and Technology*, 2004.
33. B. T. Phong. Illumination for computer generated pictures. *Commun. ACM*, 18(6): 311-317, 1975.
34. M. Proesmans and L. VanGool. Active Acquisition of 3D Shape for Moving Objects. *In ICIP*, pages 647-650, 1996.
35. S. Rusinkiewicz, O. Hall-Holt, and M. Levoy. Real-Time 3D Model Acquisition. *In SIGGRAPH '02: Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, pages 438-446, 2002.
36. J. Short, J. Kittler, and K. Messer. Comparison of Photometric Normalisation Algorithms for Face Verification. *In Proc. 6th Int. Conf. on Automatic Face and Gesture Recognition (FGR 2004)*, pp. 254-259, 2004.
37. J. P. Siebert and C. W. Urquhart. C3D: a Novel Vision-Based 3D Data Acquisition. *In Proc. Mona Lisa European Workshop, Combined Real and Synthetic Image Processing for Broadcast and Video Production*, Hamburg, Germany, 1994.
38. H. T. Tanaka, M. Ikeda, and H. Chiaki. Curvature-based face surface recognition using spherical correlation. Principal directions for curved object recognition. *In Proceedings of The Third IEEE International Conference on Automatic Face and Gesture Recognition*, pp. 372-377, 1998.
39. Y. Wang, X. Huang, C. Lee, S. Zhang, Z. Li, D. Samaras, D. N. Metaxas, A. Elgammal, and P. Huang. High resolution acquisition, learning and transfer of dynamic 3d facial expressions. *Comput. Graph. Forum*, 23(3):677-686, 2004.
40. Y. Wang, C. Chua, and Y. Ho. Facial feature detection and face recognition from 2D and 3D images. Pattern Recognition Letters, 23(10):1191-1202, 2002.
41. C. Xu, Y. Wang, T. Tan, and L. Quan. Three-dimensional face recognition using geometric model. *In Proc. SPIE Vol. 5404*, pp. 304-315, 2004.
42. M. Yang, D. J. Kriegman, and N. Ahuja. Detecting Faces in Images: A Survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(1):34-58, 2002.
43. L. Yin and M. T. Yourst. 3D face recognition based on high-resolution 3D face modeling from frontal and profile views. *In WBMA '03: Proceedings of the 2003 ACM SIGMM workshop on Biometrics methods and applications*, pages 1-8. ACM Press, 2003.
44. I.A. Ypsilos, A. Hilton, and S. Rowe. Video-rate Capture of Dynamic Face Shape and Appearance. *In Proc. 6th Int. Conf. on Automatic Face and Gesture Recognition (FGR 2004)*, pages 117-122, 2004.

45. L. Zhang and D. Samaras. Pose Invariant Face Recognition under Arbitrary Unknown Lighting using Spherical Harmonics. *In Proc. Biometric Authentication Workshop 2004, (in conjunction with ECCV2004)*, pp. 10-23, 2004.
46. L. Zhang, N. Snavely, B. Curless, and S. M. Seitz. Spacetime faces: high resolution capture for modeling and animation. *ACM Trans. Graph.*, 23(3):548-558, 2004.
47. Z. Zhang, Z. Liu, D. Adler, M.F. Cohen, E. Hanson, and Y. Shan. Robust and Rapid Generation of Animated Faces from Video Images: A Model-Based Modeling Approach. *International Journal of Computer Vision*, 58(1):93-119, 2004.
48. W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld. Face recognition: A literature survey. *ACM Comput. Surv.*, 35(4):399-458, 2003.

Chapter 2

A SURVEY ON 3D MODELING OF HUMAN FACES FOR FACE RECOGNITION

S. Huq, B. Abidi, S. G. Kong, and M. Abidi
Imaging, Robotics, and Intelligent Systems, The University of Tennessee, Knoxville, TN 37996-2100, USA. {mhuq, besma, skong, abidi}@utk.edu

Abstract: In its quest for more reliability and higher recognition rates the face recognition community has been focusing more and more on 3D based recognition. Depth information adds another dimension to facial features and provides ways to minimize the effects of pose and illumination variations for achieving greater recognition accuracy. This chapter reviews, therefore, the major techniques for 3D face modeling, the first step in any 3D assisted face recognition system. The reviewed techniques are laser range scans, 3D from structured light projection, stereo vision, morphing, shape from motion, shape from space carving, and shape from shading. Concepts, accuracy, feasibility, and limitations of these techniques and their effectiveness for 3D face recognition are discussed.

Key words: 3D face reconstruction, face recognition, laser range scanner, structured light, stereo vision, morphing, shape from shading, shape from motion.

1. INTRODUCTION

Face recognition has become one of the most active research fields in computer vision due to increasing demands in commercial and law enforcement applications. Biometrics-based techniques such as fingerprint and iris matching often require physical contact or cooperation of the user. Face recognition offers a reliable means for personal identification without requiring much of the participant's cooperation. Despite the success in various applications, recognizing human faces in an uncontrolled environment has remained largely unsolved. The appearance of a face, an inherently three-dimensional (3D) object, projected onto a two-dimensional

A. Koschan et al. (eds.), 3D Imaging for Safety and Security, 25–67.

(2D) space is sensitive to the variations in pose and illumination. Even, face variations of the same person created by pose and illumination changes could become larger than variations between individuals[1].

Face recognition techniques assisted with 3D facial models promise to overcome the difficulties and limitations associated with face recognition in 2D space. In addition to being an important additional feature in recognition, depth plays a crucial role in mitigating variations caused by pose and illumination. In the acquisition of facial images using cameras deployed for surveillance or access control, people move freely with their faces appearing at any angle. In this case, a smart face recognition system should be able to reproduce the same 2D rendering of the face as in the database for an accurate comparison. Once modeled in 3D, faces can accurately be back projected at any angle for matching.

The major goal of this chapter is to present a comprehensive review of the current technologies for modeling 3D human faces. Several related topics such as (1) the use of 3D face images in face recognition, (2) existing 3D face databases (publicly available), and (3) future trends of 3D face modeling and recognition have also been covered.

3D face modeling techniques can be divided into two classes - active and passive - depending on the imaging modalities and reconstruction methods. Active modeling techniques apply external lights such as a laser beam[2-4] or a structured light pattern[5-7] for 3D modeling. We refer the reader to an early survey on active range imaging conducted by Jarvis in 1983[8]. According to a similar survey conducted by Besl in 1988, at least six different optical principles were applied for 3D range imaging[9]. Among them, laser scanning and structured light are by far the techniques that have been most widely used for 3D modeling of human faces.

Passive modeling techniques use multiple intensity images of a scene usually captured without the sensors emitting any external lights. Images in passive reconstruction are acquired in one of three forms - single intensity images, a collection of multiple images captured simultaneously, or a time sequence of images i.e. video. Passive modeling techniques include stereo-vision[10,11], morphing[12-19], structure from motion[20,21], shape from space carving[22-24], and shape from shading[25-27]. Modeling from stereo images requires two or more images of the object acquired from different viewing points. Morphing needs one or more photographs taken from different view-points and knowledge of the generic shape of the object. Structure-from-motion methods extract motion information of feature points from multiple video frames to obtain 3D information. Space carving starts with the inclusion of the scene in a 3D space and then goes through the 3D space to decide which voxels should remain in the scene. The decision is made from more than two

2D images of the scene. Shape from Shading estimates the illumination direction in a 2D image to infer the 3D shape of the surfaces.

This chapter has been organized into eight sections. Section 2 describes 3D face model representation and introduces a few commercially available 3D face modeling devices. Sections 3 and 4 address the details of active and passive modeling techniques. Although currently active techniques are more widespread than the passive ones, we discuss both techniques in detail for two reasons: (1) we believe it is important to know which techniques have been used more than others and so the others can be assessed for their non-popularity and improvements; (2) active methods are unable to recover 3D facial structure from 2D facial images. Passive methods serve as the only alternates in this case since sometimes 2D images could be the only available data, captured by 2D sensors or scanned from identity cards, for modeling 3D face of a person. Shape from space carving has not been used for 3D modeling so far; yet we briefly present this technique for its potential in 3D face modeling. Section 5 compares face modeling techniques in terms of their performances in 3D face recognition, and discusses their accuracy and feasibility. Section 6 reviews uses of 3D face models in 3D assisted face recognition and presents a list of publicly available 3D face databases. Future trends in face modeling as they relate to the demands and goals of face recognition are discussed in section 7. Section 8 concludes this chapter.

2. DATA ACQUISITION AND REPRESENTATION

Active 3D reconstruction techniques based on laser scan and structured light have evolved into commercial devices that can acquire 3D facial data under more or less favorable lighting conditions and cooperation from the participants. Passive techniques can cope with more unfavorable conditions in real life for instance in outdoor scenarios. Gaining reliable accuracy in 3D modeling becomes relatively difficult, however. To the best of our knowledge, stereo is the only passive and commercialized technique that has been used for 3D face modeling. The following subsections give an overview of major 3D data acquisition systems and introduce several 3D data representation formats.

2.1 Image Sensors for 3D Face Reconstruction

According to Besl's survey[9] active 3D scanners can operate by TOF (Time-Of-Flight), triangulation (with a camera), Doppler phase shift, inter-ferometry, or confocal principles. Scanners with interferometry and confocal principles have depth accuracy within nano- to micro-meter range

but they work in millimeter range of field of view and depth of field. TOF and phase shift scanners work within several meters of range but their depth accuracy is in an order of centimeters (thus they are only appropriate for scanning large scenes such as buildings). Triangulation has working range in the order of meters and accuracy in micrometer to millimeter. Therefore, common active scanners for acquiring 3D facial data are laser range scanners and structured light projectors. They apply the triangulation technique.

A number of commercial range scanners with significant reconstruction accuracy are available. Minolta Vivid® 900, a CCD device, projects a horizontal stripe laser on the object and scans it by a galvanic mirror[28]. The manufacturer's claim on its depth resolution is 0.27*mm* and mean error is 0.2*mm*. Vivid acquires the texture just after acquisition of shape, which can result in subject motion to cause poor registration between the texture and shape. Cyberware™ 3030PS, another laser scanner that has been used to scan faces, is claimed to have 0.2*mm* depth resolution[4]. FastSCAN (www.polhemus.com) is a handheld laser scanner that takes around 15 seconds to scan a human face with 0.5 mm depth resolution. The commercial laser range scanner, IVP Ranger M50® senses only the laser light reflected back from the object, computes the 3D points inside the camera, and delivers them directly to the PC eliminating the need for expensive post processing or high speed frame grabbers.

3DFaceCam® is a commercial sensor that reconstructs human faces from structured light projections[5]. Figure *2-1* shows some examples of structured light patterns. Figure *2-1*(a) and Figure *2-1*(b) are two rainbow like patterns used by 3DFaceCam®. To avoid occlusion, 3DFaceCam® uses two cameras to sense the structured light pattern, builds the two symmetric halves of the face, and combines. Patterns with stripes of varying resolution are used to index the columns. Figure *2-1*(c), on the other hand, is binary coded such that the stripes can be self-indexed from a single pattern[29]. Please note that unless otherwise mentioned the authors themselves processed or produced all the results and illustrations used in this paper.

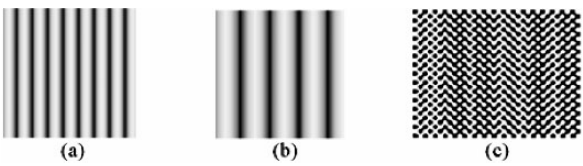


Figure 2-1. Some structured light patterns: (a) Rainbow like pattern in high resolution, (b) Rainbow like low resolution pattern, and (c) Column coded structured light pattern (picture 1(c) taken from the paper authored by Vuylsteke[29]).

Table 2-1. Commercially Available 3D Digitizers.

Techniques	Company	Model	Depth Resolution (mm)	3D Face Reconstruction Speed (sec)
Laser Scan	Konica Minolta, USA	Minolta Vivid	0.02	0.3-2.5
Laser Scan	Cyberware, Canada	3030RGB/PS	0.10	8
Laser Scan	Polhemus, USA	FastSCAN	0.5	15
Structured Light	Konica Minolta, USA	Minolta 3D 1500	N/A	0.125
Structured Light	Vitrionic, Germany	VITUS Ahead	1.0	8
Structured Light	Genex Tech., USA	3DFaceCam	N/A	5-30
NIR Structured Light	A4Vision, USA	A4Vision	N/A	0.2
Stereo	Turing Institute, Scotland	C3D	0.5	< 1
Stereo	Geometrix, Germany	FaceVision 200	N/A	60
Stereo	3DMD, USA	3DMDface	<0.5	N/A

Stereoscopic cameras are commercially available devices that use stereo techniques for 3D imaging. Two or more intensity images are captured simultaneously (for moving objects) or sequentially (for still objects) from slightly different viewpoints to allow the reconstruction of a scene by triangulation known as stereo vision[30]. Examples of commercial stereo systems for face reconstruction are FaceVision®[31], 3DMDface™ (www.3dmd.com), and C3D[32]. 3DMDface™ is developed especially for facial surgery planning and assessment but has also been used for face recognition biometrics (www.3dmd.com). It has six cameras, one color and two black and white appearing each side of the face, to capture ear to ear shape and texture.

Laser scanners, structured light systems, and stereo systems are available as commercial products built by a number of industries. There are many commercial systems for 3D reconstruction in general. Table *2-1* lists some of these that have been used for 3D face modeling. A comprehensive scanner list can be found in www.rapidform.com/220 and in a recent survey on range sensors by Blais[33].

2.2 3D Image Representations

3D images are represented as point clouds or mesh format. In the point clouds representation, the 3D model is described in distinct Cartesian (X, Y, Z) or cylindrical (h, r, φ) coordinates. Texture and normal of each point are added if available and necessary. In mesh representation, neighboring points

are connected together to tessellate the surface with triangles or polygons called faces. Mesh represented file generally contains vertices (3D points), faces, texture of each vertex, and normal of each face.

Several standard 3D data formats exist which are designed in accordance with the needs for scene details, fast visualization, space efficiency, or fast processing such as feature extraction. In *point cloud* format (*X*, *Y*, *Z*) coordinates, (R, G, B) color values, and normal of each 3D point are written explicitly in the file. PLY, developed by Stanford University, is a mesh represented format and describes only one object as a group of elements. A typical PLY file contains just two elements, the (*X*, *Y*, *Z*) triples and the triple indices for each face. STL format contains triangular mesh. The STL specifications require that all adjacent triangles share two common 3D points which facilitates extraction of features such as surface curvature. In VRML (Virtual Reality Modeling Language) format, a 3-D scene can be described in high details by a hierarchical tree structure of nodes. VRML format, known with extension .wrl, supports 54 different types of nodes - some of them for interactive visualization. To save space, oftentimes surface with less detail is represented with fewer numbers of polygons or control parameters. Smooth surfaces use fewer control parameters than surfaces with more details. Along with local shape, these parameters may have control over the global shape[34]. OBJ file format, developed by Wavefront Technologies (acquired by Silicon Graphics, CA, USA later, see www.sgi.com) supports lines, polygons described in terms of their points and free-form curves and surfaces defined with control points. MDL, developed by Cornell University, is fast to read (hence visualize) and write and is reasonably space-efficient.

One should note that many 3D scanners sample on a rectangular grid facilitating a way of more explicitly describing neighborhood connectivity among vertices. Such connectivity is often useful to preserve. Standard formats such as VRML make that connectivity difficult to recover.

Figure 2-2. 3D data representations: (a) Represented cloud of points, (b) Triangulated 3D mesh model, and (c) A shaded 3D model of a human face.

Figure *2-2* illustrates 3D image models of a human face generated by 3DFaceCam® and visualized with the 3D viewing software RapidForm (www.rapidform.com). Figure *2-2*(a) is a point clouds, un-textured representation of a 3D facial model of 5399 points, Figure *2-2*(b) is a triangular mesh representation of the 3D face with 9493 triangles, and Figure *2-2*(c) shows the shaded 3D face. In MDL format, the 3D model required 222 KB of space which was a decimated version of the initial model of 1.1 MB.

3. ACTIVE MODELING TECHNIQUES

In active modeling, an external light is projected on the object surface. Two well known active reconstruction methods, laser scanning and structured light projection, have been widely used for modeling human faces. In the following two subsections, we review these two techniques and their underlying theories.

3.1 Laser Range Scanning

In laser-scanning techniques, a laser line or a single-spot laser beam is scanned over the subject of interest to obtain a sequence of scan images from where a 3D surface of the subject is recovered[35]. The intensity of the laser needs to be low enough to be tolerable to eyes and at the same time distinguishably contrasting with the facial complexion. In a laser range profiler, the camera senses light around the wavelength of the emitted laser and thus avoids the unwanted background. As an alternate, visual images of the profiles can be captured and software filters can be applied to extract the profiles from the background[36]. The filters apply Gaussian weights based on the fact that the intensity variation across a light stripe is Gaussian in nature. Besides, a Gaussian filter can locate the profile in sub-pixel. The acquired profiles are triangulated to obtain depth information. For high performance face recognition, a higher reconstruction resolution is required. There are accuracy levels in a scanner (one micron for example) that are not necessary because faces can change several millimeters or more over time[37]. Triangulation methods can achieve resolutions in the sub-millimeter range. Chang et al.[38] have shown that this resolution is well above the minimum; only beyond this minimum recognition rate is affected.

The name ‘triangulation’ comes from the fact that depth is estimated from a triangular set-up: the laser source, the camera, and the object form a triangle. In triangulation, the location of a point on the laser profile captured by the CCD is related to the depth of the object point. When a horizontal laser line is projected on a surface having some depth variations, the line is

bent accordingly. The amount of bending depends on the depth of the surface with respect to its background. To obtain an estimation of the depth, laser profiling systems need first to be calibrated. In the calibration process, a pattern with known and marked (*X*, *Y*, *Z*) points is used to establish the relation between the image coordinates (*x*, *y*) and the world coordinates (*X*, *Y*, *Z*). For reconstruction, a video or set of images of the object is captured while scanning with the laser profile. Each frame in the video contains a view of the laser profile projected at a different location on the object's surface. During the scanning process, either both camera and laser together or the object alone are moved in linear motion. Without disturbing the setup, a scanned video of the calibration pattern is captured in the same manner. Depth can be recovered by extracting calibration parameters from the second video and applying them to the profiles in the first video.

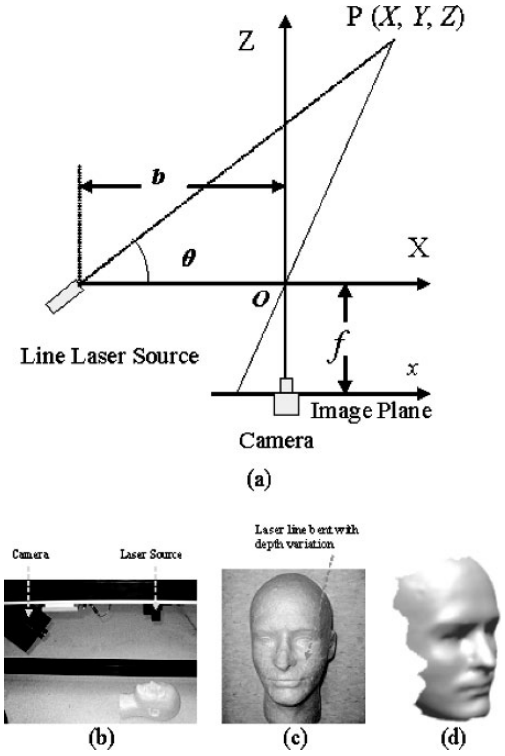


Figure 2-3. 3D Face Reconstruction using laser scanning: (a) Basic geometry of active triangulation from laser scanning, (b) a laser scan setup at the IRIS lab, (c) one frame of the laser scan sequence with the profile seen at the nose base, and (d) the reconstructed dense 3D head model.

Figure *2-3* shows the working principle of the triangulation method. The laser source is located on the baseline looking at an angle θ and at a distance b from the camera's optical axis. The center of projection of the camera is at the origin of the world coordinates (X, Y, Z), the optical axis is aligned with the Z axis, the X axis represents the baseline and is also aligned with the x axis of the image plane. The Y axis is orthogonal to the plane (X, Z) and aligned with the y axis of the image plane. The intersection of the plane of light with the scene surface is a planar curve called *stripe* and is observed by the camera. The 3D point P (X, Y, Z), whose projection on the image plane is at (x, y), is given by

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \frac{b}{(f\cot\theta - x)} \begin{pmatrix} x \\ y \\ f \end{pmatrix} \tag{1}$$

where f is the camera's focal length. In the calibration process, the unknown parameters b, θ, and f are computed from the system of nonlinear equations described in (1) with known (x, y) points and their corresponding (X, Y, Z) coordinates.

Depth recovered applying triangulation technique has high accuracy. Active triangulation for 3D facial reconstruction with laser scans was used by Xu et al.[39]. Two cameras (instead of one as shown in Figure *2-3*) were engaged to avoid occlusions. A rec
scan time of a complete face of 40 seconds were achieved.

To scan human faces, laser scanners are required to have eye-safe laser. To overcome low contrast problems of eye-safe lasers in the presence of ambient light, light in the invisible range, such as infrared with wavelengths from $0.7\mu m$ to about $0.1mm$ seems convenient. In case of visible lasers, low contrast problem can mostly be resolved by installing a laser filter in front of the sensor. While scanning, which can take several seconds to complete, the subject has to remain still - a major drawback of laser scanners. Besides, the translucency of the eyes can cause spikes in reconstructions.

3.2 Structured Light Projection

In structured light methods, a plane of light scans the surface. The scan can be accomplished serially with a single-spot beam but for convenience the scan is performed in parallel with a spatial pattern of light illuminating the whole surface of the subject simultaneously in one shot. The patterns of light are distinguishable patterns, such as lines, dots, etc., of high intensity coherent light (laser) or incoherent monochromatic light in either the visible

or the invisible range and also distinguishable from the surface of the object. Since depth can be recovered from one view of the illuminated object structured light technique waives the necessity of capturing video and allows modeling of face of a moving person such as someone accessing a restricted facility.

Technically, projecting a single laser profile and capturing a video is analogous to projecting multiple profiles simultaneously and acquiring a single shot. Therefore, the reconstruction algorithm from structured light projection is in principle similar to the technique used in laser scanning. There are two possible cases with location of the light source: the source is at infinity (i.e. the beams are parallel) or it is close to the object (i.e. the beams form a pencil). Availability of projector has made the second case more popular. In both cases, deformations of the light stripes (or patterns) contain depth information of the surface. The base line length is similar to the laser scanner in the projector case. The number of calibration parameters varies with variations in camera-projector setup. The 2D image coordinates of the points on the stripes are combined with the calibration parameters to compute the 3D coordinates of the points.

Figure *2-4* shows a 3D scene reconstruction using structured light projections. The structured light face image and its 3D model are obtained from the 3DfaceCam® by GenexTech. The structured light pattern shown in Figure *2-4*(b) is similar to the rainbow like pattern of Figure *2-1*(a) and Figure *2-1*(b). 3DfaceCam® uses Eq. (1) for reconstruction with different θ for the different stripes of the rainbow[5].

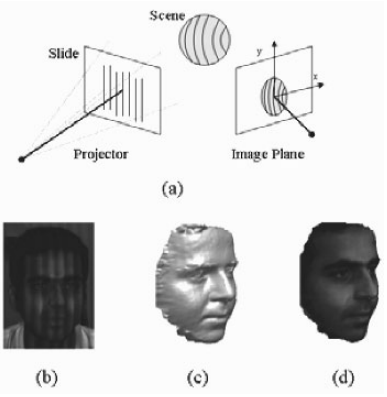


Figure 2-4. 3D Face Reconstruction from structured light projections: (a) A simple structured light projector setup, (b) Structured light projected on a face (the real structured light projected on the face is black and white, see Figure *2-1*), (c) Shaded 3D face, and (d) Textured reconstructed face. The structured light face image and its 3D model are captured with the 3DFaceCam developed by GenexTech, Inc.

3DfaceCam® projects two structured light patterns (Figure *2-1* (a) and Figure *2-1*(b)) with each of them shifted temporally and projected more than once. Although multiple shots may help the algorithm achieve more reliability and robustness in modeling, movement of the head can introduce error. 3DfaceCam® is claimed to have 0.5*mm* of accuracy. Estimation of this accuracy, however, is based on scanning of planar objects.

In structured light reconstruction, the stripes of the pattern need to be distinctly detectable. Beumier and Acheroy[6] reconstructed faces by encoding the stripes with varying thickness. Specific lenses were used to project the stripes and achieve a sufficient field of view and depth of focus. The encoded stripes were projected on a square object fixed at a known depth for calibration. The optical axes of the camera and the projector were coplanar and perpendicular to the light stripes. Such constrained setup reduced the number of calibration parameters to 7. Both the projector and the camera were rotated 45 degrees from the vertical line so that stripes projected on the vertically and horizontally aligned facial features such as mouth and eyebrows could also be easily detected. Vertical continuity constraint was enforced in the stripe detection phase. A median filter was used to smooth out local bumps. Horizontal positions of the stripes were obtained at sub-pixel locations by applying interpolation to the horizontal profiles. The reconstruction process took about 0.5 seconds. The depth resolution of the reconstruction was not discussed but the 3D face database built was successfully used in 3D face recognition[6].

Garcia and Dugelay[7] used homographies to avoid constraints on orientation of the camera and projector. In their setup, a grid pattern of vertical and horizontal lines was projected on a calibration pattern from an arbitrarily oriented projector. The projected grid appeared on the calibration pattern which was also a grid. The deformed image of the projected grid was captured by an arbitrarily oriented camera. Projection grid, calibration grid, and the image plane were related through homographies (transformations that map between points on two planes). A projective transformation between the calibration grid and the image plane was used to estimate the calibration parameters that relate the 3D space to the 2D projective plane. Faces were reconstructed partially from different angles and then combined to obtain complete 3D faces.

Structured light for 3D reconstruction, theoretically, is not affected by the movement of the object in the scene since a single view suffices to recover 3D information of the whole scene. A drawback is that sufficient number of lines for high-resolution reconstruction cannot be projected at one time. Narrow gaps between lines reduce the contrast between dark and bright lines and exaggerate smeared edges caused by light diffusion. Diffusion of light lowers the peak intensity values making the differences difficult to be

detected. Thus, the reconstruction obtained is usually sparse; while depth details inside the stripes are lost. This effect is seen in 3DfaceCam® generated models which appear as if built combining stripes. Detection of the stripes may become difficult if their colors match the facial texture or features. 3DfaceCam® gives the option to adjust intensities of the structured light for dark, brown, or fair complexion of the face. Light of the pattern of this device is in the visible range.

Some projectors use invisible light patterns. By using structured light in the near-infrared (NIR), the A4Vision projector built by A4Vision (www.A4Vision.com) enables onsite capture of 3D images. NIR has allowed this system to become more tolerant to ambient lighting conditions and independent from background color and artificial features such as make-up.

Active reconstruction techniques are currently the dominant technology for capturing 3D faces. Their geometric accuracy has continually improved. However, they are expensive and can have a number of technical limitations: (1) they are invasive hence unable to reveal the details beyond a limit; current active techniques use stripe or dot patterns that are wide hence hinder the detection of details of the surface beyond a limit; (2) they are not scalable; faces captured from long distance can not be reconstructed densely or reliably which is also true for passive reconstruction but to a lesser extent.

4. PASSIVE MODELING TECHNIQUES

This class of 3D modeling techniques comprises the following approaches: 3D from Stereo, Morphing, 3D from Motion, and 3D from Shading. Shape from space carving has not been applied for 3D modeling of human faces; however we present this technique briefly for its potential. The following subsections review these five techniques.

4.1 Stereo Vision

Stereo-vision uses triangulation with two or more 2D projections of the object to reconstruct its 3D model[40]. Triangulation is formed by two camera locations and the object. Key of depth recovery in stereo is that if a scene is captured with two cameras from slightly different viewpoints, points in one image will be shifted in the other. The amount of shift is called disparity. The higher the disparity is the smaller the depth. If correspondences between scene points of the stereo images are established well, stereo systems can recover highly accurate depth. Figure *2-5*(a) illustrates a binocular stereo setup with two identical cameras focused on the same scene.

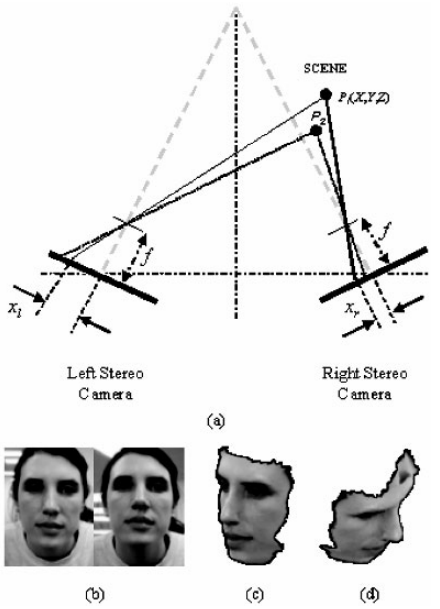


Figure 2-5. 3D Face Reconstruction from Stereo: (a) Binocular stereo vision system showing how depth is inversely proportional to disparity, (b) Stereo pair, and (c)-(d) Reconstructed 3D face.

Point $P_1(X, Y, Z)$ appears at distances x_l and x_r away from the focus point respectively in the left and right images. The disparity $|x_l - x_r|$ for P_1 is less than the disparity for point P_2 located relatively closer to the cameras. A number of parameters are needed to establish relation between the 2D coordinates of the image points (x, y) and the 3D world coordinates (X, Y, Z) of the scene. In a calibration process two types of parameters are estimated: (1) extrinsic, related to the relative position of the cameras and (2) intrinsic, related to the camera itself. The intrinsic parameters are camera characteristics such as focal length f, size of pixel in terms of length s_x, width s_y, and coordinates of the principal point (O_x, O_y).

The principal point of a camera is the point that remains unchanged with variations in zoom. The fact that identical stereo cameras have the same intrinsic parameters makes the parameters' recovery easy in the calibration process. Non-identical parameter concept is of particular interest in the case of video where objects are reconstructed even if the effective focal length of a camera changes frame to frame as the camera wants to keep a moving object in focus[41-43]. The extrinsic parameters are the three rotations and three translations of the camera with respect to the world coordinate system. In calibration, a set of linear equations comprising unknown intrinsic and extrinsic parameters that relate (x, y) to (X, Y, Z) are formed and solved with known (X, Y, Z) and (x, y). These can be obtained respectively from a

calibration pattern of known scale and the stereo images of this pattern. Neglecting the lens's radial distortion, an image warping phenomenon encountered with smaller lenses with a wide field of view, the extrinsic and intrinsic parameters can be represented by two transformation matrices M_{ext} and M_{int}. Assuming O_x=O_y=0 and s_x=s_y=1, a single transformation matrix M can be written as[44]

$$M=M_{int}M_{ext}=\begin{bmatrix}-f/s_x & 0 & O_x\\ 0 & -f/s_y & O_y\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}r_{11} & r_{12} & r_{13} & -R_1^TT\\ r_{21} & r_{22} & r_{23} & -R_2^TT\\ r_{31} & r_{32} & r_{33} & -R_3^TT\end{bmatrix}=\begin{bmatrix}m_a & m_b & m_c & m_d\\ m_e & m_f & m_g & m_h\\ m_i & m_j & m_k & m_l\end{bmatrix} \quad (2)$$

Here r_{ij} are the elements of R, rotation of the world coordinates with respect to the camera coordinates and R_i is the i-th column vector of R. T is a translation vector from the origin of the world coordinate system. In a homogeneous coordinate system, we obtain the linear matrix equation describing the perspective projection as:

$$\begin{bmatrix}u\\ v\\ w\end{bmatrix}=M\begin{bmatrix}X\\ Y\\ Z\\ 1\end{bmatrix} \quad (3)$$

The image coordinates are defined as $x = u/w$ and $y = v/w$. When unconstrained, M, with 12 parameters, describes the full-perspective camera model and is called a *projection matrix*. We can divide both sides of (3) by m_l and reduce the number of parameters to 11 leaving the reconstruction to happen up to a scale. If (x_l, y_l) is an image point in the image captured by the left camera, we can then write

$$x_l=\frac{u}{v}=\frac{m_1X+m_2Y+m_3Z+m_4}{m_9X+m_{10}Y+m_{11}Z+1} \quad (4)$$

$$y_l=\frac{v}{w}=\frac{m_5X+m_6Y+m_7Z+m_8}{m_9X+m_{10}Y+m_{11}Z+1}. \quad (5)$$

The parameters, m_1, …, m_{11} are obtained by dividing m_a, …, m_l by m_l and must be determined at least from 11 equations. 11 equations can be obtained

from five and a half points from the left image and their corresponding (*X*, *Y*, *Z*) on the calibration pattern. Similarly, another set of 11 parameters relate the right image plane to the world coordinates.

Building 3D model from stereo images requires performing two major tasks: matching and reconstruction. Stereo matching uses texture or gradient as tokens for matching. Even after imposing constraints such as ordering and disparity gradient on the matching criterion, matching two stereo face images remains difficult due to the lack of sufficient gradient in face regions. Therefore, stereo matching of human faces has often used other constrains. Based on all these constraints, matching techniques can be categorized into two - *3D model based matching* and *structured light assisted matching*. In the first category, a generalized 3D face model provides several constraints to the stereo matching algorithms for improved matching.

A 3D model-based energy minimizing algorithm was described by Lengagne et al.[45]. The purpose of the cost function was to minimize the amount of curvature between the evolving 3D face and the given 3D model. The curvature was enforced with differential constraints that were estimated from the model. This method requires registration of the 3D model with 2D images. It is not necessary to register the whole 3D model since not all the regions of human faces are difficult to match. Baker et al. have used prior knowledge of only the nose for its better reconstruction[46].

In the second category, structured light is projected on the face while capturing the stereo images. The structured light images are then matched almost explicitly. Example works are Enciso et al.[47] and D'Apuzzo[48] where the authors exploited noise like patterns for matching.

Without being enforced by 3D models or structured light matching tends to introduce false correspondences hence inconsistencies in the model. Using multiple stereo pairs may help avoid such inconsistencies. Medioni and Pesenti reconstructed faces from a sequence of face images captured by one camera[49]. The sequence was used to model large portion of the head by combining the partial models obtained by treating each two subsequent frames as stereo pair. Chen and Medioni have shown that it is possible to densely match faces in two stereo images without the assistance of structured light but by imposing additional constraints[30]. In their algorithm, starting with some seed points, matching on the whole image was spread by enforcing the disparity gradient and ordering constraints. While the disparity gradient constraint ($|\Delta d|/|\Delta x| < 1$ where d is the disparity of an image point (x, y)) limits the search area of the matching, the ordering constraint stipulates that points appear in the same order in both stereo images and helps avoid mismatches. Chen and Medioni's work has been used by GeoMetrix, Inc. to build a commercial stereo system FaceVision™ 200 for modeling 3D human faces[30]. Lao et al. reconstructed faces sparsely from trinocular stereo

images[50]. Three cameras were used to capture lower, central, and upper views of the face. Stereo correspondences were established at isoluminance lines. Isoluminance lines are the boundaries between black and white portions on the face obtained from thresholding. Edges and boundaries of facial features such as eyes, nose, and mouth reveal themselves through isoluminance lines.

In adverse lighting conditions, stereo correspondence algorithms might fail and so might the depth recovery process. Stereo systems that exploit structured light to assist in matching are more tolerant to lighting variations. Depth accuracy is also function of the distance between the face and the cameras. For a faithful reconstruction, Medioni and Waupotitsch limited the face motion to 30*cm* from its optimal reconstruction distance from the camera[51]. The 3D face in Figure *2-5* was reconstructed from a binocular stereo setup in IRIS Lab, UTK. The subject appeared at around 2 feet away from the cameras.

In more accurate modeling of faces, stereo setup has to be adjusted for first, relative positions of the cameras and second, position of both cameras relative to the face. When optical axes of the camera are parallel (called parallel axes setup), distance between the two stereo cameras (called baseline) needs to be small to obtain enough overlap among the two images. In this case, stereo matching becomes easier but estimation of depth becomes more sensitive to image resolution. The cameras are often twisted to obtain bigger overlap. Although depth becomes less sensitive to image resolution stereo matching becomes difficult. Then, structured light assisted matching can help. Commercial stereo systems 3DMDface™, 3D-Matic (www.3Q.com), and FCS2 (www.surfim.com) use random texture pattern[52]. Structured light assisted stereo is known as photogrammetry or active stereo[53]. Stereo which is not structured light or external light assisted is called passive. Huq et al. has applied active and passive stereo together to model 3D human face[54]. The objective was to fill up the holes left by undetected laser stripes on the face.

In building 3D faces from stereo, the cameras need to be located conveniently so that the whole face appears in the overlapped region. A trick to obtain relatively wide baseline avoiding occlusion is to capture the images placing the cameras vertically[30,50] (see Figure *2-5*).

It was shown by Boehnen and Flynn[37] that reconstructions with laser scanner has higher accuracies than with stereo, which is contradictory with the fact that laser scanner hides details within stripes whereas stereo gives access to pixels hence should be able to reveal more details. Details at discontinuities of surfaces are lost more or less if matching in stereo is based on window correlation. There exist pixel level matching algorithms with discontinuity preserving ability[55-57]. Graph-cuts developed by Boykov et al.[55]

and Kolmogorov and Zabih[56] is such an algorithm which has been used for fast 3D human face modeling (3 seconds/person)[58].

Stereo reconstruction could be Euclidean or Metric i.e. a scaled version of the Euclidean. Given the intrinsic parameters are already known, two stereo images suffice to estimate the external parameters hence recover depth in Metric space[30]. True Euclidean scale is not necessary for 3D assisted face recognition since all faces are usually normalized to the same scale for matching. In binocular stereo vision, both cameras must have identical intrinsic parameters in order to avoid calibration with the use of pattern[30]. If the parameters are not identical the images only allow recovery of the epipolar geometry[59]. Epipolar geometry, which is a widely studied subject in computer vision[44,60,61], tells that points on lines (called epipolar lines) in one image match on lines (epipolar lines) in the other image. Line correspondences in epipolar geometry limit the two dimensional search of stereo matching to one dimension.

4.2 Morphing using a Generalized Model

Morphing requires that a generic shape of the object to be modeled be known *a priori*. A generic shape of human face can be acquired using laser scanners or structured light based systems by capturing several models and then averaging them. A generic 3D face model can be morphed (i.e. deformed under constraints) to obtain a person-specific texured 3D model. Morphing process (i.e. deformation of the generic 3D model) is regulated in such a way that 2D projections of the evolving 3D model approach as much as possible to one or more 2D intensity images of the face[62] (Figure *2-6*). This indicates the necessity of registration between the morphable 3D model and the 2D images. Morphing starts from a number of feature points on a face registered with the generalized 3D model. Initial registration of the feature points can be done manually or automatically. The registration is then spread over the whole face and refined iteratively. A morphing process can engage more than one 2D image to avoid imposing greater number of constraints while deforming the generic shape [63]

Blanz et al. used morphing to model 3D human faces[4]. Along with a generic model they used a number of person specific laser scanned faces [100 males and 100 females] and computed the probability distribution of depth of surface points on the faces to constrain the range of allowable deformations. The key idea was that if a large database of 3D head models of a variety of faces and correspondences among them are given then it is possible to obtain the 3D model of a new person. Using such database, an analysis by synthesis loop was used to find the morphing parameters such that the rendered image of the 3D model came as close as possible to the 2D

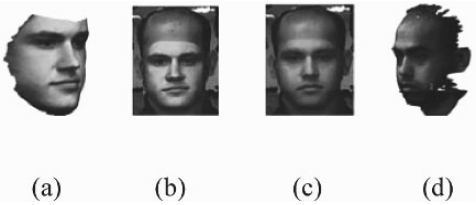


Figure 2-6. Illustration of facial reconstruction from morphing: (a) A 3D template face, (b) 3D template superimposed on a 2D facial image, (c) an intermediate stage of morphing 3D template, (d) reconstructed 3D face.

image of the person. The algorithm extracted the shape of the face using a single 2D input image which could be anywhere in the range of frontal to profile views. The output was a 3D textured model of the new face.

The technique cited above has two major registration tasks to perform - the registration among the 3D scanned head models and the registeration of the 3D generic facial model with 2D intensity images of the face that needs to be modeled. The generic 3D face model is obtained by averaging the scanned models. The registration among the 3D models is established using optical flow (described in section 4.3). A scanned 3D model is represented by $I(h, \phi) = (R(h, \phi), G(h, \phi), B(h, \phi), r(h, \phi))^T$ in a cylindrical coordinate system where h is the height along the central axis, ϕ is the angular coordinate, and r is the distance of the voxel from the central axis. R,G, and B are the color components. To establish correspondence between two models $I_1(h, \phi)$ and $I_2(h, \phi)$, the algorithm computes the flow field ($\delta h(h, \phi)$, $\delta\phi\,(h, \phi)$) such that $\|I_1(h, \phi) - I_2(h, \phi)\|$ is minimized. In high contrast regions, the flow vectors are stable but in face regions with low or no contrast at all, the optical flow algorithms tend to fail. In low contrast regions the flow vectors are computed such that a minimum-energy arrangement is achieved.

To morph the 3D generic model, which is an average of all models in the database, each model in the database is first described by its shape and texture parameters. These parameters are the first m largest eigenvalues, α for shape and β for texture. Next, the 3D model is rendered to obtain 2D images. A rendering parameter vector ρ that includes azimuth and elevation of the camera, scale, rotation and translation of the model, and intensity of ambient and directed light is introduced. Using the parameter vectors α, β, and ρ and Phong's illumination model[64] the rendered images obtained under perspective projetcion are given by

$$I_{model}(x,y) = \left(I_{R,model}(x,y), I_{G,model}(x,y), I_{B,model}(x,y)\right)^T \qquad (6)$$

where $I_{R,model}(x, y)$, $I_{G,model}(x, y)$, and $I_{B,model}(x, y)$ are the *R*, *G*, and *B* color intensity values of the rendered 2D image. The problem is formulated as obtaining the set of parameters α, β, ρ so that a posterior probability,

$$p\left(I_{input} \mid \alpha, \beta, \rho\right) \approx \exp\left(-\frac{1}{2\sigma_N^2} \left\| I_{input}(x, y) - I_{model}(x, y) \right\|^2 \right) \tag{7}$$

is maximized. σ_N is the standard deviation of the Gaussian noise in the input image. The posterior probability is maximized if the cost function *E* given by (8) is minimized.

$$E = \frac{1}{\sigma_N^2} E_I + \sum_{j=1}^{m} \frac{\alpha_j^2}{\alpha_{S,j}^2} + \sum_{j=1}^{m} \frac{\beta_j^2}{\alpha_{T,j}^2} + \sum_{k=1}^{n} \frac{\left(\rho_k - \mu_{\rho_k}\right)_j^2}{\sigma_{\rho_k}^2} \tag{8}$$

σ_N is initially set to a relatively large value and then reduced to obtain the maximum quality matching. E_I is given as $\sum_{x,y} \left\| I_{input}(x, y) - I_{model}(x, y) \right\|$. $\sigma_{S,j}^2$ and $\sigma_{T,j}^2$ are the *j*-th eigenvalues of the shape and texture covariance matrices, and μ_{ρ_k} and $\sigma_{\rho_k}^2$ are the mean and variance of the *k*-th parameter of vector ρ of size *n*. The parameter vectors α, β, and ρ are optimized using gradient descent algorithm with analysis by synthesis loop which takes about 150 seconds to converge.

A similar method was described by Hu et al.[65]. Unlike the work of Blanz et al.[4] where the deformation was done globally without extracting facial features, morphing in the work of Jiang et al.[66] were performed at the level of feature points, i.e., sparsely. 100 laser scan heads were used and only shape parameters were considered to generate a person specific face from its frontal view. Relevant features of the face were registered on the 3D models. Alternatively, a set of eigenvalues were computed iteratively so that the deviation of the deformed model from the average model resembled the frontal view. The use of shape parameters at only some selected feature points made the algorithm faster and limited the reconstruction time to 4 seconds. Since illumination factors were not considered, only frontal faces under normal illumination, i.e., in the absence of any spot lights, were valid for modeling.

Yan and Zhang used a single generic 3D model and multiple 2D images of a face taken at different poses[17]. A profile and a frontal view were used to deform the generic face into a person specific 3D face through a coarse to fine level deformation. A similar 3D facial modeling with morphing was developed by Sarris et al.[67] who used a single 3D model, a frontal view, and a profile view. In both works, the deformation was performed by minimizing

a distance based cost function. Zhang and Cohen used a cubic explicit polynomial for the deformation[10]. They used a generic 3D model and multiple 2D images of unknown facial poses.

3D face modeling from a single generic 3D model and a single image of an un-calibrated 2D face image is described by Ho and Huang[68] and Xing et al.[69]. A set of control points, *V*, was defined on the facial landmarks of a generic 3D model. A cost function was developed based on four weighted estimates: 1) distance between intensities of the given 2D image and the projected image, 2) symmetry, 3) distance between the estimated 3D points and 3D points of the generic model, and 4) model ratios. The estimates were obtained from *V* and the cost function was optimized for a set of control points, *V'*, featuring the 3D facial model of an individual. In the work of Xing et al.[69], facial feature points on the 2D image were located first and depths at those points were estimated through multi-variant linear regression. Radial Basis Function based interpolation was utilized to morph a 3D generic model. Radial Basis Function methods can interpolate points of more than one variable from a given set of irregularly positioned points. Experiments in these works have shown that the recovered model has satisfying visual effects for a broad range of poses. Although modeling a face from a single un-calibrated image may be useful in some applications, obtaining faithful reconstructions from such procedure is quite challenging, especially when illumination parameters remain unknown.

Morphing is computationally expensive due to its iterative nature. Sometimes manual manipulation in alignment of features with the generic 3D model is required. These limitations make morphing impossible to use in real time applications. Nevertheless, morphing is able to reconstruct a 3D face from a single image (up to a scale factor). Face Recognition Vendor Test (FRVT) 2002 applied morphing to reconstruct 3D face from a single image and rendered the 3D face to obtain 2D face at frontal pose. This method was proven to be effective in achieving higher recognition rate[70].

4.3 Shape from Motion

Shape-from-motion, or 3D reconstruction from video, builds a 3D model of an object from a video sequence. The video needs to have at least two frames and small disparities between points in consecutive frames. In videos with big inter frame disparities, the object can be reconstructed using stereo when the camera intrinsic parameters are known[49]. Reconstruction from video captured with a camera of unknown parameters requires self-calibration. For self-calibration, at least three frames of video are needed to estimate the calibration parameters and obtain reconstruction up to a scale factor[71,72].

Depth from a video can be recovered using one of two major approaches - either by tracking features or computing the optical flow. The feature tracking approach requires matching tokens temporally at a few selected points. The reconstruction gradually gains stability and consistency over many frames. The image coordinates of the feature points in consecutive frames are used as input to the reconstruction algorithm. In this approach, extrinsic parameters - rotational and translational vectors of the object under reconstruction - are determined from the video itself. An orthographic projection is usually assumed. Hence, camera calibration can be ignored if reconstruction is accepted only up to a scale factor.

In contrast, optical flow is a differential technique and can be computed either densely at every point or sparsely at a few feature points from spatial and temporal derivatives of image brightness. In this case, computed motion fields at each point are used as input to the reconstruction algorithm. The camera intrinsic parameters have to be known *a priori* to determine the translation and rotations needed for the reconstruction process. Unlike feature tracking, optical flow analysis does not need to be integrated over many frames.

For sparse reconstruction from feature tracking the factorization method developed by Tomasi-Kanade[73] is briefly introduced here. Say, N feature points tracked over F frames (Figure *2-7*) form a $2F \times N$ matrix, $W = [U \quad V]^T$, where U and V are respectively the mean centered x and y coordinates of the feature points.

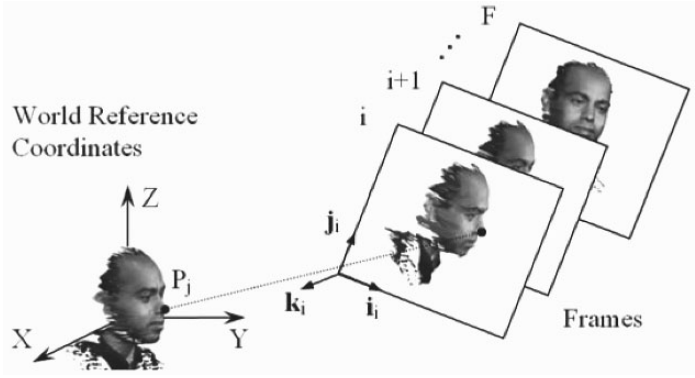


Figure 2-7. Illustration of 3D face reconstruction from motion. N number of feature points, P_j ($j = 1, \ldots, N$), are tracked over F number of frames.

Assuming a world coordinate system placed at the centroid of the object points, it was determined that W can be broken into R and S, where R is a $2F\times3$ matrix that represents the motion of the camera and S is a $3\times N$ matrix that represents the 3D coordinates of the feature points. The rank theorem stipulates that in absence of any noise, W is at most of rank three. In the presence of noise, W is highly rank deficient and can be decomposed with singular value decomposition (SVD) into a left singular $2F\times N$ matrix O_1, an $N\times N$ diagonal matrix H, and a right singular $N\times N$ matrix O_2. For an optimal R and S, the first three largest singular values from H and their corresponding parts, a $2F\times3$ matrix O_1' from O_1 and a $3\times N$ matrix O_2' from O_2 are used. W then becomes W' with the largest 3 singular values,

$$W' = R'S' = O_1'H'O_2' \tag{9}$$

where $R' = O_1'[H']^{1/2}$ and $S' = [H']^{1/2}O_2'$. R' and S' are not unique since there exist a 3×3 matrix Q such that $(R'Q)(Q^{-1}S') = (R'QQ^{-1}S') = R'S' = W'$. Q can be found by imposing the metric constraints:

$$i^TQQ^Ti = 1,\ j^TQQ^Tj = 1, \text{ and } i^TQQ^Tj = 0, \tag{10}$$

where i and j are unit vectors along the x and y axes of the image plane defined with respect to the world reference system. Initially, the image axes can be assumed to be aligned with the world reference system with $i = (1, 0, 0)^T$ and $j = (0, 1, 0)^T$. Under the orthographic projection assumption, this method is good for reconstruction of scenes viewed from long distances. The method can not reconstruct a scene having motion along the camera's optical axis. A later work by Poelman-Kanade[74] under para-perspective projections removed this limitation and dealt with images captured at relatively close distances.

A 3D face can be modeled more consistently by considering the statistics of the temporal quality of the video. Chowdhury et al. have reconstructed faces from video by estimating quality of reconstruction and using a generic model[75-77]. The generic model was applied as a *regularizer* to correct errors in local regions after recovering the reconstruction from video. The depth map was computed from two consecutive frames with respect to the camera reference frame. The camera focal length was known. Horizontal and vertical motions of feature points were estimated from two consecutive frames to solve for depths from a system of nonlinear equations. To obtain an optimum reconstruction, a Markov Chain Monte Carlo (MCMC)[78] sampling strategy was used. Instead of leading the reconstruction close to the generic model, exploiting the generic model after recovering the shape enabled the algorithm to retain person specific features.

A model-based bundle adjustment algorithm was developed by Shan et al.[79] to avoid the prior association of a 2D image to a 3D model. A deformable parametric surface automatically associated with tracked points on the face was used. The bundle adjustment algorithm constructs a cost function combining a large set of geometric parameters that includes 3D feature points and calibration parameters. Sparse bundle adjustment algorithm has been applied by Xin et al. to iteratively match 2D features, estimate camera parameters, and model the 3D face from video[80]. To further improve the model the bundle adjustment algorithm has been incorporated with silhouette information of the face image and the generic model[81]. Lee et al. has described 3D face modeling using generic face and only silhouette information extracted from images[82]. Reliable reconstruction of faces from silhouette requires images with optimal views. The method of finding optimal views is described in another paper from Lee et al.[83].

3D modeling from optical flow is based on image brightness constancy property, which assumes that the brightness of image points remains constant over time. This approximation is valid at points with high spatial gradient. Human faces have regions that have very small gradient or sometimes no visible gradient at all. Illumination in these regions varies from frame to frame and can mislead the optical flow estimation. Keeping the baseline small between frames can partially solve this problem, although a smaller baseline introduces higher sensitivity to noise. Then, a least square solution over many frames can reduce this sensitivity. Lhuillier and Quan have described a quasi-dense approach that can reconstruct a surface from relatively fewer images[84]. Photo consistency and silhouette information were included in the image correlation process based on best-first match propagation approach[85]. Both in feature tracking and optical flow estimation methods, parts of the faces might become occluded in the video. Occlusions can be dealt implicitly by treating them as high-noise measurements[82] or explicitly using 3D model based tracking[86].

Video provides the least constrained means to facial modeling in a range of applications such as surveillance[87]. However, it has been observed that modeling from video is quite ill-posed and severely affected by the presence of relatively small amounts of noise[75,76]. Pairs of sequential frames in video with wide inter-frame gaps can be treated as stereo pairs[49]

4.4 Shape from Space Carving

A considerably different approach that can potentially be used for modeling of human faces from multiple images is space carving[88,89]. In space carving framework, space is represented by an array of voxels, which must enclose the entire scene. The voxels that are not consistent with the

photographs are removed from the array one by one. The core part of space carving algorithm is the photo consistency checking i.e. if the shape can reproduce the input photographs when assigned. Kutulakoz and Seitz argued that in space carving no finite set of input images guarantees a unique 3D shape[89]. This is true if the camera is not calibrated. Camera can be calibrated using a pattern[22] or by tracking 2D features[23] to know camera positions in 3D world. Camera positions are used to check the photo consistency of the voxels. Kutulakos and Seitz have described a space carving algorithm which is dense and regularization bias free[89]. Broadhurst et al.[24] and Yezzi et al.[90] have developed a probabilistic framework to decide if a voxel is photo consistent or not. Beside calibration or probabilistic approaches, photometric information and silhouette of the objects have also been incorporated for space carving[91]. Silhouette of human face is almost always distinctively defined compared to the background hence can efficiently be used in this algorithm.

4.5 Shape from Shading (SFS)

Originating from the presence of multiple participating factors such as the direction of incident light, the surface geometry, the albedo (fraction of light reflected back by the surface), and the irradiance of the viewed image plane, shading only casts little information for one who wants to go backward and estimate these factors. Thus, shape from shading is an underdetermined problem where fewer parameters than necessary to recover the shape are known. To illustrate this, let us assume that at a surface point P, the radiance or amount of gray level intensity is $L(P)$. Assuming a *Lambertian model*[92] of surface reflectance, $L(P)$ is proportional to the cosine of the angle between the surface normal $\mathbf{n}$ and the direction of illuminant, $\mathbf{i}$:

$$L(P) = \rho \mathbf{i}^T \mathbf{n} \tag{11}$$

where ρ is the albedo. $L(P)$ is captured by the imaging system as surface irradiance at image point (x, y). Thus, the surface normal $\mathbf{n}$ is function of three unknowns and has to be determined from one linear equation even when we assume that the albedo ρ and the illuminant direction $\mathbf{i}$ are known. If the surface is expressed in terms of surface gradient instead of the normal, two unknown values $(\partial Z/\partial x, \partial Z/\partial y)$ have to be determined from given one irradiance value at (x, y); this is again an underdetermined situation. Therefore, additional constraints are imposed. For instance, the image acquisition system is assumed to be calibrated such that the irradiance of $L(.)$ at a scene point equals the irradiance at its corresponding image point. This

is known as the brightness constraint. Integrity is another constraint which stipulates that the surface maintains smoothness[93]. A comprehensive survey on shape-from-shading is available in a paper by Zhang et al.[94].

Assuming orthographic projection, if the surface is defined by $Z(x, y)$, with Z being the optical axis of the viewing camera (Figure *2-8*) then the unit normal vector at $Z(x, y)$ is given in terms of the surface gradient $(a = \partial Z/\partial x, b = \partial Z/\partial y)$ by:

$$\mathbf{n}(x, y) = \frac{1}{\sqrt{1 + a^2 + b^2}}\begin{bmatrix} -a & -b & 1 \end{bmatrix}^T \tag{12}$$

By enforcing the brightness constraint, shape-from-shading can be formulated as an optimization problem where one attempts to minimize the average error E between image brightness $I(x, y)$ at points (x, y) and $L(P)$ at corresponding scene points $P= Z(x, y)$ [95],

$$E = \iint \big(I(x, y) - L(P)\big) dxdy \tag{13}$$

Starting with an initial planar surface, Z is updated iteratively. To maintain smoothness of the surface, E is combined with a smoothness factor and an energy function is minimized to obtain the optimal shape. Several algorithms exist that describe how to estimate the light source direction $\mathbf{i}$ [96,97].

With only one image of a face available, shape-from-shading is seen as an alternative to morphing. Shape from shading techniques can be divided into four groups: minimization approaches,[93] propagation approaches[98], local approaches[99], and linear approaches[100]. Minimization approaches obtain the shape by minimizing an energy function constructed from E as seen in (10) and a smoothness constraint.

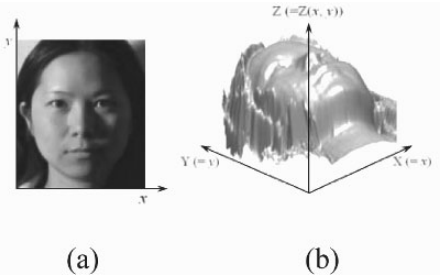


Figure 2-8. Illustration of facial reconstruction using shape from shading: (a) Face illuminated with a single light source, (b) corresponding reconstruction in world coordinate frame.

Propagation approaches propagate the shape information from a set of surface points (e.g. singular points) to the whole image. Local approaches derive shape based on assumptions on the surface type (i.e., locally spherical). Linear approaches compute a solution based on the linearization of the reflectance map[94].

The minimization algorithm needs to be initialized to estimate depth. A generic face model can be fed into the algorithm to gain speed in convergence. Atick et al.[95] reconstructed faces from an average model of 200 scanned heads. Principle component analysis was performed to estimate eigenvalues and eigenvectors for each model point $Z(x, y)$. Eigenvalues that minimized the energy function were selected iteratively to model a person specific 3D face.

At points of occlusion SFS algorithms fail to determine the surface parameters since the smoothness constraint is violated. A way to avoid this problem is to learn depths from images taken from multiple viewpoints. Fanany and Kumazawa incorporated a neural network based learning scheme in the minimization approach, propagation approach, and linear approach to model human faces[101]. In these approaches, the neural network first learned from shading at selected points of the depth maps of several calibrated multi-view images by extracting vertices and color parameters at these points. The camera extrinsic and intrinsic parameters were assumed known. Starting with a polyhedron, the network gradually transformed the polyhedron according to the learned depth maps. It was seen that the Tsai-Shah method[100] (a linear approach) and Bischel-Pentland method[98] (a propagation approach) converged to a more compact and realistic 3D face model compared to Zheng-Chellapa minimization method[97]. Theoretically, the Zheng-Chellapa method is more robust since it is a local approach that explicitly specifies the optimum selection of the albedo and the illuminant direction. However, if the albedo and the illuminant direction are wrongly selected, the result becomes inverted[101]. Kouzani et al. have described a 3D model based modeling of human faces from shading[102]. The face model was recovered by matching the 2D image of a face with projections of a generic 3D model viewed from different angles.

SFS techniques do not deliver an accurate model when the constraints can not be imposed strictly. The smoothness constraint would not properly model surfaces having discontinuity or sharp depth variations. Human faces have such discontinuities around facial landmarks which are important for machine based recognition. Youwei et al. applied additional constraint such as a parametric representation of generic 3D shape to retain discontinuities in the modeling[103]. Brightness and weak perspective camera model are two other constraints. Faithful modeling of faces requires that the faces remain close to the sensors. For close objects having a wide angle of view, the weak

perspective model assumption introduces higher inaccuracies in modeling. One image does not work well for all SFS algorithms since these algorithms are highly sensitive to light source location, albedo, and intensity bias.

5. RECONSTRUCTION QUALITY AND COMPARISONS

Experiments have been conducted to determine accuracies of 3D face scanning technologies. Boehnen and Flynn performed *absolute accuracy* (the accuracy of the scanner with respect to a physical reference subject) and *repeatability accuracy* (the degree of consistency between scans) tests on five different commercially available face scanners[37]. One of the scanners was stereo based system - 3DMDface™ (also known as Qlonerator™); two were laser scanners - Minolta Vivid® 910 and FastSCAN™ (also known as FastTRAK™); one was structured light pattern based; one was infrared laser scanner. The last two were mentioned as anonymous, FR1 and FR2, in the paper since they did not represent the manufacturer's current technologies when the experiments were conducted. In the experiments, first, faces of ten different individuals of varying races were scanned by Minolta Vivid® 910.

These scans served as ground truth. They were processed by 3D Systems Thermojet rapid prototyping machine (www.3dsystems.com) to build personalized synthetic mask faces from durable plastic polymer molds. The mask faces were test subjects that were again scanned by the scanners to test accuracies. The experiments suggest that Minolta Vivid® has the highest absolute accuracy (0.18*mm* mean global accuracy) followed by 3DMDface (0.24*mm*), FastSCAN (0.28*mm*), FR1 (1.21*mm*), and FR2 (1.78*mm*). The repeatability accuracy was observed in the same order - Minolta Vivid® (0.003*mm*), 3DMDFace (0.016*mm*), FastSCAN (N/A), FR1 (0.06*mm*), and FR2 (0.23*mm*). An interesting test in the experiment was checking the effect of pose variation on reconstruction accuracies. It was seen that, overall, the accuracies of the scans were slightly higher at frontal pose than at other poses.

Two types of resolutions are of particular interest for the assessment of the quality of 3D facial reconstruction: *depth resolution* and *spatial resolution*. Chang et al. have shown that depth resolutions of up to 0.5 mm on the whole face surface result in recognition rates within 94 to 95%[38]. The reconstructed face was a 640×480 array of 3D points scanned with Minolta Vivid® 900. As the resolution becomes coarser, the face surface becomes overly contoured, therefore, reducing the between class discrimination and making the recognition rate fall sharply. Spatial resolution of the reconstruction is defined as the number of voxels per unit square area of a

surface. Higher spatial resolution improves recognition. Highest recognition rates seemed to remain unchanged for up to 60% decrease in spatial resolution of the original 640×480 array of 3D points. After that, for each 1% decrement in resolution, recognition rates seemed to drop by 1%.

In laser scan reconstruction, depth and spatial resolutions are limited by the number of frames in the video scan and the thickness of the laser profile. In structured light projection the limiting factor is the resolution of lines or dots in the pattern. When the stripes (or dots) are too wide, depth discontinuities within them remain undetected[53] and the resultant reconstruction may seem to be built from a combination of stripes (or dots). Two other factors, movement of the object and baseline length, have significant impact on depth accuracy. Laser scanners require the participants to be cooperative i.e. remain still during the scan.

In stereo, a relatively wide baseline length ensures a higher accuracy although it also may introduce occlusions in the reconstruction and make stereo correspondences more difficult to establish due to brightness changes caused by non-Lambertian reflectance[104]. Inaccurate correspondences, on the other hand, lower the reconstruction accuracy. Thus, a compromise must be made between the baseline length and the robustness of the correspondence algorithm for an optimal reconstruction. The reconstruction may be dense or sparse. A dense model can easily be tessellated with accurate texture mapping from correspondences of points.

Reconstruction resolution and accuracy of other techniques, such as morphing, shape from motion, and shape-from-shading, are related to the working principles of their underlying algorithms. The objects are not required to remain still and texture, like in stereo, is readily available in all these techniques. The capability of morphing and shape-from-shading to reconstruct a 3D object from a single image is useful when the person is not available for registration in the 3D reconstruction process (called enrollment) and only one image of the face is available.

In security applications, such as video surveillance, a 3D face of a person can be modeled from video captured by a surveillance camera and added to a 3D face gallery for future recognition. When reconstructed with respect to the camera coordinate system, i.e., X and Y axes are aligned with the image plane and Z with the optical axis, depth becomes very sensitive to noise and confusion between rotation and translation of the camera may occur. This problem can be avoided by modeling the depth recovery process with respect to a world coordinate system[73].

Table *2-2* shows a comparison between different 3D modeling techniques with respect to their application in 3D face recognition. Both the gallery and the probe faces were reconstructed in 3D for recognition. Independently from the recognition principles and facial features used in different works, it

is seen from the table that laser scans, structured light, stereo vision, morphing, and shape from shading can be listed in the order of decreasing recognition performances. Note that however, recognition performances can be poor due to a particular recognition method used in an experiment. To the best of our knowledge, no 3D face recognition work has been done where the face was modeled from video. In the work of Smith and Hancock[25] the face database was built from laser scans and only the probe faces from shape from shading. More on 3D face recognition can be found in a survey by Bowyer et al.[105]. An earlier survey by Zhao et al. discusses 2D, 3D face recognition and the challenges and psychophysical issues with face recognition[106].

Table 2-2. Comparison of different 3D face reconstruction techniques with respect to the performance of 3D facial recognition.

Recognition Principle (3D-3D)	Technique	Paper	Recon. Time (sec/face)	Facial features	Database Size	Recognition Rate
Nearest Neighbor (NN), SVM	Laser Range Scan	Srivastava et al.[107]	2.5	Optimal Subspace of Range Data	67	NN:99% SVM:94.03% within rank 1
Volume Difference	Structured Light Projection	Xu et al.[108]	0.5	Surface Mesh	30	98% within rank 5
Distance Map	Stereo Vision	Medioni and Waupotitsch[51]	9	Surface Mesh	700	97.5% within rank 1
Mahalanobis like Distances	Morphing	Blanz and Vetter[109]	150	Shape + Texture	68	95.9% within rank 1
Bhattacharya Distance	Shape from Shading	Smith and Hancock[25]	N/A	Gaussian Curvature	12	50% within rank 5

6. USE OF 3D FACE MODELS IN FACE RECOGNITION

In 3D based face recognition approaches, depth information introduces more information in the recognition process hence faces that could otherwise be similar in 2D become more discriminated. Facial pose recovery, which is an essential step in 2D to 3D face recognition, has acquired more accuracy applying 3D facial model[110]. Generic 3D facial model has been used for pose and illumination compensation of 2D images[1,111,112]. While in facial pose recovery three angles of rotation are estimated, in pose compensation, a 2D face posed at some angles is projected at frontal pose. The final 2D probe face can be obtained from projection on a generic 3D face which, however, is different from the actual 3D of the probe face. To alleviate such problem,

Blanz, Romdhani, and Vetter have used morphing to transform the generic 3D face into person specific 3D face[4,70].

Illumination compensation synthesizes standard 2D prototype images from a given 2D image affected by illumination. For the compensation, surface normal and the light source (i.e. direction of light) need to be known. Knowing surface normals it is possible to estimate the light source. Surface normals can be approximately known from a generic 3D model. This approach has been used by Zhao and Chellappa[1] and Malassiotis and Strintzis[112]. Zhao and Chellappa used the symmetry assumption to avoid the estimation of albedo. The symmetry assumption requires that the facial pose be determined *a priori*. The pose was provided manually in their work.

There are algorithms that have used morphing to estimate pose from a single image however[113]. Morphing, as described earlier, warps a generic 3D face into a person specific 3D face addressing all three problems – 3D face modeling, pose recovery, and illumination compensation[4,70,114].

In an evaluation of 2D and 3D face recognition, Chang et al. have found that 2D and 3D have similar recognition performances when considered individually[115]. This experiment, however, was only limited to frontal poses. When pose variation is allowed, 3D model based face recognition shows better performance than 2D based recognition as observed by Blanz et al.[4]. While 3D to 3D face recognition (Table *2-2*) seems to have high recognition rates, as an online recognition solution 2D to 3D face recognition is more promising due to their practicality. In many applications, mostly where face recognition is used as a security measure, online 3D face reconstruction is impractical due to lengthy processing time or impossible due to improper lighting conditions. In 2D to 3D recognition approaches[13,14,109], however, a 3D database of the individuals can be maintained, pose of person captured in 2D can be estimated, and then the 2D image of the probe face can be matched with the projection on the 3D face database. Figure *2-9* illustrates the block diagram of such 2D to 3D face recognition approach.

With the advances in 3D face recognition research, a number of 3D face databases have become available which we find worth mentioning. Table *2-3* lists several 3D face databases that have been made public to the computer vision community. FRGC 2.0[116] is the biggest database publicly available. It is in point cloud format with spikes and holes in them and contains coordinates of manually labeled 4 facial landmarks: two eye corners, nose, and chin.

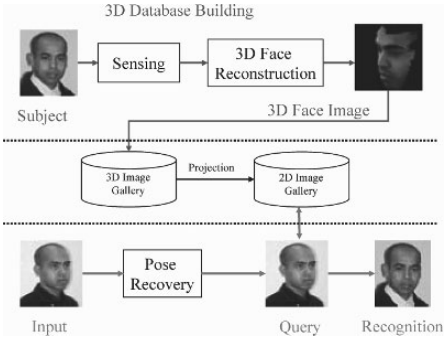


Figure 2-9. 3D face reconstruction and pose recovery for 2D-3D face recognition under uncontrolled environment.

Table 2-3. Publicly available 3D face databases.

Database name	Reconstruction method	Size (# of images)	# of people	3d data representation	Subject variations
FRGC 2.0 (Validation Dataset)[116]	Laser Scan (Minolta Vivid 900/910 series sensor)	4007	466	Point Clouds (both range and texture)	Expressions
GavabDB[117]	Laser Scan (Minolta VI-700)	427	61	Mesh (texture omitted)	6 poses, 3 expressions
XM2VTS[118,119]	Stereo	295	295	VRML (both range and texture)	Talking face with some head rotation
3D_RMA[6]	Structured Light	360	120	Point Clouds (only range)	3 poses, no expressions
3D Face DB of York Univ.*	Laser Scan (Cyber ware 3030RGB/PS)	970	97	Point Clouds (both range and texture)	10 poses, 2 expressions
3D Face DB at Notre Dame**	Laser Scan (Minolta Vivid 900)	1906	277	Point Clouds (both range and texture)	Frontal pose and neutral expr. taken at different sessions

*Department of Computer Science, The University of York (http://www.users.cs.york.ac.uk/~tomh/3DFaceDatabase.html)
**University of Notre Dame Biometrics Database Distribution (http://www.nd.edu/%7Ecvrl/UNDBiometricsDatabase.html)

7. FUTURE TRENDS

The future of research in 3D face modeling is governed by the need for face recognition as a means to accurate, inexpensive, flexible, and unobtrusive biometrics. Face recognition is rapidly becoming a dominant

biometric alongside fingerprints, iris, and voice recognition[120]. Based on experiments on 3D face recognition conducted by FRVT (Face Recognition Vendor Test) 2002[121,122], the organizers of FRVT 2006 under the Face Recognition Grand Challenge (FRGC) project decided to include larger facial database (50000 facial images.) that included 3D scans along with high resolution still imagery. The 3D scans are taken with laser range scanners Minolta Vivid® 900/910. The goal of FRGC is to reduce the error rate in face recognition systems by an order of magnitude over FRVT 2002 results. FRGC will answer some important questions such as if recognition from 3D imagery is more effective than recognition from high resolution (5 to 6 mega pixels) 2D frontal pose imagery or multi-sample still facial imagery[116].

In fact, in a recent FRGC workshop, it was shown that at *frontal pose* the 2D face recognition algorithm, hierarchical graph matching (HGM), based on a universal object recognition approach[123], is more accurate than 3D face recognition. However, it was also shown that fusion of 3D with 2D recognition improved the recognition rate significantly[124]. The goal of 3D face recognition is to make the recognition invariant to pose and illumination - the two factors that severely affect 2D face recognition.

Lately, experiments on 3D face recognition have been performed combining 3D geometry information with 2D texture[13,14,38,50,115,125]. In 2D face recognition both visual and IR imagery have been used to achieve higher recognition rate[126-128]. In a recent experiment, IR information has been used with 3D data. Encouraging recognition rate (98.22% with rank one on 88 subjects) has been observed in the presence of facial expression. The IR information was used to test aliveness of the face (to avoid scanning of fake face such as mannequin by intruders), to avoid hair and other accessories on the face, and to extract facial vasculature in the forehead region to provide more discriminative information on the subject[129]. Multimodal face recognition approaches usually exhibit a superior performance to using each modality alone. However, small sample sizes (466 and 295 individuals as far as known to be the big sizes so far[118,119]) and inability to handle facial expressions (with the exceptions of the works of Wang et al.[13] and Bronstein et al.[130]) contribute to currently inconclusive results[129].

Shape and texture both channels have been applied together to achieve better recognition rate[138]. But, when visible light is used, laser scanners and structured light projectors need to capture shape and texture channels at two different times leaving a possibility of poor registration between shape and texture due to movement of the face. Near infrared light (NIR) is invisible to camera that works in the visible range, which allows capturing both patterns and texture at the same time using two sensors. Then, the sensors have to be located at different view points to work at the same time; the 3D model has

to be transformed (translated, rotated, etc.) such that viewpoints of the pattern sensor matches with viewpoint of the texture sensor - a work not seen in the literature or industry. NIR is more tolerant to ambient lights and artificial facial features such as make up. NIR has been used in structured light based face scanner A4Vision (www.a4vision.com). A4Vision simultaneously capture a color texture image with the 3D shape with texture sensor placed on top of the pattern sensor. It is not mentioned if they have fused the two viewpoints into one.

Speed has become a major concern in 3D face modeling hence in 3D based face recognition. According to A4Vision on their 3D face recognition system, optimal ergonomics allow for instant recognition within less than 1/2 of second from the time a subject appears within the view field. Although some access-control applications do not need prompt responses, current face modeling or recognition techniques have not reached online or real time performances demanded by many other applications. 3D face modeling takes time in the order of seconds (see Table *2-1*). On the other hand, in FRVT 2002 number of 2D faces compared was 15,555/sec. With 3D involved with its current modeling speed, this rate is likely to fall sharply. Therefore, fast modeling of faces currently seems to be forthcoming demand.

In addition to biometrics, 3D face modeling is becoming popular in medical applications for diagnosis and surgery. To visualize temperature changes on skin thermal image has been mapped on 3D face to develop 3D Thermography Imaging for inflammation diagnosis[131,132]. As a pre-operative surgical planning 3D faces are used to extract 3D surface measurements in orthognathic surgery for detection or correction of facial deformity[133,134].

8. CONCLUSION

3D face modeling is the very first step towards 3D assisted face recognition and a major requirement for pose invariant face recognition. It also provides the possibility to correct for illumination problems of a 2D face using its pose and depth information from person-specific or a generic 3D human face[1]. The accuracies in modeling, spatial resolution, sensitivity to noise and illumination, flexibility in capturing data, texture deliverability with 3D data, data format, and modeling speed are considered important factors in determining the merits of a 3D face modeling technique.

Modeling from laser scans is highly accurate but the subject has to be cooperative remaining still to avoid global distortions in models. In experiments on head motion during PET (Positron Emission Tomography) imaging, it has been observed that the head can move about 5 millimeters

and rotate about 3 degrees during a 30 second scan even when the subject is cooperative[135,136]. Thus, although global distortions seem difficult to avoid entirely they can be minimized if the scan time can somehow be reduced, for instance, by engaging multiple laser sources in parallel. Structured light reconstruction avoids global distortions but the reconstruction is less dense. Stereo vision techniques avoid global distortions since the stereo images are captured simultaneously in one shot.

In the presence of a sufficient gradient, stereo techniques can perform 3D reconstruction at higher depth resolutions than laser scans or structured light. Depth at sub-pixel locations can be computed by interpolation from depths of neighboring points. Such processing does not introduce new information in the model but can still improve the recognition rate[38]. Sub-pixel accuracy can be achieved by locating image points (x, y) between pixels. In commercial laser scanners an accuracy of about 0.25 mm is a common target and a half pixel may correspond to up to $0.5mm$[44]. With laser scans and structured light sub pixels can be found at the peak intensity locations determined across a profile using a Gaussian weighted mask. In stereo, interpolations from stereo matching coefficients measured at neighboring pixels can be used to estimate the location of the best correspondence location. Alternatively, intensities can be interpolated and used in similarity estimation so that the estimate remains insensitive to image sampling.[137]

Under controlled lighting conditions, laser profiles and structured light patterns projected on facial skin are detectable and do not interfere with the accuracy of reconstruction. It is claimed that a wide variation in illumination does not affect reconstruction from stereo vision either[51]. When using low and medium illuminance stereo reconstruction during enrollment, a statistically significant difference between verification attempts made at low, medium, and high illuminance is observed. However, for high illuminance reconstruction during enrollment, there is no statistically significant difference between verification attempts made at low, medium, or high illuminance[31]. Morphing from a single image is sensitive to illumination and requires illumination modeling[4]. In contrast, illumination is an essential entity for modeling objects with shape from shading.

For building 3D face databases usual practice is to arrange enrollment session to assure modeling of high quality 3D faces from user cooperative setups in laboratory environment. Stereo vision has fewer constraints on their subject cooperation than laser scan and structured light. Morphing and shape from shading are useful to modeling face from a single image[109]. Depth accuracy may not be high but the modeled face can still be stored in a 3D face database and used for pose invariant recognition. Therefore, a single modeling technique is not sufficient to build a complete 3D face database. Oftentimes, subjects are not available for enrollment. In biometrics based

security applications, the only images available for reconstruction could be a scan from a driving license, an image sequence from a video surveillance camera, or images captured in any other uncontrolled environment. Shape from motion can reconstruct faces from video and the reconstructed faces can be stored in 3D face databases.

As evidenced from Table *2-2*, recognition rates using 3D faces acquired from laser scans, structured light projection, and stereo vision are higher than those from morphing and shape from shading. It is evident that when texture is added, 3D face recognition shows better performances[138]. Laser scan and structured light projection based commercial systems deliver both shape and texture. All other modeling techniques rely on texure images for depth recovery hence also deliver textured models. Faithful facial modeling from video captured in an uncontrolled security environment is often impossible since the image quality in such videos is poor and the resolution of faces is small (often 28×28). However, for building 3D faces in laboratory (i.e. controlled) environment, SFM could be quite flexible and inexpensive.

Experiments suggest that laser scanners are able to build a 3D face with accuracy higher than the other modeling techniques[37]. Establishing feature correspondences in uncontrolled environment reliably and accurately is difficult hence the reason for active techniques gaining popularity more than the passive techniques. However, active techniques alone may not be able to build a complete 3D face database that, in many biometrics applications, may require including every individual of a particular group of people. Therefore, passive modeling is expected to continue with more or less attention alongside active modeling techniques.

ACKNOWLEDGEMENT

This work was supported by ONR, URPR, and NSF.

REFERENCES

1. W. Zhao and R. Chellappa. 3D Model Enhanced Face Recognition. Proc. of Int'l Conf. on Image Processing (ICIP), 3: 50–53, 2000.
2. G. Gordon. Face Recognition Based on Depth Maps and Surface Curvature. SPIE Proc. on Geometric Methods in Computer Vision, 1570: 234–247, 1991.
3. C. Chua, F. Han, and Y. Ho. 3D Human Face Recognition Using Point Signature. Proc. of the 4th IEEE Int'l Conf. on Automatic Face and Gesture Recognition, pp. 233–238, 2000.

4. V. Blanz, S. Romdhani, and T. Vetter. Face Identification across Different Poses and Illuminations with a 3D Morphable Model. Proc. of the 5th IEEE Conf. on Automatic Face and Gesture Recognition, pp. 202–207, 2002.
5. J. Geng, P. Zhuang, P. May, S. Yi, and D. Tunnell. 3D FaceCam: a fast and accurate 3D facial imaging device for biometrics applications. Proc. of SPIE, 5404: 316–327, 2004.
6. C. Beumier and M. Acheroy. Automatic 3D face authentication. Image and Vision Computing, 18(4): 315–321, 2000.
7. E. Garcia and J. Dugelay. Low cost 3D face acquisition and modeling. Proc. of Int'l Conf. on Information Technology: Coding and Computing, pp. 657–661, 2001.
8. R. Jarvis. A Perspective on Range Finding Techniques for Computer Vision. IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI), 5(2): 122–139, 1983.
9. P. Besl. Active, Optical Range Imaging Sensors. Machine Vision and Applications, 1(2): 127–152, 1988.
10. C. Zhang and F. Cohen. 3–D face structure extraction and recognition from images using 3–D morphing and distance mapping. IEEE Trans. on Image Processing, 11(11): 1249–1259, 2002.
11. A. Eriksson and D. Weber. Towards 3–Dimensional Face Recognition. Proc. of the 5th IEEE AFRICON: Conference in Africa, 1: 401–406, 1999.
12. T. Jebara, K. Russel, and A. Pentland. Mixture of Eigenfeatures for Real–Time Structure from Texture. Proc. of the 6th Int'l Conf. on Computer Vision, pp. 128–135, 1998.
13. Y. Wang, C. Chua, and Y. Ho. Facial feature detection and face recognition from 2D and 3D images. Pattern Recognition Letters, 23(10): 1191–1202, 2002.
14. Y. Wang, C. Chua, Y. Ho, and Y. Ren. Integrated 2D and 3D Images for Face Recognition. Proc. of the 11th Int'l Conf. on Image Analysis and Processing, pp. 48–53, 2001.
15. R. Hsu and A. Jain. Semantic Face Matching. Proc. of IEEE Int'l Conf. on Multimedia and Expo., 2: 145–148, 2002.
16. J. Huang, V. Blanz, and B. Heisele. Face recognition using component–based SVM classification and Morphable models. Lecture Notes in Computer Science (LNCS), Springer–Verlag, 2388: 334–341, 2002.
17. J. Yan and H. Zhang. Synthesized virtual view–based eigenspace for face recognition. Proc. of the 5th IEEE Workshop on Applications of Computer Vision, pp. 85–90, 2000.
18. B. Hwang and S. Lee. Reconstruction of partially damaged face images based on a morphable face model. *IEEE Trans. on Pattern Analysis and Machine Intelligence* (*PAMI*), 25(3): 365–372, 2003.
19. B. Hwang and S. Lee. Face reconstruction with a morphable face model. *Proc. of the 16th Int'l Conf. on Pattern Recognition*, 2: 366–369, 2002.
20. V. Kruger, R. Gross, and S. Baker. Appearance–based 3D Face Recognition from Video. *Proc. of the 24th German Symposium on Pattern Recognition*, pp. 566–574, 2002.
21. S. Zhou, V. Krueger, and R. Chellappa. Face Recognition from Video: A CONDENSATION Approach. *Proc. of the 5th IEEE Int'l Conf. on Automatic Face and Gesture Recognition*, pp. 221–228, 2002.
22. A. Montenegro, P. Carvalho, L. Velho, M. Gattass. Space carving with a hand–held camera. *Proc. of the 17th Brazilian Symposium on Computer Graphics and Image Processing*, pp. 396–403, 2004.
23. M. Sainz, N. Bagherzadeh, and A. Susin. Carving 3D Models from Uncalibrated Views. *Proc. of the 5th IASTED Int'l Conf. Computer Graphics and Imaging* (*CGIM*), *ACTA Press*, pp. 144–149, 2002.

24. A. Broadhurst, T. Drummond, and R. Cipolla. A probabilistic framework for space carving. *Proc. of the 8th IEEE Int'l Conf. on Computer Vision* (*ICCV*), 1: 388–393, 2001.
25. W. Smith and E. Hancock. Face Recognition using Shape–from–Shading. *Proc. of the 13th British Machine Vision Conference*, 2: 597–606, 2002.
26. D. Gelli and D. Vitulano. Surface recovery by self shading projection. *Signal Processing*, 84(3): 467–473, 2004.
27. J. Villa and J. Hurtado–Ramos. Surface shape estimation from photometric images. *Optics and Lasers in Engineering*, 42(4): 461–468, 2004.
28. T. Sohmura, M. Nagao, M. Sakai, K. Wakabayashi, T. Kojima, S. Kinuta, T. Nakamura, and J. Takahashi. High–resolution 3–D shape integration of dentition and face measured by new laser scanner. *IEEE Trans. on Medical Imaging*, 23(5): 633–638, 2004.
29. P. Vuylsteke and A. Oosterlinck. Range image acquisition with a single binary-encoded light pattern. *IEEE Trans. on Pattern Analysis and Machine Intelligence* (*PAMI*), 12(2): 148–164, 1990.
30. Q. Chen and G. Medioni. Building 3–D Human Face Models from Two Photographs. *The Journal of VLSI Signal Processing, Kluwer Academic Publishers*, 27(1–2): 127–140, 2001.
31. E. Kukula, S. Elliot, R. Waupotitsch, B. Pesenti. Effects of Illumination Changes on the Performance of Geometrix Face Vision 3D FRS. *Proc. of the 38th IEEE Int'l Conf. on Security Technology*, pp. 331–337, 2004.
32. X. Ju, T. Boyling, P. Siebert. A High Resolution Stereo Imaging System. *Proc. of 3D Modeling*, Paris, France, 2003.
33. F. Blais. Review of 20 Years of Range Sensor Development. *Journal of Electronic Imaging*, 13(1): 231–240, Jan 2004.
34. M. Jackowski, M. Satter, and A. Goshtasby. Approximating Digital 3D Shapes by Rational Gaussian Surfaces. *IEEE Trans. on Visualization and Computer Graphics*, 9(1): 56–69, 2003.
35. J. Davis and X. Chen. A laser range scanner designed for minimum calibration complexity. *Proc. of the 3rd Int'l Conf. on 3D Digital Imaging and Modeling*, pp. 91–98, 2001.
36. J. Forest, J. Salvi, E. Cabruja, and C. Pous. Laser stripe peak detector for 3D scanners. A FIR filter approach. *Proc. of the 17th Int'l Conf. on Pattern Recognition* (*ICPR*), 3: 646–649, 2004.
37. C. Boehnen and P. Flynn. Accuracy of 3D scanning technologies in a face scanning context. *Proc. of 5th Int'l Conf. on 3D Digital Imaging and Modeling*, 2005.
38. K. Chang, K. Bowyer, and P. Flynn. Face Recognition Using 2D and 3D Facial Data. *Proc. of ACM Workshop on Multimodal User Authentication*, pp. 25–32, 2003.
39. Y. Xu, C. Xu, Y. Tian, S. Ma, and M. Luo. 3D face image acquisition and reconstruction system. *Proc. of Instrumentation and Measurement Technology Conf.* (*IMTC*), 2: 1121–1126, 1998.
40. H. Longuet–Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293(10): 133–135, 1981.
41. M. Polleyfeys, R. Koch, and L. Van Gool. Self-calibration and metric reconstruction in spite of varying and unknown internal camera parameters. *Proc. of the IEEE Int'l Conf. on Computer Vision* (*ICCV*), pp. 90–95, Bombay, India, 1998.
42. M. Polleyfeys, L. Van Gool, and M. Proesmans. Euclidean 3D reconstruction from image sequences with variable focal lengths. *Proc. of European Conf. on Computer Vision* (ECCV), pp. 31–42, 1996.

43. M. Polleyfeys, R. Koch, L. Van Gool. Self–Calibration and Metric Reconstruction in spite of Varying and Unknown Internal Camera Parameters. *Int'l Journal of Computer Vision*, 32(1): 7–25, 1999.
44. E. Trucco and A. Verri. Introductory Techniques for 3D Computer Vision. *Prentice–Hall*, New Jersey, USA, 1998.
45. R. Lengagne, P. Fua, and O. Monga. 3D face modeling from stereo and differential constraints. *Proc. of the 3rd IEEE Int'l Conf. on Automatic Face and Gesture Recognition*, pp. 148–153, 1998.
46. J. Baker, V. Chandran, and S. Sridharan. Techniques for improving stereo depth maps of faces. *Proc. of IEEE Int'l Conf. on Image Processing*, pp. 3279–3282, 2004.
47. R. Enciso, J. Li, D. Fidaleo, T. Kim, J. Noh, and U. Neumann. Synthesis of 3D Faces. *Proc. of the 1st Int'l Workshop on Digital and Computational Video*, pp. 8–15, 1999.
48. N. D'Apuzzo and E. Zürich. Modeling human faces with multi–image photogrammetry. *Proc. of SPIE Three–Dimensional Image Capture and Applications*, 4661: 191–197, 2002.
49. G. Medioni and B. Pesenti. Generation of a 3–D face model from one camera. *Proc. of the 16th Int'l Conf. on Pattern Recognition* (*ICPR*), 3: 667–671, 2002.
50. S. Lao, M. Kawade, Y. Sumi, and F. Tomita. 3D Template Matching for Pose Invariant Face Recognition Using 3D Facial Model Built with Isoluminance Line Based Stereo Vision. *Proc. of Int'l Conf. on Pattern Recognition*, 2: 911–916, 2000.
51. G. Medioni and R. Waupotitsch. Face Recognition and Modeling in 3D. *IEEE Int'l Workshop On Analysis and Modeling of Faces and Gestures* (*AMFG*), pp. 232–233, 2003.
52. J. Kittler, A. Hilton, M. Hamouz, and J. Illingworth. 3D Assisted Face Recognition: A Survey of 3D Imaging, Modelling, and Recognition Approaches, *Proc. of the IEEE Computer Society Conf. On Computer Vision and Pattern Recognition* (*CVPR*), 3: 114–120, Jun 2005.
53. J. Batlle, E. Mouaddib, and J. Salvi. Recent progress in coded structured light as a technique to solve the correspondence problem: a survey. *Pattern Recognition*, 31(7): 963–982, 1998.
54. S. Huq, B. Abidi, A. Goshtasby, and M. Abidi. Stereo Matching with Energy Minimizing Snake Grid for 3D Face Modeling. *Proc. of SPIE, Defense and Security Symposium*, 5404: 339–350, 2004.
55. Y. Boykov, O. Veksler, and R. Zabih. Fast Approximate Energy Minimization via Graph Cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (*PAMI*), 23(11): 1222–1239, 2001.
56. V. Kolmogorov and R. Zabih. Computing Visual Correspondence with Occlusions using Graph Cuts. *Int'l Conf. on Computer Vision*, pp. 508–515, 2001.
57. S. Birchfield and C. Tomasi. Depth Discontinuities by Pixel–to–Pixel Stereo. *Proc. of 6th Int'l Conf. on Computer Vision*, pp. 1073–1080, 1998.
58. G. Dainese, M. Marcon, A. Sarti, and S. Tubaro. Accurate Depth-map estimation for 3D face modeling. *The 2005 European Signal Processing Conference* (*EUSIPCO*), Antalya, Turkey, Sep 2005.
59. Z. Zhang, R. Deriche, O. Faugeras, and Q. Luong. A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry. *Artificial Intelligence*, 78(1–2): 87–119, 1995.
60. R. Hartley and A. Zisserman. Multiple View Geometry in Computer Vision. Cambridge University Press, ISBN 0–521–62304–9, 2000.

61. O. Faugeras, Q. Luong, and T. Papadopoulo. The Geometry of Multiple Images. MIT Press, ISBN 0262062208, 2001.
62. J. Yan, W. Gao, B. Yin, and Y. Song. Deformable model–based generation of realistic 3–D specific human face. *Proc. of the 4th Int'l Conf. on Signal Processing* (*ICSP*), 2: 857–860, 1998.
63. A. Ansari and M. Mottaleb. 3D face modeling using two orthogonal views and a generic face model. *Proc. of Int'l Conf. on Multimedia and Expo* (*ICME*), 3: 289–292, 2003.
64. B. Phong. Illumination for computer generated pictures. *Communications of the ACM*, 18(6): 311–317, 1975.
65. Y. Hu, D. Jiang, S. Yan, L. Zhang, and H. Zhang. Automatic 3D Reconstruction for Face Recognition. *Proc. of 6th IEEE Int'l Conf. on Automatic Face and Gesture Recognition* (*FG*), Seoul, South Korea, 2004.
66. D. Jiang, Y. Hu, S. Yan, L. Zhang, H. Zhang, and W. Gao. Efficient 3D reconstruction for face recognition. *Pattern Recognition*, 38(6): 787–798, 2005.
67. N. Sarris, N. Grammalidis, and M. Strintzis. Building Three–Dimensional Head Models. *Graphical Models*, 63(5): 333–368, 2001.
68. S. Ho and H. Huang. Facial modeling from an uncalibrated face image using a coarse–to–fine genetic algorithm. *Pattern Recognition*, 34(5): 1015–1031, 2001.
69. Y. Xing, C. Guo, and Z. Tan. Automatic 3D facial model reconstruction from single front–view image. *Proc. of 4th Int'l Conf. On Virtual Reality and Its Applications in Industry, SPIE*, 5444: 149–152, 2004.
70. V. Blanz, P. Grother, J. Phillips, and T. Vetter. Face recognition based on frontal views generated from non–frontal images. *IEEE Computer Society Conf. on Computer Vision and Pattern Recognition* (*CVPR*), pp. 446–453, 2005.
71. Q. Luong and O. Faugeras. Self–Calibration of a Moving Camera from Point Correspondences and Fundamental Matrices. *Int'l Journal of Computer Vision*, 22(3): 261–289, 1997.
72. L. Quan. Invariants of six points and projective reconstruction from three uncalibrated images. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (*PAMI*), 17(1): 34–46, 1995.
73. C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: A factorization method. *Int'l Journal of Computer Vision*, 9(2): 137–154, 1992.
74. C. Poelman and T. Kanade. A Paraperspective Factorization Method for Shape and Motion Recovery. *Proc. of 3rd European Conf. on Computer Vision* (*ECCV*), 2: 97–108, 1994.
75. A. Chowdhury, R. Chellappa, S. Krishnamurthy, and T. Vo. 3D face reconstruction from video using a generic model. *Proc. of IEEE Conf. on Multimedia and Expo*, 1: 449–452, 2002.
76. A. Chowdhury and R. Chellappa. Face Reconstruction from Monocular Video using Uncertainty Analysis and a Generic Model. *Computer Vision and Image Understanding*, 91: 188–213, 2003.
77. A. Chowdhury, R. Chellappa, and T. Keaton. Wide baseline image registration with application to 3–D face modeling. *IEEE Trans. on Multimedia*, 6(3): 423–434, 2004.
78. W. Gilks, S. Richardson, and D. Spiegelhalter. Markov Chain Monte Carlo in Practice, Chapman and Hall, London, 1996.
79. Y. Shan, Z. Liu, and Z. Zhang. Model–based bundle adjustment with application to face modeling. *Proc. of IEEE Int'l Conf. on Computer Vision* (*ICCV*), 2: 644–651, 2001.
80. L. Xin, Q. Wang, J. Tao, X. Tang, T. Tan, and H. Shum. Automatic 3D Face Modeling from Video. *Proc. of IEEE Int'l Conf. on Computer Vision* (*ICCV*), 2: 1193 – 1199, 2005.

81. C. Cheng and S. Lai. An integrated approach to 3D face model reconstruction from video. *Proc. of IEEE Int'l Conf. on Computer Vision* (*ICCV*) *Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real–Time Systems*, pp. 16–22, 2001.
82. B. Moghaddam, J. Lee, H. Pfister, and R. Machiraju. Silhouette–based 3d face shape recovery. *Graphics Interface*, 2(9): 61–68, 2003.
83. J. Lee, B. Moghaddam, H. Pfister, R. Machiraju. Finding Optimal Views for 3D Face Shape Modeling. *Proc. of the Int'l Conf. on Automatic Face and Gesture Recognition* (*FGR*), pp. 31–36, 2004.
84. M. Lhuillier and L. Quan. A Quasi–Dense Approach to Surface Reconstruction from Uncalibrated Images. *IEEE Trans. on Pattern Analysis and Machine Intelligence* (*PAMI*), 27(3): 418–433, 2005.
85. M. Lhuillier and L. Quan. Match Propagation for Image–Based Modeling and Rendering. *IEEE Trans. on Pattern Analysis and Machine Intelligence* (*PAMI*), 24(8): 1140–1146, 2002.
86. J. Strom, T. Jebara, S. Basu, and A. Pentland. Real time tracking and modeling of faces: an EKF–based analysis by synthesis approach. *Proc. of IEEE Int'l Workshop on Modeling People*, pp. 55–61, 1999.
87. R. Gross, I. Matthews, and S. Baker. Eigen Light–Fields and Face Recognition Across Pose. *Proc. of the 5th IEEE Int'l Conf. on Automatic Face and Gesture Recognition*, 2002.
88. G. Slabaugh, B. Culbertson, T. Malzbender, and R. Shafer. A survey of methods for volumetric scene reconstruction from photographs. *Proc. of Int'l Workshop on Volume Graphics*, 2001.
89. K. Kutulakos and S. Seitz. A theory of shape by space carving. *Proc. of the 7th IEEE Int'l Conf. on Computer Vision* (*ICCV*), 1: 307–314, 1999.
90. A. Yezzi, G. Slabaugh, A. Broadhurst, R. Cipolla, and R. Schafer. A surface evolution approach of probabilistic space carving. *Proc. of the 1st Int'l Symposium on 3D Data Processing, Visualization, and Transmission*, pp. 618–621, 2002.
91. A. Montenegro, P. Carvalho, M. Gattass, and L. Velho. Adaptive space carving. *Proc. of the 2nd Int'l Symposium on 3D Data Processing, Visualization and Transmission*, (*3DPVT*), pp. 199–206, 2004.
92. J. Lambert. Photometria de Mensura et Gratibus Luminis, Colorum et Umbrae, Eberhard Klett, Augsburg, Germany, 1760. Translation in W. Engleman (1982), Lambert's Photometrie, Liepzig.
93. K. Ikeuchi and B. Horn. Numerical shape from shading and occluding boundaries. *Artificial Intelligence*, 17(3): 141–184, 1981.
94. R. Zhang, P. Tsai, J. Cryer, and M. Shah. Shape from Shading: A Survey. *IEEE Trans. on Pattern Analysis and Machine Intelligence* (*PAMI*), 21(8): 690–706, 1999.
95. J. Atick, P. Griffin, and A. Redlich. Statistical approach to shape from shading: Reconstruction of 3–dimensional face surface from single 2–dimensional images. *Neural Computation*, 8(6): 1321–1340, 1996.
96. A. Pentland. Finding the illumination direction. *Journal of Optical Society of America*, 72: 448–455, 1982.
97. Q. Zheng and R. Chellappa. Estimation of Illuminant Direction, Albedo, and Shape from Shading. *IEEE Trans. on Pattern Analysis and Machine Intelligence* (*PAMI*), 13(7): 680–703, 1991.
98. M. Bischel and A. Pentland. A Simple Algorithm for Shape from Shading. *Proc. Of IEEE Conf. on Computer Vision and Pattern Recognition* (*CVPR*), pp. 459–465, 1992.
99. A. Pentland. Local Shading Analysis. *IEEE Trans. on Pattern Analysis and Machine Intelligence* (*PAMI*), 6(2): 170–187, 1984.

100. P. Tsai and M. Shah. Shape from Shading using Linear Approximation. *Image and Vision Computing Journal*, 12(8): 487–488, 1994.
101. M. Fanany and I. Kumazawa. Analysis of shape from shading algorithms for fast and realistic 3D face reconstruction. *Proc. of Asia–Pacific Conf. on Circuits and Systems*, 2: 181–185, 2002.
102. A. Kouzani and F. He, and K. Sammut. Example–based shape from shading: 3D heads form 2D face images. *IEEE Int'l Conf. on Systems, Man, and Cybernetics*, 4: 4092–4097, 1998.
103. Y. Youwei, Y. Lamei, and M. Deris. SFS based neural algorithm for robust 3D face shape recovery. *Proc. of the 9th Int'l Conf. on Neural Information Processing* (*ICONIP*), 2: 665–669, 2002.
104. S. Sakamoto, I. Cox, J. Tajima. A multiple–baseline stereo for precise human face acquisition. *Pattern Recognition Letters*, 18(9): 923–931, 1997.
105. K. Bowyer, K. Chang, and P. Flynn. A Survey of Approaches To Three-Dimensional Face Recognition. *Int'l Conf. on Pattern Recognition* (*ICPR*), I: 358–361, 2004.
106. W. Zhao, R. Chellappa, P. Phillips, and A. Rosenfeld. Face Recognition: A literature Survey. *ACM Computing Surveys*, 35(4), 2003.
107. A. Srivastava, X. Liu, and C. Hesher. Face Recognition Using Optimal Linear Components of Range Images. *Journal of Image and Vision Computing* (*IVC*), 24(3): 291–299, 2006.
108. C. Xu, Y. Wang, T. Tan, and L. Quan. Three–dimensional face recognition using geometric model. *Proc. of SPIE on Biometric Technology for Human Identification*, 5404: 304–315, 2004.
109. V. Blanz and T. Vetter. Face Recognition Based on Fitting a 3D Morphable Model. *IEEE Trans. on Pattern Analysis and Machine Intelligence* (*PAMI*), 25(9): 1063–1074, 2003.
110. K. Choi, M. Carcassoni, and E. Hancock. Recovering facial pose with the EM algorithm. *Pattern Recognition*, 35(10): 2073–2093, 2002.
111. W. Zhao and R. Chellappa. SFS–based View Synthesis for Robust Face Recognition. *Proc. of IEEE Conf. on Automatic Face and Gesture Recognition* (*AFGR*), pp. 285–293, 2000.
112. S. Malassiotis and M. Strintzis. Pose and Illumination Compensation for 3D Face Recognition. *Proc. of IEEE Int'l Conf. on Image Processing* (*ICIP*), 1: 91–94, 2004.
113. X. Chai, L. Qing, S. Shan, X. Chen, and W. Gao. Pose Invariant Face Recognition under Arbitrary Illumination based on 3D Face Reconstruction. Proc. *Of the 5th Int'l Conf. on Audio– and Video–Based Biometric Person Authentication* (*AVBPA*), 3546: 956–965, 2005.
114. J. Lee, R. Machiraju, H. Pfister, and B. Moghaddam. Estimation of 3D Faces and Illumination from Single Photographs Using a Bilinear Illumination Model. *Eurographics Symposium on Rendering* (*EGSR*), 2005.
115. K. Chang, K. Bowyer, and P. Flynn. An evaluation of multi–modal 2D+3D face biometrics. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (*PAMI*), 27(4): 619–624, 2005.
116. P. Phillips, P. Flynn, T. Scruggs, K. Bowyer, J. Chang, K. Hoffman, J. Marques, J. Min, and W. Worek. Overview of the Face Recognition Grand Challenge. *IEEE Conf. on Computer Vision and Pattern Recognition* (*CVPR*), pp. 947-954, 2005.
117. A. Moreno and A. Sánchez. GavabDB: a 3D Face Database. *Proc. of 2nd COST 275 workshop on Biometrics on the Internet. Fundamentals, Advances, and Applications*, Spain, pp. 75–80, 2004.

118. K. Messer, J. Matas, J. Kittler, J. Luettin, and G. Maitre. XM2VTSDB: The extended M2VTS database. *2nd Int'l Conf. on Audio and Video–based Biometric Person Authentication*, pp. 72–77, 1999.
119. J. Matas, M. Hamouz, K. Jonsson, J. Kittler, Y. Li, C. Kotropoulos, A. Tefas, I. Pitas, T. Tan, H. Yan, F. Smeraldi, J. Bigun, N. Capdevielle, W. Gerstner, S. Ben–Yacoub, Y. Abdeljaoued, and E. Mayoraz. Comparison of face verification results on the XM2VTS database. *Proc. of the 15th Int'l Conference on Pattern Recognition*, 4: 858–863, 2000.
120. IMTCE 2005. Putting Your Finger on Biometrics. *Presentation of the 29th Annual Law Enforcement Information Management Training Conference and Exposition* (*IMTCE*), Greensboro, NC, USA, 2005.
121. P. Phillips, P. Grother, R. Micheals, D. Blackburn, E. Tabassi, J. Bone. Face Recognition Vendor Test 2002. *Evaluation Report*, *NISTIR* 6965, 2003.
122. P. Grother, R. Micheals, and P. Phillips. Face Recognition Vendor Test 2002 Performance Metrics. *Proc. 4th Int'l Conf. on Audio Visual Based Person Authentication*, 2003.
123. L. Wiskott, J. Fellous, N. Krüger, and C. Malsburg. Face recognition by elastic bunch graph matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (*PAMI*), 19(7): 775–779, 1997.
124. M. Hüsken, M. Brauckmann, S. Gehlen, and C. Malsburg. Strategies and Benefits of Fusion of 2D and 3D Face Recognition. *Proc. of the IEEE Computer Society Conf. on Computer Vision and Pattern Recognition* (*CVPR*), pp. 174, 2005.
125. F. Tsalakanidou, S. Malasiotis, and M. Strintzis. Face localization and authentication using color and depth images. *IEEE Transactions on Image Processing*, 14(2): 152–168, 2005.
126. S. Kong, J. Heo, B. Abidi, J. Paik, and M. Abidi. Recent Advances in Visual and Infrared Face Recognition - A Review. *ELSEVIER Computer Vision and Image Understanding*, 97(1):103-135, 2005.
127. X. Chen, P. Flynn, and K. Bowyer. Visible–light and infrared face recognition. *Proc. Of Multimodal User Authentication Workshop* (*MMUA*), Santa Barbara, CA, USA, pp. 48–55, Dec 2003.
128. J. Heo, B. Abidi, S. Kong, and M. Abidi. Performance Comparison of Visual and Thermal Signatures for Face Recognition. *The Biometric Consortium Conference*, pp. 1, Crystal City, VA, USA, 2003.
129. I. Kakadiaris, G. Passalis, T. Theoharis, G. Toderici, I. Konstantinidis, and N. Murtuza. Multimodal face recognition: combination of geometry with physiological information. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (*CVPR*), 2: 1022–1029, 2005.
130. A. Bronstein, M. Bronstein, and R. Kimmel. Expression–invariant 3D face recognition. *Proc. Of Audio Video–based Biometric Person Authentication* (*AVBPA*), Guildford, UK, pp. 62–69, 2003.
131. X. Ju, J. Nebel, and J. Siebert. 3D thermography imaging standardization technique for inflammation diagnosis. *Proc. of SPIE*, Vol. 5640–46, Photonics Asia, Nov 2004.
132. P. Aksenov, I. Clark, D. Grant, A. Inman, L. Vartikovski, and J. Nebel. 3D Thermography for the quantification of heat generation resulting from inflammation. *Proc. Of 8th 3D Modeling symposium, Paris, France*, 2003.
133. B. Khambay, J. Nebel, J. Bowman, A. Ayoub, F. Walker, and D. Hadley. A pilot study: 3D stereo photogrammetric image superimposition on to 3D CT scan images – the future

of orthognathic surgery. *Int'l Journal of Adult Orthodontics & Orthognathic Surgery*, 17(4): 331–341, 2002.

134. M. Hajeer, D. Millett, A. Ayoub, W. Kerr, and M. Bock. 3 dimension soft–tissue changes following orthognathic surgery – a preliminary report. *British Orthodontic Conference*, Harrogate, 2001.
135. U. Ruttimann, P. Andreason, and D. Rio. Head motion during positron emission tomography: Is it significant? *Psychiatry Research: Neuroimaging*, 61: 43–51, 1995.
136. H. Patterson, G. Clarke, R. Guy, and W. McKay. Head movement in normal subjects during simulated brain imaging. *Journal of Nuclear Medicine Technology*, 26(4): 257–261, 1998.
137. S. Birchfield and C. Tomasi. A Pixel Dissimilarity Measure that is Insensitive To Image Sampling. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (*PAMI*), 20(4): 401–406, 1998.
138. C. Liu and H. Wechsler. A Shape and Texture Based Enhanced Fisher Classifier for Face Recognition. *IEEE Trans. on Image Processing*, 10(4): 598–608, 2001.

Chapter 3

AUTOMATIC 3D FACE REGISTRATION WITHOUT INITIALIZATION

A. Koschan, V. R. Ayyagari, F. Boughorbel, and M. A. Abidi
Imaging, Robotics, and Intelligent Systems Laboratory, The University of Tennessee, 334 Ferris Hall, Knoxville, TN 37996, USA.

Abstract: Recently 3D face reconstruction and recognition has gained an important role in computer vision and biometrics research. Depth information of a 3D face can aid solving the uncertainties in illumination and pose variation associated with face recognition. The registration of data that is usually acquired from different views is a fundamental element of any reconstruction process. This chapter focuses on the problem of automatic registration of 3D face point sets through a criterion based on Gaussian fields. The method defines a straightforward energy function, which is always differentiable and convex in a large neighborhood of the alignment parameters; allowing for the use of powerful standard optimization techniques. The introduced technique overcomes the necessity of close initialization, which is a requirement when applying the Iterative Closest Point algorithm. Moreover, the use of the Fast Gauss Transform reduces the computational complexity of the registration algorithm.

Key words: 3D registration, range, Gaussian fields, 3D face reconstruction.

1. INTRODUCTION

The need for a robust and effective biometric system for security application has been highlighted by security agencies all over the world. The human face seems to be one of the most effective biometric features even in the uncooperative environment. Although many security systems based on

A. Koschan et al. (eds.), 3D Imaging for Safety and Security, 69–93.

2D analysis of faces are prevalent in the market, most of them suffer from the inherent problem of illumination and pose[3, 23]. This is one of the main motivating factors for research in 3D face reconstruction and recognition for security purposes. The field of 3D face reconstruction has been rapidly growing during the recent past as range scanners became more accurate, affordable, and commercially available. In fact, its applications are not restricted just to recognition, but spread over wide areas ranging from medical systems to computer animation, from video surveillance to lip reading systems, from video teleconferencing to virtual reality[8].

Automatic reconstruction of 3D face models typically involves three stages: a data acquisition stage, wherein the samples of the face are collected from different views using sensors; a data registration stage, which aligns the different 3D views into a common coordinate system; and an integration stage, which simplifies the aligned views into parametric models. Generally, some parts of the face will be unobservable from any given position, either due to occlusion or limitations in the sensor's field of view. When seen from a slightly different viewpoint, the missing data in unobserved regions is readily apparent. However, these different views will be in their local coordinate system and some transformations have to be employed to align these views in a common coordinate system. It is in this capacity that registration becomes an integral part of the reconstruction process.

In this chapter, the automatic registration problem is addressed at the point level without any explicit point correspondence. The main contribution of this work is the design of a point set criterion that is differentiable and convex in the large neighborhood of the aligned position, overcoming the shortcomings of standard registration techniques, and in particular the Iterative Closest Points (ICP) registration. The ICP algorithm is a locally convergent scheme that requires parameter initialization close to the aligned position. First described by Besl and McKay[2], ICP is the standard solution to register two roughly aligned 3D point sets D_1 and D_2. Each point of D_1 is paired with the closest point in D_2 at each ICP iteration and a transformation is computed that minimizes the mean squared error (MSE) between the paired points. The new transformation is applied to D_1 and MSE is updated. The above steps are iterated until the MSE falls below a certain threshold or a maximum number of iterations is reached. Without *a-priori* approximate estimate of the transformation, the ICP technique often ends in a local minimum instead of the global minimum which represents the best transformation.

The applied energy function is convex in the neighborhood of the solution and always differentiable, allowing for the use of a wide range of well proven optimization techniques. A straightforward sum of Gaussian distances is used which is defined for point sets with associated attributes - local

moments in this case. More importantly, the computational complexity of this criterion is reduced using the numerical technique known as the Fast Gauss Transform (FGT). The obtained results confirm that the proposed criterion can be used for accurate registration of 3D face datasets, while at the same time extending the region of convergence, avoiding the need for close initialization. In the following sections, the related work in this area is presented first. Then the Gaussian criterion and the used local attributes are described, followed by an overview of the FGT evaluation method. In the results section, an analysis of the approach is presented considering (a) the effect of the Gaussian parameter σ on the registration accuracy, (b) the robustness of the proposed algorithm to different levels of noise, and (c) the influence of the data resolution on the results. Furthermore, the performance of the proposed algorithm is compared to the performance of the standard ICP algorithm. Finally, this chapter ends with a few concluding remarks.

2. RELATED WORK

Data acquisition techniques for 3D face reconstruction can be broadly grouped into active and passive methods, based on their imaging modalities[4]. Active acquisition techniques such as laser scan and structured light use external sources of illumination for reconstruction. Passive techniques such as stereo vision, morphing, structure from motion, etc. do not depend on external sources of illumination. Most of the above mentioned methods make use of registration techniques in the process of building a complete face model.

The majority of the registration algorithms attempt to solve the classic problem of absolute orientation: finding a set of transformation matrices that will align all the data sets into a world coordinate system. Given a set of corresponding points between two 3D data sets, Horn[16] derived a closed from solution to the absolute orientation problem. Similar results were also obtained in [11]. Automatically establishing the set of correspondences to be used in such algorithm is a common interest to both registration and object recognition tasks. Several feature descriptors were used to represent free-form surfaces and point sets. In the class of global descriptors spherical representations such as the Spherical Attribute Image (SAI), which mapped surface curvature values into a tessellated sphere, were employed for 3D registration[15, 14]. Also to this category belongs the work of Lucchese et al.[17] extending frequency-domain methods to range data registration. Park and Subbarao[19] employed the invariant Stable Tangent Plan (STP) for crude registration.

In the literature, a common distinction is found between fine and coarse registration methods[5], which are often used in a two stage fashion: a coarse registration followed by fine registration using the ICP and its variants. The ICP algorithm was first introduced by Besl and MacKay[2]. Its basic version aligns a set $S = \{s_1,...,s_{N_s}\}$ of 3D scene points with a geometric model $M = \{m_1,...,m_{N_m}\}$, by minimizing the sum of the squared distances between the scene points and the model. For every point $s_i \in S$, the distance to M is defined as: $d(s_i, M) = \min_{m \in M} \|s_i - m\|$.

The ICP algorithm can be summarized as follows:

1. Start with an initial transformation (R_0, t_0).
2. For $k = 1,...,k_{\max}$ or until stopping criteria met do:
   2.1. Compute $s_i^{k-1} = R_{k-1} s_i + t_{k-1}$.
   2.2. Build the correspondence set
   $$C^{k-1} = \bigcup_{s_i \in S} \{(s_i^{k-1}, \arg\min_{m \in M} \|s_i^{k-1} - m\|)\}.$$
   2.3. Using the pairs C compute the transformation that minimizes the sum of squared distances[16].

The ICP algorithm was shown to converge monotonically to a local minimum. Therefore, the initial estimate of the transformation should be sufficiently close to the correct registration. Another limitation of the original version is that it requires large overlap between the datasets to be aligned. Step 2.3 in the algorithm is commonly solved by applying a feature matching techniques. Modifications to the original ICP algorithm have been made to improve the convergence and register partially overlapping datasets. Chen and Medioni[7] used an iterative refinement of initial coarse registration between views to perform registration utilizing the orientation information. They devised a new least square problem where the energy function being minimized is the sum of the distances from points on one view surface to the tangent plane of another views surface. This approach allowed the incorporation of local shape information, as well as the handling of partially overlapping datasets. Zhang[24] proposed a method based on heuristics to remove inconsistent matches by limiting the maximum distance between closed points allowing registration of partially overlapping data. While the basic ICP algorithm was used in the context of registration of cloud of

points, Turk and Levoy[22] devised a modified registration metric that dealt with polygon meshes. They used uniform spatial subdivision to partition the set of mesh vertices to achieve efficient local search.

Masuda and Yokoya[18] used a Least Mean Square (LMS) error measure that is robust to partial overlap to improve the robustness of ICP. Additional methods were designed to further improve the robustness of registration like, for example, the Minimum Variance Estimate (MVE) of the registration error proposed by Dorai et al.[9] and Least Median Squares (LMedS) proposed by Trucco et al.[21]. Moreover, some other variants were introduced for reducing the computational complexity such as the use of *k*-D trees to partition datasets[24] and the use of spatial subdivision to partition mesh vertices[22].

Stoddart et al.[20] studied the relationship between surface shape complexity and registration accuracy, and devised a force based optimization method to register the datasets. Early work by Arun et al.[1] on estimating 3D rigid body transformations presented a solution using the singular value decomposition (SVD). The method requires a connected set of correspondences and accurately registers the 3D data. Faugeras and Hebert[11] employed the quaternion method to solve the registration problem directly.

Eggert et al.[10] proposed a method in which data from each view is passed through Gaussian and Median filters, and point position and surface normal orientation are used to establish correspondence between points. Chen et al.[6] proposed a random sample consensus (RANSAC) scheme that is used to check all possible data-alignments of two data sets. The authors claim that their scheme works with featureless data and requires no initial pose estimate.

The non differentiability of the ICP cost function imposes the use of specialized heuristics for optimization. Addressing the registration in the context of gradient-based optimization has attracted some interest recently. In his work, Fitzgibbon[12] showed that a Levenberg-Marquardt approach to the point set registration problem offers several advantages over current ICP methods. The proposed method uses Chamfer distance transforms to compute derivatives and Huber kernels to widen the basins of convergence of existing techniques. The limitations of the ICP algorithm are overcome by introducing a straightforward differentiable cost function, explicitly expressed in terms of point coordinates and registration parameters.

3. GAUSSIAN FIELDS FOR 3D FACE REGISTRATION

The main idea employed in this 3D registration method is to make use of Gaussian fields to measure both the spatial proximity and the visual similarity of the two datasets in the point form.

3.1 Gaussian fields and energy function

A criterion is introduced on two point-sets, $M = \{(P_j, S(P_j))\}$ and $D = \{(Q_j, S(Q_j))\}$, with their associated attribute vectors. As the datasets are considered in point form, 3D moments are utilized as attributes. However, the attributes can also include curvature for smooth surfaces and curves, invariant descriptors, and color attributes when available. The Gaussian measure is given by

$$F(P_i, Q_j) = exp(-\frac{d^2(P_i, Q_j)}{\sigma^2} - \frac{(S(P_i) - S(Q_j))^T \Sigma_a^{-1} (S(P_i) - S(Q_j))}{C_a^2}) \quad (1)$$

with $d(P_i, Q_j)$ being the Euclidean distance between the points and C_a being the attribute confidence parameter. By analogy to particle physics, expression (1) can be seen as a force field whose sources are located at one point and are decaying with distance in Euclidean and attribute space. An energy function can now be defined that measures the registration of *M* and *D* as

$$E(Tr) = \sum_{\substack{i=1...N_M \\ j=1...N_D}} exp(-\frac{d^2(P_i, Tr(Q_j))}{\sigma^2} - \frac{(S(P_i) - S(Tr(Q_j)))^T \Sigma_a^{-1} (S(P_i) - S(Tr(Q_j)))}{C_a^2}) \quad (2)$$

where *Tr* is the transformation that registers the two point-sets. The force range parameter σ controls the region of convergence, while the parameter Σ_a normalizes the differences in the attributes, and the parameter C_a compensates the effect of noise on the features used in Gaussian criterion. If we choose the decay parameters very small, the energy function *E* will just 'count' the number of points that overlap at a given pose. This is due to

exponential being very small except for $P_i = (RQ_j + t)$ and $S(P_i) = S(Q_j)$. In particular, if *M* is a subset of *D* it holds at the registered position

$$(R^*, t^*): \lim_{\substack{\sigma \to 0 \\ \Sigma \to 0}} E(R^*, t^*) = N_M . \tag{3}$$

Thus, for this case the registration is defined as maximization of both overlap and local shape similarity between the datasets.

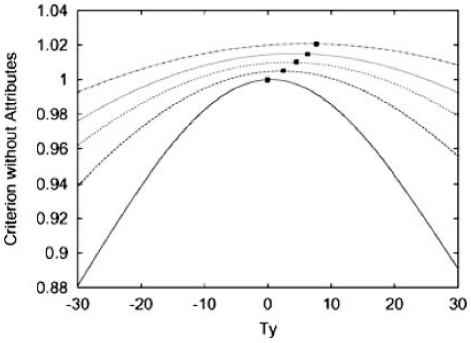
The Gaussian energy function is convex and is differentiable in a large region around the registered position, allowing us to make use of the standard optimization techniques such as Quasi-Newton methods. As mentioned earlier, the parameter σ controls the convex safe region of convergence. The higher its value, the larger will be the region of convergence, but this generally comes at the expense of reduced localization accuracy. However, the region of convergence can be extended considerably with limited reduction in localization accuracy if the datasets have sufficient shape complexity and many independent local descriptors are used.

This tradeoff can be illustrated by the behavior of the matching criteria with and without attributes as pointed out in Figure *3-1*. The profile of the criterion with increasing values of sigma is plotted for the relative displacement of the two point sets shown in Figure *3-1* (a). It is noticed in Figure *3-1* (b) that for the non-attributed case, the width of the Gaussian bell increases as σ increases, but the maximum starts to drift away from the correct position. However, when the Gaussian criterion is applied with moment invariants as attributes associated with the points, the maximum is much more stable for the same values of σ, as can be seen in Figure *3-1* (c). Instead of just continuously incorporating additional information from the point sets, a strategy is employed that tunes the parameter σ to increase the ROC without losing localization accuracy. A rough alignment is performed initially using a large sigma and then its value is decreased for future refinement steps.

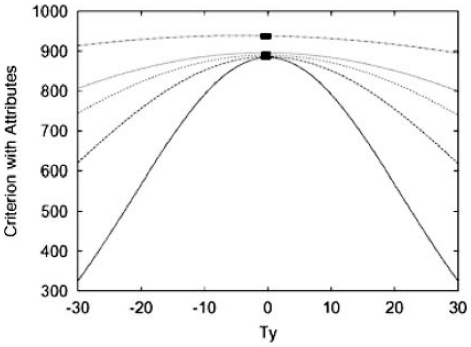
3.2 The Fast Gauss Transform

The registration criterion has a computational cost of $O(N_M \times N_D)$, being a mixture of N_D Gaussians evaluated at N_M points then summed together, which is very high for large datasets. This problem, which is also encountered in other computer vision applications, can be solved by a new numerical technique called as the Fast Gauss Transform (FGT). The method, introduced by Greengard and Strain[13], is derived from a new class of fast evaluation algorithms known as "fast multipole" methods and can reduce the computational complexity of the Gaussian mixture evaluation to $O(N_M \times N_D)$.

(a)



(b)



(c)

Figure 3-1. Profiles of the Gaussian energy function for a displacement around the registered position of the dataset shown in (a). In (b) the profiles are plotted in the case without attributes for σ = 30,50,70,90,150 (from narrowest to widest). Plots with moment invariants as attributes for the same values of σ are shown in (c) (For (b) magnitudes were rescaled for comparison). The largest dimension of the face is 200. The maximum for each curve is also marked.

The basic idea is to exploit the fact that all calculations are required only up to certain accuracy. In this framework, the sources and targets of potential fields were clustered using suitable data structures, and the sums were replaced by smaller summations that are equivalent to a given level of precision.

The Fast Gauss Transform is employed to evaluate sums of the form

$$S(t_i) = \sum_{j=1}^{N} f_j \exp\left(-\left(\frac{s_j - t_i}{\sigma} \right)^2 \right), \; i = 1,\ldots,M, \tag{4}$$

where $\{s_j\}$, $_{j=1,\ldots,N}$, are the centers of the Gaussians known as sources and $\{t_i\}$, $_{i=1,\ldots,M}$, are the targets. The following shifting identity and expansion in terms of Hermite series are used:

$$\exp\left(\frac{-(t-s)^2}{\sigma^2} \right) = \exp\left(\frac{-(t - s_0 - (s - s_0))^2}{\sigma^2} \right)$$

$$= \exp\left(\frac{-(t-s_0)^2}{\sigma^2} \right) \sum_{n=0}^{\infty} \frac{1}{n!} \left(\frac{s - s_0}{\sigma} \right)^n H_n\left(\frac{t - s_0}{\sigma} \right), \tag{5}$$

where H_n are Hermite polynomials. Given that these series converge rapidly, and that only few terms are needed for a given precision, this expression can be used to replace several sources by s_0 with a linear cost at the desired precision. These clustered sources can then be evaluated at the targets. For a large number of targets, the Taylor series (6) can similarly be used to group targets together at a cluster center t_0, further reducing the number of computations:

$$\exp\left(\frac{-(t-s)^2}{\sigma^2} \right) = \exp\left(\frac{-(t - t_0 - (s - t_0))^2}{\sigma^2} \right)$$

$$\approx \sum_{n=0}^{p} \frac{1}{n!} h_n \left(\frac{s - t_0}{\sigma} \right) \left(\frac{t - t_0}{\sigma} \right)^n, \tag{6}$$

where the Hermite functions $h_n(t)$ are defined by $h_n(t) = e^{-t^2} H_n(t)$. The method was shown to converge asymptotically to a linear behavior as the number of sources and targets increases.

4. EXPERIMENTAL ANALYSIS

In these experiments, a synthetic dataset of a mannequin head (Figure *3-1*.a) is used as well as real datasets from the IRIS 3D face database. The 3D faces in the IRIS database were scanned using a Genex 3DFaceCam, which operates on the principle of structured light. The data acquisition system uses three high resolution Charge Coupled Device (CCD) sensors and a color encoded pattern projection system. A 3D surface map is generated using the *RGB* information from each pixel and multiple 3D views are combined to generate a 3D model having ear to ear coverage. Since the 3D FaceCam system employs three CCD cameras to cover the face, frame data correspondence and registration is performed to generate the 3D face model. It is a valuable three dimensional surface profile measurement system capable of acquiring full frame dynamic 3D images of objects with complex surface geometry at a high speed. The key benefits of the 3D FaceCam are its small image acquisition time of 400-500 msec and its fast processing time of approximately 30 seconds. Nevertheless, the 3D FaceCam has practical limitations in terms of its work space which is restricted to a volumetric box of approximately 50 cm width, 40 cm height, and 30 cm depth. The minimum and maximum working distances are 80 cm and 120 cm respectively.

Fig. *3-2* shows three different 3D views of a face where the objective is to align these three data sets to reconstruct the 3D head. Note that in our experiments the texture is discarded. Some of the reconstructed models from the IRIS 3D face database are depicted in Figure *3-3*. The experiments include an investigation of the robustness of the proposed algorithm and the identification of the conditions where the registration accuracy degrades.

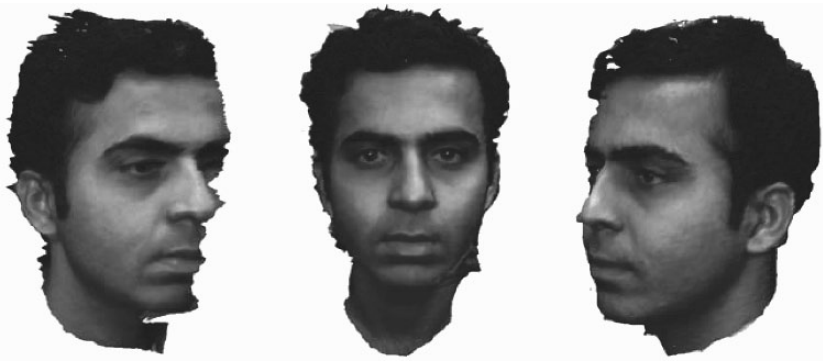


Figure 3-2. Example of three different 3D views of a face that need to be registered to generate a complete 3D face model. (See also Plate 1 in the Colour Plate Section)

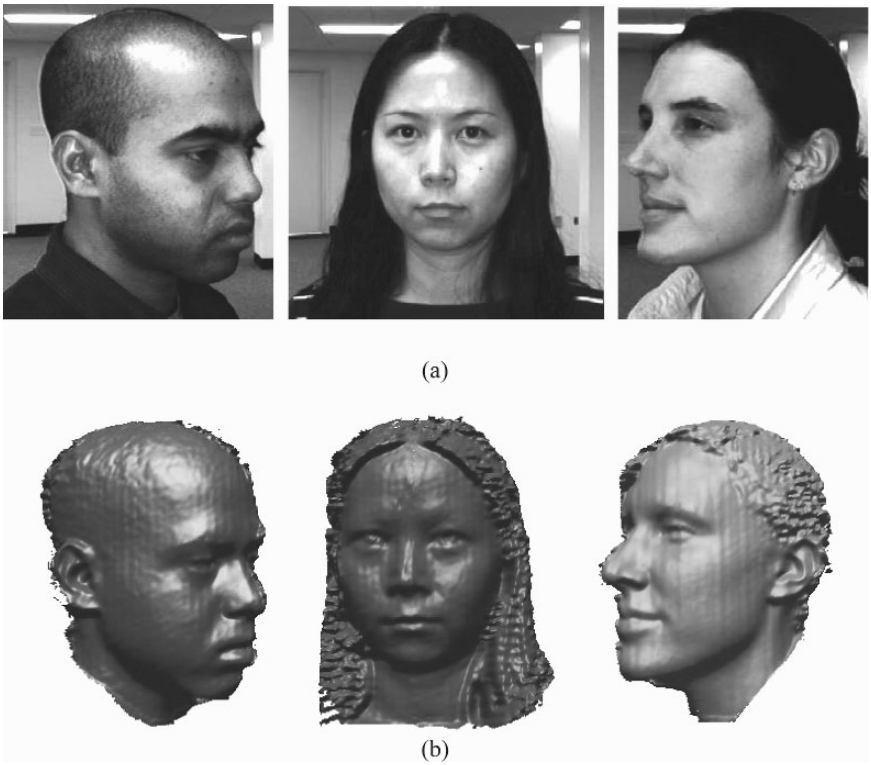


Figure 3-3. Reconstructed 3D models from partial views in the IRIS 3D database. The two dimensional color images (a), along with their associated 3D models (b).
(See also Plate 2 in the Colour Plate Section)

4.1 Effect of the parameter σ

The parameter σ controls the region of convergence which should be large for better practical applications. However, increasing the value of σ without any constraints causes a decrease in the localization accuracy. It is this bias that motivates analyzing the effect of varying σ on the registration accuracy using the synthetic and 3D face dataset from the IRIS 3D face database. The results of this experiment are shown in Figure *3-4*.

Note that both the models exhibit similar trends in the sense that the registration error increases linearly as a function of σ. However, the rate of increase slows down for larger values of σ and tends towards an asymptotic limit. This can be the explained by the fact that as σ exceeds the average distance between the points in the datasets the exponential can be approximated by its first order development

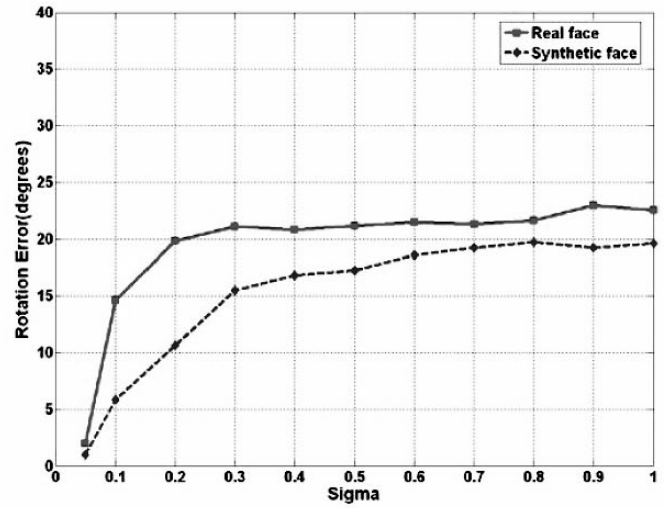
$$\exp\left(-\frac{d^2(Tr(P_i),Q_j)}{\sigma^2}\right) \approx 1-\frac{d^2(Tr(P_i),Q_j)}{\sigma^2}. \qquad (7)$$

The optimization problem now reduces to minimizing the sum of average distances from one point set to other dataset and does not depend anymore on the parameter σ. Hence, the registration error is bounded. Based on this behavior, an algorithm can be developed that starts with the initial rough alignment using a large σ and then it continues with a refinement step where σ is significantly decreased leading to a small registration error.

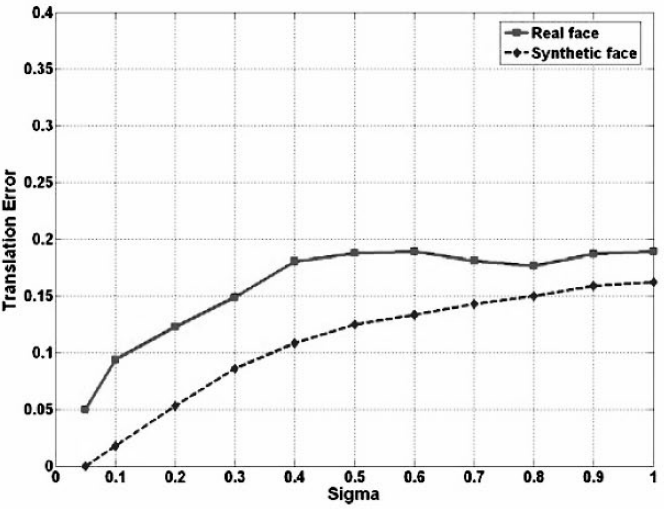
4.2 Resolution analysis

The main criteria of a high-quality registration method are the level of accuracy and the computational complexity involved. Although there are many optimization techniques which could reduce the computational complexity; the sub-sampling of the datasets would lead to a further computational gain. However, the number of points in the datasets should be sufficient to maintain the accuracy level. Hence, this turns out to be a tradeoff between the computational complexity and the level of accuracy. It was this factor which drove us to experiment on the minimum number of points in space required for an effective 3D registration.

The dataset utilized was taken from the IRIS 3D face database where the average model has around 75,000 points. In every level of sub-sampling the number of points is reduced by half. To study the influence of the reduction in resolution, the datasets is sub-sampled in three different ways:



(a)



(b)

Figure 3-4. Plots showing the rotation (a) and translation error (b) of the real face data and the synthetic face as a function of parameter σ. The parameter sigma and translation error are in terms of fraction of the length of the face model. (See also Plate 3 in the Colour Plate Section)

a) Uniform sampling, where the points are sampled at equal intervals;
b) Curvature based sampling, where points in high curvature regions are retained and points in low curvature region are thinned in order to maintain the accuracy of the curvature line; and

c) Random sampling, where the points are randomly sampled throughout the dataset.

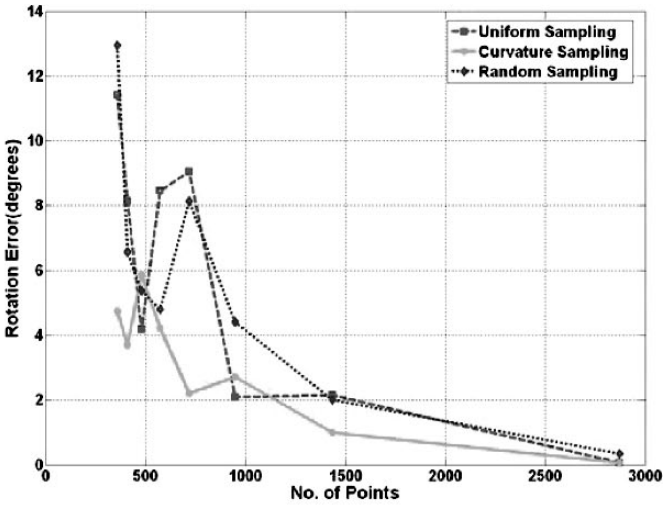
For the numerical analysis we start with a relatively small number of 3000 points for each view and then reduced the sampling by half to obtain the next pairs until we reach 350 points. Although curvature sampling provides a slight advantage over others at higher levels of sampling (lower number of points; see Figure *3-5*), no particular method can be considered superior to others. The reason that no particular sampling method can be attributed as best is due to the following reasons:

a) Uniform sampling has better spatial distribution of points but this may lead to coarser description of objects.
b) Curvature sampling has better visual description but may sometimes lead to complications due to clustering of points in certain areas.
c) Random sampling may create complications due to uneven distribution of points.

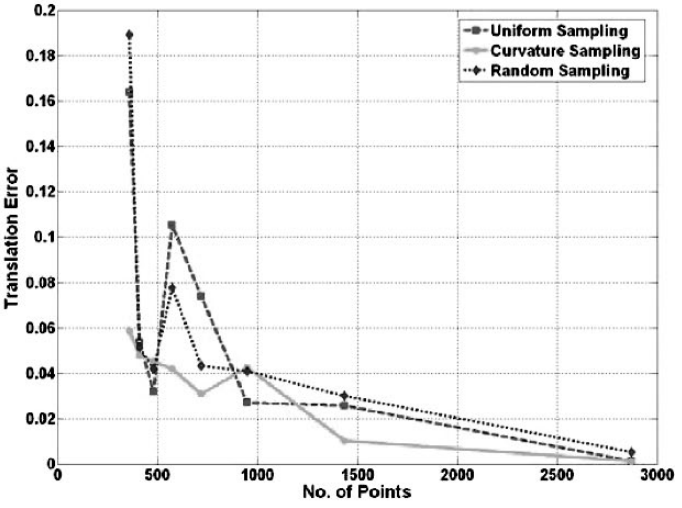
Furthermore, we can observe from Figure *3-5* that the criterion does not break down even at higher levels of sampling and remains intact even for a small number of points around 800.

4.3 Analysis of noise

Noise in the acquired data can have a significant effect on the 3D registration process, especially in the Gaussian criterion framework, because it influences both the position of the point-sets as well as the descriptors computed from them. In practical applications, noise is more dominant in the radial direction with respect to camera's coordinate frame. However, here the experimental analysis focuses on uniform noise to study the worst case scenario. As mentioned in Section 3.1, the attribute confidence parameter C_a is added to the criterion to compensate the effect of descriptors which become practically insignificant at very high levels of noise. This is achieved by forfeiting a part of discriminatory power that the descriptors add at higher levels of noise. For practical applications the confidence level factor is typically chosen to be around 10^{-3} for datasets with low noise levels and around unit value for higher noise values. For the purpose of noise analysis uniform noise is added of amplitude ranging up to 10% of the length of the face to both the models.



(a)

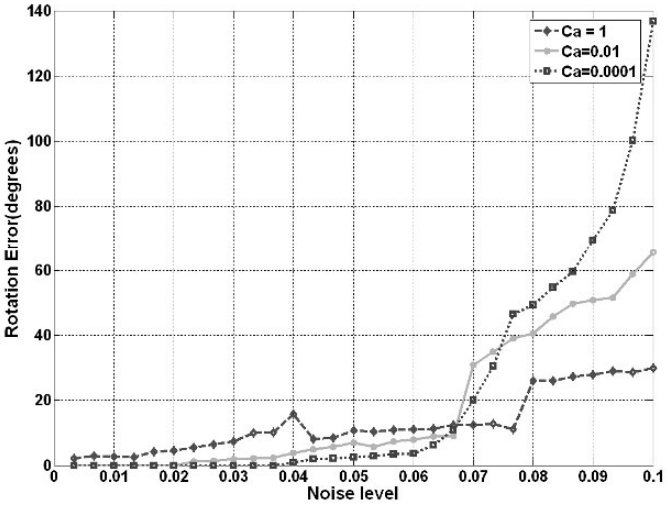


(b)

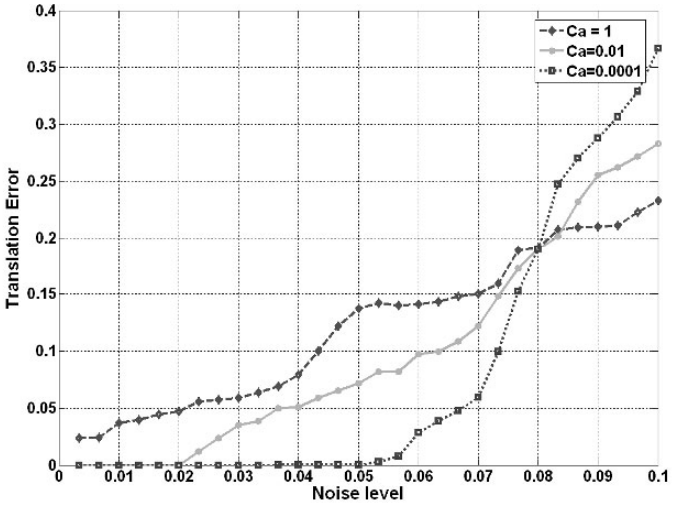
Figure 3-5. Effect of sampling on the registration accuracy. (a) Rotation error and (b) translation error as a function of number of points for three different sampling methods. (See also Plate 4 in the Colour Plate Section)

The effect of uniform noise on the drift in the maximum of the criterion can be studied from the plots shown in Figure *3-6*. The first conclusion made from the plots is that the algorithm is robust for levels of uniform noise up to ±7%, which is very high by any practical standards. The effect of C_a in

moderating the effect of registration accuracy at higher levels of noise can also be seen.



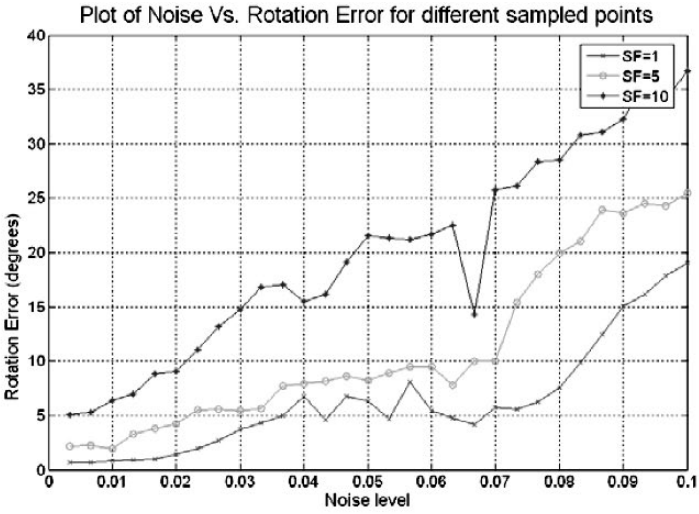
(a)



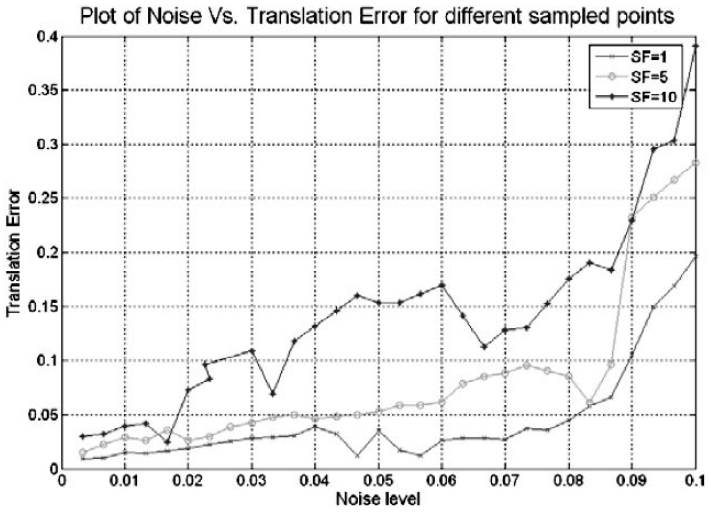
(b)

Figure 3-6. Registration error versus uniform noise level: (a) Rotation error in degrees. (b) Translation error as a fraction of the length of the face model. Plots are shown for three values of the confidence parameter.

In addition, an experimental analysis was performed to analyze the drift in the maximum of the Gaussian criterion by performing sampling and adding noise in parallel. Uniform noise was added to the models which were sub-sampled by the sampling factors (SF) SF = 5 and SF = 10. The residual error remains small for noise level up to 6% of the length of the head and then increases drastically as seen in Figure *3-7*.



(a)



(b)

Figure 3-7. Effect of noise on (a) the rotation error and (b) the translation error for different sampling factors in terms of fraction of the dimensions of the face.
(See also Plate 5 in the Colour Plate Section)

4.4 Effect of data overlap

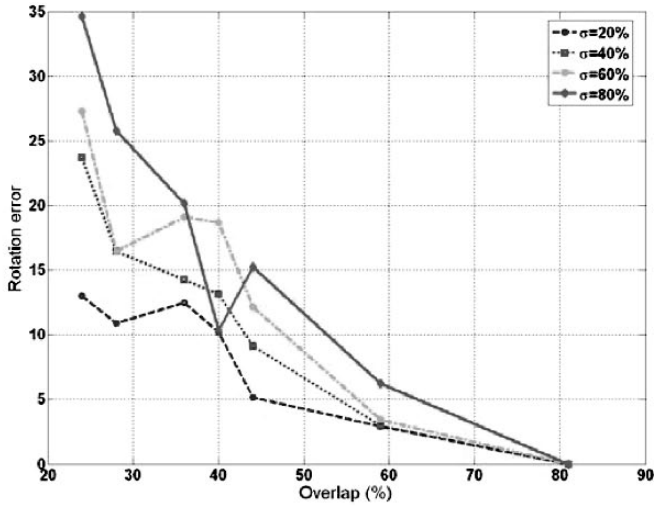
The amount of overlap between the different datasets to be registered plays an important role in the accuracy of the registration. In other terms, the lower the relative overlap between the two datasets, the higher the uncertainty of registration. The outliers which can be defined as area of datasets not shared by the datasets causes the drift in the maximum of the Gaussian criterion from the correct position, but this can be compensated by a suitable choice of force range parameter.

To study the effects of overlap, partial face models with different levels of overlap ranging from 25% to 80% were generated. The drift of the criterion maximum caused by the outliers is studied for four different values of the force range parameter σ (20%, 40%, 60%, and 80%). The translation error is computed in terms of the percentage of the largest dimensions of the box bounding the model.

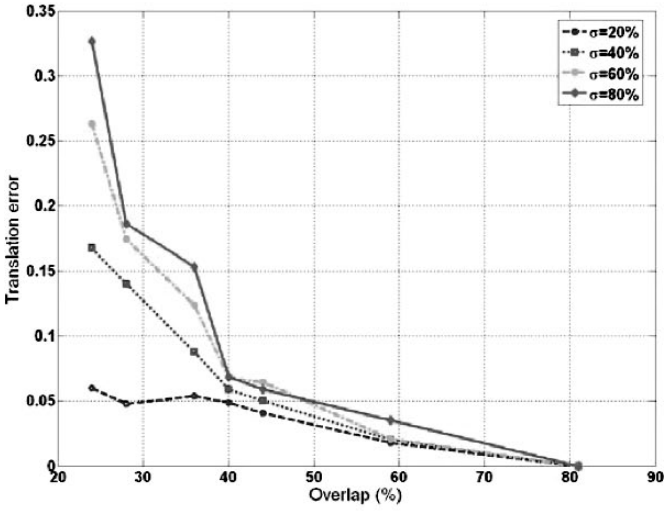
The results are summarized in of Figure *3-8* where the plots show that the algorithm is stable for up to 40% overlap and the registration accuracy decreases rapidly for overlap less than 30%. This is due to the effect of outliers and by the term outliers we mean the area which is not common in both the models. These outliers shift the maximum of the Gaussian away from its true maximum and this effect can be overridden by decrease in the force range parameter or increase in the information content. The slowest drift in the localization error occurs for the curve having low gamma which strengthens the theoretical claim about the effect of force range parameter. Hence, it can be concluded that for practical applications it is suitable to have at least around 40% to 50% overlap between the data sets that should be registered.

The next experiment was performed to analyze the effect of noise on different overlapping models used for registration. Different levels of uniform noise were added to both the face models to be registered and then the Gaussian criterion was applied on them. It is seen from Figure *3-9* that the localization error increases as the level of noise in the models increases. This increase is much higher for the face models having lower amount of overlap. At lower amount of overlap, the localization error shows an oscillating behavior. The criterion is stable to noise levels up to 6% of the length of the model.

A similar kind of experiment was also conducted to study the effect of sampling and different levels of overlap on the localization error. We start with the relatively low number of 3000 points for each view, then sample by two to obtain the next pairs until we reach 300 points. It can be seen from the experimental results shown in Figure *3-10* that the localization error increases as the number of points in the face datasets decreases.

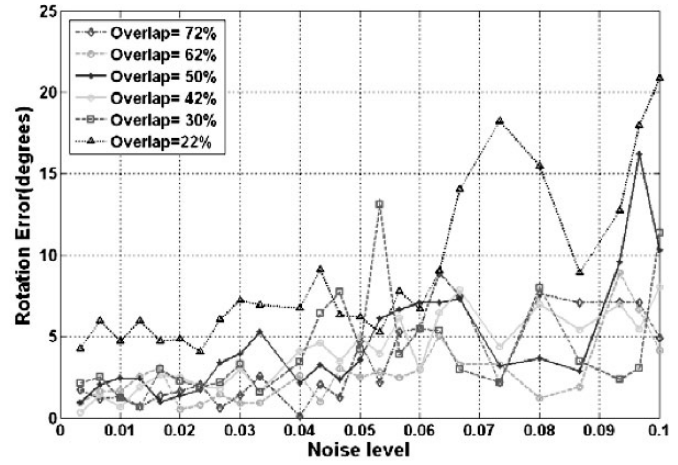


(a)

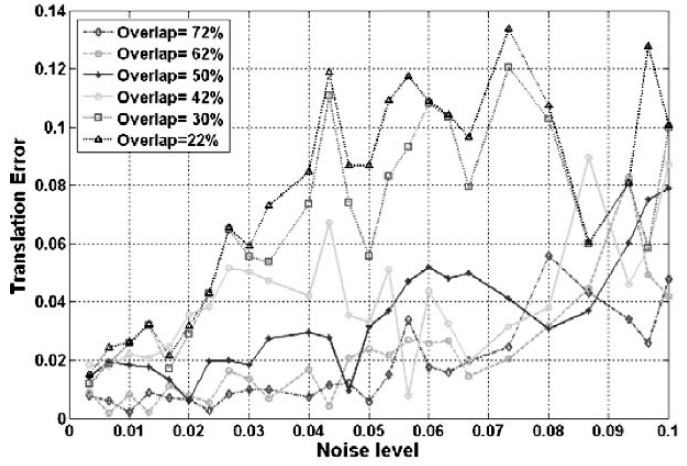


(b)

Figure 3-8. Effect of amount of overlap between two faces on the registration accuracy. Plots of (a) rotation error and (b) translation error for different percentage of overlap.
(See also Plate 6 in the Colour Plate Section)



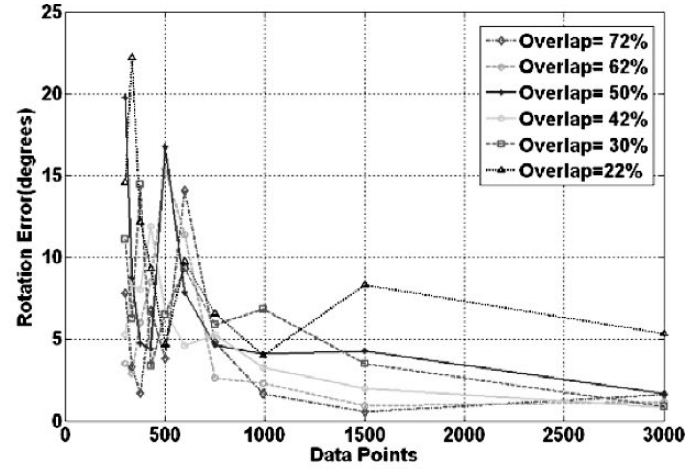
(a)



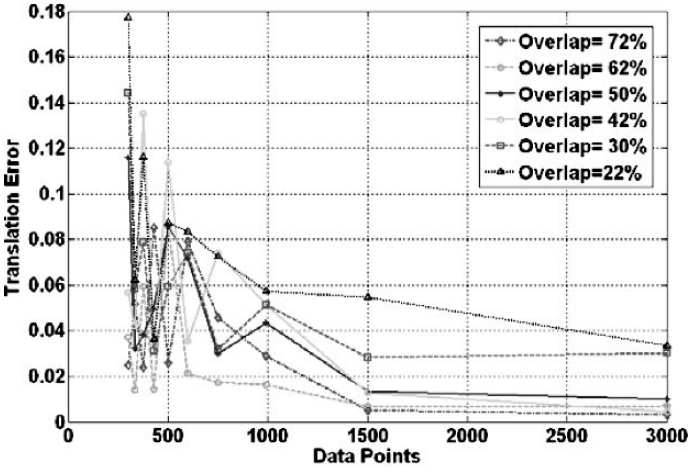
(b)

Figure 3-9. Effect of noise on the registration accuracy for models with different amount of overlap; (a) rotation error and (b) translation error for different values of overlap. (See also Plate 7 in the Colour Plate Section)

From Figure *3-10* in can be concluded that the models with smaller overlap have higher localization errors when compared to models containing the same number of points but having larger overlap. Furthermore, the criterion is stable for face models up to 700-800 points and the localization error increases drastically below that. Thus, for practical purpose it would be suitable to have an overlap of more than 40% between the data sets to be registered and a minimum number of points in each data set of at least 800 points.



(a)



(b)

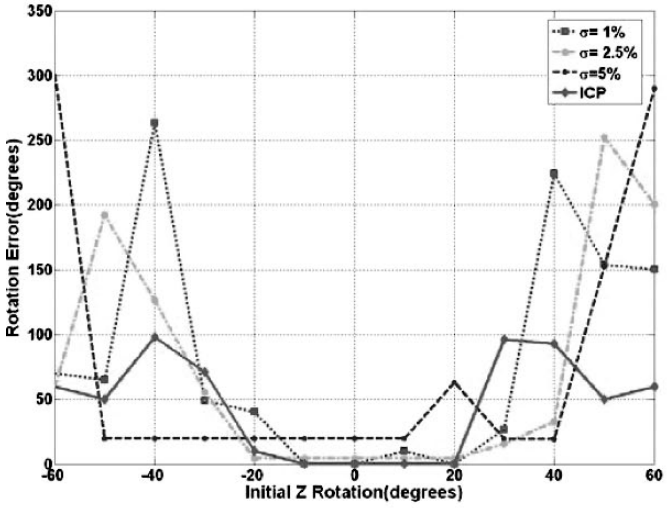
Figure 3-10. Effect of sampling on the registration accuracy for different overlap models; (a) rotation error and (b) translation error in terms of length of the model.
(See also Plate 8 in the Colour Plate Section)

4.5 Comparison with ICP

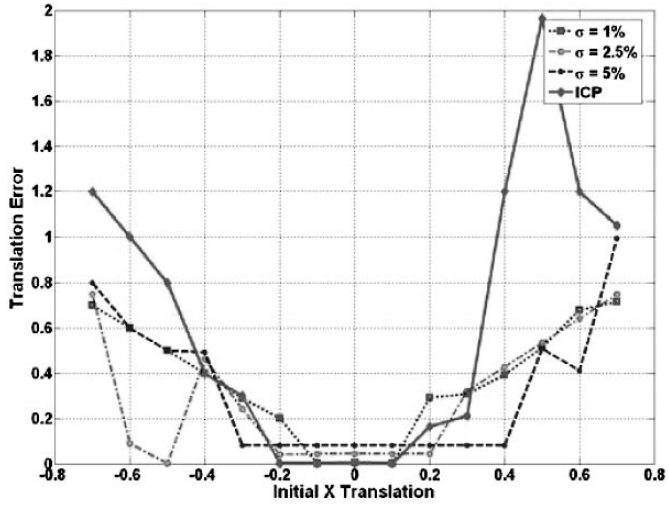
In order to study the effect of σ on the region of convergence and to demonstrate its advantages over the ICP algorithm, the basins of convergence of the algorithm are studied for the 3D face dataset. A relationship between the initial value of transformation parameters provided

to the algorithm and the residual error at the end of the process with different values of σ can be seen in Figure *3-11*.

These plots confirm the tradeoff between a large basin of convergence for a large value of σ associated with a large residual error as well, and a smaller basin of convergence for a small value of σ that comes with better registration accuracy.



(a)



(b)

Figure 3-11. Comparison of the proposed method's basin of convergence to that of ICP; rotation error (a) and translation error (b) for three different values of σ and the ICP. (See also Plate 9 in the Colour Plate Section)

It can also be seen from Figure *3-11* that the widths of the basins grow fast at first but then do not increase much after a certain value of the force range parameter. Also, when these basins are compared with that of ICP, it is found that they are wider even for small values of σ. This can be attributed to the fact that ICP is a locally convergent scheme and needs close initialization. However, the ICP has a small residual error except when compared with algorithm tuned for close Gaussian fields. Thus, a balance between the residual error and the region of convergence can be obtained by a suitable adaptive optimization scheme.

5. CONCLUSIONS

A new automatic registration method has been presented that is based on Gaussian Fields applied to 3D face reconstruction. The method overcomes the close initialization limitation of ICP and avoids the two stage registration process employed by the other algorithms. Moreover, the method allows us to start from an arbitrary initial position and converge to the registered position. A simple energy function is utilized, and by the application of the Fast Gauss Transform the computational complexity is reduced to linear level. The experiments performed on real, noisy 3D face datasets demonstrate the effectiveness of the proposed method.

6. ACKNOWLEDGEMENTS

This work is supported by the University Research Program in Robotics under grant DOE-DE-FG02-86NE37968 and by the DOD/RDECOM/NAC/ ARC Program, R01-1344-18.

REFERENCES

1. K. S. Arun, T. S. Huang, and S. D. Blostein, “Least-Squares Fitting of Two 3-D Point Sets,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 9, no. 5, pp. 698–700, 1987.
2. P. J. Besl and N. D. McKay, “A Method for Registration of 3-D Shapes,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 14, no.2, pp. 239-256, 1992.
3. K. W. Bowyer, K. I. Chang, and P. J. Flynn, “A Survey of Approaches to Three-Dimensional Face Recognition,” *Proc. Int’l Conf. on Pattern Recognition*, Cambridge, vol. 1, pp. 358-361, 2004.

4. A. Bronstein, M. Bronstein and R. Kimmel, "Expression -invariant 3D face recognition," *Proc. Audio and Video-based Biometric Person Authentication*, Lecture Notes in Comp. Science 2688, Springer, pp. 62-69, 2003.
5. R. Campbell and P. Flynn, "A Survey of Free-form Object Representation and Recognition Techniques," *Computer Vision and Image Understanding*, vol. 8, no. 2, pp. 166-210, 2001.
6. C. Chen, Y. Hung, and J. Cheng, "RANSAC-based DARCES: A New Approach for Fast Automatic Registration of Partially Overlapping Range Images," *IEEE Trans. on Pattern Analysis and Machine Vision*, vol. 21, no. 11, pp. 1229-1234, 1999.
7. Y. Chen and G. Medioni, "Object modeling by registration of multiple range images," *Image and Vision Computing*, vol.10, pp.145–155, 1992.
8. N. D'Apuzzo, "Modeling Human Faces with Multi-Image Photogrammetry," *Three-Dimensional Image Capture and Applications V, Proc. of SPIE*, San Jose, California, vol. 4661, pp. 191-197, 2002.
9. C. Dorai, J. Weng, and A. K. Jain, "Optimal Registration of Object Views using Range Data," *IEEE Trans. on Pattern Analysis Machine Intelligence*, vol. 19, no. 10, pp. 1131-1138, 1997.
10. D.W. Eggert, A.W. Fitzgibbon, and R. B. Fisher, "Simultaneous registration of multiple range views for use in reverse engineering of CAD models," *Computer Vision and Image Understanding*, vol. 69, pp 253–272, 1998.
11. O. D. Faugeras and M. Hebert, "The representation, recognition and positioning of 3-D shapes from range data," in Takeo Kanade, editor, *Three-Dimensional Machine Vision*, Kluwer Academic Publishers, Boston, Massachusetts, pp 301-353, 1987.
12. A. W. Fitzgibbon, "Robust registration of 2D and 3D Point Sets," *Image and Vision Computing*, vol. 21, pp. 1145-1153, 2003.
13. L. Greengard and J. Strain, "The Fast Gauss Transform," *SIAM J. Scientific Computing*, vol. 12, pp. 79-94, 1991.
14. M. Hebert, K. Ikeuchi, and H. Delingette, "A Spherical Representation for Recognition of Free-form Surfaces," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 7, pp. 681-690, 1995.
15. K. Higuchi, M. Hebert, and K. Ikeuchi, "Building 3-D Models from Unregistered Range Images," *Graphical Models and Image Processing*, vol. 57, no. 4, pp. 315-333, 1995.
16. B. K. P. Horn, "Closed-form Solution of Absolute Orientation using Unit Quaternions," *Journal of the Optical Society of America*, vol. 4, no. 4, pp. 629-642, 1987.
17. L. Lucchese, G. Doretto, and G. M. Cortelazzo, "A frequency domain technique for range data registration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 11, pp.1468-1484, 2002.
18. T. Masuda and N Yokoya, "A Robust Method for Registration and Segmentation of Multiple Range Images," *Computer Vision and Image Understanding*, vol. 61, no. 3, pp. 295-307, 1995.
19. S. Y. Park and M. Subbarao, "Automatic 3D Reconstruction based on Novel Pose Estimation and Integration Techniques," *Image and Vision Computing*, vol. 22, no. 8, pp. 623-635, 2004.
20. A. J. Stoddart and A. Hilton, "Registration of multiple point sets," *Proc. 13th Int'l Conf. on Pattern Recognition*, Vienna, Austria, vol. 2, pp. 40-44, 1996.
21. E. Trucco, A. Fusiello, and V. Roberto, "Robust motion and correspondence of noisy 3-D point sets with missing data," *Pattern Recognition Letters*, vol. 20, pp. 889-898, 1999.
22. G. Turk and M. Levoy, "Zippered Polygon Meshes from Range Images," *Proc. of Siggraph 94*, Orlando, FL, pp. 311-318, 1994.

23. W. Zhao and R. Chellappa, "3D Model Enhanced Face Recognition," *Proc. of Int'l Conf. on Image Processing*, vol. 3, pp. 50-53, 2000.
24. Z. Zhang, "Iterative Point Matching for Registration of Free-Form Curves and Surfaces," *Int'l Journal of Computer Vision*, vol. 13, no. 2, pp. 119-152, 1994.

Chapter 4

A GENETIC ALGORITHM BASED APPROACH FOR 3D FACE RECOGNITION

Using Geometric Face Modeling and Labeling

Y. Sun and L. Yin
Computer Science Department , State University of New York at Binghamton, Binghamton, New York 13902 USA

Abstract: The ability to distinguish different people by using 3D facial information is an active research problem being undertaken by the face recognition community. In this paper, we propose to use a generic model to label 3D facial features. This approach relies on our realistic face modeling technique, by which the individual face model is created using a generic model and two views of a face. In the individualized model, we label face features by their principal curvatures. Among the labeled features, "good features" are selected by using a Genetic Algorithm based approach. The feature space is then formed by using these new 3D shape descriptors, and each individual face is classified according to its feature space correlation. We applied 105 individual models for the experiment. The experimental results show that the shape information obtained from the 3D individualized model can be used to classify and identify individual facial surfaces. The rank-4 recognition rate is 92%. The 3D individualized model provides consistent and sufficient details to represent individual faces while using a much more simplified representation than the range data models. To verify the accuracy and robustness of the selected feature spaces, a similar procedure is applied on the range data obtained from the 3D scanner. We used a subset of the optimal feature space derived from the Genetic Algorithm, and achieved an 87% rank-4 recognition rate. It shows that our approach provides a possible way to reduce the complexity of 3D data processing and is feasible to applications using different sources of 3D data.

Key words: genetic algorithm; 3D face recognition; feature selection; generic model; geometric modeling and labeling.

A. Koschan et al. (eds.), 3D Imaging for Safety and Security, 95–118.

1. INTRODUCTION

3D face recognition[1-10, 28] has attracted much attention in recent years. It seems to be superior to 2D face recognition due to its less affection by imaging conditions and facial pose variations[11, 12]. Currently, most research for exploring 3D face information focuses on the investigation of 3D range data obtained by 3D digitizers due to the difficulty of 3D reconstruction from 2D images[3, 5, 10, 13]. Despite its high fidelity representation of faces, 3D range data is not feasible for many application scenarios. Firstly, an ideal environment with the cooperation of subjects is needed to capture 3D information using a 3D scanner. Secondly, the range data obtained is usually in a raw format, and different persons' faces may have a different number of vertices. Thus, it is hard to compare from face to face without building consistent correspondences between two subjects. Therefore, further processing is needed in order to extract the feature regions and segment the facial surface. Finally, although the high-resolution range data provides a realistic appearance that is good for perception and animation in the entertainment industry, it contains redundant information, which may be unnecessary or could make the face shape comparison noise sensitive.

In order to reduce the complexity of data processing and to provide the model correspondence easily, a template model could be used for fitting a generic model to the face image[14, 15]. Currently, there are some successful systems that utilize 3D wire-frame models to represent the face shape for recognition. However, the accuracy of 3D feature representation is still limited by the selected simple models in which only a small number of vertices are presented. Hence, it is hard to label the face surface and characterize its property precisely.

In this paper, we present a more accurate face representation method using a generic model with about 3000 vertices. The utilization of the face model with certain accuracy can help us describe the face surface more precisely. We label the face surface by its principal curvatures based on the generated individual facial model. Since every individualized model has the same number of vertices, the correspondence between different individual models is inherently established. Thus, the 3D shape comparison can be carried out accordingly. We investigate a number of different metrics to measure the facial model disparity, and choose to use the correlation metric as our model similarity measurement for efficiency and simplicity. Our 3D face recognition approach has been evaluated based on various databases, including FERET, XM2VTS and our in-house data set. Finally, our selected optimal features are assessed by using 3D range models captured by a 3D face imaging system to validate the applicability of such features to high-resolution 3D face range data.

The paper is organized as follows: In Section 2, we will overview our proposed 3D face recognition frameworks. Section 3 will introduce the procedure of our 3D face model creation using our existing face modeling tool. The algorithms for face model labeling and good feature selection are described in Section 4 and Section 5, respectively. Section 6 explains the similarity measurement using a number of metrics, followed by the experiments and performance analysis in Section 7. Finally, the concluding remarks are given in Section 8.

2. FRAMEWORK OF THE SYSTEM

Figure *4-1* shows the general framework of our proposed system. In this system, a 3D face model database is firstly created based on two views of face images of each subject and a generic face model. The face modeling algorithm consists of several key components, including (1) facial feature analysis and detection from two views' of a face; (2) a generic face model instantiation by adapting the model to the two views' facial images, and (3) the facial geometric model synthesis using two individual-view adjusted models.

After the individual model is created, we implement a cubic approximation method on each 3D individualized model to analyze the principal curvatures (i.e., the maximum and minimum curvatures at each vertex's location). After labeling the facial surface by using eight categories of surface types (e.g. concave, convex, and saddle, etc.), we are able to segment different regions and extract the 3D shape descriptors using their surface features.

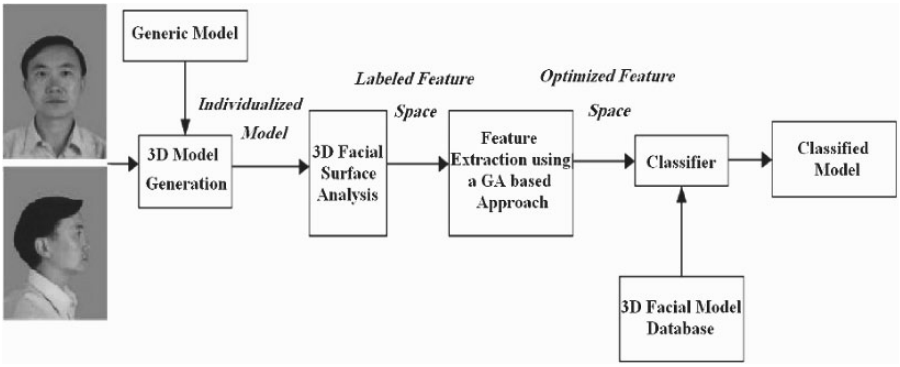


Figure 4-1. Framework of proposed 3D face labeling and recognition system.

These 3D shape descriptors are the simplified representation of each facial instance, which preserve the surface varieties of different faces. Among the 3D descriptors, only the partial features ("good features") are selected. We apply a feature selection procedure based on the Genetic Algorithm to form a feature vector that is composed of the selected "good features". As a result, the dimension of the 3D descriptors is greatly reduced. Finally, the composed new feature vectors are used for a similarity measurement to classify individual facial models based on their feature space correlations. One subject is classified the same as that which ranks the highest among all other subjects.

3. 3D FACIAL MODEL CREATION

We created a 3D facial model by modifying a generic facial model to customize the individual person's face, given two views of images, i.e., a front view and a side view of the face. This approach is based on recovering the structure of selected feature points in the face and then adjusting a generic model using these control points to obtain the individualized 3D head model. The algorithm is implemented by three major components. First of all, a set of fiducial points are extracted from a frontal view and a side view of face images using a maximum curvature tracing algorithm. Then, the generic face model is adjusted to fit the face images in two views separately. Finally, the two view's models are combined together to generate the individualized 3D facial model. The blended textures from the two views are synthesized and mapped onto the instantiated model to generate the face images in arbitrary views.

We apply a local maximum curvature tracing (LMCT) algorithm to identify features on the profile such as chin tip, mouth, nose tip, and nose bridge[16]. Eight salient feature points are exhibited on the profile of a human face (referred by Ip and Yin[16]). Given the vertical positions of the eyes, nose, mouth, and chin as determined by profile analysis, we can limit the search area for interior feature points of the front view face, and features of each organ can be extracted within the restricted window area. The orientation of the face can be determined by the key features within each image. For the frontal view, the eye centers and nose center are used. For the profile view, the tip of the nose and the tip of the chin are used.

The next stage is to extract the shape for the feature measurement and the model adaptation. The shape of facial features mainly refers to the contour of eyes, eyebrows, mouth, nose (nostril and nose side), and chin. Given the restricted feature region, the deformable template matching method[17] seems a good choice to extract the feature shape because the initial location of the

template is close to the target position. Since the feature regions have been restricted in the corresponding window areas, the feature shape can be extracted effectively.

The model adaptation scheme intends to deform the face model into the non-rigid face area. Based on the previously detected the feature shapes, the model adaptation process can then be applied. By fitting the shape of the individual 3D model to the frontal and profile view, and combining the two results, a reasonable geometric description of the objects of interest in the scene is automatically acquired. Our model adaptation procedure consists of two major steps:

(1) Coarse adaptation, which applies the dynamic mesh method to make the large movement converge quickly to the region of an object.

(2) Fine adaptation, which applies the EOM method to finely adjust the mesh obtained after the first step and makes the adaptation more "tight" and "accurate". The extended dynamic mesh will produce much more accurate and realistic face representation than the existing methods. The detailed algorithm can be found in Yin and Basu[18]. An example is shown in Figure *4-2.*

Since the generic model has been modified separately for the front view and the side view in the previous stage, two sets of 2D coordinates of all the vertices are obtained. They are (x_f, y_f) in the front view plane and (z_s, y_s) in the side view plane. Therefore, the 3D coordinates of all the vertices can be simply estimated as follows:

$$(x, y, z) = (x_f, \frac{y_f + y_s}{2}, z_s) \tag{1}$$

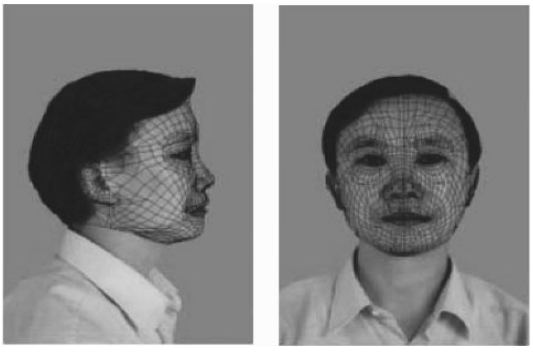


Figure 4-2. Example of face model adaptation onto two views of faces.

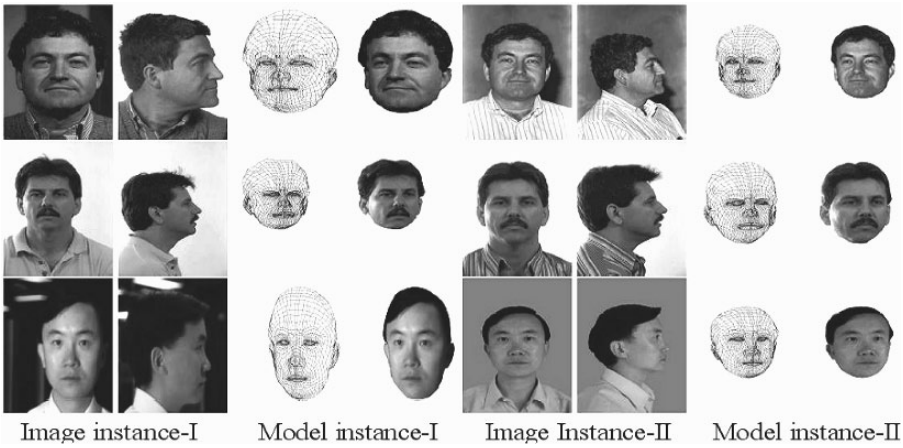


Figure 4-3. Examples of 3D model database created from the frontal and profile views of each subject. Row 1 to Row 3 shows three subjects; each subject has two instance models generated from corresponding image instances.

The resulting individual 3D Facial models are shown in Figure *4-3* with texture mapping. Our generic wire-frame facial model consists of 2953 vertices, which represent the face surface with sufficient accuracy. This model instantiation approach is simple and efficient. The created model is accurate enough to characterize features of the individual face. Our 3D facial model database is generated using 105 pairs of face images from 40 subjects. These source image pairs are mainly chosen from the database of FERET[19] and XM2VTS[20], and some additional images are captured by our research group. For each subject, there are two or three pairs of frontal and profile images, which were taken under different imaging conditions (e.g., different poses, lighting and facial appearances at different time-period).

4. 3D FACIAL MODEL LABELING

To represent the information of 3D faces, a naïve method is to represent each face as a vector, where the (x, y, z) coordinate value of each vertex is the component of that vector. However, such a representation is very sensitive to the surface's change and noises.

Curvature values of each vertex on the model may also form a feature vector to represent the facial surface. It seems to be less influenced by the surface's change or the noise as compared to the coordinate representation. However, it does not provide sufficient stability to represent individual facial surfaces too. In fact, our preliminary study has shown that these two

descriptors are not able to distinguish the faces of different subjects. Thus, we need to explore a more robust 3D feature descriptor to represent the local 3D facial surface.

In order to better characterize 3D features of the facial surface, each vertex on the individual model is labeled by one of the eight label types (i.e., convex peak, convex cylinder/cone, convex saddle, minimal surface, concave saddle, concave cylinder/cone, concave pit and planar). Therefore, the facial model is represented by a set of labels. Our labeling approach is based on the estimation of principal curvatures of the facial surface, which has desirable computational and perceptual characteristics[13, 21], and is independent of rigid transformation. It is believed that a set of curvature-based surface labels is a good reflection of local shapes of the facial surface.

Given a face mesh model of vertices and polygons approximating the facial smooth surface, we need to calculate accurate estimates of the principal curvatures and their directions at points on the facial surface. There are a number of methods for estimating the principal curvatures and principal directions of the underlying surface, including the normal curvature approximation, quadratic surface approximation, cubic approximation, and the higher order methods[22], since the estimation is performed on the sparse mesh model rather than the dense range data model, the minor normal curvature approximation errors could be magnified into large errors in the estimated principal directions. Therefore, we need to choose a high order approximation method. Considering the trade-off between the estimation accuracy and the computation complexity, we adopt the cubic approximation approach to calculate the principal curvatures and their directions on each vertex of the 3D model. The calculation is briefly described as follows:

Let p denote a point on a surface S, N_p denote the unit normal to S at point p, and $X(u,v)$ be a local parameterization of surface S at p. Using $X_u(p)$, $X_v(p)$ and N_p as a local orthogonal system, we can obtain the principal curvatures and the principal directions by computing he eigenvalues and eigenvectors of the Weingarten curvature matrix:

$$W = \begin{bmatrix} \dfrac{eG - fF}{EG - F^2} & \dfrac{fE - eF}{EG - F^2} \\ \dfrac{fG - gF}{EG - F^2} & \dfrac{gE - fF}{EG - F^2} \end{bmatrix} \tag{2}$$

Where, $e = N_P \cdot X_{uu}(p), E = X_u(p) \cdot X_u(p)$, $f = N_p \cdot X_{uv}(p)$, $F = X_u(p) \cdot X_v(p)$, $g = N_p \cdot X_{vv}(p)$ and $G = X_v(p) \cdot X_v(p)$

Note that if we properly choose $X_u(p)$ and $X_v(p)$ to be orthogonal unit vectors, the Weingarten curvature matrix can be simplified as a symmetric matrix.

$$W = \begin{pmatrix} A & B \\ B & C \end{pmatrix} \tag{3}$$

Where, A=e, B=f, and C=g. The eigenvalues λ_1 and λ_2 of the matrix W are the maximum and minimum principal curvatures of the surface S at point p. The eigenvectors λ_1 and λ_2, which are represented in a global coordinate system, are the corresponding maximum and minimum principal directions. In order to estimate the Weingarten curvature matrix accurately, we apply the cubic approximation approach to fit a local cubic surface onto the face model surface centered on each vertex. To do so, we firstly transform the adjacent vertices of p to a local system with the origin at p and with the positive z axis along the estimated normal N_p. Based on the cubic surface equation, we can derive its surface normal and establish two equations for each normal as following.

We can use a cubic surface to fit in the local vertices data. Firstly, we transform each adjacent vertex q_i to the local coordinate system (x_i, y_i, z_i). In the local system, p is the original point (0, 0, 0) and the estimated normal $N_p^{'}$ is along the positive z-axis.

$$z = S(x, y) = \tfrac{1}{2}Ax^2 + Bxy + \tfrac{1}{2}Cy^2 + Dx^3 + Ex^2y + Fxy^2 + Gy^3 \tag{4}$$

Since the normal to the surface can be represented as

$$N(x, y) = (S_x(x, y), S_y(x, y), -1) = (-\tfrac{a_i}{c_i}, -\tfrac{b_i}{c_i}, -1)$$

$$= \left(Ax + By + 3Dx^2 + 2Exy + Fy^2, Bx + Cy + Ex^2 + 2Fxy + 3Gy^2, -1\right) \tag{5}$$

Where (a_i, b_i, c_i) denote the normal at point (x_i, y_i, z_i) in the local coordinate system. Let $\alpha = (A \quad B \quad C \quad D \quad E \quad F \quad G)^T$, then we can rewrite equation (4) as

$$\left(\tfrac{1}{2}x_i^2 \quad x_i y_i \quad \tfrac{1}{2}y_i^2 \quad x_i^3 \quad x_i^2 y_i \quad x_i y_i^2 \quad y_i^3\right) \cdot \alpha = z_i \tag{6}$$

And rewrite equation (5) as two equations

$$\begin{pmatrix} x_i & y_i & 0 & 3x_i^2 & 2x_i y_i & y_i^2 & 0 \end{pmatrix} \cdot \alpha = -\frac{a_i}{c_i} \tag{7}$$

$$\begin{pmatrix} 0 & x_i & y_i & 0 & x_i^2 & 2x_i y_i & 3y_i^2 \end{pmatrix} \cdot \alpha = -\frac{b_i}{c_i} \tag{8}$$

Then a linear regression method can be applied to solve the equation groups (6), (7), and (8), and the elements of the Weingarten matrix can be determined (see details in Goldfeather and Interrante[22]). Figure *4-4* shows examples of the obtained maximum and minimum principal directions for the vertices of one instance models from a same subject using the cubic approximation method. Note that this model is generated from a sequential facial motion (expressional changes) of the same subject.

After estimating the principal curvatures and directions of each vertex, we can further categorize each vertex into one of the eight distinct label types. The labels are classified according to the relation of the principal curvatures, which are expounded in Table *4-1*.

Since every vertex is represented by a surface label, each labeled vertex provides a feature component in the facial feature space. The reason that we use the surface labels to represent the local shape is that the defined basic surface categories are relatively insensitive to the small shape distortion and locations. As such, once the reconstruction from two orthogonal views of images to the 3D individualized model is accurate enough to preserve the curvature types of the vertices, it will not influence the classification result. The usage of surface labeling can help improve the robustness to the facial expression variation. We will illustrate this property in Section 6.

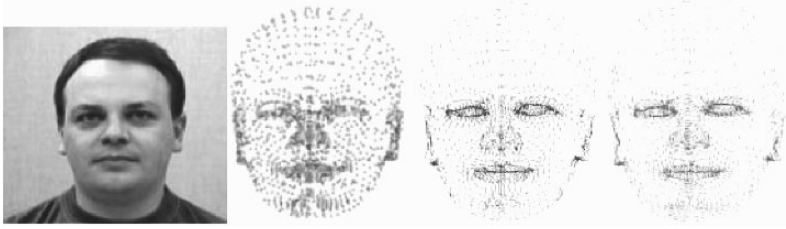


Figure 4-4. From left to right, a frontal view of the image, a labeled model, and models showing the maximum principal direction and the minimum principal direction respectively.

Table 4-1. Eight distinct types of facial surface labels classified by their principal curvatures.

	Surface label	Principal curvatures
(1)	Convex (peak)	$\lambda_1, \lambda_2 < 0$
(2)	Convex cylinder/cone	$\lambda_1 = 0,\ \lambda_2 < 0$
(3)	Convex saddle	$\lambda_2 < 0 < \lambda_1$, and $\lvert\lambda_1\rvert < \lvert\lambda_2\rvert$
(4)	Minimal surface	$\lvert\lambda_1\rvert = \lvert\lambda_2\rvert$
(5)	Concave saddle	$\lambda_2 < 0 < \lambda_1$, and $\lvert\lambda_1\rvert > \lvert\lambda_2\rvert$
(6)	Concave cylinder/cone	$\lambda_2 = 0,\ \lambda_1 < 0$
(7)	Concave (pit)	$\lambda_1, \lambda_2 > 0$
(8)	Planar	$\lambda_1, \lambda_2 > 0$

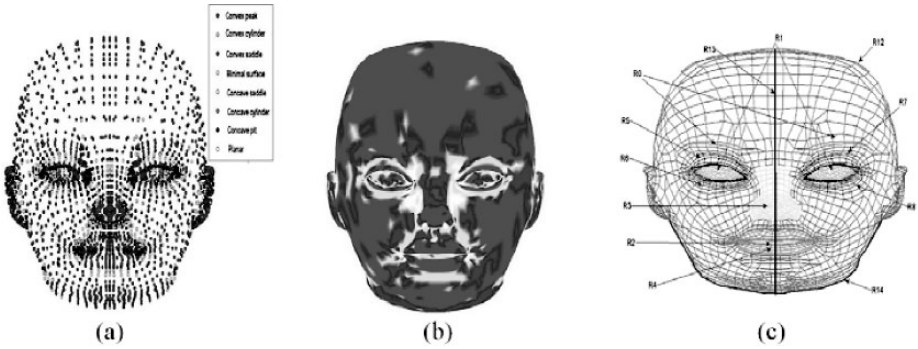


Figure 4-5. (a) 3D model labeling using eight label types based on the principal curvatures. Each dot corresponds to one vertex, which is labeled by one color. (b) 3D labeled model after rendering. (c) Sub-regions defined on the generic facial model.
(See also Plate 10 in the Colour Plate Section)

Figure *4-5*(a) shows the result of surface labeling on the generic model using the cubic approximation method. Four label types (i.e. convex (peak), convex saddle, concave (pit) and concave saddle) are dominant on this model as a whole. In order to understand the label distribution on different regions of a face, we conduct histogram statistics on different regions of the individual models. Figure *4-5*(b) is the rendered result of the 3D labeled model. The statistical results show that different facial regions of the same subject have different distributions of the label types. Furthermore, the label distributions of the same facial region for different subjects are also clearly distinguishable. As an example, Figure *4-6* illustrates the label histograms of nine subjects' facial models on two selected regions (i.e., eyeballs and eyelids). As seen from (a) and (b), the label histograms in the same region are different from one person to another among the nine subjects, e.g., the dominant label in the region of eye-lids is the convex saddle for Subject 9, while it is the concave saddle for Subject 6. The label histograms are also different from region to region in the same subject's model, as compared between region (a) eyeballs and (b) eye-lids.

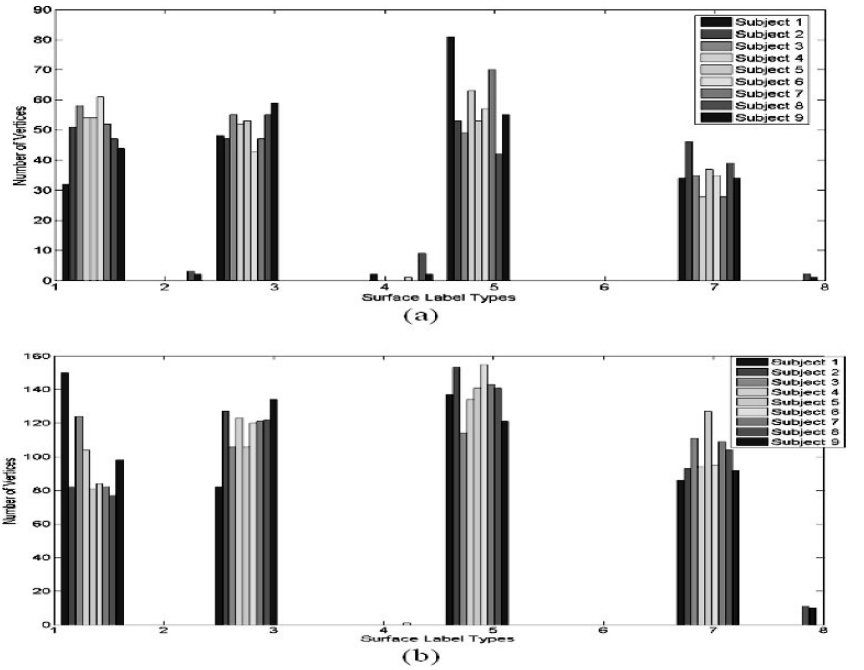


Figure 4-6. Histograms of the vertex label types in regions of (a) eyeballs and (b) eye-lids of nine subjects. Eight label types 1-8 are defined in the Table *4-1*.
(See also Plate 11 in the Colour Plate Section)

5. GOOD FEATURE SELECTION USING A GA-BASED APPROACH

After the stage of facial model labeling, each individual face model is represented by a set of labels (i.e., 2953 labeled vertices in each model). Among the set of labels, only the labels located in certain regions are of our most interest. Some non-feature labels could be noises that could blur the individual facial characteristics. Therefore, we need to apply a feature screening process to select features in order to better represent the individual facial traits for maximizing the difference between different subjects while minimizing the size of the feature space. We refer to the selected features as the "good features". In order to select the good features, we partition the face model into 15 sub-regions based on their physical structures (there are overlaps between some of the regions), which is similar to the region components used by Juang et al.[2], as expounded in Table *4-2* and Figure *4-7*(c). These sub-regions represent the anatomic structure of the human face. However, they are not equally robust for characterizing the traits of each

individual subject. For example, some regions are sensitive to local expression variation, and they may not be the good features for discriminating different faces.

Since not all the sub-regions contribute to the recognition task, and not all the vertices within one sub-region contribute to the classification, we need to select (1) the best set of vertex labels within each sub-region, and (2) the best set of sub-regions. The purpose of the feature selection is to remove the irrelevant or redundant features which may degrade the performance of face classification. Since it is a difficult task to examine all possible combinations of feature sets, an efficient strategy for optimal feature selection is highly necessary. The genetic algorithms (GA) have been used successfully to address this type of problem[23]. We choose to use a GA-based method to select the components that contribute the most to our face recognition task. GA provides a learning method analogous to biological evolution.

Table 4-2. Selected 15 sub-regions (from R0 to R14) for the feature extraction.

R0	Eye brows	R5	Upper eye lid of the left eye	R10	Lower eye lids(R6+R8)
R1	Eye balls	R6	Lower eye lid of the left eye	R11	Eye lids (R9+R10)
R2	Lip	R7	Upper eye lid of the right eye	R12	Frontal head contour
R3	Nose	R8	Lower eye lid of the right eye	R13	Central facial profile
R4	Mouth	R9	Upper eye lids(R5+R7)	R14	Chin contour

Initialization

Let the **EER** denote as a fitness function
Compute the fitness function for each chromosome and rank them
Feature space = the chromosome with the highest rank fitness value
For each chromosome left in the current sub-region
  Do it in the feature space, and compute the **EER** of the current feature space
    If **EER** has increased,
      Remove the added chromosome

Keep the final regional feature space as optimal feature space of the current sub-region

Figure 4-7. Pseudo-procedure of proposed Genetic Algorithm based feature selection approach.

It is an evolutionary optimization approach, which is an alternative to traditional optimization methods. GA is the most appropriate for complex non-linear models in order to find the location of the global optimum[24]. The solution of GA is identified by a fitness function where the local optima are not distinguished from other equally fit individuals. Those solutions closer to the global optimum will have higher fitness values, and the outcomes will tend to get closer to the global optimum. Here, we chose to use the Equal Error Rate (EER) as the fitness function. For each sub-region, we obtained an EER when we performed the face similarity measurement based on the sub-region only, given the training set of the 3D facial model database. The similarity measure is based on the feature space's correlation, which will be introduced in the next section. The procedure for the feature selection consists of two parts: (1) vertices selection in each sub-region using a Genetic Algorithm (GA) approach, and (2) the integration of sub-regions.

5.1 Vertices selection for each sub-region

The GA-based approach is described as follows: Let ϕ denote the target optimal feature space, which is initialized as an empty set, and $X = (x_1, x_2, ...x_n)$ denote the full candidate feature space. Firstly, the algorithm calculates values of the fitness function (e.g., equal error rates) using each instance feature x_i (we call it chromosome). Secondly, the algorithm ranks the features in the candidate feature space according to their calculated EER values. Thirdly, the algorithm adds the features that achieve the lowest EER value to the initial feature space ϕ. The highly ranked chromosome is sequentially added to the target optimal feature space until the achieved EER value is increased because the case with a higher EER value may not be suitable for the recognition task. Figure *4-7* depicts the pseudo-procedure of the Genetic Algorithm based feature selection.

5.2 The integration of sub-regions

Following the previous step, all the sub-regions have been reduced to the regional feature spaces. At this point, we may use the EER to determine which regional feature spaces should be selected as our final feature spaces. Here we choose those regional feature spaces that have a high-ranking fitness value (EER) as the final feature space. Figure *4-8*(a) shows the EER for each sub-region after the vertices are selected from the training set. Figure 4-8(b) depicts the selected sub-regions on the face model. Note that in this training stage, the EER values are obtained by testing on the training set, which contains 33 face models from 15 subjects.

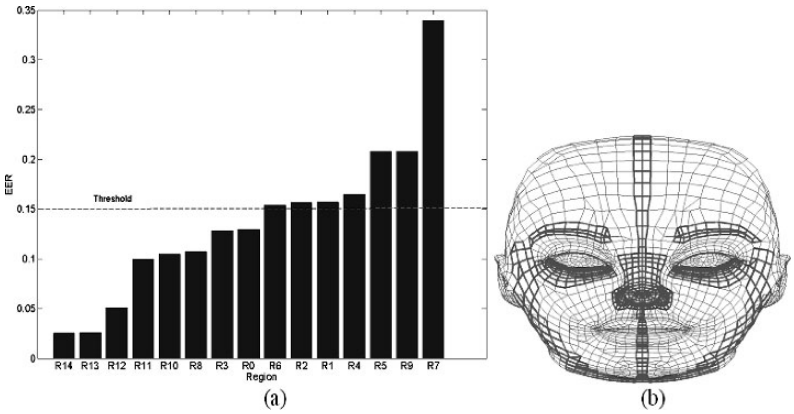


Figure 4-8. (a) EERs for 15 sub-regions, which are sorted in a non-decreasing order; (b) The optimized feature space (within the regions marked with red color) after performing the GA-based method on the training set. (See also Plate 12 in the Colour Plate Section)

We set the threshold at the EER value of 0.15 (i.e., the first eight-ranks) based on the mean of the fitness values. As a result, the first 8 sub-regions are selected, which are sub-regions 0, 3, 8, 10, 11, 12, 13 and 14 as shown in Figure *4-8*(a) and (b). The numbers of vertices selected from the sub-regions are: 15 vertices for R0; 45 for R3; 11 for R8; 13 for R10; 23 for R11; 42 for R12; 15 for R13 and 21 for R14. Since there are overlaps for different regions, the optimized feature space contains 137 vertices. Although using only one sub-region, R14 for instance, may achieve a lower EER rate, it might contain too few of vertices that makes it relatively sensitive to noises and dependent on the training data. Note that the optimized feature space represents the individual facial traits. Because the majority of these features are located in the ``static'' facial regions, the feature space is less influenced by the facial skin deformation caused by facial expressions.

6. FACE MODEL MATCHING

After selecting the optimal features in the previous training stage, the components (i.e., vertex labels) in the final feature space form a feature vector for each individual facial model. The similarity of two individual facial models can be measured by the similarity of their feature vectors.

The computation of feature similarities can be described as follows: given two feature vectors X and Y from two individual models, we can represent the feature vectors as $X = \{x_1, x_2, ..., x_m\}$ and $Y = \{y_1, y_2, ..., y_m\}$, where m is the number of vertices in the region. x_i and y_i (i=1,…,m) are the

labeled types (denoted from 1 to 8) for the corresponding vertex. To measure the similarity between two feature vectors, we can use different similarity measure.

The Euclidean distance and inner-product of two vectors could be the direct solutions to calculate the similarity score. Symmetric relative entropy[25] is also a common used method to compare the distance between two vectors. We used inner-product, symmetric relative entropy, and correlation-based similarity measurement for the test. Comparing these approaches, we found correlation-based approach shows the best property in classifying 3D face models using our selected feature space. Next, we will give a brief introduction of these approaches.

6.1 The inner product based metrics function

Inner product based similarity function treats X and Y as two vectors in a space, and uses the angle between these vectors as the similarity measurement. The definition is as follow:

$$f(X,Y) = \frac{X \cdot Y}{|X| \cdot |Y|} \tag{10}$$

Since the inner product based metric takes X and Y as independent vectors, it does not consider the statistics information in each vector set. Therefore, it is not suitable to distinguish models of different subjects. Figure *4-11* shows the similarity of different subjects using inner product metrics function. In Figure *4-9*, the 64 green-labelled rectangles represent 64

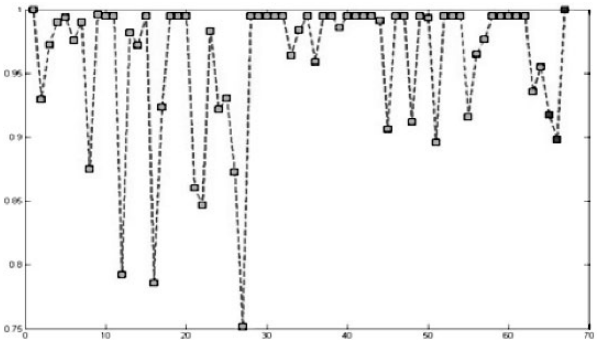


Figure 4-9. Similarity of different subjects using inner product metrics function.
(See also Plate 13 in the Colour Plate Section)

instance models of the same subject A, the three blue-labelled rectangles represent models of three different subjects, say B, C and D. We use the first instance model of A as the base model and calculate the similarity values between A and the other 67 models including the base model. From this figure, we are not able to separate instance of A from B, C or D.

6.2 Symmetric relative entropy

Symmetric relative entropy is also known as Kullback-Liebler distance, which is commonly used to compare the histograms of two vectors. Let $a(i)$ denote the probability of label "*i*" appearing in vector X, and $b(i)$ be the probability of label "*i*" appearing in vector Y. It is defined as

$$D(a,b) = ((a \,||\, b) + (b \,||\, a))/2 \tag{11}$$

Where

$$(a \,||\, b) \stackrel{\Delta}{=} \sum_{i=1}^{N} a(i) \cdot \log(a(i)/b(i)) \tag{12}$$

Since $D(a,b)$ is larger than or equal to 0, we can transform it to [0, 1] by the following equation

$$k(X,Y) = 2^{-\sqrt[4]{D(a,b)}} \tag{13}$$

Then, we can use k as a metric to measure the similarity of distributions between vector X and vector Y.

As shown in Figure *4-10*, this metric function is not able to distinguish the instance model of subject A with that of B, C, and D. It is not suitable for our similarity measurement.

6.3 Correlation based similarity measurement

Considering the statistical property of feature label space, we choose to use the correlation[26] to measure the similarity of two feature vectors (the experimental results of different measures is shown below). The usage of correlation-based similarity measurement is based on the observation that the facial surface primitive labels are highly correlated across the instances of the same individual. Since the inner-product function takes X and Y as independent vector sets and does not consider the statistics information in

each vector set, correlation may be a better measure function. The correlation coefficient ρ is calculated by

$$\rho(X,Y)=\frac{\sum_{i=1}^{m}(x_i-\bar{x})(y_i-\bar{y})}{\sqrt{(\sum_{i=1}^{m}(x_i-\bar{x})^2)\cdot(\sum_{i=1}^{m}(y_i-\bar{y})^2)}} \tag{14}$$

This correlation coefficient ρ satisfies the following conditions

1. $\rho(X,Y)=\rho(Y,X)$
2. $-1\le\rho\le 1$
3. $\rho(X,Y)=0$, if X and Y are independent

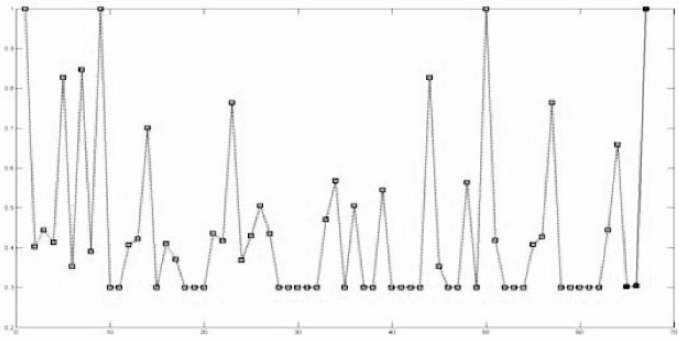


Figure 4-10. Similarity of different subjects using symmetric relative entropy (Kullback-Liebler distance). (See also Plate 14 in the Colour Plate Section)

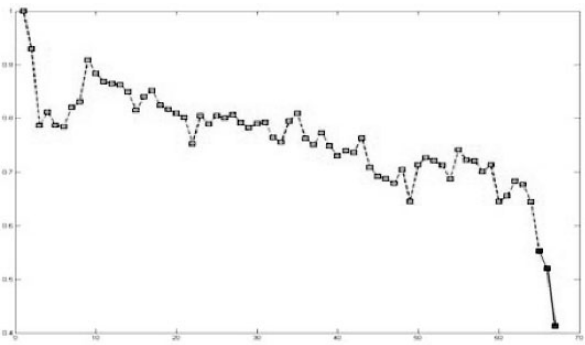


Figure 4-11. Similarity of different subjects using correlation coefficient function. (See also Plate 15 in the Colour Plate Section)

The high correlation coefficient of two feature vectors indicates the high similarity of two individual models. As a result, the individual models can be classified according to their correlation values. Figure *4-11* depicts the similarities among different subjects using the correlation function. As Figure *4-11* illustrates, the correlation coefficients allow us to distinguish the 64 instance models (green dots representing different expressions of a same subject) with the other three subjects (blue dots).

7. EXPERIMENTS AND ANALYSIS

7.1 Experiments on generated facial models

In our 3D face model database (see Section 2), there are 40 subjects with total 105 instance models (note that each subject has two or three generated instance models depending on two or three pairs of instance images available.) We chose 33 generated instance models from 15 subjects as a training set, and 72 generated models from 25 subjects as a test set. As described in Section 4, after the training stage, 137 feature vertices are finally included in the optimal feature space.

The performance of the 3D facial model classification is illustrated by a ROC curve, which is obtained by conducting the classification on the test set (72 instance models). As shown in Figure *4-12*(a), the EER is 9%. Figure *4-12*(b) shows the correct recognition rates using the score vs. rank curve. Our system can achieve almost a 92% correct recognition rate if the first candidates (rank=4) are selected.

We also applied the Principal Component Analysis (PCA) method on the optimal feature space to reduce the dimensionality of feature vector from 149 to 55. As seen from the results shown in Figure *4-13*, both the EER error (10% compared with 9%) and the rank-four correct recognition (87% compared with 92%) are degraded. However, the dimensionality reduction approach can reduce the computation load while keep the reasonably good performance in face classification.

In order to investigate how critical the facial expression could affect the feature space and how well our feature labels could represent the individual characteristics, we compared both the labeled models from different subjects and the models from different expressions of the same subject. Among our test set, the feature space difference between expression models of the same subject is much smaller than the difference between the different subjects.

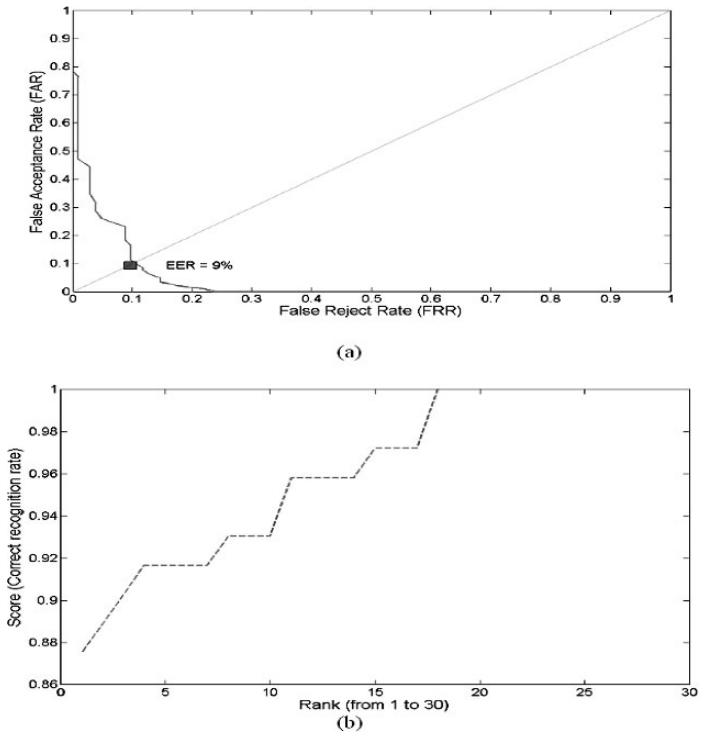


Figure 4-12. Experimental results: (a) ROC curve (marked with FAR, FRR, and EER = 9%) and (b) Ac-cumulative score vs. rank (The rank 1 score is 88%, and the rank 4 score is about 92%) using the optimal feature space. (See also Plate 16 in the Colour Plate Section)

Figure *4-14* shows one example with two different subjects (i.e., one subject has one instance model (a) and the other has two instance models (b-c) with two different expressions). As seen, in the nose region (marked with a black box), the distribution of feature labels has a distinct pattern in (a) as compared to the patterns in (b) and (c). The label compositions of (b) and (c) are very similar although the nose shapes have been changed because of the smiling expression. This suggests that without large expressional variations, the similarity between models of the same subject (inner-class) is higher than the similarity between models of different subjects (intra-class). The reason for that lies in (1) the surface labeling at each vertex location is stable to the surface change if the expression is performed in a non-exaggerate fashion and (2) the set of good features selected by our system are less affected by the expression change.

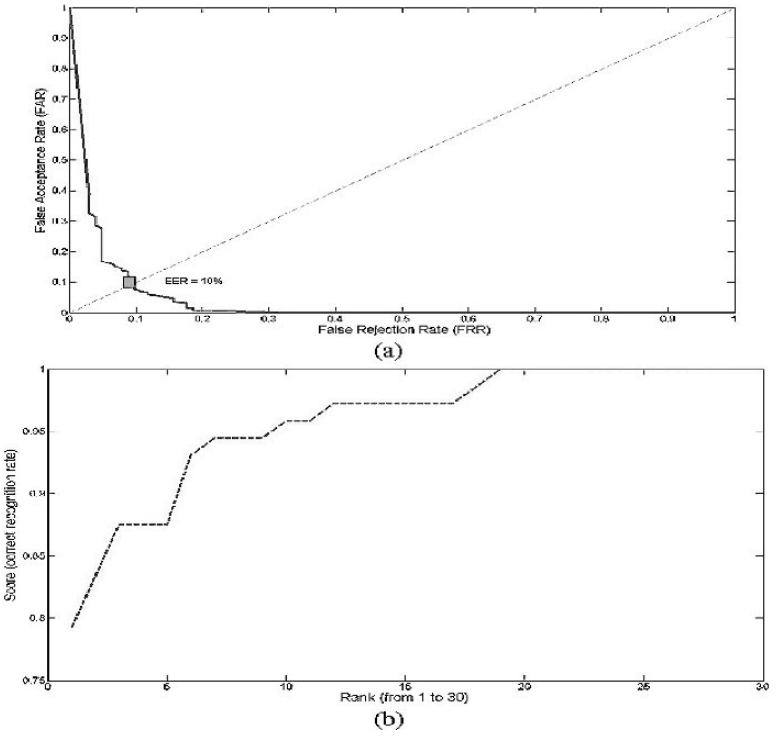


Figure 4-13. Experimental results using the transformed feature space via PCA decomposition. (a) ROC curve (marked with FAR, FRR, and EER = 10%) and (b) Accumulative score vs. rank curve (The rank 4 score is about 87%). (See also Plate 17 in the Colour Plate Section)

7.2 Applicability to 3D range data

The obtained facial feature space has been applied to the generic model based individual facial surfaces. We further investigate the applicability of such features to the other 3D data, such as range data from 3D imaging systems. To verify it, we applied the subset of obtained optimal feature space from the generic model on the real 3D range data and conducted a comparison study.

A set of 3D range facial models are generated using a 3Q imaging system[27]. The data set includes 105 range models from 15 subjects. Each subject has various facial expressions. Figure *4-15* shows one instance of one subject with anger expression. To apply the optimal feature space on the 3D range data facial surface, we need to adapt the generic model to the 3D range data. We adapted the surface onto the facial region below the eyebrows area (including the eyebrows).

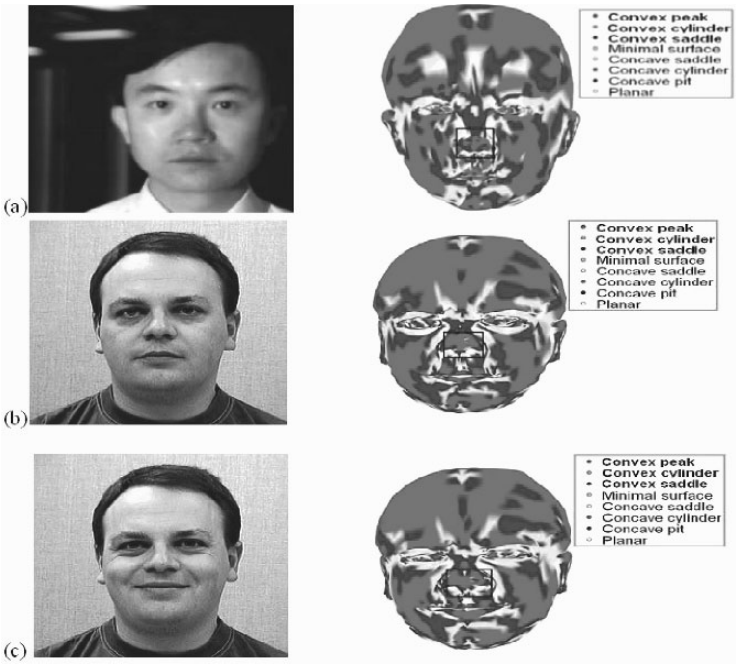


Figure 4-14. (a) Example of two subjects with three instance models: In order to better show the label distribution, we render the labels on both vertices and polygons by using linear interpolation. The labeled models are then displayed in a continuous mode. The black box delineates the nose feature region. (See also Plate 18 in the Colour Plate Section)

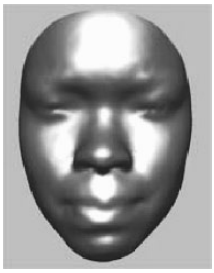


Figure 4-15. An instance of the 3D facial range models.
(See also Plate 19 in the Colour Plate Section)

The reasons to use the sub-set of the feature space are twofold: (1) due to the limitation of current imaging system, facial hair (e.g., dark area in front of the forehead) is hard to reconstruct; (2) the selected facial region covers the most of the facial region, which represent the characteristics of each individual face. To extract the features on the range models, we map the

generic model onto the 3D range model. To do so, we used 83 key points on the generic model to control the adaptation (referred in Sun and Yin[30]). After mapping the feature points from the generic model to the range model, we derived 101 feature points (a subset of 137 feature points used before) on the range model using the interpolation technique.

We applied the same set of feature points to both the generic models and the range models, and conducted experiments for face classification. For the wire-frame models generated from our two-view based modeling system, the rank-four correct recognition rate is about 91%. For the range models obtained from the 3D imaging system, the rank-four correct recognition rate is about 87%.

8. CONCLUSIONS AND FUTURE WORK

We proposed a system for recognizing human faces based on the 3D model labeling technique. The generated 3D facial models derived from two facial views and the label-based feature space has shown to perform well for characterizing the individuals' 3D features. The system achieved a 92% correct recognition rate at the fourth rank and 9% equal error rate when 105 models were tested. The proposed work has certain existing implications and practices: (1) image pairs (frontal and profile views) are the most commonly used data source for personal records, which are available from existing police and federal government databases. (2) The setup of two surveillance video cameras (front and profile) to simultaneously capture two views of a face is feasible in many public sites, such as security entrances and check-in points or airports and federal government buildings, where each individual must pass through a security gate or check point one-by-one. The performance of our system relies on the quality of the reconstructed 3D models. Low quality of input images will degrade the accuracy of the individual model representation, and may increase the possibility of misclassification. To verify the feasibility of our optimal feature space using the GA approach, we implemented further experiments to apply the feature space on the real 3D range data, and the experimental results show the rank-four correct recognition rate is about 87%. This shows the applicability of the derived optimal feature space for 3D face recognition tasks. Our future work is to design a better feature selection algorithm, by incorporating multiple feature descriptors combined with normal maps, curvature maps, and label maps together and by using a multi-classifier strategy to enhance the system performance.

Our future work will also include the expansion of the existing data set and conduct an intensive test for the data obtained under variable imaging

conditions (e.g., various face sizes, variable lighting conditions and poses, etc.). We plan to compare our system with some other range data based systems (e.g., Phillips[29]) to refine the feature space in order to further improve the recognition performance.

ACKNOWLEDGEMENT

We would like to thank the National Science Foundation for the support of this work under grants IIS-0414029 and IIS-0541044, and the support from NYSTAR's James D. Watson program. We would also like to thank Dr. Jonathan Phillip for the FERET database and Dr. Josef Kittler for the XM2VTS data set for our experiment.

REFERENCES

1. X. Lu, R. Hsu, A.K. Jain, B. Kamgar-parsi, and B. Kamgar-parsi, Face Recognition with 3D Model-based Synthesis, *ICBA*, 2004, pp. 139-146.
2. J. Juang, V. Blanz, and B. Heisele, Face Recognition with Support Vector Machines and 3D Head Models, *International Workshop on Pattern Recognition with Support Vector Machines (SVM2002) ,* 2002, Niagara Falls, Canada. pp. 334-341.
3. A. Moreno, A. Sanchez, A. F. Velez, and F. J. Diaz, Face recognition using 3D surface-extracted descriptors, *Irish Machine Vision and Image Processing Conference, 2003*.
4. T. Heseltine, N. Pears, and J. Austin, Three Dimensional Face Recognition: An Eigensurface Approach, http://www-users.cs.york.ac.uk/~tomh/.
5. V. Blanz and T. Vetter, Face Recognition Based on Fitting a 3D Morphable Model, *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 2003, **25**(9), pp. 1063-1074.
6. K. Chang, K. W. Bowyer, P. J. Flynn, X. Chen, Multi-biometrics Using Facial Appearance, Shape, and Temperature, *IEEE International Conference on Face and Gesture Recognition*, 2004.
7. T. Heseltine, N. Pears, and J. Austin, Three Dimensional Face Recognition: A Fishersurface Approach, *IEEE International Conference of Image Processing*, 2004.
8. C. Zhang and F. Cohen, 3-D face structure extraction and recognition from images using 3-D morphing and distance mapping, *IEEE Transactions on Image Processing*, 2002, pp. 1249-1259.
9. A. M. Bronstein, M. M. Bronstein, and R. Kimmel, Expression Invariant 3D Face Recognition, *Proceedings of 4th International Conference on VBPA,* 2003.
10. A. Godil, S. Ressler, and P. Grother, Face Recognition using 3D surface and color map information: Comparison and Combination, *SPIE's symposium on Biometrics Technology for Human Identification*, Orlando, FL, April 12-13, 2004.
11. W. Zhao, R. Chellappa, P.J. Phillips, and A. Rosenfeld, Face Recognition: A Literature Survey, *ACM Computing Surveys*, **35**(4), pp. 399-458, 2003.
12. P.J. Phillips, H. Moon, S. A. Rizvi, and P. J. Rauss, The FERET Evaluation Methodology for Face-Recognition Algorithms, *IEEE Trans. on PAMI,* 2000, **22**(10).

13. G. Gordon, Face Recognition Based on Depth and Curvature Features, *IEEE International Conference on Computer Vision and Pattern Recognition, 1*992, pp. 808-810.
14. R. Hsu and A. Jain, Face Modeling for Recognition, *International Conference on Image Processing (ICIP)*, Greece, October 7-10, 2001, pp. 693-696.
15. A. Ansari and A. Abdel-Mottaleb, 3-D Face Modeling Using Two Views and a Generic Face Model with Application to 3-D Face Recognition, *IEEE Conference on AVSS'03*.
16. H. Ip and L. Yin, Constructing a 3D Individualized Head Model from Two Orthogonal Views, *The Visual Computer*, 12 , Springer-Verlag, (1996), pp. 254-266.
17. A. Yuille, P. Hallinan, and D. Cohen, Feature Extraction from Faces Using Deformable Templates, *International Journal on Computer Vision*, **8**(2), pp. 99-111, 1992.
18. L. Yin and A. Basu, Generating realistic facial expressions with wrinkles for model based coding, *Computer Vision and Image Understanding*, **84**(11), pp. 201-240, 2001.
19. The FERET Database, http://www.itl.nist.gov/iad/humanid/feret/feret_master.html .
20. The XM2VTS Database, http://www.ee.surrey.sc.uk/Research/VSSP/xm22vtsdb/ .
21. H. Tanaka, M. Ikeda, and H. Chiaki, Curvature-based face surface recognition using spherical correlation, *IEEE Intern. Conference on Face and Gesture Recognition*, 1998, pp. 372-377.
22. J. Goldfeather and V. Interrante, A Novel Cubic-Order Algorithm for Approximating Principal Direction Vectors, *ACM Transactions on Graphics*, **23**(1), pp. 45-63, 2004.
23. G. Van Dijck, M. M. Van Hulle, and M. Wevers, Genetic Algorithm for Feature Subset Selection with Exploitation of Feature Correlations from Continuous Wavelet Transform: a real-case Application, *International Journal of Computational Intelligence*, **1**(1), 2004, pp. 1-12.
24. D. Goldberg, Generic and Evolutionary Algorithms Come of Age, *Communications of the ACM*, **37**(3), 1994, pp. 113-119.
25. J. T. Foote, A Similarity Measure for Automatic Audio Classification, *Proc. of AAAI Spring Symposium on Intelligent Integration and Use of Text, Image, Video and Audio Corpora*, 1997.
26. http://www.nvcc.edu/home/elanthier/methods/correlation.htm
27. http://www.3dmd.com
28. J. Czyz, J. Kittler, and L. Vandendorpe, Combining face verification experts, *Proc. International Conference on Pattern Recognition*, Vol. **2**, 2002, pp. 28-31.
29. P. J. Phillips, P.J. Flynn, T. Scruggs, K. W. Bowyer, J. Chang, K. Hoffman, J. Marques, J. Min, and W. Worek, Overview of the Face Recognition Grand Challenge, *In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2005.
30. Y. Sun and L. Yin, Evaluation of 3D Face Feature Selection, *Technical Report,* Computer Science Department, SUNY at Binghamton, 2005.

Chapter 5

STORY OF CINDERELLA

Biometrics and Isometry-Invariant Distances

Alexander M. Bronstein, Michael M. Bronstein and Ron Kimmel
Department of Computer Science, Technion – Israel Institute of Technology, Haifa 32000, Israel

Abstract: In this chapter, we address the question of what are the facial measures one could use in order to distinguish between people. Our starting point is the fact that the expressions of our face can, in most cases, be modeled as isometries, which we validate empirically. Then, based on this observation, we introduce a technique that enables us to distinguish between people based on the intrinsic geometry of their faces. We provide empirical evidence that the proposed geometric measures are invariant to facial expressions and relate our findings to the broad context of biometric methods, ranging from modern face recognition technologies to fairy tales and biblical stories.

Key words: biometrics, isometry, face recognition, facial expression, multidimensional scaling, intrinsic geometry.

1. INTRODUCTION

Most of us are familiar with the story of the handsome prince who declares that he will marry the girl whose foot fits into the glass slipper she lost at his palace. The prince finally finds Cinderella by matching the slipper to her foot, and they live happily ever after. This is how Charles Perrault's story ends. One of the stepsisters of the German Cinderella, according to the Brothers Grimm version, succeeds in fitting her foot into the slipper by cutting off a toe, in an attempt to create a fake biometric signature. Cinderella's stepsister was not the first – it appears that identity frauds date back to biblical times. In Genesis, Jacob stole his father's blessing, the privilege of the elder, by pretending to be his firstborn brother Esau. By

A. Koschan et al. (eds.), 3D Imaging for Safety and Security, 119–131.

hand scan Isaac wrongly verified his sons Esau's identity, since smooth-skinned Jacob wrapped kidskin around his hands to pose as his brother. Based on face recognition, another biometric technology, Little Red Riding Hood makes the unavoidable conclusion that it was the wolf she was talking to rather than her grandmother.

With this example, we leave the fairy-tale world and enter into the realm of modern biometric technologies. In this chapter, we focus on the problem of three-dimensional face recognition, though the approach we introduce is general and can be applied to any non-rigid surface comparison problem under reasonable assumptions. The main questions we will try to answer is what are facial measures we could use in order to distinguish between people and how could we use them for that goal.

2. UBIQUITOUS ISOMETRIES

Recently, a team of French surgeons has reconstructed the face of a woman by transplanting donor tissues. This remarkable operation raised controversial questions regarding the identity of the patient: will she recover the lost identity or look like the donor? Apparently, the lady's face has preserved its original features: though the skin tone may have changed, the geometry of the patient's face remained (at least partially) more or less intact. The reason is that the rigid structure of the scull was unaltered, which preserved the form of the overlaying tissues. Obviously, the geometry of the face reflects important information; uniquely describing our identity.

At this point, the term "geometry" requires a more accurate definition. Due to the fact that Nature provides us with rich facial expressions, our face undergoes complicated non-rigid deformations. Emotions may drastically change the way the facial surface is embedded in the ambient three-dimensional Euclidean space. Such changes are called *extrinsic*. Clearly, the extrinsic geometry is not preserved by facial expressions. Yet, if we restrict our measurements of distance to the facial surface, we notice that distances measured on the surface (that is, the lengths of the shortest paths on the surface, referred to as geodesic distances) remain almost unaltered. This happens due to the fact that our skin and underlying tissues exhibit only slight elasticity: they can be bent but not too much stretched.

In order to validate this claim, we marked tens of fiducial points on a subject's face and scanned its geometry under various expressions (see Figure *5-1*). We then compared the absolute change in both geodesic and Euclidean distances between the points. Figure *5-2* demonstrates the result of this experiment. Although there is some change in the geodesic distances between corresponding points in different expressions, considering the 3D

scanner accuracy, these distances are approximately preserved. Geodesic distances exhibit smaller variability compared to Euclidean ones, as depicted in Figure *5-2*.

Geometric quantities that can be expressed in terms of geodesic distances are referred to as the *intrinsic geometry* and appear to be insensitive to facial expressions. Consequently, facial expressions can be modeled as near-isometric deformations of the face, i.e. such deformations that approximately preserve the distances on the surface. Stated differently, the intrinsic geometry reflects the subject's identity, whereas the extrinsic geometry is the result of the facial expression.

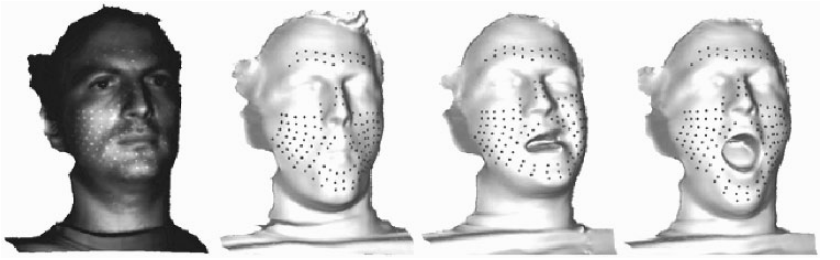


Figure 5-1. Isometric model validation experiment. Left: facial image with the markers. Right: example of one moderate and two strong facial expressions with marked reference points.

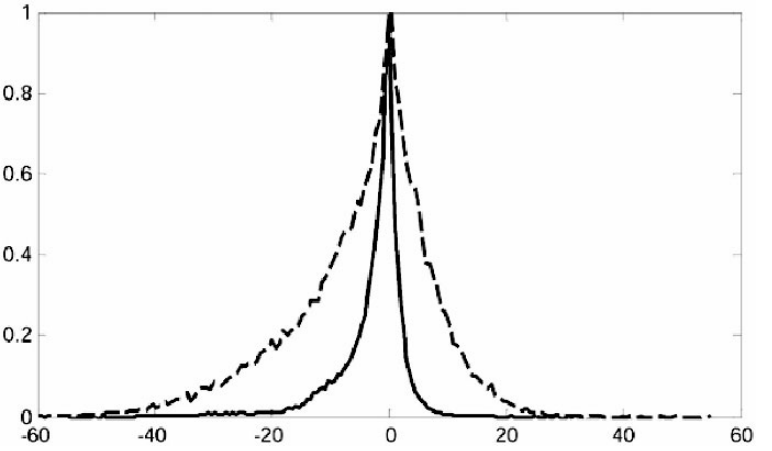


Figure 5-2. Histogram of geodesic distance deviation from the isometric model (solid); for comparison, a histogram for the Euclidean distances is shown (dashed).

In order to compare between faces in a way insensitive to facial expressions, we have to find invariants that uniquely represent the intrinsic geometry of a surface, without being affected by its extrinsic geometry. We model the two faces that we would like to compare as smooth Riemannian surfaces S and Q, with the geodesic distances, d_S and d_Q, respectively. In what follows, we devise computational methods for the comparison of intrinsic geometric properties of two faces.

3. FLAT EMBEDDING AND CANONICAL FORMS

Our first attempt of expression-invariant face recognition was based on replacing the intrinsic geometry of the surface by a Euclidean one by a process known as *isometric embedding*. The first use of this ideas in computer vision dates back to Schwartz et al.[1], who tried to analyze the brain cortical surface by embedding it into the plane. Revisiting the ingredients of this method, Zigelman et al.[2] introduced a texture mapping procedure, in which the geodesic distances are computed with a numerically consistent efficient scheme[3].

Embedding into the plane can be thought of as an invariant parameterization of the surface. However, in order to compare between intrinsic properties of two surfaces, the dimension of the embedding space has to be at least three, or, using a fancier terminology, the *co-dimension* has to be at least one. Elad and Kimmel[4] proposed to embed the metric structure of the two dimensional surface S into $\mathbf{R}^n$, where $n \geq 3$. Formally, we try to find a map $\varphi: S \to \mathbf{R}^n$, such that $d_{\mathbf{R}^n}(\varphi(s_1), \varphi(s_2)) = d_S(s_1, s_2)$ for every $s_1, s_2 \in S$. Such a map is called isometric embedding. However, for a general non-flat surface, a truly isometric embedding usually does not exist; all we can find is a *minimum-distortion embedding*.

Practically, the surface is sampled at a set of m points $\{s_1, ..., s_m\}$, and we find a configuration of points $\{x_1, ..., x_m\}$ in $\mathbf{R}^n$ by solving the following optimization problem,

$$\{x_1, ..., x_m\} = \arg \min_{x_1, ..., x_m} \sum_{i<j} (d_{\mathbf{R}^n}(x_i, x_j) - d_S(s_i, s_j))^2. \quad (1)$$

Here, $x_i = \varphi(s_i)$ are the images of the samples of S under the embedding φ. We try to find such a configuration of points that the Euclidean distances $d_{\mathbf{R}^n}$ between each pair of image points is as close as possible to their corresponding original geodesic distances, d_S. A numerical

procedure solving the above optimization problem is known as *multidimensional scaling* (MDS).

$\{x_1,...,x_m\}$ can be thought of as an approximation of the intrinsic properties of S. We call it the *canonical form* of the surface. The comparison of canonical forms is a significantly simpler task than comparison of the intrinsic geometries of the non-rigid surfaces themselves. Indeed, for canonical forms there is no difference between extrinsic and intrinsic geometries. Unlike the rich class of non-rigid isometric deformations the original surfaces can undergo, the only degrees of freedom for the canonical forms are the rigid transformations (translation, rotation and reflection), which can be easily solved for using efficient rigid surface matching methods such as the iterative closest point (ICP) algorithm[6, 7] or moments signatures[4,5]. The latter method is especially attractive, since it produces a simple signature describing the geometry of the canonical form that can be efficiently compared to a large data base of signatures representing other faces. Canonical forms cast the original problem of non-rigid surface matching to the simpler problem of rigid surface comparison. Figure *5-3* depicts faces with various expressions embedded into $\mathbf{R}^3$ by the described procedure. Note how even strong expressions of the same subject have just little influence on the canonical forms.

Based on this approach, we built a prototype face recognition system that achieved sufficient accuracy to tell apart identical twins, even in the presence of extreme facial expressions[8]. Nevertheless, the canonical form approach is limited in two aspects. First, the inevitable distortion introduced by the embedding sets an artificial threshold to the sensitivity of the method. Second, in order to perform an accurate matching, the support of the surfaces S and Q must be the same. For that purpose, a pre-processing by means of a consistent cropping of S and Q is required. Changing the surface support regions would generally yield different canonical forms, an undesired property, since in many practical applications, matching of partially missing or partially overlapping surfaces is required.

4. SPHERICAL CANONICAL FORMS

In order to reduce the distortion of embedding facial surfaces into a Euclidean space, we should search for better spaces than the Euclidean ones. A simple space with non-Euclidean geometry, in which the geodesic distances are given analytically is the *n*-dimensional sphere $\mathbf{S}^n$. There exist almost straightforward generalizations of the MDS methods suitable for embedding into $\mathbf{S}^n$. Given the control over the sphere radius R, the spherical geometry constitutes a richer choice, since it includes the Euclidean case at

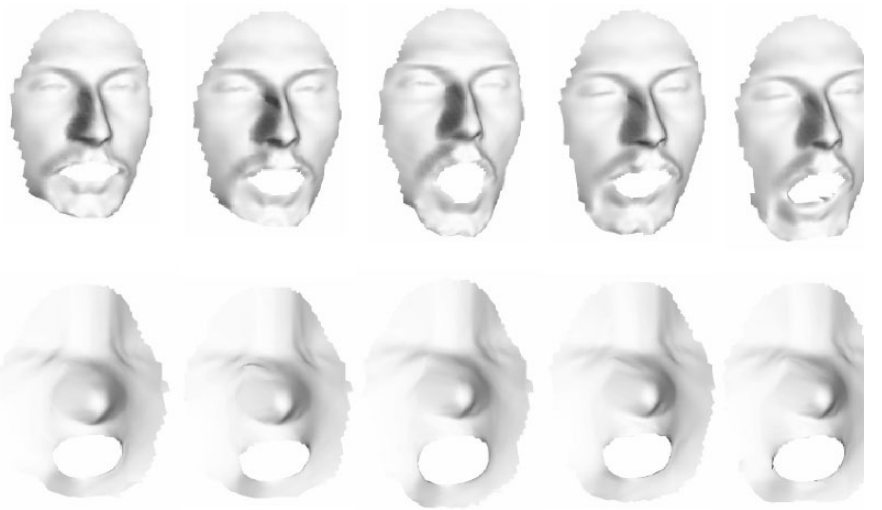


Figure 5-3. Facial expressions (top) and their corresponding canonical forms (bottom).

the limit $R \to \infty$. Once embedded into a sphere, the spherical canonical forms have to be matched. For that goal we developed various tools that can be found in Bronstein et al.[9]

Figure *5-4* demonstrates that embedding into spherical spaces has smaller distortions for some range of radii similar to the radius of an average human face. Moreover, the recognition rates exhibit a clear correlation with the embedding distortion: the lower is the distortion; the more accurate is the recognition. This gives an empirical justification to the pursuit of better embedding spaces.

Although there is an improvement in the recognition rates, the spherical embedding is not the end of our journey. We are still occupied with the problem of partial matching and that of the unavoidable embedding distortions even when selecting a somewhat more favorable embedding space.

5. GENERALIZED MULTIDIMENSIONAL SCALING

Replacing the Euclidean geometry of the embedding space by the spherecal one usually leads to smaller metric distortions and, consequently, to better isometry-invariant representation of surfaces, while maintaining

practically the same computational complexity compared to the Euclidean MDS algorithm. Nevertheless, spherical embedding cannot completely avoid the distortion.

It is, however, possible to eliminate the need of intermediate space by choosing one of the surfaces, say Q, as the embedding space. In other words, we intend to embed S directly into Q. The embedding can be achieved by solving an MDS-like problem,

$$\{q_1,...,q_m\} = \arg \min_{q_1,...q_m \in Q} \sum_{i>j} (d_S(s_i,s_j) - d_Q(q_i,q_j))^2, \quad (2)$$

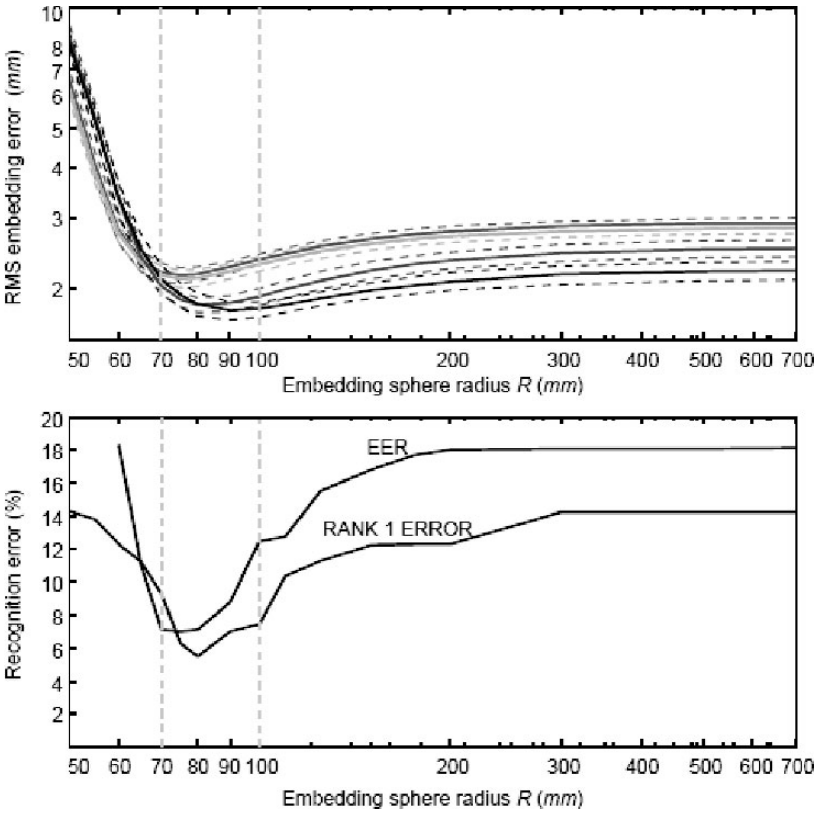


Figure 5-4. First row: embedding error versus the embedding sphere radius for four different subjects (colors denote different subjects, dashed lines indicate 95% confidence intervals). Second row: Equal-error and rank-1 error rates versus the embedding sphere radius. The asymptote $R \to \infty$ corresponds to embedding into $\mathbf{S}^n$.
(See also Plate 20 in the Colour Plate Section)

that we term as the *generalized multidimensional scalin*g or GMDS for short [10, 11]. As in the standard MDS procedure, we find a configuration of points $\{q_1,...,q_m\}$ on the surface Q, that represents the intrinsic geometry of S as accurately as possible. The points q_i are the images of s_i under the embedding $\varphi: S \to Q$. The minimum achievable value of the cost function in (2) quantifies how much the metric of S has to be distorted in order to fit into Q. If the two surfaces are isometric, such an embedding will be distortion-less; otherwise, the distortion will measure the dissimilarity between S and Q. This dissimilarity is related to the *Gromov-Hausdorff distance*, first used in the context of the surface matching problem by Mémoli and Sapiro[12].

So far, the embedding distortion has been an enemy that was likely to lower the sensitivity of the canonical form method; now it has become a friend that tells us how different the surfaces are. For this reason, GMDS is essentially the best non-Euclidean embedding, in the sense that it allows to completely avoid unnecessary representation errors stemming from embedding into an intermediate space. Strictly speaking, we do not use canonical forms anymore; the measure of similarity between two surfaces is obtained from the solution of the embedding problem itself.

Another important advantage of GMDS is that it allows for local distortion analysis. Indeed, defining the *local distortion* as

$$\sigma_i = \sum_j \left| d_S(s_i, s_j) - d_Q(q_i, q_j) \right|^2, \tag{3}$$

we create a map $\sigma: S \to Q$ quantifying the magnitude of the change the metric of S undergoes in every point in order to be embedded into Q (Figure 5-5). Practically, it allows us to determine how much two faces are dissimilar, and also identify the regions with the largest dissimilarities. Last, GMDS enables *partial matching* between non-rigid surfaces, that is, matching a part of S to Q. Partial matching is of paramount importance in practical applications, where due to the limitations of physical acquisition devices, parts of the facial surface may be occluded.

Although GMDS looks like a powerful instrument for isometry-invariant surface matching, there is some cost for its advantages. First, unlike the Euclidean or the spherical cases, we gave up the luxury of computing the distance in the embedding space analytically. Nevertheless, geodesic distances on arbitrarily complex surfaces can be efficiently approximated.[10, 11] The overall complexity of GMDS is comparable to that of the standard MDS algorithms. Another, shortcoming stems from the fact that every time we need to compare between two faces, we have to solve a new embedding problem.

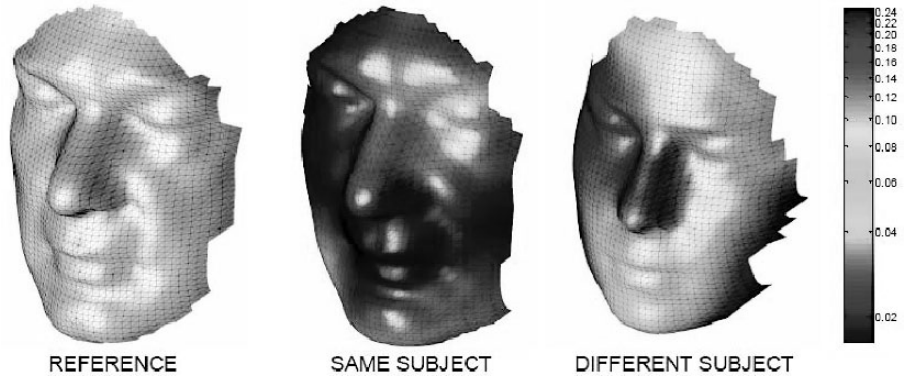


Figure 5-5. Local distortion map obtained by embedding two faces of two different subjects into a face of a reference subject. (See also Plate 21 in the Colour Plate Section)

This makes one-to-many comparison scenarios with large databases improbable. A hierarchical matching strategy or a combination of GMDS with the canonical form approach provides some remedy to this difficulty.

6. COMPARISON OF PHOTOMETRIC PROPERTIES

So far, we focused our attention on the recognition of the intrinsic facial geometry that appeared to be insensitive to expressions. However, our face is also endowed with photometric characteristics that provide additional useful information for the recognition task. A somewhat simplified model incorporating both geometric and photometric properties of the face consists of a non-rigid surface S and a scalar field $\rho : S \to [0,1]$ associated with it. The scalar field ρ measures the *reflectance coefficient* or *albedo* at each point on the surface, that is, the fraction of the incident light reflected by the surface. If the acquisition device is capable of sensing multiple color channels, ρ can be replaced by a vector field (for example, $\rho : S \to [0,1]^3$ in case of a standard tri-chromatic camera) measuring the reflectance coefficient for different light wave lengths. Using the computer graphics jargon, ρ is the *texture* of S.

In practice, the albedo cannot be directly measured by a camera; what we observe is the *brightness* of the surface, or, in simple words, the amount of radiation scattered from it in the camera direction. However, it appears that our skin behaves approximately like a diffusive reflector, which means that its apparent brightness is roughly the same regardless of the observer's

viewing direction. This fact allows using the Lambertian reflection law to estimate the reflectance coefficient ρ given the surface normal field. Clearly, such information is unavailable in two-dimensional face recognition methods, which are based on the brightness image of the face.

In this setting, the problem of expression-invariant face recognition aims at measuring the similarity of two faces, based on the similarity of their intrinsic geometries (S, d_S) and (Q, d_Q), and their photometric properties, ρ_S and ρ_Q. However, in order to be able to compare between ρ_S and ρ_Q, we have to bring them first to some common coordinates system, in which the facial features coincide.

Following the steps we took for comparing the intrinsic geometry; let us briefly go through the same evolution for the texture. Common coordinates can first be found by a common parameterization of S and Q into some planar domain. Such a parameterization should be invariant to facial expressions, which according to our isometric model is not influenced by the extrinsic geometry of the surface. After the surfaces S and Q are re-parameterized, ρ_S and ρ_Q can be represented in the common parameterization domain that makes the comparison trivial using standard image matching techniques. Expression-invariant comparison of the photometric properties of the faces therefore reduces to finding an isometry-invariant "canonical" parameterization of the facial surfaces.

The simplest way to construct such a parameterization is by embedding the surface into $\mathbf{R}^2$ using an MDS algorithm. The problem is very similar to the computation of the canonical form, except that now the embedding space, serving as the parameterization domain, is restricted to be two-dimensional. We refer to such an embedding as *zero co-dimensional*. As the embedding is based on the intrinsic geometry only, such a parameterization will be invariant to isometries, and consequently, the reflectance image in the embedding space will be insensitive to facial expressions. We term such an image the *canonical image* of the face. However, recall that the embedding into $\mathbf{R}^2$ is defined up to rigid isometry, implying that the canonical images can be computed up to planar rotation, translation and reflection, which has to be resolved. Also, the inevitable distortion of the metric introduced by the embedding into a plane makes the canonical image only approximately invariant.

A partial fix for the latter problem comes from non-Euclidean embedding, for example, into the two-dimensional sphere $\mathbf{S}^2$. Since a face is more similar to a sphere than to a plane, spherical embedding produces canonical images with lower distortion. A clear relation between better representation and better recognition is observed again[11]. Another advantage of the spherical embedding is that the obtained spherical canonical images

(Figure 5-6) can be represented using a signature of the spherical harmonic coefficients, that are known to be invariant to rigid isometries on $\mathbf{S}^2$. A property analogous to the translation invariance of the magnitude in the Fourier transforms.

All the approaches described so far provide only approximate isometry-invariance, since a fixed embedding space implies necessarily embedding distortion. As an alternative, we can resort yet again to using the GMDS for embedding S into Q. In addition to quantifying the similarity of the two intrinsic geometries, the minimum-distortion embedding $\varphi: S \to Q$ would also bring ρ_S and ρ_Q to the same coordinates system in Q. The photometric distance between ρ_Q and $\rho_S \circ \varphi$, measured either locally or globally, provides additional information about the similarity of the two faces. Such an approach is inherently metric distortion-free and naturally allows for partial comparison of both photometric and geometric information.

7. CONCLUSIONS

We started with fairy tales, and like most fairy-tales, we are at the happy ending part of our story. We hope the reader found the plot illuminating and rewarding. We first claimed that our face can be described by the isometric model and validated this claim empirically. We studied a simple isometry invariant signature obtained by replacing the intrinsic geometry by a Euclidean one. We applied this process in a prototype face recognition system, which was able to distinguish between identical twins (the first two authors). This was just the beginning of our journey. Soon after, we noted that embedding into non-Euclidean spaces provides smaller embedding errors and consequently better recognition rates.

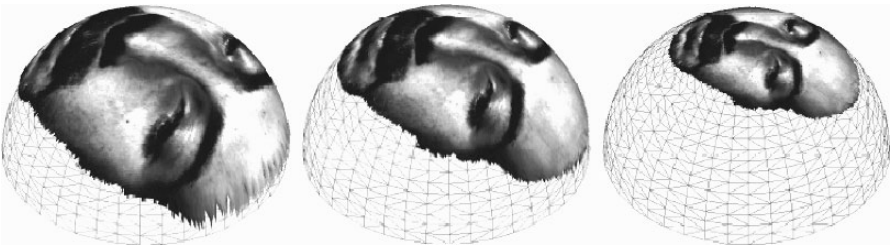


Figure 5-6. Canonical images of a face in $\mathbf{S}^2$ with different radii.

In both cases, the numerical tools we used for the embedding are members of the well known family of multidimensional scaling algorithms. The next step was to eliminate the embedding error altogether by harnessing it to our needs. We utilized the minimal distortion of embedding one surface into another as a measure of similarity between the two surfaces. The new numerical tool, used for solving the embedding problem, is a generalized MDS procedure.

In this chapter, we looked through a keyhole at the world of metric geometry, where objects are non-rigid and isometries introduce new challenges into our life. This is a new playground for engineers and it conceals numerous enigmae and problems waiting to be solved. Our goal was to let the reader touch some of these problems. The methods we discussed could one day help each of us find his Cinderella, or at least buy slippers of the right shape...

REFERENCES

1. E. L. Schwartz, A. Shaw, and E. Wolfson, "A numerical solution to the generalized mapmaker's problem: flattening nonconvex polyhedral surfaces," *IEEE Trans. PAMI*, 11:1005–1008, 1989.
2. G. Zigelman, R. Kimmel, and N. Kiryati, "Texture mapping using surface flattening via multi-dimensional scaling," *IEEE Trans. Visualization and Computer Graphics*, 9(2):198–207, 2002.
3. R. Kimmel and J. A. Sethian, "Computing geodesic paths on manifolds," *PNAS*, 95(15):8431–8435, 1998.
4. A. Elad and R. Kimmel, "On bending invariant signatures for surfaces," *IEEE Trans. PAMI*, 25(10):1285–1295, 2003.
5. M. Elad, A. Tal, and S. Ar, "Content based retrieval of VRML objects-an iterative and interactive approach," *EG Multimedia*, 39:97–108, 2001.
6. Y. Chen and G. Medioni, "Object modeling by registration of multiple range images," *Proc. IEEE Conference on Robotics and Automation*, 1991.
7. P. J. Besl and N. D. McKay, "A method for registration of 3D shapes," *IEEE Trans. PAMI*, 14:239–256, 1992.

8. A. M. Bronstein, M. M. Bronstein, and R. Kimmel, "Three-dimensional face recognition," *International Journal of Computer Vision (IJCV)*, 64(1):5–30, August 2005.
9. A. M. Bronstein, M. M. Bronstein, and R. Kimmel, "Expression-invariant representations of faces," *IEEE Trans. Imag. Proc.*, 16(1):188-197, 2007.
10. A. M. Bronstein, M. M. Bronstein, and R. Kimmel, "Generalized multidimensional scaling: a framework for isometry-invariant partial surface matching," *PNAS*, 103: 1168–1172, 2006.
11. A. M. Bronstein, M. M. Bronstein, and R. Kimmel, "Efficient computation of isometry-invariant distances between surfaces," *SIAM Journal on Scientific Computing*, 28(5): 1812-1836, 2006.
12. F. Mémoli and G. Sapiro, "A theoretical and computational framework for isometry invariant recognition of point cloud data," *Foundations of Computational Mathematics*, 5(3):313-347, 2005.
13. A. M. Bronstein, M. M. Bronstein, and R. Kimmel, "Expression-invariant face recognition via spherical embedding," *Proc. IEEE International Conf. Image Processing (ICIP)*, Vol. 3, 756–759, 2005.

Chapter 6

HUMAN EAR DETECTION FROM 3D SIDE FACE RANGE IMAGES

H. Chen and B. Bhanu
Center for Research in Intelligent Systems, University of California, Riverside, CA 92521, USA, {hui, bhanu}@cris.ucr.edu

Abstract: Ear is a new class of relatively stable biometrics which is not affected by facial expressions, cosmetics and eye glasses. To use ear biometrics for human identification, ear detection is the first part of an ear recognition system. In this chapter we propose two approaches for locating human ears in side face range images: (a) template matching based ear detection and (b) ear shape model based detection. For the first approach, the model template is represented by an averaged histogram of shape index that can be computed from principal curvatures. The ear detection is a four-step process: step edge detection and thresholding, image dilation, connect-component labeling and template matching. For the second approach, the ear shape model is represented by a set of discrete 3D vertices corresponding to ear helix and anti-helix parts. Given a side face range image, step edges are extracted and then the edge segments are dilated, thinned and grouped into different clusters which are the potential regions containing an ear. For each cluster, we register the ear shape model with the edges. The region with the minimum mean registration error is declared as the detected ear region; during this process the ear helix and anti-helix parts are identified. Experiments are performed with a large number of real side face range images to demonstrate the effectiveness of the proposed approaches.

Key words: ear biometrics; ear detection; range images; shape model; shape index.

1. INTRODUCTION

Ear is a viable new class of biometrics since ears have desirable properties such as universality, uniqueness and permanence[1, 2]. The ear has certain

A. Koschan et al. (eds.), 3D Imaging for Safety and Security, 133–155.

advantages over other biometrics. For example, ear is rich in features; it is a stable structure which does not change with the age. It does not change its shape with facial expressions. Furthermore, the ear is larger in size compared to fingerprints and can be easily captured although sometimes it can be hidden with hair and earrings. Although it has certain advantages over other biometrics, it has received little attention compared to other popular biometrics such as face, fingerprint and gait[3,4,5,6,7,8].

The current research has used intensity images and, therefore, the performance of the recognition systems is greatly affected by imaging problems such as lighting and shadows[3,4,5]. Range sensors that are insensitive to above imaging problems can directly provide us 3D geometric information[9,10]. Therefore it is desirable to design a human ear recognition system from 3D side face range images obtained at a distance. Human ear detection is the first task of a human ear recognition system and its performance significantly affects the overall quality of the system.

In this chapter, we propose two techniques for locating human ears in side face range images: template matching based ear detection[11] and ear shape model based detection[12]. The first approach has two stages: off-line model template building and on-line ear detection. Ear can be thought of as a rigid object with much concave and convex areas. As compared to other approaches for object detection in range images[13,14,15,16,17], we use the averaged histogram of shape index to represent the ear model template since shape index is a quantitative measure of the shape of a surface[20,21]. During the on-line detection, we first perform the step edge computation and thresholding since there is a sharp step edge around the ear boundary, and then we do image dilation and connected-component analysis to find the potential regions containing an ear. Next for every potential region, we grow the region and compute the dissimilarity between each region's histogram of shape indexes and the model template. Finally among all of the regions, we choose the one with the minimum dissimilarity as the detected region that contains ear.

For the second approach, the ear shape model is represented by a set of discrete 3D vertices corresponding to ear helix and anti-helix parts. Since the two curves formed by ear helix and anti-helix parts are similar for different people, we do not take into account the small deformation of two curves between different persons, which greatly simplifies our ear shape model. Given side face range images, step edges are extracted; then the edge segments are dilated, thinned and grouped into different clusters which are the potential regions containing an ear. For each cluster, we register the ear shape model with the edges. The region with the minimum mean registration error is declared as the detected ear region; the ear helix and anti-helix parts are identified in this process.

The rest of chapter is organized as follows. Section 2 introduces the related work, motivation and contributions. Section 3 describes our first approach to detect ears using template matching based method. Section 4 describes our second approach to build the ear shape model and detect human ears in side face range images. Section 5 gives the experimental results to demonstrate the effectiveness of two approaches. Finally, Section 6 provides the conclusions.

2. RELATED WORK, MOTIVATION AND CONTRIBUTIONS OF THE CHAPTER

2.1 Related work

There are only a few papers dealing with object detection from range images. In the following we give a brief review of object detection techniques from range images.

Keller et al.[13] introduced a fuzzy logic system for automatic target detection from LADAR images. They used two fuzzy logic detection filters and one statistical filter to create pixel-based target confidence values which are fused by the fuzzy integral to generate potential target windows. Features extracted from these windows are fed to a neural network post-processor to make a final decision.

Meier and Ade[14] proposed an approach to separate image features into ground and road obstacles by assuming the road was flat. They distinguished obstacles and road pixels using the separating plane. The plane model is updated by fitting a plane through all pixels marked as ground. Connected component analysis is used to partition detected obstacles into different objects.

Sparbert et. al.[15] presented a method to detect lanes and classify street types from range images. First they calculated the lane's width, curvature and relative position to the car, then compared them with a prior knowledge on construction rules of different street types, and finally achieved street type based on the mean value of lane's width.

Garcia et. al.[16] generated a unique signature of a 3D object by the Fourier transform of the phase-encoded range image at each specific rotation. The signature defined in a unit sphere permitted the detection of 3D objects by correlation techniques.

Heisele and Ritter[17] proposed a method for segmenting temporal sequences of range and intensity images. The fusion of range and intensity data for segmentation is solved by clustering 4D intensity/position features.

Kalman filters are then used to stabilize tracking by predicting dynamic changes in cluster positions.

Boehnen and Russ[18] proposed an algorithm to utilize the combination of the range and registered color images for automatically identifying facial features. The foreground is first segmented using the range data and then the skin color detection is used to identify potential skin pixels, which are further refined using z-based range erosion to compute the eye and mouth maps. Finally, the geometric-based confidence of candidates is computed to aid in the selection of best feature set.

Tsalakanidou et. al.[19] introduced a face localization procedure combining both depth and color data. First the human body is easily separated from the background by using the depth information and its geometric structure; and then the head and torso can be identified by modeling the 3D point distribution as a mixture model of two Gaussians; and finally the position of the face is further refined using the color images by exploiting the symmetric structure of the face.

2.2 Motivation

The anatomical structure of the ear is shown in Figure *6-1*. The ear is made up of standard features like the face. These include the outer rim (helix) and ridges (anti-helix) parallel to the helix, the lobe and the concha that is a hollow part of ear. From Figure *6-1*, we clearly see that two curves formed by ear helix and anti-helix parts are easily identified. We can use these two curves to develop the procedures to locate the ear in side face range images.

2.3 Contributions of the Chapter

The contributions of the chapter are: (a) Based on the geometric characteristics, we develop a template matching based approach for ear detection in side face range images. (b) We also propose a ear shape model based approach for locating 3D ears more accurately in side face range images. (c) We present many examples to illustrate the performance of these two approaches.

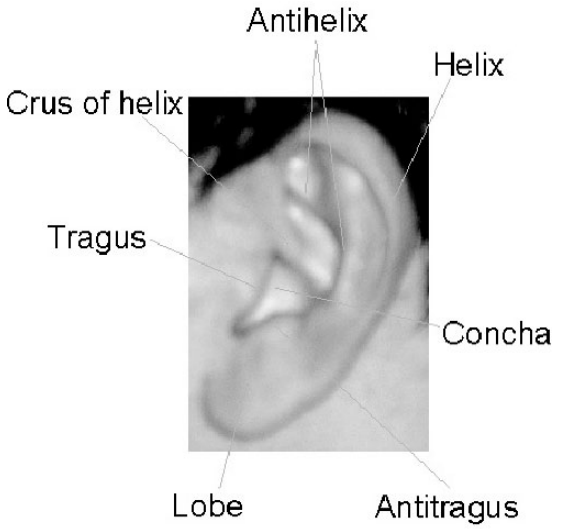


Figure 6-1. The external ear and its anatomical parts.

3. TEMPLATE MATCHING BASED EAR DETECTION

3.1 Off-line model template building

3.1.1 Shape Index

Shape index S_i, a quantitative measure of the shape of a surface at a point p, is defined by Eq. (1).

$$S_i = \frac{1}{2} - \frac{1}{\pi} \tan^{-1} \frac{k_1(p) + k_2(p)}{k_1(p) - k_2(p)} \tag{1}$$

where $k_1(p)$ and $k_2(p)$ are maximum and minimum principal curvatures respectively[20]. With this definition, all shapes can be mapped into the interval $S_i \in [0,1]$. The shape categories and corresponding shape index range are listed in Table *6-1*[20,21]. From Table *6-1*, we can see that larger shape index values represent convex surfaces and smaller shape index values represent concave surfaces.

Table 6-1. Surface shape categories and the range of shape index values.

Shape Category	S_i Range
Spherical cup	[0, 1/16)
Trough	[1/16, 3/16)
Rut	[3/16, 5/16)
Saddle rut	[5/16, 7/16)
Saddle	[7/16, 9/16)
Saddle ridge	[9/16, 11/16)
Ridge	[11/16, 13/16)
Dome	[13/16, 15/16)
Spherical cap	[15/16, 1]

Ear has significant convex and concave areas, which gives us a hint to use the shape index for ear detection. The original ear range image and its shape index image are shown in Figure *6-2*. In Figure *6-2*(b), the brighter points denote large shape index values that correspond to ridge and dome surfaces. We believe the ridge and valley areas form a pattern for ear detection. We use the distribution of shape index as a robust and compact descriptor since 2D shape index image is much too detailed. The histogram h can be calculated by $h(k) = \#$ of points with shape index $\in bin(k)$. The histogram is normalized during the implementation.

3.1.2 Curvature Estimation

In order to estimate curvatures, we fit a quadratic surface $f(x, y) = ax^2 + by^2 + cxy + dx + ey + f$ to a 5×5 window centered at the surface point of interest and use the least squa re method to estimate the parameters of the quadratic surface. After we get the parameters, we use differential geometry to calculate the surface normal, Gaussian and mean curvatures and principal curvatures[6,22].

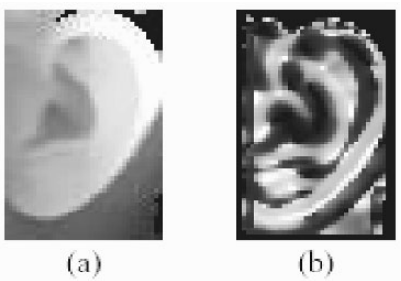


Figure 6-2. (a) Ear range image. Darker pixels are away from the camera and the lighter ones are closer. (b) Its shape index image. Darker pixels correspond to concave surfaces and lighter ones correspond to convex surfaces.

3.1.3 Building Model Template

Given a set of training side face range images, first we extract ears manually, and then calculate its shape index image and histogram the shape index image. After we get the histograms for each training image, we average the histograms and use the averaged histogram as our model template. Figure *6-3* shows the model template, obtained using 20 training images, in which the two peaks correspond to the convex and concave regions of the ear.

3.2 Step Edge Detection, Thresholding and Dilation

The step edge magnitude, denoted by M_{step}, is calculated as the maximum distance in depth between the center point and its neighbors in a small window[23]. M_{step} can be written as Eq. (2):

$$M_{step}(i, j) = \max |z(i, j) - z(i+k, j+l)|, \quad where \quad -(w-1)/2 \le k, l \le (w-1)/2 \tag{2}$$

In Eq. (2) w is the width of the window and $z(i, j)$ is the z coordinate of the point (i, j). To get the step edge magnitude image, a $w \times w$ window is translated over the original side face range image and the maximum distance calculated from Eq. (2) replaces the pixel value of the pixel covered by the center of the window. The original side face range image and its step edge magnitude image are shown in Figure *6-4* (a) and (b).

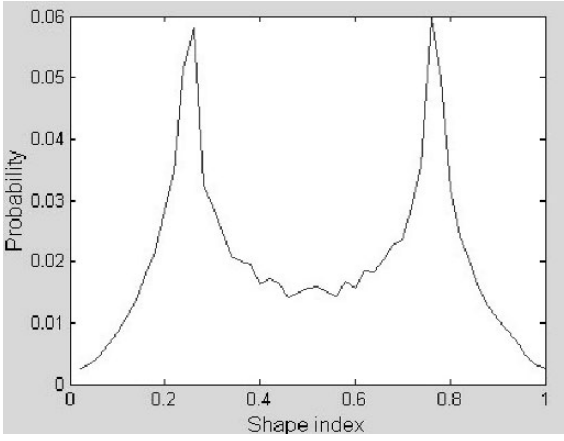


Figure 6-3. Model template (discretized into 50 bins).

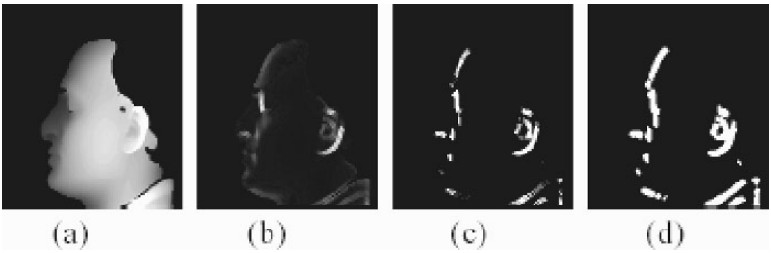


Figure 6-4. (a) Original side face range image. (b) Its step edge magnitude image. (c) Thresholded binary image. (d) Dilated image.

From Figure *6-4*(b), we clearly see that there is a sharp step edge around the ear boundary since brighter points denote large step edge magnitude. The step edge image is thresholded to get a binary image that is shown in Figure *6-4*(c). The threshold is set based on the maximum of M_{step}. Therefore, we can get a binary image by using Eq. (3). There are some holes in the thresholded binary image and we want to get the potential regions containing ears. We dilate the binary image to fill the holes using a 3×3 structure element. The dilated image is shown in Figure *6-4*(d).

$$F_T(i,j) = \begin{cases} 1 & \text{if} \quad M_{step}(i,j) \geq \eta \, Max\{M_{step}\} \\ & \quad 0 \leq \eta \leq 1 \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

3.3 Connected Component Labeling

Using the above result, we proceed to determine which regions can possibly contain human ears. To do so, we need to determine the number of potential regions in the image. By running connected component labeling algorithm, we can determine the number of regions. We used an 8-connected neighborhood to label a pixel. We remove smaller components whose areas are less than since the ear region is not small. The labeling result is shown in Figure *6-5*(a) and the result after removing smaller components is shown in Figure *6-5*(b).

After we get regions, we need to know the geometric properties such as the position and orientation. The position of a region may be defined using the center of the region. The center of area in binary images is the same as the center of the mass and it is computed by Eq. (4)

Figure 6-5. (a) Labeled image (12 components). (b) Labeled image after removing smaller components (6 components). (See also Plate 22 in the Colour Plate Section)

$$\bar{x} = \frac{1}{A}\sum_{i=1}^{n}\sum_{j=1}^{m} jB[i, j] \qquad \bar{y} = \frac{1}{A}\sum_{i=1}^{n}\sum_{j=1}^{m} iB[i, j] \tag{4}$$

where B is $n \times m$ matrix representation of the region and A is the size of the region.

For the orientation, we find the axis of elongation of the region. Along this axis the moment of the inertia will be the minimum. The axis is computed by finding the line for which the sum of the squared distances between region points and the line is minimum. The angle of θ is given by Eq. (5).

$$\theta = \frac{1}{2}\tan^{-1}\frac{b}{a-c} \tag{5}$$

The parameters a, b and c are given by Eq. (6), Eq. (7) and Eq. (8) respectively.

$$a = \sum_{i=1}^{n}\sum_{j=1}^{m} (x_{ij}^{'})^2 B[i, j] \tag{6}$$

$$b = 2\sum_{i=1}^{n}\sum_{j=1}^{m} x_{ij}^{'} y_{ij}^{'} B[i, j] \tag{7}$$

$$c = \sum_{i=1}^{n}\sum_{j=1}^{m} (y_{ij}^{'})^2 B[i, j] \tag{8}$$

where $x' = x - \bar{x}$ and $y' = y - \bar{y}$. θ gives us the hint about the direction along which region growing must take place.

3.4 Template Matching

As mentioned in Section 3.1.1, the model template is represented by an averaged histogram of shape index. Since histogram can be thought of as an approximation of probability density function, it is natural to use the $\chi^2 - divergence$ function Eq. (9)[24].

$$\chi^2(Q,V) = \sum_1 \frac{(q_i - v_i)^2}{q_i + v_i} \tag{9}$$

where Q and V are normalized histograms. From Eq. (9), we know the dissimilarity is between 0 and 2. If the two histograms are exactly the same, the dissimilarity will be zero. If the two histograms do not overlap with each other, it will achieve the maximum value 2.

From Section 3.3, we get the potential regions which may contain the ears. For each region, we find a minimum rectangular box to include the region, then we grow this region based on the angle θ. If $0 \le \theta \le \pi/2$, we grow the rectangle by moving the top-right vertex right, up and anti-diagonal and moving the bottom-left vertex left, down and anti-diagonal. If $\pi/2 \le \theta \le \pi$, we grow the rectangle by moving the top-left vertex left, up and diagonal and moving the bottom-right vertex right, down and diagonal. For every region, we choose the grown rectangular box with the minimum dissimilarity as the candidate ear region. Finally over all of the candidate regions, we select the one with the minimum dissimilarity as the detected region. We set a threshold γ for region growing, which controls the size of the region.

4. SHAPE MODEL BASED EAR DETECTION

4.1 Shape Model Building

Considering the fact that the curves formed by ear helix and anti-helix parts are similar for different people, we construct the ear shape model from one person only. We plan to work on building a generic ear model from multi persons. We extract ear helix and anti-helix parts by running a step edge detector with different thresholds and choose the best edge extraction result which detects ear helix and anti-helix parts. We define the ear shape

model s as 3D coordinates $\{x, y, z\}$ of n vertices that lie on the ear helix and anti-helix parts. The shape mode s is represented by a $3n \times 1$ vector $\{x_1, y_1, z_1, x_2, y_2, z_2, \ldots, x_n, y_n, z_n\}$. Figure *6-6* shows the 3D side face range image with textured appearance, in which the ear shape model s marked by yellow vertices is overlaid.

4.2 Step Edge Detection and Thresholding

Given the step face range image, the step edge magnitude, denoted by M_{step}, can be calculated as described in Section 3.2. One example of step edge magnitude image is shown in Figure *6-7*(b). In Figure *6-7*(b), larger magnitudes are displayed as brighter pixels. We can clearly see that most of the step edge magnitudes are small values. To get edges, the step edge magnitude image can be segmented using a threshold operator. The selection of threshold value is based on the cumulative histogram of the step edge magnitude image which is different from Section 3.2. Since we are interested in larger magnitudes, in our approach the top α ($\alpha = 3.5\%$) pixels with the largest magnitudes are selected as edge points. We can easily determine the threshold by investigating the cumulative histogram. The thresholded binary image is shown in Figure *6-7*(c), while the original side face range image is shown in Figure *6-7*(a).

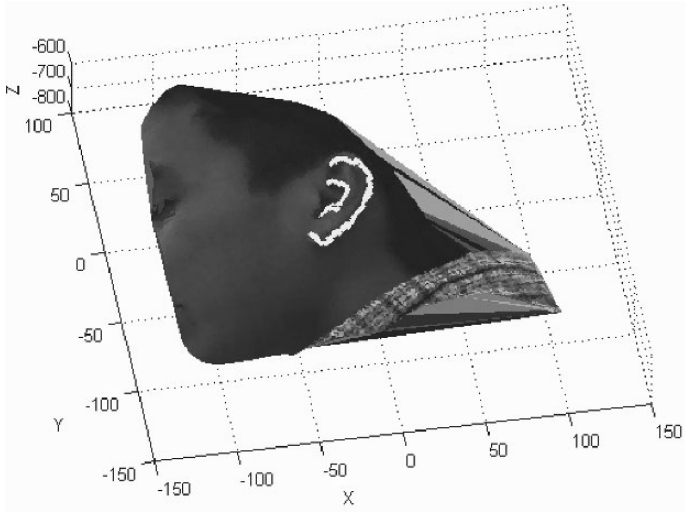


Figure 6-6. The textured 3D face and overlaid ear shape model (The units of X, Y and Z are mm).

Figure 6-7. (a) Original side face range image. (b) Its step edge magnitude image. (c) Its step edge image.

4.3 Edge Thinning and Connected Component Labeling

Since some step edge segments are broken, we dilate the binary image to fill the gaps using a 3×3 structure element. The dilated image is shown in Figure *6-8*(a). We proceed to do edge thinning and the resulting image is shown in Figure *6-8*(b). The edge segments are labeled by running connected component labeling algorithm and some small edge segments (less than 15 pixels) are removed. The edge segments that are left are shown in Figure *6-8*(c).

4.4 Clustering Edge Segments

Since the ear region contains several edge segments, we group edge segments which are close to each other into different clusters. The clustering procedure works as follows:

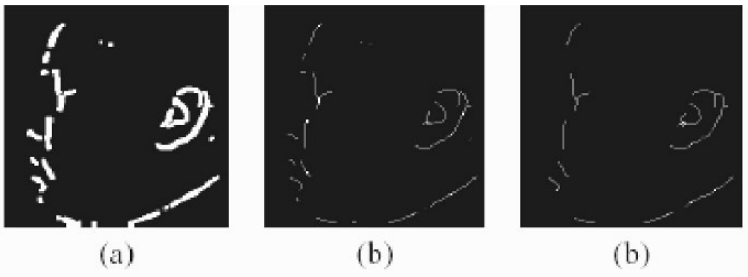


Figure 6-8. (a) Dilated edge image. (b) Thinned edge image. (c) Edge segments left after removal of small segments.

1. For each edge segment e_i
2. Initialize e_i into a cluster C_i, calculate its centroid $\{\mu_{xi}, \mu_{yi}\}$
3. For each edge segment e_j while $i \neq j$
   - (a) calculate its centroid $\{\mu_{xj}, \mu_{yj}\}$
   - (b) if $\max\{|\mu_{xi} - \mu_{xj}|, |\mu_{yi} - \mu_{yj}|\} \leq \varepsilon_1$, put e_j into cluster C_i, remove e_j and update the centroid of C_i

In the implementation, ε_1 =36 pixels since we want to put ear helix and anti-helix parts into a cluster. Three examples of clustering results are shown in the second row of Figure *6-9*, in which each cluster is bounded by a red rectangular box. The first row of Figure *6-9* shows side face range images.

4.5 Locating Ears by Use of the Ear Shape Model

For each cluster obtained in the previous step, we extract step edges around ear helix and anti-helix parts. We use a threshold ε_2 = 1.9 mm since the step edge magnitudes of vertices in ear anti-helix are at least 1.9 mm and the magnitude of vertices in anti-helix part is smaller than that of vertices in the helix part for the collected data. The problem of locating ears is to minimize the mean square error between ear shape model vertices and their corresponding edge vertices in the bounded rectangular box.

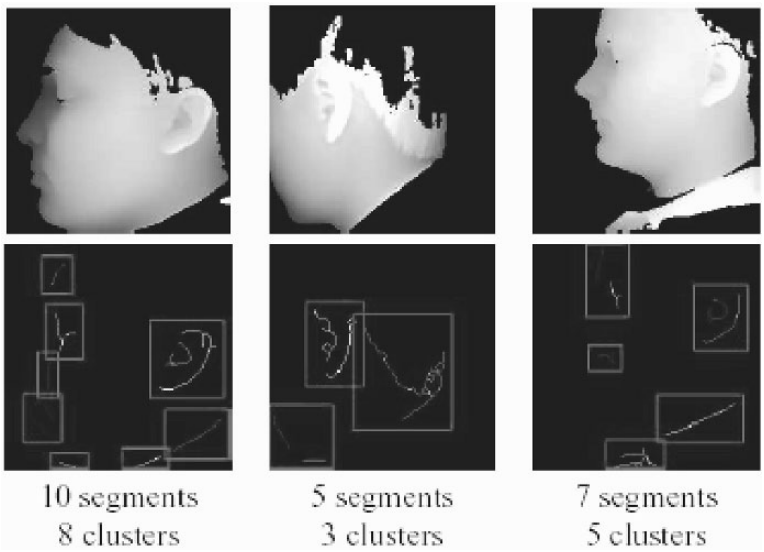


Figure 6-9. Examples of edge clustering results.

$$E = \frac{1}{n}\sum_{l=1}^{n} | T_r(s_i) - I(s_i) |^2 \quad (10)$$

where T_r is the rigid transformation and $I(s_i)$ is vertex in the 3D side face image closest to the $T_r(s_i)$. Iterative Closest Point (ICP) algorithm developed by Besl and Mckay[25] is well-known method to align 3D shapes. However, ICP requires that every point in one set have a corresponding point on the other set, we can't guarantee that edge vertices in the potential regions satisfy this requirement. Therefore, we use a modified ICP algorithm presented by Turk[26] to register the ear shape model with the edge vertices. The steps of modified ICP algorithm to register a test shape Y to a model shape X are:

- 1) Initialize the rotation matrix R_0 and translation vector T_0.
- 2) Given each point in Y, find the closest point in X.
- 3) Discard pairs of points that are too far apart.
- 4) Find the rigid transformation (R, T) such that E is minimized.
- 5) Apply the transformation (R, T) to Y.
- 6) Goto step 2) until the difference $| E_k - E_{k-1} |$ in two successive steps falls below a threshold or the maximum number of iterations is reached.

By initializing the rotation matrix R_0 and translation vector T_0 to the identity matrix and difference of centroids of two vertex sets respectively, we run ICP iteratively and finally get the rotation matrix R and translation vector T, which brings the ear shape model vertices and edge vertices into alignment. The cluster with minimum mean square error is declared as the detected ear region; the ear helix and anti-helix parts are identified in this process.

5. EXPERIMENTAL RESULTS

5.1 Data Acquisition

We use real range data acquired by Minolta Vivid 300. During the acquisition, 52 subjects sit on the chair about 0.55 ~ 0.75m from the camera. The first shot is taken when subject's left side face was approximately parallel to the image plane; two shots are taken when the subject was asked to rotate his/her head to left and right side within 35 degrees with respect to his/her torso. The same acquisition procedure was repeated once for another three shots. Six images per subject are recorded. Therefore, we have 312

images in total. Each range image contains 200×200 grid points and each grid point has a 3D coordinate $\{x, y, z\}$. The ear shape model is built from a side face range image described in Section 4.1. Examples of side face range images are shown in Figure *6.10*.

5.2 Results for the Template Matching Approach

We test the template matching based detection method on 312 side face range images. The parameters are $w = 5$ pixels, $\eta = 0.35$, $\beta = 99$ pixels and $\gamma = 35$ pixels. Figure *6-11* shows examples of positive detection in which the detected ears are bounded by rectangular boxes. If the detected region contains part of ear, we think it is a positive detection; otherwise it is a false detection. From Figure *6-11*, we observe that the ear region is correctly detected.

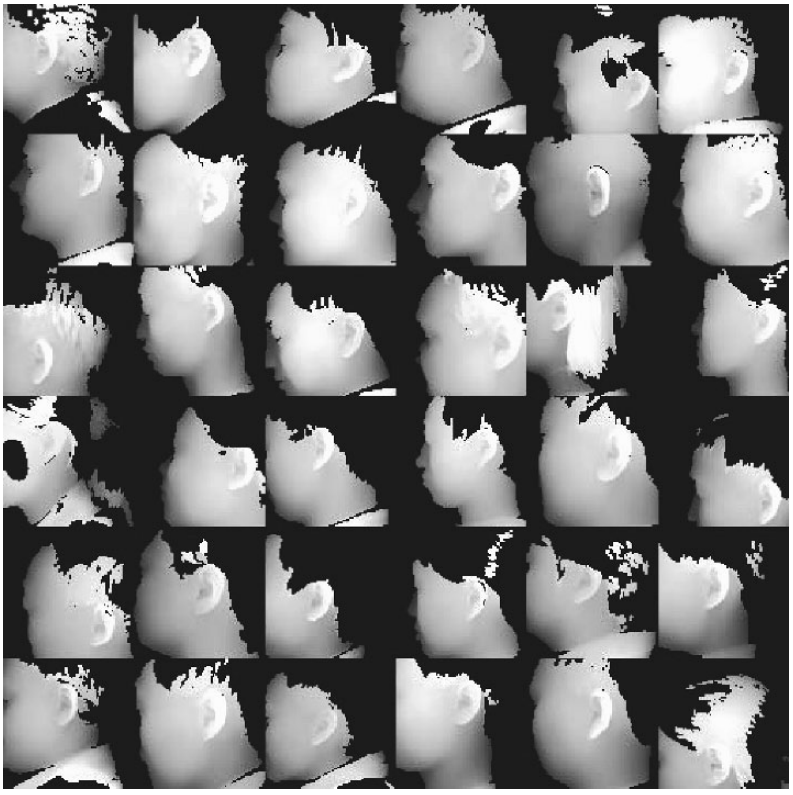


Figure 6-10. Examples of side face range images.

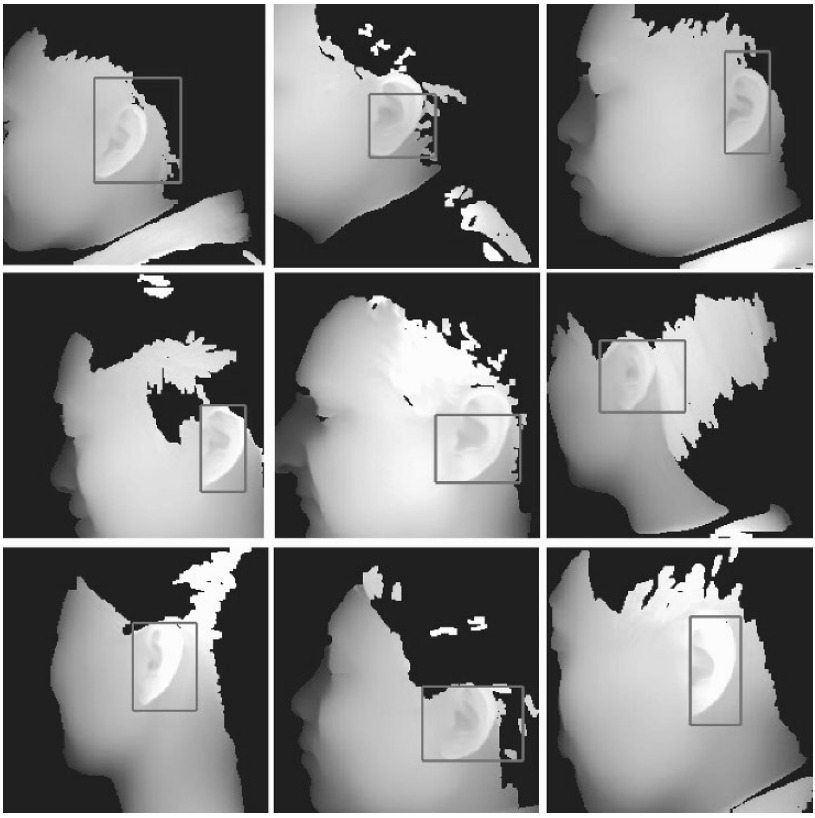


Figure 6-11. Examples of the positive detection using the template matching approach.

However, we may obtain a part of an ear; we may obtain parts that do not belong to an ear. Figure *6-12* shows examples of false detection. Each column in Figure *6-12* shows the step edge magnitude image, the dilated binary edge map and the detection result respectively. Since the ear helix part cannot be extracted, we made false detections. The average time to detect an ear from a side face range image is 5.2 seconds with Matlab implementation on a 2.4G Celeron CPU. We achieve 92.4% detection rate.

5.3 Results for the Shape Model-Based Approach

We test the ear shape model based detection method on 312 side face range images. If the ear shape model is aligned with the ear helix and anti-helix parts, we classify it as a positive detection; otherwise false detection.

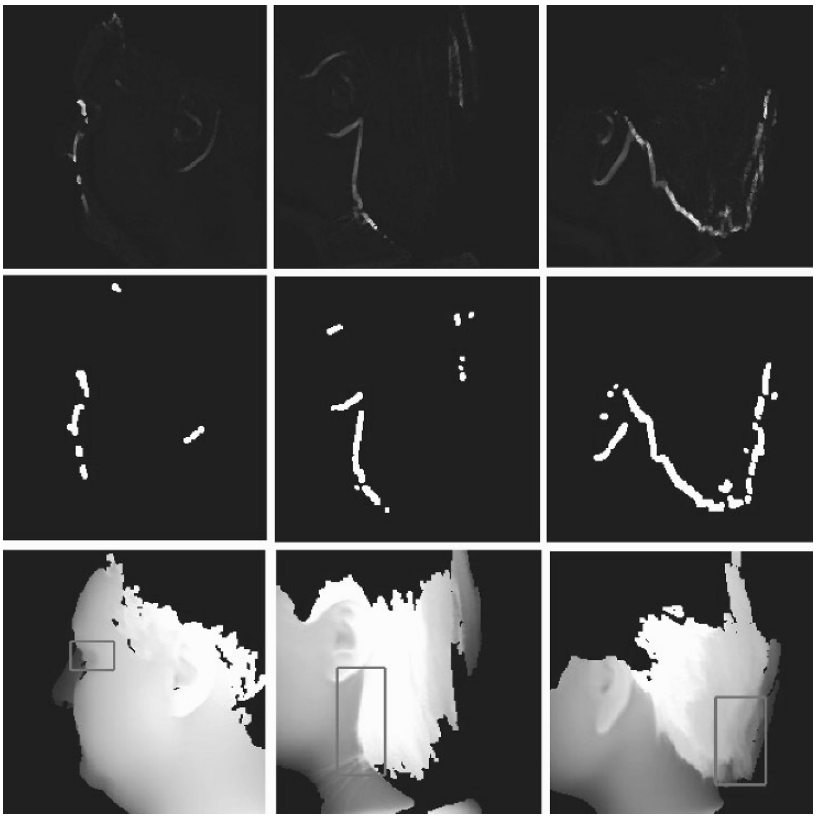


Figure 6-12. Examples of false detection using the template matching approach. Each column shows the step edge magnitude image, the dilated binary edge map and the detection result respectively.

In our experiments, the number of vertices of the ear shape model is 113; the average number of edge segments is 6 and the average number of clusters is 4. The average time to detect an ear from a side face range image is 6.5 seconds with Matlab implementation on a 2.4G Celeron CPU. Examples of positive detection results are shown in Figure *6-13*. In Figure *6-13*, the transformed ear shape model marked by yellow points is superimposed on the corresponding textured 3D face. From Figure *6-13*, we can observe that the ear is correctly detected and the ear helix and anti-helix parts are identified from side face range images. The distribution of mean square error defined in Eq. (10) for positive detection is shown in Figure *6-14*. The mean of mean square error is 1.79 mm. We achieve 92.6% detection rate. After we locate the ear helix and anti-helix parts in a side face range image, we put a minimum rectangular bounding box that contains the detected ear.

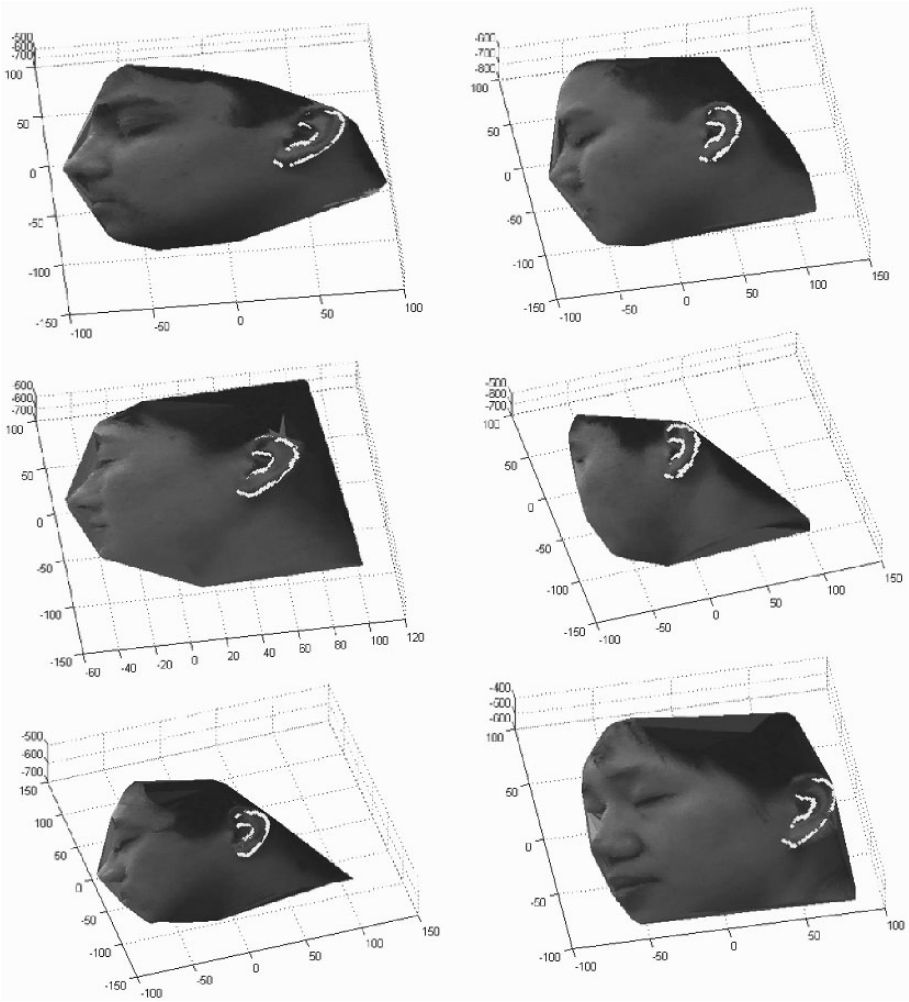


Figure 6-13. Examples of positive detection results using the shape model-based approach (The 3D axes and units are the same as in Figure 6-6).

Figure *6-15* shows the examples of detected ears with a red bounding box. From Figure *6-15*, we observe that the ears are accurately located from side face range images. For the failed cases, we notice that there are some edge segments around the ear region caused by hair, which bring more false edge segments or results in the cluster that cannot include the ear helix and anti-helix parts.

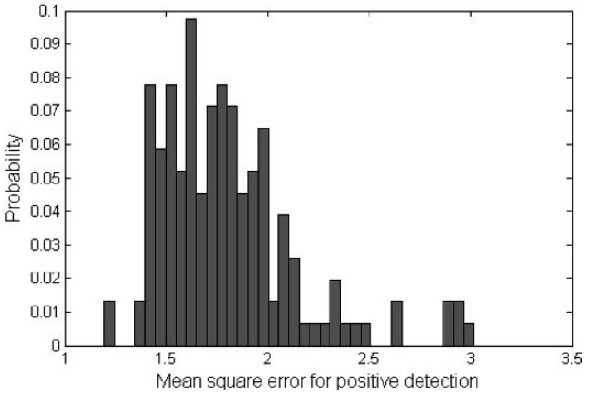


Figure 6-14. Distribution of the mean square error for positive detection using the shape model-based approach.

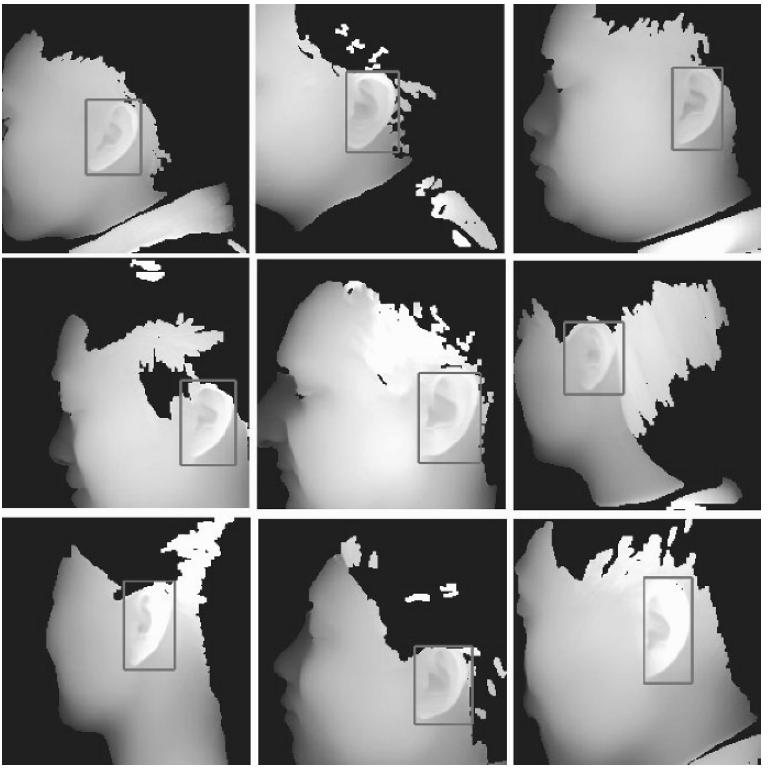


Figure 6-15. Examples of positive ear detection using the shape model-based approach.

Since ICP algorithm cannot converge due to the existence of outliers, the false detection happens; these cases are shown in Figure *6-16* and Figure *6-17*. The original face range images and corresponding edge clusters are shown in Figure *6-16*. In this figure, the first row shows face images; the second row shows edge clustering results. The textured 3D faces with overlaid detected ear helix and anti-helix are shown in Figure *6-17*.

5.4 Comparison of the Two Approaches

Table *6-2* lists the comparison of two approaches in terms of positive detection rate and average detection time. From this table, we observe that the second approach performs slightly better than the first approach and it is a little bit slower. Note that for the template matching approach, if the ear region is roughly detected, it is a positive detection; while for the ear shape model based approach, if the ear helix and anti-helix parts are aligned with the shape model, it is a positive detection. If we compare Figure *6-11* and Figure *6-15*, we see that we can locate ears more accurately by using the shape model-based approach, which can provide us more helpful cues for ear recognition.

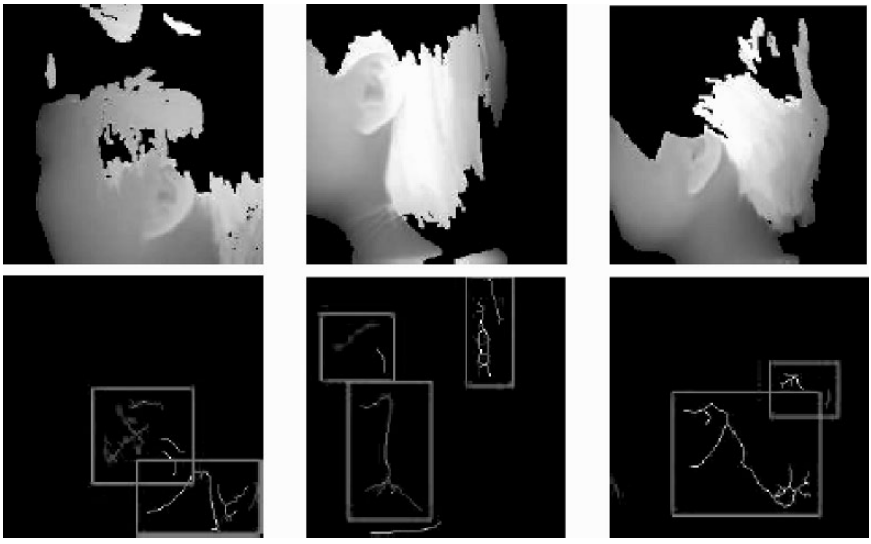


Figure 6-16. Examples of failed cases using the shape model-based approach. Each column shows the range image and the edge clustering result respectively.

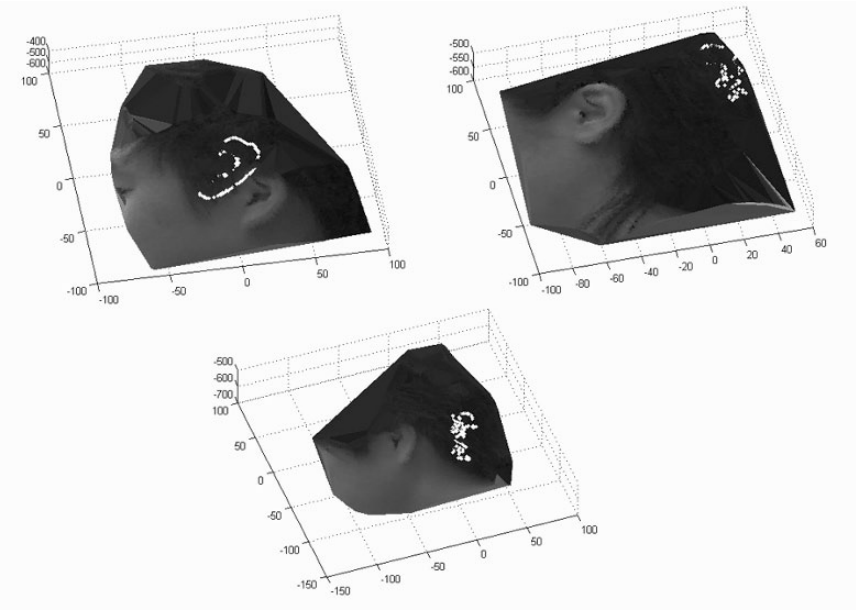


Figure 6-17. Examples of false detection results using the shape model-based approach.

Table 6-2. Comparion of two approaches.

	Detection Rate	Detection Time
Template matching	92.4%	5.2sec
Ear shape model	92.6%	6.5sec

6. CONCLUSIONS

We have proposed two techniques; template matching based detection and ear shape model based detection, to locate ears from side face range images. The success of the first approach relies on two facts: 1) there is a sharp step edge around the ear helix that can be easily extracted; 2) shape index is a good measurement to capture the geometric characteristics of ears since the ear has much ridge and valley areas. The first approach is also simple, effective and easy to implement. For the second approach, the ear shape model is represented by a set of discrete 3D vertices corresponding to ear helix and anti-helix parts. Given side face range images, step edges are extracted, dilated, thinned and grouped into different clusters which are potential regions containing ears. For each cluster, we register the ear shape model with the edges. Our method not only detects the ear region, but also

identifies the ear helix and anti-helix parts. Experimental results on real side face range images demonstrate the effectiveness of our proposed approaches.

REFERENCES

1. A. Iannarelli, Ear Identification, Forensic Identification Series, Paramont Publishing Company, 1989.
2. A. Jain, Personal Identification in Network Society, Kluwer Academic, 1999.
3. D. Hurley, M. Nixon, and J. Carter, Automatic ear recognition by force field transformations, IEE Colloquium on Visual Biometrics, 7/1 --7/5, 2000.
4. M. Burge and W. Burger, Ear biometrics in computer vision, Proc. Int. Conf. on Pattern Recognition, vol. 2, 822-826, 2000.
5. K. Chang, K. Bowyer, S. Sarkar, and B. Victor, Comparison and combination of ear and face images in appearance-based biometrics, IEEE Trans. Pattern Analysis and Machine Intelligence, 25(9), 1160--1165, 2003.
6. B. Bhanu and H. Chen, Human ear recognition in 3D, Workshop on Multimodal User Authentication, 91--98, 2003.
7. H. Chen and B. Bhanu, Contour matching for 3D ear recognition, 7th IEEE Workshops on Application of Computer Vision, vol. 1, 123--128, 2005.
8. P. Yan and K. W. Bowyer, Multi-Biometrics 2D and 3D ear recognition, Audio and Video based Biometric Person Authentication, 503-512, 2005.
9. B. Bhanu, Representation and shape matching of 3-D objects, IEEE Trans. Pattern Analysis and Machine Intelligence, 6(3): 340-351, 1984.
10. B. Bhanu and L. Nuttall, Recognition of 3-D objects in range images using a butterfly multiprocessor, Pattern Recognition, 22(1): 49-64, 1989.
11. H. Chen and B. Bhanu, Human ear detection from side face range images, Proc. Int. Conf. on Pattern Recognition, vol.3, 574--577, 2004.
12. H. Chen and B. Bhanu, Shape model-based 3D ear detection from side face range images, Proc. IEEE Conf. Computer Vision and Pattern Recognition workshop on Advanced 3D Imaging for Safety and Security, 2005.
13. J. Keller, P. Gader, R. Krishnapuram, and X. Wang, A fuzzy logic automatic target detection system for LADAR range images, IEEE International Conference on computatinoal intelligence, pp. 71-76, 1998.
14. E. Meier and F. Ade, Object detection and tracking in range images sequences by separation of image features, IEEE International conference on Intelligent Vehicles, 176-181, 1998.
15. J. Sparbert, K. Dietmayer, and D. Streller, Lane detection and street type classification using laser range images, IEEE Intelligent Transportation Systems conference proceedings, 454-459, 2001.
16. J. Garcia, J. Valles, and C. Ferreira, Detection of three-dimensional objects under arbitrary rotations based on range images, Optics Express, 11(25), 3352-3358, 2003.
17. B. Heisele and W. Ritter, Segmentation of range and intensity image sequences by clustering, Proc. IEEE Conf. on Information Intelligence and Systems, 223-227, 1999.
18. C. Boehnen and T. Russ, A fast Multi-Modal approach to facial feature detection, 7th IEEE Workshops on Application of Computer Vision, 1:135-142, 2005.
19. F. Tsalakanidou, S. Malasiotis, and M. G. Strintzis, Face localization and authentication using color and depth images, IEEE Trans. on Image Processing, 14(2):152-168, 2005.

20. C. Dorai and A. Jain, COSMOS-A representation scheme for free-form surfaces, Proc. Int. Conf. on Computer Vision, 1024-1029, 1995.
21. J. J. Koenderink and A. V. Doorn, Surface shape and curvature scales, Image Vision Computing, 10(8), 557--565, 1992.
22. P. Flynn and A. Jain, On reliable curvature estimation, Proc. IEEE Conf. Computer Vision and Pattern Recognition, 110-116, 1989.
23. N. Yokoya and M. D. Levine, Range image segmentation based on differential geometry: A hybrid approach. IEEE Trans. Pattern Analysis and Machine Intelligence, 11(6), 643-649, 1989.
24. B. Schiele and J. Crowley, Recognition without correspondence using multidimensional receptive field histograms, International Journal of Computer Vision, 36(1), 31-50, 2000.
25. P. Besl and N. D. Mckay, A method of registration of 3-D shapes, IEEE Trans. Pattern Analysis and Machine Intelligence, 14(2), 239-256, 1992.
26. G. Turk and M. Levoy, Zippered polygon meshes from range images, Proceedings of Conf. on Computer Graphics and Interactive Techniques, 311--318, 1994.

Part II

Safety and Security Applications

Chapter 7

SYNTHETIC APERTURE FOCUSING USING DENSE CAMERA ARRAYS

V. Vaish*, G. Garg†, E.-V. Talvala†, E. Antunez†, B. Wilburn†,
M. Horowitz†, and M. Levoy*
*Computer Science Department, Stanford Univeristy
†Department of Electrical Engineering, Stanford University.

Abstract: Synthetic aperture focusing consists of warping and adding together the images in a 4D light field so that objects lying on a specified surface are aligned and thus in focus, while objects lying off this surface are misaligned and hence blurred. This provides the ability to see through partial occluders such as foliage and crowds, making it a potentially powerful tool for surveillance. In this paper, we describe the image warps required for focusing on any given focal plane, for cameras in general position without having to perform a complete metric calibration. We show that when the cameras lie on a plane, it is possible to vary the focus through families of frontoparallel and tilted focal planes by shifting the images after an initial recitification. Being able to vary the focus by simply shifting and adding images is relatively simple to implement in hardware and facilitates a real-time implementation. We demonstrate this using an array of 30 video-resolution cameras; initial homographies and shifts are performed on per-camera FPGAs, and additions and a final warp are performed on 3 PCs.

Key words: light fields, synthetic aperture, projective geometry, real-time system

1. INTRODUCTION

Synthetic aperture focusing (also called dynamically reparametrized light fields) is a technique for simulating the defocus blur of a large aperture lens using multiple images of a scene, such as from a light field[3, 4]. The process consists of acquiring images of a scene from different viewpoints, projecting them onto a desired focal surface, and computing their average. In the

A. Koschan et al. (eds.), 3D Imaging for Safety and Security, 159–172.

resulting image, points that lie on the focal surface are aligned and appear sharp, whereas points off this surface are blurred out due to parallax (Figure *7-1* (a)). Researchers in computer vision and graphics have used synthetic aperture focusing to blur out occluders in front of desired focal planes, enabling them to see objects behind dense foliage[3, 6]. This ability to see behind occluders makes synthetic aperture focusing an attractive tool for surveillance.

One challenge in using synthetic aperture focusing for surveillance of dynamic scenes has been the amount of computation required. Constructing a synthetically focused image for a given focal plane requires applying a homography to each camera's image and computing their mean. The homography required for each image depends on the camera parameters and the focal plane. If we wish to change the focal plane, we need to apply different homographies to all the images. This requires substantial computation and may be difficult to achieve in real-time. However, in certain cases, we can change the focal plane without having to apply a new projective warp to the images. Consider the case when the cameras lie on a plane, and their images have been projected onto a parallel plane Π_0 (Figure *7-1* (b)). To focus on any other plane parallel to the camera plane, we need to just shift the projected images and add them[6]. In other words, we have factorized the homographies for focusing on frontoparallel planes into an initial projection (onto Π_0) followed by shifts. The initial projection needs to be applied only once.

In this paper, we explore such a factorization the case of arbitrary camera configurations and focal planes. We show that the homographies required for focusing can be factorized into an initial projection, as before, followed by a *planar homology* (a special projective warp[1]). Varying the focal plane requires varying the homologies applied to the projected images. We prove a rank-1 constraint on homology parameters to characterize homologies required for varying the focal plane. This lets us focus on any plane, for any camera configuration, without having to perform a metric calibration of the cameras; while letting a user specify focal planes in a geometrically intuitive way.

Interestingly, there are camera configurations and families of tilted focal planes for which the homologies reduce to shifts (Figure *7-1* (c)), just as in the frontoparallel case. This shear-warp factorization into an initial projection independent of the focal plane, followed by shifts to vary the focal plane, is well suited for real-time implementation in hardware. The initial projection may be implemented via a lookup table, and does not need to be changed to vary the focus. It is relatively simple to vary the focus by shifting the images in hardware. We demonstrate real-time synthetic aperture focusing with an array of 30 video cameras. The initial projection and shifts

are implemented in per-camera FPGAs, and addition of images (with an optional warp applied to the final image) is done on a cluster of 3 PCs.

Our work builds upon two important concepts in multi-view geometry: the notion of plane + parallax[2, 5] which simplifies geometric analysis by projecting images from different views onto a reference plane; and the study of the space of all homologies by Zelnik-Manor et al.[9] They show that homologies lie in a 4-D space. By representing the homolgies differently, and by factoring out the epipoles we show the homology parameters actually live in a 1-D space. This helps us in specifying arbitrary focal planes.

The remainder of this paper is structured as follows. We study the homologies required for synthetic aperture focusing in Section 2 and enumerate the cases in which the factorization described above could lead to greater efficiency. Section 3 describes our real-time synthetic focus system, and results. We conclude with a discussion in Section 4.

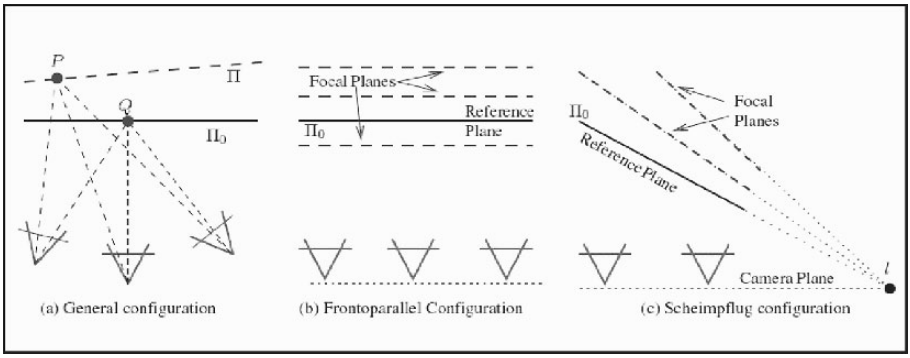


Figure 7-1. (a) Synthetic aperture focusing consists of projecting camera images on to a plane Π_0 and computing their mean. Point Q on the plane Π_0 is in focus; point P not on this plane is blurred due to parallax. Projection on to a focal plane requires applying homographies to the camera images. (b) If the cameras lie on a plane and their images are projected on to a parallel reference plane via homographies, then we can vary the focus through frontoparallel planes by just shifting and adding the images. This is simpler than having to apply different homographies for every focal plane. (c) We show that there exist camera configurations and families of tilted focal planes for which the focusing homographies can be decomposed into an initial projection followed by shifts. Varying the focus requires merely shifting the images, as in the frontoparallel case, plus a final warp after adding the images together.

2. REFOCUSING WITH HOMOLOGIES

We will now study how to factorize the homographies required for synthetic aperture focusing into an initial projection followed by a homology. Consider an array of cameras with centers $C_0, \ldots, C_N$ whose images have been projected onto some reference plane Π_0 (Figure *7-1* (a)). Let I_i denote the projected image from the i^{th} camera. If we compute the average of the projected images I_i, we get an image focused on the reference plane.

Suppose we wish to focus on a different plane, Π. One could do so by applying different homographies to the camera images. Another way would be to *reproject* each of the projected images I_i from the reference plane onto Π through center of projection C_i (Figure *7-2*). This reprojection is called a *planar homology* and can be described by a 3x3 matrix. In this section, we describe the homology matrices for different focal planes. We establish a new rank-1 constraint on homology parameters, we show how to compute the homologies without requiring metric calibration, and we enumerate the configurations in which these homologies are reduced to affine or simpler transforms.

We begin our study of homologies by defining the coordinate systems we will use. Assume that there is a given 2D coordinate system on the reference plane. The pixels of the projected images I_i are specified in this reference coordinate system.

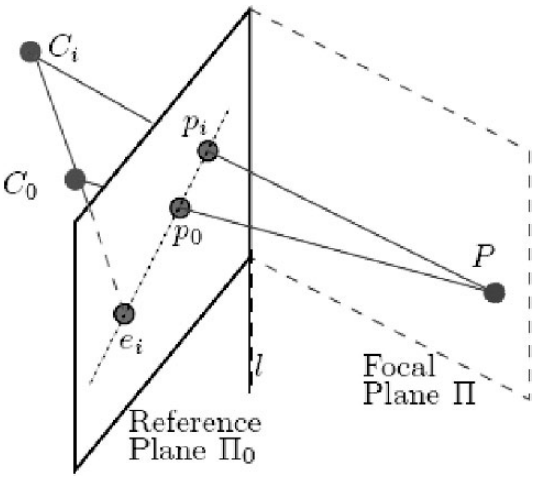


Figure 7-2. To change the focus from reference plane Π to plane Π_0, we need to reproject the image from camera center C_i onto Π. This projection is called a *homology*.
(See also Plate 23 in the Colour Plate Section)

We also need to pick a coordinate system on the new focal plane Π. To do so, we pick a reference camera - say C_0. A point P on Π will be assigned the coordinates of its projection p_0 on the reference plane through C_0 (Figure *7-2*). Thus, we are projecting the coordinate system on the reference plane onto the new focal plane through center of projection C_0.

Let G_i be the homology required to project I_i onto Π, $1 \leq i \leq N$ ($G_0 = I$, the identity matrix). G_i maps point p_i on the reference plane to the point P on Π (Figure *7-2*). Since P has the same coordinates as p_0, we may write $G_i\, p_i \equiv p_0$ where G_i denotes the homology matrix, points and lines on the reference plane are represented in homogeneous coordinates and ≡ denotes equality up to a scale. For the ensuing analysis, it will be simpler to work with the inverse homologies $K_i = G_i^{-1}$, i.e. K_i projects points on Π onto the reference plane through center C_i and $K_i\, p_0 \equiv p_i$.

We now proceed to characterize the 3x3 homology matrices for changing the focal plane. From projective geometry[1, 5], we know that the homology K_i can be written as:

$$K_i = I + \mu_i\, e_i\, l^T \qquad (1)$$

Here I is the identity, e_i the epipole associated with cameras C_0, C_i and the reference plane, l the line of intersection of the reference plane and Π, and μ_i is a scalar. Geometrically, varying μ_i while keeping e_i, l fixed corresponds to rotating the focal plane about axis l and moving p_i along the epipolar line through e_i and p_0. Choosing a value for μ_i amounts to choosing a focal plane (through l) for camera C_i. Suppose we are given the epipoles e_i, $1 \leq i \leq N$. We would like to characterize $\mu_1, \ldots, \mu_N$ which are consistent, i.e. correspond to the same choice of focal plane Π. Let us call $[\, \mu_1 \ldots \mu_N \,]$ the μ-vector for homologies induced by a focal plane Π. The following result helps us characterize which vectors of $\mathbf{R}^N$ are μ-vectors for some focal plane.

Theorem. *Given epipoles $e_1, \ldots, e_N$ on a reference plane as defined above, the μ-vectors for all focal planes lie in a 1-D space, i.e. μ-vectors for any two planes are equal up to scale.*

Remark: In homogeneous coordinates, points and lines can be scaled arbitrarily. The μ-vector for a focal plane will change if we change the scale of any of the epipoles or the line l. It is assumed in the theorem that we have chosen a fixed scale for each epipole e_i, this scale could be arbitrarily chosen but must be the same for all homologies we compute. If we change the scale for any of the epipoles, we will change the 1D space the μ-vectors lie in. However, as long as the scales are fixed they will still lie in a 1D space.

The proof of this theorem is presented in Appendix A.

2.1 Focusing and calibration

Let us see how to use the preceding analysis for user-driven change of the focal plane. Suppose we know the epipoles $e_1, \ldots, e_N$ and the μ-vector $\boldsymbol{\mu} = [\, \mu_1 \ldots \mu_N \,]$ for the homologies induced by some focal plane. To specify a new focal plane, the user first chooses a line l on the reference plane (l could also be the line at infinity) through which the focal plane passes. Every focal plane through l has to have a μ-vector equal to $f\boldsymbol{\mu}$ for some scalar f. By picking a value for f, the user selects a particular focal plane through l. The homologies for this focal plane are $K_i = I + f\, \mu_i\, e_i\, l^T$. The synthetic aperture image with this focal plane can be computed:

$$I_{sap} = \frac{1}{N+1} \sum_{i=0}^{N} K_i^{-1} o I_i \qquad (2)$$

Varying f amounts to rotating the focal plane about axis l. At f=0, the focal plane coincides with the reference plane. In our system, the user can either specify f interactively by moving a slider and getting feedback from the synthetic aperture image for the corresponding focal plane, or specify a range of values of f to compute a sequence of synthetically focused images with the focal plane rotating about l (Figure *7-3*). f is analogous to the depth of the plane in focus of the lens being simulated by the camera array.

It should be clear from this discussion that to vary the focus it suffices to know the epipoles, a μ-vector for any one focal plane and the initial homographies required to project camera images onto a reference plane. These quantities may be computed using any of the projective calibration methods[1, 5] in the literature; no metric calibration (camera intrinsics or Euclidean pose) is required.

Figure 7-3. User-driven change of focus. (a) An image from a light field showing a toy humvee at an angle to the plane of cameras. (b) Synthetically focused image on reference plane. Note that the left side of the humvee is out of focus, since it is not on the reference plane. (c) If we rotate the focal plane about the line indicated, we can get the full side of the humvee in focus with a tilted focal plane. (d) Plan view of our setup.

At the minimum, one would need to know the images of four points on a plane (for reference plane homographies) plus images of at least two points not on this plane to compute the epipoles[1] and μ-vector. For most of our experiments, the cameras (to a good approximation) do lie on a plane, and we can use the method in Vaish et al.[6] for the necessary calibration.

2.2 Simpler configurations

We now enumerate the cases when the homologies are not projective warps, but affine transforms or simpler. For each of these configurations, the homologies $\boldsymbol{K_i}$ lie in a proper subgroup of the group of planar homologies. (Unless otherwise stated, we will assume we have established an affine coordinate system on the reference plane).

General affine transforms: When the camera centers lie on a plane, and the reference plane is parallel to this plane, the epipoles $e_i = [\ e_i^{(x)}\ \ e_i^{(y)}\ \ 0\ ]^T$ are points at infinity. The bottom row of the homology matrix $I + \mu\ e_i\ l^T$ becomes $[\ 0\ \ 0\ \ 1\ ]$. Hence, homologies for any focal plane are affine transforms. Note that this holds for arbitrary focal planes.

Scale and shift: When the focal plane and reference plane are parallel, their intersection is the line at infinity $l = [\ 0\ \ 0\ \ 1\ ]^T$ on the reference plane. Let $e_i = [e_i^{(x)}\ \ e_i^{(y)}\ \ e_i^{(z)}\ ]$ denote the epipoles, then the homologies are of the form:

$$K_i = I + \mu\ e_i\ l^T = \begin{bmatrix} 1 & 0 & \mu e_i^{(x)} \\ 0 & 1 & \mu e_i^{(y)} \\ 0 & 0 & 1+\mu e_i^{(z)} \end{bmatrix} \tag{3}$$

This is just a scale followed by a shift. Thus, if we wish to vary the focus through a family of planes parallel to the reference plane, we need to just scale and shift the images before computing their average. Note that this holds for arbitrary camera positions.

Shifts (Scheimpflug Configuration): Consider the case when the cameras lie on a plane,. and the camera plane, reference plane and desired focal plane intersect in the same line l (Figure *7-1* (c)). The epipoles lie on this line l. Suppose we redefine the coordinate system on the reference plane so that $l \equiv [\ 0\ \ 0\ \ 1\ ]^T$. This combines the previous two conditions, and the homologies are reduced to shifts:

$$\mathbf{K_i} = \mathbf{I} + \boldsymbol{\mu} \begin{bmatrix} e_i^{(x)} \\ e_i^{(y)} \\ 0 \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & \mu e_i^{(x)} \\ 0 & 1 & \mu e_i^{(j)} \\ 0 & 0 & 1 \end{bmatrix} \quad (4)$$

This condition is analogous to the Scheimpflug condition in photography, which is required to focus on tilted planes. Specifically, the lens plane, sensor plane and focal plane must intersect in a common line[7]. This case is well suited for a real-time implementation in hardware as the image shifts required to vary the focal plane are easy to realize in hardware. This generalizes the frontoparallel case studied in Vaish et al[6] (Figure *7-1* (b)). After shifting and adding the images, we can warp the resulting image back to the original coordinate system on the reference plane if desired.

In fact, all configurations for which varying the focus requires only shifts have to be Scheimpflug configurations. $K_i = I + \mu e_i l^T$ is a translation only if $l = [\,0 \;\; 0 \;\; 1]^T$ and $e_i^{(z)} = 0$. This means the epipoles lie on l, i.e. the camera plane intersects the reference plane in the same line l as the focal plane.

3. REAL-TIME SYNTHETIC FOCUS

We have implemented synthetic aperture video on an array of 30 cameras in our laboratory. Our camera array is based on the architecture described in Wilburn[8], with a video resolution of 320x240 grayscale, 30 frames/sec. Here we will concentrate on how the shear-warp factorization lets us vary the focal plane in real-time.

The processing pipeline is shown in Figure *7-4*. The video stream from each camera is sent to its capture board. The FPGA applies the initial homography required for projection onto the reference plane using a precomputed lookup table stored in RAM. The warped frames are shifted, MPEG compressed and transmitted to the host PC. Each host PC receives streams from 15 cameras, which are decoded, added and sent over the network to a master PC. The master PC adds the streams from the hosts and displays the final synthetic aperture video. It also warps the final image back to the original coordinate system of the reference plane, if desired, as described in the previous section.

In general, varying the focal plane requires changing the homographies being applied in the FPGAs. However, loading new lookup tables into the RAM takes about 30 seconds; we cannot change the focus interactively this way. This is where the shear-warp factorization is useful.

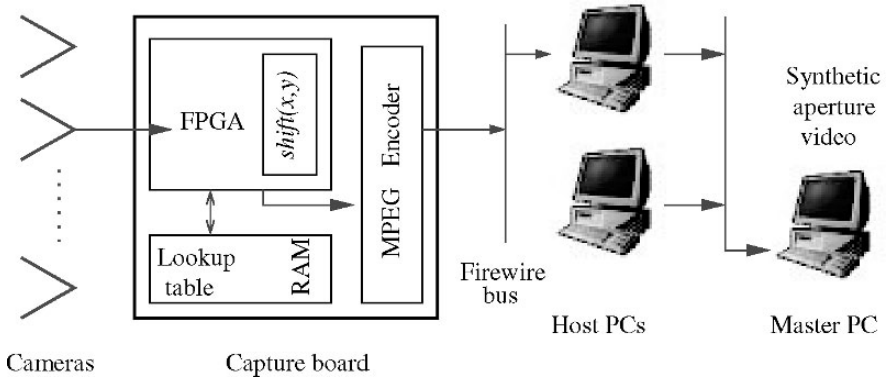


Figure 7-4. Overview of our real-time system. Each video stream goes to a capture board where a homography is applied (using a lookup table) followed by a shift *(x, y)*. The MPEG-compressed video is sent over firewire to a host PC, where warped frames from different cameras are added. A master PC adds the streams from all host PCs, displaying synthetic aperture video. The focal plane can be changed by varying the shifts.

For the Scheimpflug configuration, the homographies can be factored into a reference plane homography followed by a shift. Changing the focal plane through the family of planes described in the previous section only requires changing the shifts in the FPGAs. This is easy to do in real-time. In our interactive system, the focal plane is changed by having the user move a slider, which updates the shifts in the FPGAs.

Instead of using a lookup table, one could try to implement projective warps by multiplying each pixel coordinate with a 3x3 homography matrix and finding the intensity values at the resulting point in the captured frame (backward warp). This avoids the expense of changing a large lookup table for changing the focus. However, this approach requires multipliers and at least one divide per pixel. Implementing these in hardware would use a substantial fraction of our FPGA, and might not fit at all given more modest per-camera hardware. In contrast, using a lookup table is relatively simple to realize in the FPGAs, and could let us apply arbitrary warps which would be necessary anyway if we wished to correct for lens distortion or have non-planar focal surfaces. The drawback is that it constrains the ways in which we can vary the focus.

Building the system we have described has several challenges. They include fitting the framebuffers and warping circuitry in the FPGA, decoding multiple MPEG streams on the host PCs in real-time, and integrating the different components in the pipeline. A more detailed description of our system design describe and the architectural trade-offs we made to meet these challenges is described in Wilburn[8].

3.1 Results

We show two examples of our system in action. The first scene consists of a person moving forward towards the cameras, with people walking in front of him (Figure *7-5*). By adjusting the focal plane with a slider as the subject moves, the user is able to keep him in focus while the people occluding him are blurred out.

The second scene consists of a suitcase placed at an angle to the camera array (Figure *7-6*). Although it is not possible to bring the entire suitcase into good focus using frontoparallel planes, using tilted focal planes from a Scheimpflug configuration, we can bring the entire suitcase into focus, while people in front of it are blurred out. (We urge the reader to view a video demonstrating the capabilities of our system in action at *http://graphics.stanford.edu/papers/shear-warp/a3diss.avi*.)

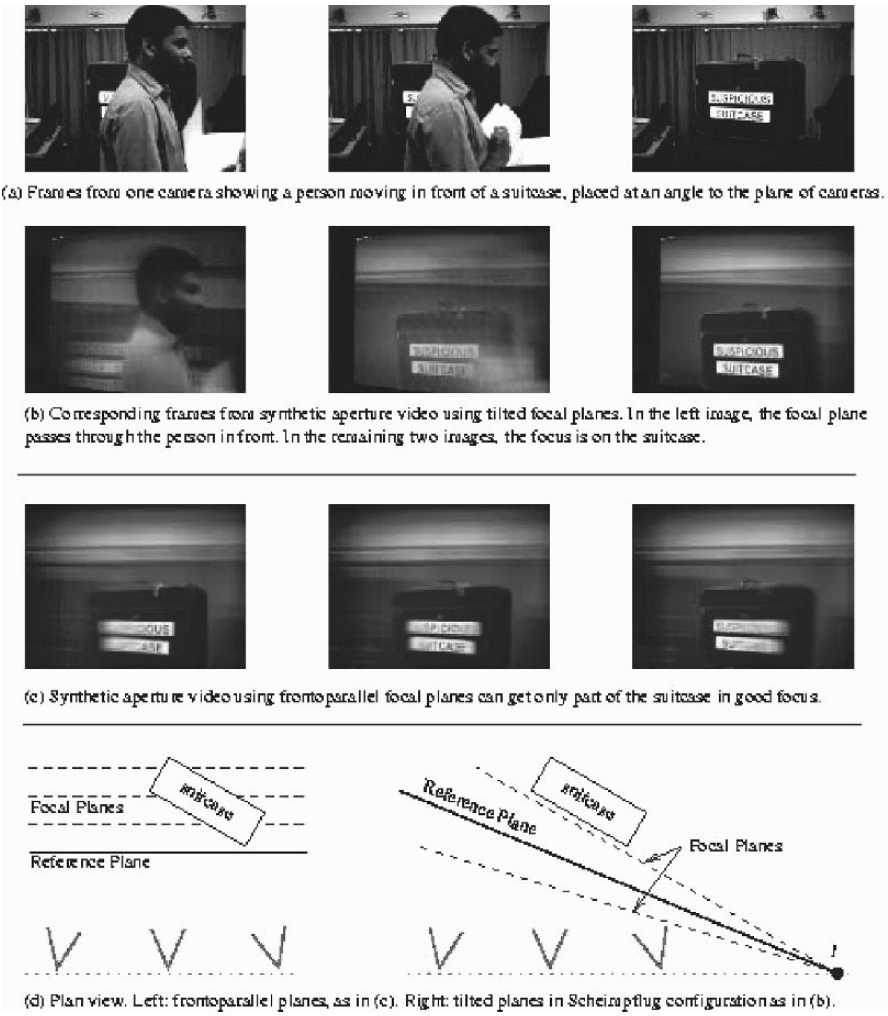
(a) Our array of 30 cameras with a 1m aperture.

(b) System screenshot, showing person to be tracked.

(c) Frames from one camera, showing the subject being tracked as people move in front of him.

(d) Corresponding frames from the synthetic aperture video. The focal plane follows the subject as he moves forward.

Figure 7-5. Real-time synthetic aperture video, used to track a person moving through a crowd of people.



(a) Frames from one camera showing a person moving in front of a suitcase, placed at an angle to the plane of cameras.

(b) Corresponding frames from synthetic aperture video using tilted focal planes. In the left image, the focal plane passes through the person in front. In the remaining two images, the focus is on the suitcase.

(c) Synthetic aperture video using frontoparallel focal planes can get only part of the suitcase in good focus.

(d) Plan view. Left: frontoparallel planes, as in (c). Right: tilted planes in Scheimpflug configuration as in (b).

Figure 7-6. Focusing on tilted planes.

4. CONCLUSIONS

In this paper, we have shown the homographies required for synthetic aperture focusing on arbitrary focal planes can be factorized into an initial projection followed by a homology. We have categorized the camera and focal plane configurations for which homologies are affine or simpler warps. For cameras and focal planes in the Scheimpflug configuration, these homologies are reduced to shifts, facilitating a hardware implementation in

which we can change the focal plane in real-time. Given the ability of synthetic aperture imaging to see around occluders, we feel this system would be useful for surveillance and reconnaissance. Our analysis also shows how to implement synthetic aperture focusing without having to perform a metric calibration.

The main limitation of the system is that we are restricted to a family of focal planes that pass through a line (or parallel focal planes, if this line is at infinity). To change this family - for example, to switch from frontoparallel focal planes to tilted planes - we need to update the lookup tables for all our cameras, which takes about 30 seconds.

We would like to extend our system to work with more cameras and handle non-planar focal surfaces. An interesting challenge would be to automatically track moving objects in real-time, even if they get occluded in most of the cameras. In our experiments with manually tracking people through crowds, we learnt that it is difficult to have the tracked person in perfect focus. Thus, computer-assisted focusing would be a desirable addition to our system.

ACKNOWLEDGEMENTS

We wish to thank Augusto Roman, Billy Chen and Abhishek Bapna for assisting in acquisitions. This work was partially supported by grants NSF IIS-0219856-001, DARPA NBCH 1030009, a gift from Bosch RTC, and the Reed-Hodgson Stanford Graduate Fellowship.

REFERENCES

1. R. Hartley and A. Zisserman. Multiple View Geometry, *Cambridge University Press*, 2000.
2. M. Irani, P. Anandan and D. Weinshall. From Reference Frames to Reference Planes: Multi-View Parallax Geometry and Applications. In *Proc. of ECCV*, 1996.
3. A. Isaksen, L. McMillan and S. Gortler. Dynamically Reparametrized Light Fields. In *Proc. of ACM SIGGRAPH*, 2000.
4. M. Levoy and P. Hanrahan. Light Field Rendering. In *Proc. of ACM SIGGRAPH*, 1996.
5. B. Triggs. Plane + Parallax, Tensors and Factorization. In *Proc. of ECCV,* 2000.
6. V. Vaish, B. Wilburn, N. Joshi, M. Levoy. Using Plane + Parallax to Calibrate Dense Camera Arrays. In *Proc. of IEEE CVPR*, 2004.
7. R. Wheeler. Notes on Viewcam Geometry, *available online at* http://www.bobwheeler.com/photo/ViewCam.pdf
8. B. Wilburn. High Performance Imaging Using Arrays of Inexpensive Cameras. *PhD Thesis, Stanford University*, 2004.
9. L. Zelnik-Manor and M. Irani. Multi-view Subspace Constraints on Homographies. In *Proc. of IEEE ICCV*, 1999.

APPENDIX A

Here we prove the theorem stated in Section 2.1. To show the μ-vectors live in a 1D space, it suffices to show that the ratio μ_i/μ_j, $1 \le i < j \le N$ is independent of the plane Π.

Without loss of generality, we may take i=1, j=2. Let P be a point on Π, and p_0, p_1, p_2 be its projections onto the reference plane through centers of projection C_0, C_1, C_2 (Figure *7-7*). Let e_{12} be the epipole associated with camera centers C_1, C_2; e_{12} lies on the reference plane. The epipoles e_1, e_2, e_{12} lie on the line of intersection of the reference plane and the plane through the camera centers C_0, C_1, C_2 (Desargues' theorem). By epipolar geometry, e_{12}, p_1, p_2 are collinear. Let $|\ a\ b\ c\ |$ denote the determinant of the matrix whose columns are the 3-vectors a, b, c, and note that $|\ a\ b\ c\ | = 0$ if a, b, c are collinear. We have

$$p_1 \equiv K_1 p_0 = p_0 + \mu_1 e_1 l^T p_0 \tag{5}$$

$$p_2 \equiv K_2 p_0 = p_0 + \mu_2 e_2 l^T p_0 \tag{6}$$

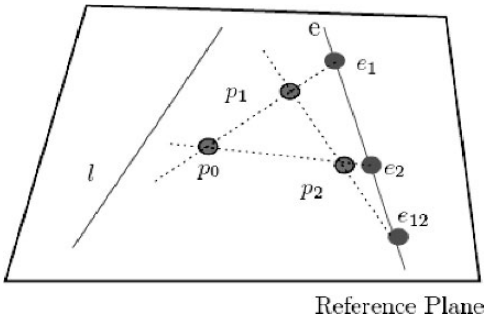


Figure 7-7. Geometry underlying proof of Theorem 1. p_0, p_1, p_2 are the images of a point P on the desired focal plane Π in cameras C_0, C_1, C_2 (see Figures *7-1*(b) and *7-2* for sectional and oblique views). e_1, e_2, e_{12} are the epipoles. The plane through the camera centers C_0, C_1, C_2 intersects the reference plane in a line **e** containing these epipoles. l is the intersection of the focal plane with the reference plane. (See also Plate 24 in the Colour Plate Section)

$$| e_{12}\ e_1\ e_2 | = 0 \tag{7}$$

$$| e_{12}\ p_1\ p_2 | = 0 \tag{8}$$

where the first two relations follow from the homologies K_1, K_2 and the last two from collinearity. If we substitute (5) and (7) in (8), expand using properties of determinants and (7), we get

$$0 = | e_{12}\ p_1\ p_2 |$$

$$= | e_{12}\ \ p_0 + \mu_1 e_1\ (l^T p_0)\ \ \ p_0 + \mu_2 e_2\ (l^T p_0) |$$

$$= | e_{12}\ \ \mu_1 e_1\ (l^T p_0)\ \ \ p_0\ | + | e_{12}\ \ p_0\ \ \mu_2 e_2\ (l^T p_0) |$$

$$= \mu_1 | e_{12}\ \ e_1\ \ \ p_0\ | + \mu_2 | e_{12}\ \ p_0\ \ e_2\ |$$

$$= \mu_1 | e_{12}\ \ e_1\ \ \ p_0\ | - \mu_2 | e_{12}\ \ e_2\ \ p_0 | \tag{9}$$

This yields

$$\mu_1/\mu_2 = | e_{12}\ e_2\ \ p_0 | / | e_{12}\ e_1\ p_0 | \tag{10}$$

The right hand side does not depend on the plane Π (interestingly, we can also show it does not depend on p_0). This completes the proof.

Chapter 8

DYNAMIC PUSHBROOM STEREO VISION

Dynamic Pushbroom Stereo Vision for Surveillance and Inspection

Z. Zhu[1], G. Wolberg[1], and J. R. Layne[2]
[1]*Department of Computer Science, The City College of New York, New York, NY 10031*
[2]*Air Force Research Laboratory, 2241 Avionics Circle, WPAFB, Ohio 45433-7318*

Abstract: We present a dynamic pushbroom stereo geometry model for both 3D reconstruction and moving target extraction in applications such as aerial surveillance and cargo inspection. In a dynamic pushbroom camera model, a "line scan camera" scans across the scene. Both the scanning sensor and the objects in the scene are moving, and thus the image generated is a "moving picture" with one axis being space and the other being time. We study the geometry under a linear motion model for both the sensor and the object, and we investigate the advantages of using two such scanning systems to construct a dynamic pushbroom stereo vision system for 3D reconstruction and moving target extraction. Two real examples are given using the proposed models. In the first application, a fast and practical calibration procedure and an interactive 3D estimation method are provided for 3D cargo inspection with dual gamma-ray (or X-ray) scanning systems. In the second application, dynamic pushbroom stereo mosaics are generated by using a single camera mounted on an airplane, and a unified segmentation-based stereo matching algorithm is proposed to extract both 3D structures and moving targets from urban scenes. Experimental results are given.

Key words: video mosaicing, motion analysis, stereo vision, 3D reconstruction, moving target extraction, pushbroom stereo.

1. INTRODUCTION

Pushbroom images (or mosaics, when generated from video sequences) with parallel-perspective projections are very suitable representations for surveillance and/or security applications where the motion of the camera has a dominant translational direction. Examples include satellite pushbroom

A. Koschan et al. (eds.), 3D Imaging for Safety and Security, 173–199.

imaging[7], airborne video surveillance[24, 25], 3D reconstruction for image-based rendering[1], road scene representations[22, 24], under-vehicle inspection[3, 13], and 3D measurements of industrial parts by an X-ray scanning system[6, 15]. A pushbroom image/mosaic is a *parallel-perspective* image which has parallel projection in the direction of the camera's motion and perspective projection in the direction perpendicular to that motion. A pair of pushbroom stereo images/mosaics can be used for both 3D viewing and 3D reconstruction when they are obtained from two different oblique viewing angles. An advantageous feature of the pushbroom stereo model is that depth resolution is independent of depth[1, 25]. Therefore, better depth resolution can be achieved than with perspective stereo or the recently developed multi-perspective stereo with circular projection[12, 17, 18], given the same image resolution. We note that multi-perspective stereo with circular projection that is based on wide-baseline line cameras can achieve very accurate depth resolution for far-range airborne scenes[12]. However, in such a configuration depth resolution is still proportional to the square of depth. Therefore, the depth accuracy varies greatly for cases when ranges of depths are large, in such applications as cargo inspection or ground robot surveillance. In addition, the circular motion that is required is not the best form for scanning long cargo containers, or walking through large scale 3D scenes.

Using pushbroom *stereo* images/mosaics for 3D viewing and/or 3D reconstruction has been studied for satellite imaging, airborne video mosaicing, under-vehicle inspection, street scene modeling, and industrial quality assurance. However, as far as we know, previous work on the aforementioned stereo panoramas (mosaics) only deals with static scenes. Most of the approaches for moving target tracking and extraction, on the other hand, are based on interframe motion analysis and expensive layer extraction[21, 23]. In security and inspection applications, quite a few X-ray or gamma-ray cargo inspection systems have been put to practical uses[7, 8, 9]. In the past, however, cargo inspection systems have only had two-dimensional capabilities, and human operators made most of the measurements. No moving target detection capability has been explored. If we could build an accurate geometry model for a X-ray/gamma-ray imaging system, which turns out to be a linear pushbroom scanning sensor, accurate three-dimensional (3D) and possible dynamic measurements of objects inside a cargo container can be obtained when two such scanning systems with different scanning angles are used to construct a *dynamic linear pushbroom stereo system*. The 3D and motion measurements of targets add more value to today's cargo inspection techniques, as indicated in some online reports[7, 8, 9].

In this work, we present a unified geometric model for a dynamic linear pushbroom stereo system for applications where the movements of both

sensors and objects are involved. This raises very interesting problems since a pushbroom image is a spatio-temporal image with moving viewpoints, and thus the behavior of a moving object in the pushbroom image will be more complicated than in a conventional 2D spatial image captured in a single snapshot. We study the accurate geometric model of a linear pushbroom sensor when both the sensor and a 3D point move in 3D space at constant velocities. Then, we discuss why this model can be used for real applications where the motion of the objects can be well-approximated as piecewise linear within short periods of time.

In particular, we will study two examples using this model. In the first application, issues on 3D measurements using a linear pushbroom stereo system are studied for X-ray/gamma-ray cargo inspection. The closest work to ours is the x-ray metrology for industrial quality assurance by Noble et al.[15]. However, to our knowledge, our research presents the first piece of work in using linear pushbroom stereo for 3D gamma-ray or X-ray inspection of large cargo containers. Furthermore, the proposed dynamic pushbroom stereo model enables moving target extraction within cargo containers. In our study, we use the gamma-ray scanning images provided by the Science Applications International Corporation (SAIC)[16]. However, this does not imply an endorsement of this gamma-ray technology over others, for example, the X-ray technologies. In fact, the algorithms developed in this work can be used for pushbroom images acquired by X-ray or other scanning approaches as well.

In the second application, we study a dynamic pushbroom stereo mosaic approach for representing and extracting 3D structures and independent moving targets from urban 3D scenes. Our goal is to acquire panoramic mosaic maps with motion tracking information for 3D (moving) targets using a light aerial vehicle equipped with a video camera flying over an unknown area for urban surveillance. In dynamic pushbroom stereo mosaics, independent moving targets can be easily identified in the matching process of stereo mosaics by detecting the "out-of-place" regions that violate epipolar constraints and/or give 3D anomalies. We propose a segmentation-based stereo matching approach with natural matching primitives to estimate the 3D structure of the scene, particularly the ground structures (e.g., roads) on which humans or vehicles move, and then to identify moving targets and to measure their 3D structures and movements.

The rest of the text is organized as follows. In Section 2, the principle of the dynamic linear pushbroom imaging and then the model of dynamic pushbroom stereo are described. Section 3 discusses sensor calibration and 3D measurement issues for the gamma-ray cargo inspection application. Section 4 presents the stereo mosaicing approach to generate pushbroom stereo images from a single video sequence of an urban scene, and then to

extract both 3D and moving targets from a pushbroom stereo pair. Section 5 provides some concluding remarks and discussion.

2. DYNAMIC PUSHBROOM STEREO GEOMETRY

We start with the proposed dynamic linear pushbroom imaging geometry for both static and dynamic scenes. Then, we study pushbroom stereo geometry for static scenes followed by dynamic pushbroom stereo geometry for dynamic scenes.

2.1 Dynamic linear pushbroom geometry

A 3D point at time $t = 0$ is denoted as $P = (x,y,z)^t$ in a world coordinate system *o-xyz* (Figure *8-1*). It is viewed by a moving camera O_c-$X_cY_cZ_c$ with an optical axis Z_c and a focal length *f*, but only points on the $O_cY_cZ_c$ plane are recorded on a 1D scan line in the direction of *v*. The center of the linear image in the *v* direction is defined by a vertical offset p_v. The relation between the world coordinate system and the camera coordinate system located at the time $t = 0$ is represented by a rotational matrix **R** and a translational vector **T**. Both the camera and the point translate at constant velocities, $\mathbf{V_c}$ and $\mathbf{V_o}$ respectively. Both of them are represented in the world coordinate system. The relative motion between them is denoted as **V** in the camera coordinate system. Therefore, we have

$$\mathbf{V} = (V_x, V_y, V_Z)^t = \mathbf{R}(\mathbf{V_c} + \mathbf{V_o}) \tag{1}$$

Using the linear pushbroom model proposed in Gupta and Hartley[7], we have our *dynamic* linear pushbroom camera model:

$$\begin{pmatrix} u \\ wv \\ w \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & f & p_v \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1/V_x & 0 & 0 \\ -V_y/V_x & 1 & 0 \\ -V_z/V_x & 0 & 1 \end{pmatrix} (\mathbf{R}\,|\,\mathbf{RT}) \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \tag{2}$$

Note that the form is the same, but here we assume that both the camera and the point are moving during imaging, while in Gupta and Hartley[7] only the camera moves at a constant velocity.

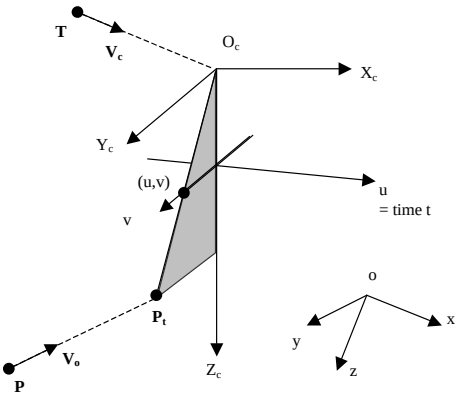


Figure 8-1. Dynamic linear pushbroom imaging geometry. A 3D point at time $t = 0$ is denoted as $P = (x,y,z)^t$ in the world coordinate system o-xyz. The scene is viewed by a moving camera O_c-$X_cY_cZ_c$ with an optical axis Z_c, but only those points that pass the $O_cY_cZ_c$ plane are recorded on a 1D scan line in the direction of v. Both the camera and the point translate at constant velocities, V_c and V_o respectively.

The linear pushbroom model for static scenes with general parameters has been studied thoroughly in Gupta and Hartley[7], including camera calibration, the epipolar geometry and fundamental matrix for a pushbroom stereo system. In our work, we are more interested in investigating the behavior of a moving target under a linear pushbroom imaging system, therefore extend the model to a dynamic version. Since it will be very complicated with a general model, for simplify the discussion we make the following assumptions of the orientation and motion direction of the camera, i.e.

$$\mathbf{R} = \begin{pmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{pmatrix}, \; V_c = (S,0,0) \tag{3a}$$

That is, the camera moves in the direction of the x-axis of the world coordinate system, and there are no tilt and roll angles between the two coordinate systems; the only rotation is around the y-axis. Suppose that $T = (T_x, T_y, T_z)^t$ and

$$V_o = (W_x, W_y, W_z)^t, \tag{3b}$$

then Eq. (2) can be written as our dynamic linear pushbroom imaging model:

$$u = \frac{(x - T_x) - (z - T_z)\tan\theta}{S + W_x - W_z \tan\theta},\ v = f\cos\theta \frac{y - T_y - W_y u}{z - T_z - W_z u} + p_v \qquad (4)$$

Note that the linear pushbroom camera system has parallel projection in the *u* direction, but has perspective projection in the *v* direction. Recall that *(x,y,z)* is the "initial" point location in the word at the time *t = 0*. The current location seen in the image (at the time *t = u*) can be calculated as

$$P_t = (x, y, z)^t - u(W_x, W_y, W_z)^t \qquad (5)$$

However, the moving point does not have to keep the linear motion starting from time *t=0* to make Eq. (4) valid. The time *t = u* is the only time it is seen in the current temporal-spatio *u-v* image, so the initial location is only an assumed location that will be useful in the dynamic pushbroom stereo model in the Section 2.3.

2.2 Linear pushbroom stereo for static scenes

First, we want to look at a point in a static scene, i.e., when $\mathbf{V_o} = (W_x, W_y, W_z)^t = 0$. If we have two linear pushbroom scanning systems that have two different sets of parameters (θ_k, $\mathbf{T}_k$, f_k, p_{vk}), k = 1, 2, we have from Eq. (4) the relations for a pair of correspondence points (u_k,v_k), k =1,2:

$$\begin{aligned} u_k &= \frac{x - T_{xk} - (z - T_{zk})\tan\theta_k}{S_k} \\ v_k &= f_k \cos\theta_k \frac{y - T_{yk}}{z - T_{zk}} + p_{vk} \end{aligned}, \ (k=1,2) \qquad (6)$$

Using sensor calibration (Section 3.1), we can find these parameters. Therefore, the depth of the point can be recovered as

$$z = \frac{d - d_0}{\tan\theta_1 - \tan\theta_2} \qquad (7)$$

where

$$d = S_2 u_2 - S_1 u_1 \qquad (8)$$

is defined as the *visual displacement* (measured in metric distance) of the point (*x,y,z*) as observed in the pair of stereo images, and

$$d_0 = (T_{x1} - T_{z1}\tan\theta) - (T_{x2} - T_{z2}\tan\theta_2)$$

is the fixed offset between two images. Note that Eq. (7) is acquired by only using the *u* coordinates of the stereo images, and the depth of any point is proportional to its visual displacement in the stereo pair. Thus, the depth resolution in a linear pushbroom stereo system is independent of depth. The epipolar geometry can be derived from Eq. (6), which has been proved to be of hyperbolic curves[7].

2.3 Linear pushbroom stereo for dynamic scenes

For a dynamic scene, with a moving point $\mathbf{V_o} = (W_x, W_y, W_z)^t \neq 0$, we need to estimate not only the depth *z* but also the motion $\mathbf{V_o}$ of the dynamic point. Without loss of generality, we assume $\mathbf{T} = (T_X, T_Y, T_Z)^t = \mathbf{0}$, $p_v = 0$, i.e., the camera is at the origin of the world coordinate system at time *t = 0*, and we have to find the center of perspective image by calibration. Therefore we have

$$u = \frac{x - z\tan\theta}{S + W_x - W_z\tan\theta}, \quad v = f\cos\theta\frac{y - W_y u}{z - W_z u} \tag{9}$$

Assume that two linear pushbroom scanning systems with two different sets of parameters (θ_k, f_k) scan the scene at the same time, starting at the same camera location (Figure *8-2*). For a pair of correspondence poins (u_k,v_k), *k*=1,2, of a 3D moving point *P (x,y,z)*, we have the following dynamic linear pushbroom stereo model:

$$u_k = \frac{x - z\tan\theta_k}{S + W_x - W_z\tan\theta_k}, v_k = f_k\cos\theta_k\frac{y - W_y u_k}{z - W_z u_k}, \ k = 1,2 \tag{10}$$

Hence the depth can be represented as

$$z = \frac{S + W_x}{\tan\theta_1 - \tan\theta_2}d_u + \frac{u_1\tan\theta_1 - u_2\tan\theta_2}{\tan\theta_1 - \tan\theta_2}W_z \tag{11}$$

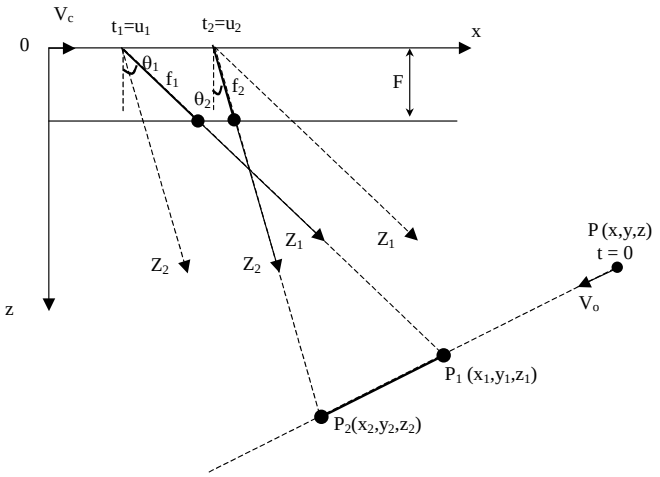


Figure 8-2. Dynamic linear pushbroom stereo geometry. The *y* axis points out of the paper.

where

$$d_u = u_2 - u_1 \tag{12}$$

is the *temporal displacement* (measured in numbers of scan lines) as observed in the stereo pair. In the dynamic linear pushbroom stereo model, we only need to assume that the point *P* moves at a constant velocity $\mathbf{V_o}$ from time $t = u_1$ to time $t = u_2$. These are the only two time instants that points can be seen by the stereo system (Figure *8-2*). The locations of the points at these two time instants are

$$P_k = (x, y, z)^t - u_k (W_x, W_y, W_z)^t \text{ , k = 1, 2} \tag{13}$$

In real applications with a moving video camera, if we extract two scanlines in each 2D perspective image to simulate the two scanners with two angles[26], we will have F= $f_k \cos\theta_k$, k=1,2, where F is the real focal length of the perspective camera, and its optical axis is parallel to the z axis of the world (Figure *8-2*).

There are two remaining questions: how can we know a point is a dynamic point, and then how can we estimate the motion velocity of the point? We know that if the point is static, the correspondence point of (u_1, v_1) in the second image, (u_2,v_2), will on its parabolic epipolar line. Or in the ideal model of Eq. (10) with F= $f_k \cos\theta_k$, k=1, 2, on a horizontal straight line (i.e., $v_1 = v_2$). Therefore, in the general case (i.e. $W_y \neq 0$ or $W_z \neq 0$), a moving point will violate the epipolar geometry of the static scene condition.

Note that if the point moves in *xoy* plane, the correspondence point will be on a straight line $v_2 - v_1 = -FW_y(u_2 - u_1)/z$ instead of on the parabolic epipolar curve. However, when the motion of the point is in the direction of the camera's motion, i.e., both W_y and W_z are zeros, the moving point will also obey the epipolar geometry of static assumption. But in this special case we can use a depth anomaly test (for aerial surveillance where moving objects are on the ground) or depth difference constraints (for known objects), as below.

In summary, the velocity of the dynamic point can be estimated as the following two cases, depending on if $W_z=0$ or not.

Case1. The point moves in the *xoy* plane (i.e., $W_z=0$). The following two steps will be used in estimating the motion of the point:

(1). If z can be obtained from its surroundings, or if the depth difference between two points (e.g. "height" of an object) is known, then W_x can be calculated from Eq . (11);

(2). Given z, both y and W_y can be obtained from the equations for v_k (k=1, 2) in Eq. (10) with a pair of the linear equations for v_1 and v_2.

Case 2. General case ($W_z \neq 0$). We also have two steps to fulfil the task:

(1). If three points on the same moving object have the same depth (or two points have known depths), then z, W_x and W_y can be obtained from Eq. (11) with a linear equation system, and then x from an equation for u_k (k = 1 or 2) in Eq. (10);

(2). Given W_x, x and z, both y and W_y can be obtained from the equations for v_k (k=1,2) in Eq. (10) with a pair of linear equations for v_1 and v_2.

3. GAMMA-RAY LINEAR PUSHBROOM STEREO

The system diagram of the gamma-ray cargo inspection system is shown in Figure *8-3* (b). A 1D detector array of 256 NaI-PMT probes counts the gamma-ray photons passing through the vehicle/cargo under inspection from a gamma-ray point source. Either the vehicle/cargo or the gamma-ray system (the source and the detector) moves in a straight line in order to obtain a 2D scan of gamma-ray images. The geometry of the system is shown in Figure *8-3* (a), which is essentially the same as in Figure *8-1*.

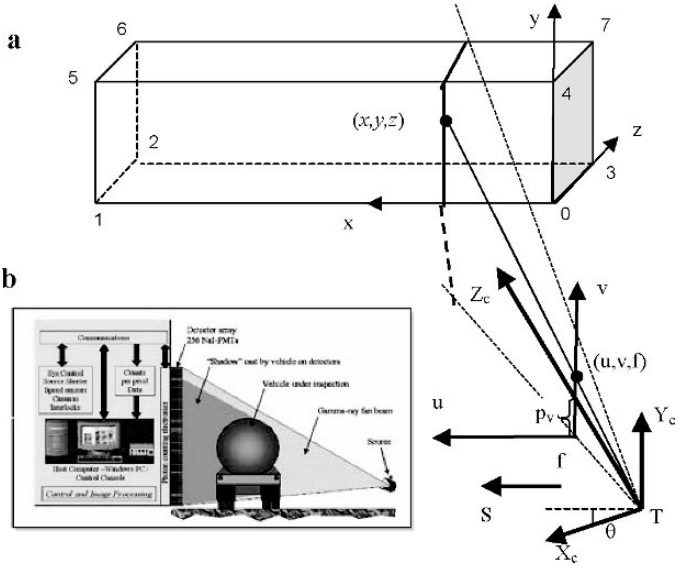


Figure 8-3. Linear pushbroom sensor model of a gamma-ray cargo inspection system. (a) The model; (b) the system (Courtesy SAIC, San Diego, CA, USA).

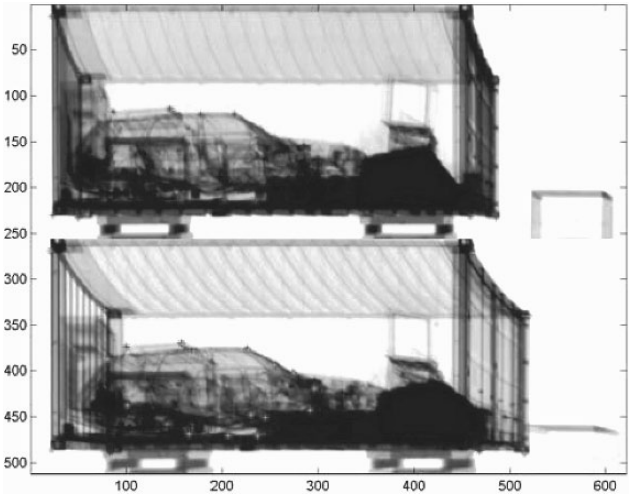


Figure 8-4. Real gamma-ray images from two different scanning angles - ten and twenty degrees (Images courtesy SAIC, San Diego, CA, USA). Each image has a size of 621x256 pixels, i.e., 621 scans of the 256-pixel linear images. This figure also shows the matching of two sets of points in the two images. (See also Plate 25 in the Colour Plate Section)

A dual-scanning system is a *linear pushbroom stereovision system*. It can be constructed with two approaches: two linear pushbroom scanning sensors with different scanning angles, or a single scanning sensor to scan the same cargo twice with two different scanning directions. The first approach can be used to detect moving targets inside a cargo container. Figure *8-4* shows two real gamma-ray images, with different scanning angles – ten and twenty degrees, respectively. Each image has a size of 621x256 pixels, i.e., 621 scans of the 256-pixel linear images. For static scene points, depths (z) can be calculated by using Eq. (7), and further x and y coordinates can be calculated by Eq. (6). Note that here we assume that the two scans are independent of each other and thus have two different sets of imaging parameters. Two important steps are described in the following for 3D measurements in cargo inspection: sensor calibration and stereo matching.

3.1 Sensor calibration

For each scanning setting, the following parameters are required for 3D estimation: the focal length f, the image center p_v, the scanning angle θ, the scanning speed S, and the initial sensor location (T_x,T_y,T_z). In order to fulfill this task, we need to know a set of 3D points $\{(x_i, y_i, z_i)\}$ and their corresponding image points $\{(u_i, v_i)\}$, i=1, 2,…, N. Our calibration method only needs to know the dimension of the container, which is

length(x)*height(y)*depth(z)=20*8*8 (ft^3).

Then we locate the 8 vertices of the rectangular container (refer to Figure *8-3* (a)) in each gamma-ray image by manually picking up the 8 corresponding image points.

An interesting property of the linear pushbroom sensor is that the two equations in Eq. (4) can work independently (with $W_x=W_y=W_z=0$). Therefore, in calibrating the sensor, we first obtain the "parallel projection parameters" from u and then the "perspective projection parameters" from v. The parallel projection equation can be turned into a linear equation with three unknowns, i.e., S, $\tan\theta$ and T_x-$T_z\tan\theta$:

$$u_i S + z_i \tan\theta + (T_x - T_z \tan\theta) = x_i \tag{14}$$

Given more than three pairs of points (i=1, 2,…, N where $N \geq 3$)., we can solve the linear system to find the three unknowns by using the least square method. Similarly, the perspective equation leads to a linear equation with five unknowns, i.e. f, fT_y, p_v, p_vT_z and T_z:

$$(y_i \cos\theta) f - \cos\theta(fT_y) + z_i p_v - (p_v T_z) + v_i T_z = v_i z_i \quad (15)$$

With the known θ and given more than five pairs of points ($i=1, 2,\ldots, N$ where $N \geq 5$), we can solve the linear equation system. Note that from Eq. (14) we can only find the values of the speed S and the angle θ and a combined parameter T_x-$T_z tan\theta$. Nevertheless, this is sufficient for obtaining the depths of points using Eq. (7). Table *8-1* shows the results of the "parallel parameters" for the two settings corresponding to the two images in Figure *8-4*. All the rest of the parameters, including T_x, can be obtained after solving Eq. (15), in order to calculate the x and y coordinates of 3D points by using Eq. (6). Table *8-2* shows the "perspective parameters" and the T_x values for the two settings.

Table *8-3* shows the 3D measurements using the *image* point pairs used for calibration between two views, the ten-degree and the twenty-degree images. The purpose is to show how accurate the pushbroom stereo model and the calibration results are. The numbers of the points listed in Table *8-3* are labelled in Figure *8-3* (a) for comparison. For the container with a dimension of 20x8x8 ft^3, the average errors in depth z, length x and height y are 0.064 ft, 0.033 ft, and 0.178 ft, respectively, indicating that the pushbroom modeling and calibration is accurate enough for 3D measurements.

Table 8-1. Parallel projection parameters

Images	*S (ft/pixel)*	*tanθ*	*θ (degrees)*	T_x-$T_z tan\theta$
10-degrees	0.04566	0.16552	9.3986	-7.283
20-degrees	0.04561	0.34493	19.031	-7.309

Table 8-2. Perspective projection parameters

Images	*F (pixels)*	T_y *(ft)*	p_v*(pixels)*	$p_v T_z$	T_z *(ft)*	T_x*(ft)*
10-degrees	441.24	-0.42881	17.787	-191.78	-15.141	-9.789
20-degrees	456.18	-0.41037	19.250	-198.03	-15.000	-12.48

Table 8-3. 3D measurements of the test points (all measurements are in feet)

No	*x*	*y*	*z*	*dx*	*dy*	*dz*
0	-0.033	-0.179	-0.063	-0.033	-0.179	-0.063
1	20.033	-0.177	0.063	0.033	-0.177	0.063
2	19.967	-0.152	7.936	-0.033	-0.152	0.064
3	0.033	-0.204	8.064	0.033	-0.204	0.064
4	-0.033	7.787	-0.063	-0.033	-0.213	-0.063
5	20.033	7.856	0.063	0.033	-0.144	0.063
6	19.967	7.799	7.936	-0.033	-0.201	0.064
7	0.033	7.844	8.064	0.033	-0.156	0.064

Note that the accuracy of the estimation only reflects the errors in sensor modeling and calibration. No image localization errors have been included. The depth error δz introduced by image localization error δu can be estimated as the first derivative of z with respect to u using Eqs. (7) and (8), that is

$$\delta z = \frac{S}{\tan\theta_1 - \tan\theta_2}\delta d \tag{16}$$

In Eq. (16) we assume that two scans share the same speed (i.e. $S_1=S_2=S$), which are almost true for our example in Figure *8-4* (see Table *8-1*). In this example, one-pixel image localization error introduces an error of 0.254 ft in depth estimation, using the parameters in Table *8-1*. Analysis in more details on the calibration accuracy can be found in Zhu et al.[27].

3.2 3D measurements and visualization

Fully automated 3D measurements of objects from gamma-ray radiographic images are difficult since the objects in the images are "transparent." Some work has been reported in literature (e.g., Mayntz et al.[14]) in using optical flow on X-ray fluoroscopy images for restoration of motion blur, but the motion parallax in their case is small. However, in our case of widely separated parallel viewing for 3D reconstruction two different views will give very different object adjacencies and occlusions. This is an important issue and will be our future work. In our current work, we have tested an interactive approach for stereo matching and visualization.

Our semi-automated stereo matching approach includes three steps: interactive point selection, automated matching, and interactive matching correction. Instead of generating a dense "depth" map from a pair of gamma-ray images, we have designed an interactive user interface for selecting and measuring objects of interest. For the automated stereo matching step, we use sum of square difference (SSD) criterion on normalized images.

Figure *8-4* shows the process of semi-automated stereo matching for the pair of the ten- and twenty-degree images. After a point in the first image is picked up by the user (marked by a red star in the first image of Figure *8-4*), its match in the second image is automatically searched along the epipolar line of the pushbroom stereo, derived from Eq. (6). The search range is pre-determined from Eq. (7) by using the knowledge that all the objects are within the cargo container. The size of the correlation window can be determined by the user interactively. We have tried different window sizes (3x3, 9x9, 11x11, etc.) and found that 11x11 was the best for this example.

The automated matches are marked by blue stars in the second image of Figure *8-4*.

After the program finds the automated matching points, the user could correct the match if necessary (marked by green stars in the second image of Figure *8-4*). In Figure *8-4*, most of the automated matches are "considered" to be correct where the green marks completely overlap the blue marks. The points that are considered incorrect are those whose matches could be identified by human eyes but whose appearances are quite different between two images for automated matching. On the other hand, a few point matches that are considered to be "correct" might be incorrect; but, we have no way to correct them due to the large differences between two views (e.g., the point pair identified by arrows). In Figure *8-4*, all eight vertices of the cargo container are selected for stereo matching as well as a few points around the boundary of a vehicle inside the cargo container. Note that the four of the eight points on the top of the container we select here are slightly different from the ones for calibration due to the requirements of an 11x11 window centered at each point.

Together with the stereo matching interface, the reconstructed 3D structures are rendered as wire frames in 3D. For each set of points that are selected for stereo matching, a connected 3D line-frame representation is generated (Figure *8-5*). In Figure *8-5*, the black rectangular frame is the reconstruction of the cargo container using the calibration image data for the ten-and twenty-degree images. The red line frame is generated from the 3D measurements by the automated stereo match algorithm.

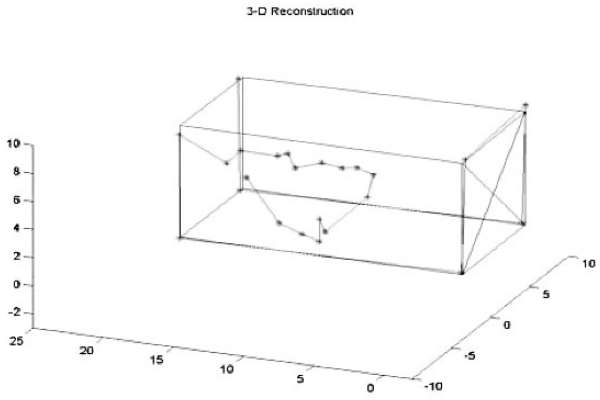


Figure 8-5. 3D measurements and visualization of objects inside the cargo container. The black rectangular frames show the cargo container constructed from the test data in Table *8-3*. The red lines (with stars) show the 3D estimates from automated stereo matches, for the cargo container and an object inside.

It is clearly shown that the automated stereo matches provide very good 3D measurements for the cargo container and the objects inside. With a 3D visualization, 3D measurements, for example, of sizes and shapes are made simple by using the most convenient views. Object measurements and identification will be our future work.

4. DYNAMIC STEREO MOSAICS FROM VIDEO

In this section, we will see how we can use the dynamic linear pushbroom stereo model by generating pushbroom stereo mosaics from an aerial video sequence. Our goal is to acquire geo-referenced mosaic maps with motion tracking information for 3D (moving) targets using a light aerial vehicle flying over an unknown 3D urban scene with moving targets (vehicles). The motion of the camera has a dominant direction of motion (as on an airplane or ground vehicle), but 6 DOF motion is allowed.

4.1 Linear pushbroom mosaic generation

First, for introducing the principle, we assume the motion of a camera is an ideal 1D translation with constant speed, the optical axis is perpendicular to the motion, and the frames are dense enough. Then, we can generate two spatio-temporal images by extracting two columns of pixels (perpendicular to the motion) at the leading and trailing edges of each frame in motion (Figure *8-6*). The mosaic images thus generated are *parallel-perspective*, which have perspective projection in the direction perpendicular to the motion and parallel projection in the motion direction.

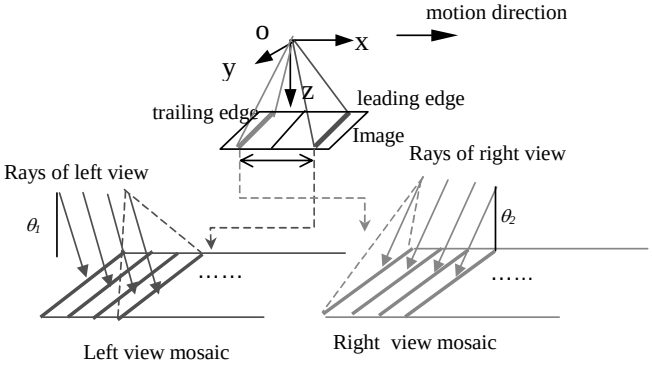


Figure 8-6. Principle of the parallel-perspective pushbroom stereo mosaics.

In addition, these mosaics are obtained from two different oblique viewing angles (θ_1 and θ_2) of a single camera's field of view, so that a stereo pair of "left" and "right" mosaics captures the inherent 3D information. The geometry in this ideal case (i.e., 1D translation with constant speed) is the same as the linear pushbroom stereo model represented in Eq. (10), where $\boldsymbol{T} = 0$, and the camera focal length $F = f_k \cos_k$, $k=1, 2$.

In real applications, there are two challenging issues. The first problem is that the camera usually cannot be controlled with ideal 1D translation and camera poses are unknown; therefore, camera orientation estimation (i.e., dynamic calibration) is needed. Bundle adjustment techniques[20] can be used for camera pose estimation, sometimes integrated with the geo-referenced data from GPS and INS when available. The second problem is to generate dense parallel mosaics with a sparse, uneven, video sequence, under a more general motion. The ray interpolation approach we proposed in Zhu et al.[26] can be modified to generate a pair of linear pushbroom stereo mosaics under the obvious motion parallax of a translating camera.

Figure *8-7* shows how the modified ray interpolation approach works for 1D cases. The 1D camera has two axes – the optical axis (Z) and the X-axis. Given the known camera orientation at each camera location, one ray with a given oblique angle θ can be chosen from the image at each camera location to contribute to the parallel mosaic with this oblique angle θ. The oblique angle is defined against the direction perpendicular to the *mosaicing direction*, which is the dominant direction of the camera path (Figure *8-7*, same as in Figure *8-2*). But the problem is that the "mosaiced" image with only those existing rays will be sparse and uneven (i.e., not linear) since the camera arrays are usually not regular and dense.

Therefore, *interpolated* parallel rays between a pair of existing parallel rays (from two neighboring images) are generated by performing local matching between these two images so that we generate rays at dense and equal intervals along the mosaicing direction, i.e., with a linear pushbroom geometry as if from a moving camera with constant velocity. Note that the velocity is measured in meters per scanline (ray) instead of time. The assumption is that we can find at least two images to generate the parallel rays. Such interpolated rays are shown in Figure *8-7*, where Ray I_1 is interpolated from Image A and Image B for a static point P_1 by back-projecting its corresponding pair in images A and B, and Ray I_2 is interpolated from Image C and Image D for a moving point P_2 in the same way. By assume that the moving point undergoes a linear motion between the two views (from P_2^C to P_2^D), the interpolated ray captures its correct location (P_2^I) at that ray (i.e., "time").

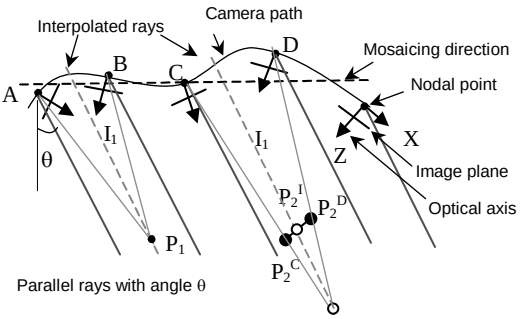


Figure 8-7. Ray interpolation for linear pushbroom mosaicing with parallel projection.

The extension of this approach to 2D images is straightforward, and a region triangulation strategy similar to the one proposed in Zhu et al.[26] can be applied here to deal with 2D cases. Since the process of parallel ray interpolation as a "temporal re-sampling" synthesizes each mosaic with a linear time axis in the direction of *mosaicing*, for both the static and moving objects, a pair of such mosaics satisfies dynamic linear pushbroom stereo geometry represented by Eq. (9). In the following text, we assume that pushbroom stereo mosaics have been generated and that the epipolar geometry obeys that of the linear pushbroom stereo under 1D translation. We will then focus on the method to perform both 3D reconstruction and moving target extraction from urban scenes from a pair of dynamic pushbroom stereo mosaics.

4.2 Segmentation-based stereo matching

Simple window-based correlation approaches do not work well for man-made scenes, particularly across depth boundaries and for textureless regions. An adaptive window approach[10] has been proposed which selects at each pixel the window size that minimizes the uncertainty in disparity estimates in stereo matching. A nine window approach has also been proposed by Fusiello et al.[4] in which the point in the right image with the smallest SSD error amongst the 9 windows and various search locations is chosen as the best estimate for the given point in the left image. Recently, color segmentation has been used as a global constraint for refining an initial depth map to get sharp depth boundaries and to obtain depth values for textureless areas[19] and for accurate layer extraction[11]. In this text, we provide a segmentation-based approach using *natural matching primitives* to extract 3D and motion of the targets. The segmentation-based stereo matching algorithm is proposed particularly for the dynamic pushbroom stereo geometry to facilitate both 3D reconstruction and moving target extraction

from 3D urban scenes. However, the proposed natural matching primitives are also applicable to more general scenes and other types of stereo geometry.

Dynamic pushbroom stereo mosaics provide several advantages for 3D reconstruction and moving target extraction (please refer to Section 2.3). The stereo mosaics can be aligned on a dominant plane (e.g., the ground), as in Zhu et al.[26]. All the static objects obey the epipolar geometry, i.e., along the epipolar lines of pushbroom stereo. An independent moving object, on the other hand, either violates the epipolar geometry if the motion is not in the direction of sensor motion or at least exhibits 3D anomaly - hanging above the road or hiding below the road even if motion happens to be in the same direction of the sensor motion[28]. With all these geometric constraints in mind, we propose a segmentation-based approach to integrate the estimation of 3D structure of an urban scene and the extraction of independent moving objects from a pair of dynamic pushbroom stereo mosaics. The approach starts with one of the mosaics, for example, the left mosaic, by segmenting it into homogeneous color regions that are treated as planar patches. We apply the mean-shift-based approach proposed by Comanicu and Meer[2] for color segmentation. Then, the stereo matching is performed based on these patches between two original color mosaics. The basic idea is to only match those pixels that belong to each region (patch) between two images in order to both produce sharp depth boundaries for man-made targets and to facilitate the searching and discrimination of the moving targets (each covered by one or more homogeneous color patches). The proposed algorithm has the following five steps.

Step 1. *Matching primitive selection.* After segmenting the left image using the mean-shift method, homogenous color patches and then the natural matching primitives are extracted.

Step 2. *Epipolar test*. Using pushbroom epipolar geometry in stereo matching, static objects will find correct matches but moving objects will be outliers without correct "matches".

Step 3. *3D anomaly test*. After ground surface fitting (and road detection), moving objects in the same motion direction will exhibit wrong 3D characteristics (hanging above roads or hiding below roads).

Step 4. *Motion extraction.* Search matches for outliers (which could be moving objects) with a 2D and larger search range, or along the road directions (if available).

Step 5. *3D estimation.* Using the dynamic pushbroom stereo method proposed in Section 2.3, the 3D structures and motion of moving objects could be derived.

In the following two subsections, we detail two important issues in the segmentation-based stereo matching approach: natural matching primitive selection, and an integrated analysis of 3D structure and motion for both static and moving targets.

4.3 Natural matching primitives

We use color segmentation to obtain natural matching primitives for both 3D reconstruction and moving target extraction. The selection and matching of the natural matching primitives includes the following five sub-steps.

(1) *Segmentation and Interest point extraction*. The left mosaic is segmented into homogeneous color regions using the mean-shift approach by Comanicu and Meer[2]. We assume that each homogeneous color region (patch) is planar in 3D. However, each planar surface in 3D may be divided into several color patches. Then the boundary of each region is traced as a close curve. All the neighborhood regions are also connected with the region in processing for further use. Finally we use a line fitting approach to extract interest points along the region's boundary. The boundary is first fitted with connected straight-line segments using an iterative curve splitting approach. The connecting points between line segments are defined as interest points.

(2) *Natural window definition*. Each interest point *p(u,v)* of a region **R** in consideration will be used as the center of an *m*×*m* rectangular window in the left mosaic. Only those points that are within the window, inside the regions, or on the boundary will be used for matching (Figure *8-8*) in order to keep sharp depth boundaries. The window is defined as a *natural matching window*, and the set of pixels involved in matching is called a *natural matching primitive*. To facilitate the computation of correlation for stereo matching, we define a region mask ***M*** of size *m*×*m* centered at that interest point such that

$$M(i,j) = \begin{cases} 1, \text{if } (\mathrm{u}+\mathrm{i}, \mathrm{v}+\mathrm{j}) \in \mathbf{R} \\ 0, \text{otherwise} \end{cases} \tag{17}$$

We changed the size m of the natural window depending on the sizes of the regions. In our experiments, we use m = 23 for large regions (with diameter >=23) and m = 15 for small regions. We also want to include a few more pixels (1-2) around the region boundary (but not belonging to the region) so that we have sufficient image features to match. Therefore, a dilation operation will be applied to the mask ***M*** to generate a region mask covering pixels across the depth boundary. Figure *8-8* illustrates four such windows for the four interest points on the top region of the box. Note the yellow-shaded portions within each rectangular window, indicating that the pixels for stereo matching cover the depth boundaries.

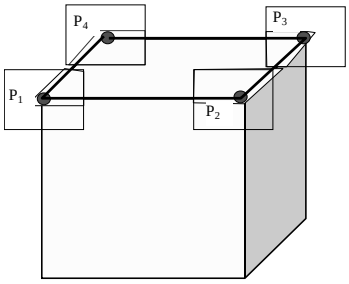


Figure 8-8. Natural matching primitive. (See also Plate 26 in the Colour Plate Section)

(3) *Natural window-based correlation.* Let the left and right mosaics be denoted as $\boldsymbol{I_1}$ and $\boldsymbol{I_2}$, respectively. The weighted cross-correlation, based on the natural window centered at *p(u,v)* in the left mosaic, is defined as

$$C(d_u, d_v) = \frac{\sum_{i,j} M(i,j) I_1(u+i, v+j) I_2(u+i+d_u, v+j+d_v)}{\sum_{i,j} M(i,j)} \tag{18}$$

Note that we still carry out correlation between two color images (mosaics), but only on those interest points on each region boundary and only with those pixels within the region and on the boundaries. This equation works for both static objects when the searching of correspondences is along epipolar lines of the pushbroom stereo and also for moving targets when the searching should be in 2D and with a larger search range. In the real implementation, we first perform matches with epipolar constraints of the pushbroom stereo, and those without good matches with be treated as "outliers" for further examination to see whether or not they are moving objects.

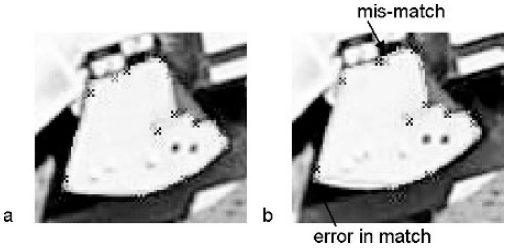


Figure 8-9. An example of region matching results. The matches are marked as "X", with corresponding colors.

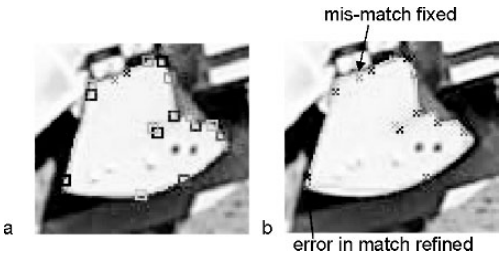


Figure 8-10. An example of surface fitting results.

Figure *8-9* shows a real example of natural-window-based stereo matching result for a static object (top of a building). The 19 selected interest points and their correspondences are marked on the boundaries in the left and right images, respectively. One mismatch and a small error in match are also indicated on images.

4.4 Surface fitting and motion estimation

Assuming that each homogeneous color region is planar in 3D, we fit a 3D plane for each region after we obtain the 3D coordinates of the interest points of the region using the pushbroom stereo geometry (assuming that it is static, i.e., $\boldsymbol{W}= \mathbf{0}$ in Eqs. (10) and (11). Seed points (≥ 3) are selected for plane fitting based on their correlation values. Then, the 3D coordinates of all the interest points are refined by constraining them on the fitted plane. Then, using the 3D plane information the region in the left mosaic can be warped to the right image to evaluate the matching and the fitting. Figure *8-10* shows the results of fitting and back-projection of the fitted region onto the right image. The 15 seed interest points (out of 19) used for planar fitting are indicated on the left image as squares. Both the mismatch and the small error in the initial match are fixed. Note that an iterative approach could be applied here to refine the matches after the initial surface fitting by using the evaluation of the warping from the left to the right mosaics and also by using the occlusion constraints from neighborhood regions, which have been obtained in the region tracing step in Section 4.3. For example, the selection of the seed points for surface fitting can be refined by removing those points that could be on occluding boundaries after we check the initial 3D surface relations and adding some other points that have reliable matches after image warping evaluation. Neighborhood regions can also be merged into a single plane if they share the same planar equation, with some error tolerant range.

After region refinement and merging, large and (near) horizontal ground regions can be easily identified. It is also possible to analyze the shape of the

ground regions to estimate road directions in which vehicles move. For those smaller neighborhood regions that happen to move in the same direction as the camera, it will have large depth differences from the surrounding ground regions when treated as static objects. This 3D anomaly can be used to identify those regions as moving objects. By assuming that their depths are the same as the surroundings, their motion parameters can be estimated using the method described in Section 2.3. For those "outliers" that do not find matches in the first pass (along epipolar lines), searching for matches can be performed along possible road directions (if obtained from the surrounding ground regions), or simply performed in a much larger 2D searching range.

4.5 Experimental results

We have performed preliminary experiments for stereo matching and moving object extraction on pushbroom stereo mosaics from real video sequences. Figure *8-11* (a) shows a pair of stereo mosaics from a video sequence that was taken when the airplane was about 300 meters above the ground. The viewing angles of the two parallel viewing directions for this pair of pushbroom mosaics are about 0.0 degrees ("left view") and 1.4 degrees ("right view"), respectively. Figure *8-11* (b) shows the corresponding dense depth map for the entire mosaic (about 4K*1.5K pixels), with sharp depth boundaries and dense depth values for buildings with large depth ranges (0-50 meters) and textureless surfaces.

Figure *8-12* shows the results of a close-up window indicated in the stereo mosaics in Figure *8-11* (a). In Figures *8-12* (a) and (b), the dynamic pushbroom stereo pair has both stationary buildings and ground vehicles moving in different directions. Figures *8-12* (c) and (d) show the segmentation result of the left image in Figure *8-12* (a), where the color label image is shown in (c) and the region boundaries are shown in (d). Note that a planar 3D region may be segmented into several color patches.

Figure *8-12* (e) shows the depth map of the static regions. Note that many regions, particularly those on top of each building are correctly merged, and the depth boundaries are clearly sharp and accurate. Figure *8-12* (f) shows the matched moving targets marked on the left mosaiced image, in blue and red, respectively. The moving vehicles exhibit much larger motion magnitudes and obvious different motion directions from the camera' motion direction.

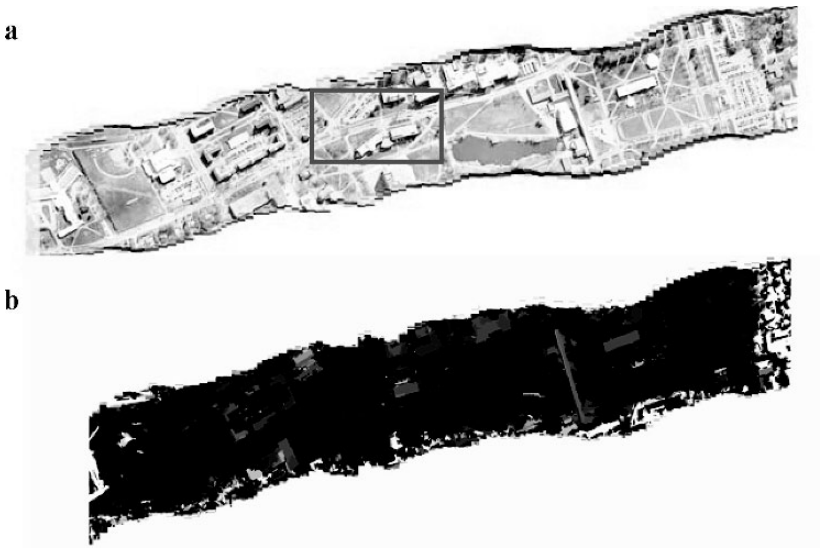


Figure 8-11. Dynamic pushbroom stereo mosaics and depth map (a) stereo mosaics: left view in the green/blue channels and right view in the red channel of a RGB image for stereo viewing; (b) depth map. (See also Plate 27 in the Colour Plate Section)

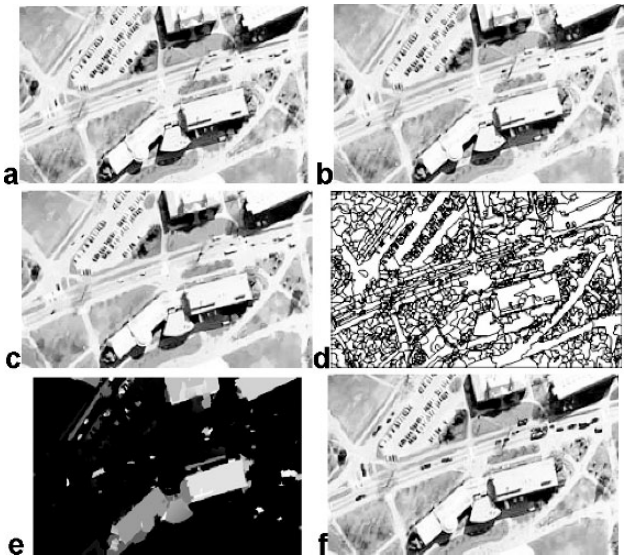


Figure 8-12. Content-based 3D mosaic representation: results for a portion of the stereo mosaics marked in Figure *8-11*: (a) left color mosaic; (b) right color mosaic; (c) and (d) left color labels and region boundaries; (e) depth map of static regions; (f) moving targets (motion: blue to red). Note how close the color label image to the original color image is.

5. CONCLUSIONS AND DISCUSSIONS

We have built the imaging geometry model of dynamic scenes observed by a linear pushbroom imaging sensor, in which only one scanline is obtained at each time instant. Examples of this kind of imaging can be find in applications such as X-ray/gamma-ray cargo inspection and airborne or ground surveillance where a moving sensor is applied, and the observed scenes include moving targets of interest. We generalize the linear pushbroom camera model proposed by Gupta and Hartley[5] for satellite imaging to the proposed *dynamic linear pushbroom imaging model* to deal with close-range scenes and moving targets. Since each image captured by such a sensor is a spatio-temporal image, where different parts of a moving object are viewed at different times and different viewpoints, it is interesting to study the 3D and motion estimation problems with a stereo vision system with such imaging geometry.

We present our *dynamic pushbroom stereo vision model* under a linear camera motion model and piece-wise object motion models. We have shown that such a stereo system has uniform depth resolution. We also provide methods to extract both 3D and motion information from a dynamic pushbroom stereo image pair.

Based on the proposed models, we have studied two examples for surveillance and inspection. In the first example, we present a practical approach for 3D measurements in gamma-ray (or X-ray) cargo inspection. Thanks to the constraints of the real scanning system, we model the system by using a linear pushbroom sensor model with only one rotation angle instead of three. This greatly simplifies the calibration procedure and increases the robustness of the parameter estimation. Using only the knowledge of the dimensions of the cargo container, we can automatically calibrate the sensor and find all the sensor parameters, including the image center, the focal length, the 3D sensor starting location, the viewing direction, and the scanning speed. The sensor modeling and calibration is accurate enough for 3D measurements. Then, a semi-automated stereo reconstruction approach is proposed to obtain 3D measurements of objects inside the cargo. With both the interactive matching procedure and the 3D visualization interface, the 3D measurements for cargo inspection could be put into practical use.

In the second example, we present a new approach to extract both 3D structure and independent moving targets from long video sequences captured by an airborne camera. The principle of dynamic pushbroom stereo mosaics is presented. Based on the properties of the dynamic pushbroom stereo, we propose a new segmentation-based stereo matching approach for both 3D reconstruction and moving target extraction from a pair of dynamic

pushbroom stereo mosaics for urban scenes. A simple yet effective natural matching primitive selection method is provided. This method is effective for stereo matching of man-made scenes, particularly when both 3D facilities and moving targets need to be extracted. We discussed the natural-primitive-based matching approach in the scenario of parallel-perspective pushbroom stereo geometry, but apparently the method is also applicable to other types of stereo geometry such as perspective stereo, full parallel stereo, and circular projection panoramic stereo.

The dynamic linear pushbroom stereo model provides a new imaging geometry that can find applications in widely used scanning systems for surveillance and inspection, including X-ray, gamma-ray, visible video, and IR video where the sensors are moving. This model unifies both the advantages of uniform depth resolution of the pushbroom stereo and the capability of independent moving target detection of the extended model to a dynamic version.

We have shown some early but promising results for two real applications. Several research issues need to be further investigated. First, we only provide methods to infer 3D and motion for dynamic objects using some physical constraints (Section 2.3). More general methods are needed to infer both 3D and independent motion using the dynamic pushbroom stereo. Second, little work exists in performing stereo matching on gamma-ray or X-ray images. We have not applied the dynamic model in this type of imaging system (Section 3). Automated stereo matching and moving target extraction methods for X-ray /gamma-ray pushbroom stereo will be a further research issue. Knowledge of physics and optics in generating the radiographic images could be very helpful in advancing this direction of research. Third, the experiments we performed for 3D and moving target extraction from aerial video (Section 4) are based on the model of generalized stereo mosaics we have previously proposed[26], in which the viewpoints are along a 3D curve instead of a more desirable straight line as in the linear pushbroom imaging model we proposed in this text. Automated generation of truly linear pushbroom stereo mosaics from a video sequence requires accurate camera orientation estimation of many frames of a lone video sequence and full ray interpolation to generate true linear pushbroom geometry. This will allow an analysis of the accuracy of the 3D and motion recovery using the dynamic pushbroom stereo model. All of these are our on-going efforts.

ACKNOWLEDGEMENTS

This work is supported by the Air Force Research Laboratory (AFRL) under the RASER Program (Award No. FA8650-05-1-1853) and the PSC-CUNY Research Award Program. It is also partially supported by an AFRL Human Effectiveness Program (Award No. F33615-03-1-63-83), by the NSF CRI Program under Grant No. CNS-0551598, by Army Research Office (ARO) under Award No. W911NF-05-1-0011, by the CUNY Graduate Research Technology Initiative (GRTI) program and PSC-CUNY. Thanks are also given to Hao Tang, Bing Shen, Li Zhao and Jiaye Lei for implementing the algorithms of the two applications, and to Robert Hill for proofreading the manuscript, all at the City College of New York. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. However, the views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the U.S. Government.

REFERENCES

1. J. Chai and H.–Y. Shum, 2000. Parallel projections for stereo reconstruction. In *Proc. CVPR'00*: II 493-500.
2. D. Comanicu and P. Meer, 2002. Mean shift: a robust approach toward feature space analysis. *IEEE Trans. PAMI*, May 2002.
3. P. Dickson, J. Li, Z. Zhu, A. Hanson,, E. Riseman, H. Sabrin, H. Schultz, and G. Whitten, Mosaic generation for under-vehicle inspection. *IEEE Workshop on Applications of Computer Vision*, Orlando, Florida, Dec 3-4, 2002.
4. A. Fusiello,. V. Roberto, and E. Trucco, 1997. Efficient stereo with multiple windowing, In *IEEE CVPR*: 858-863
5. R. Gupta and R. Hartley, 1997. Linear pushbroom cameras, *IEEE Trans PAMI*, 19(9), Sep. 1997: 963-975
6. R. Gupta, A. Noble, R. Hartley, J. Mundy, A. Schmitz, 1995. Camera calibration for 2.5-D X-ray metrology. In *Proc. ICIP'95*, Vol. 3, Oct 23 - 26, 1995 Washington D.C.
7. W. Hardin, Cargo Inspection: Imaging Solutions Wait for Government's Call, Machine Vision Online, Dec 2002.
8. W. Hardin, US Seaports: Finding the Needle in Hundreds of Haystacks, Machine Vision Online, June 2004.
9. Hitachi, Cargo container X-ray inspection systems, *Hitachi Review*, 53(2) June 2004: 97-102. http://www.hitachi.com/rev/field/industriasystems/2006638_12876.html
10. T. Kanade and M. Okutomi, 1991. A Stereo Matching Algorithm with an Adaptive Window: Theory and Experiment, In *Proc. IEEE ICRA'91*, Vol. 2: 1088-1095
11. Q. Ke and T. Kanade, A subspace approach to layer extraction, In *Proc. IEEE CVPR'01,* 2001.

12. R. Klette, G. Gimel'farb, R. Reulke, Wide-angle image acquisition, analysis and visualization. Proc. 14th Internat. Conf. Vision Interface (VI'2001), Ottawa, Canada, June 2001, 114-125.
13. A. Koschan, D. Page, J.-C. Ng, M. Abidi, D. Gorsich, and G. Gerhart, SAFER under vehicle inspection through video mosaic building," *International Journal of Industrial Robot*, September 2004, 31(5): 435-442.
14. C. Mayntz, T. Aach, D. Kunz, and J-M. Frahm, Motion blur in fluoroscopy: effects, identification, and restoration, SPIE Medical Imaging 2000, San Diego, CA.
15. A. Noble, R. Hartley, J. Mundy, and J. Farley. X-Ray Metrology for Quality Assurance, In *Proc IEEE ICRA'94*, vol 2, pp 1113-1119.
16. V. J. Orphan, R. Richardson, and D. W. Bowlin, 2002. VACIS^{TM} – a safe, reliable and cost- effective cargo inspection technology, *Port Technology International*, p. 61-65. www.porttechnology.org/journals/ed16/section02.shtml
17. S. Peleg, M, Ben-Ezra, and Y. Pritch, 2001. Omnistereo: panoramic stereo imaging, *IEEE Trans. PAMI*, 23(3): 279-290.
18. H.-Y. Shum and R. Szeliski, 1999. Stereo reconstruction from multiperspective panoramas. In *Proc. ICCV'99*: 14-21.
19. H. Tao, H. S. Sawhney, and R. Kumar, 2001. A global matching framework for stereo computation, In *Proc. ICCV'01.*
20. B. Triggs, P. McLauchlan, R. Hartley, and A. Fitzgibbon, Bundle Adjustment – A Modern Synthesis, In *Vision Algorithms: Theory and Practice*, Lecture Notes in Computer Science, vol 1883, pp 298--372, 2000, eds. B. Triggs, A. Zisserman and R. Szeliski", Springer-Verlag.
21. J. Xiao and M. Shah, 2004. Motion layer extraction in the presence of occlusion using graph cut, In *Proc. CVPR'04.*
22. J. Y. Zheng and S. Tsuji, Panoramic representation for route recognition by a mobile robot. *International Journal of Computer Vision*, 9(1), 1992: 55-76.
23. Y. Zhou and H. Tao, 2003. A background layer model for object tracking through occlusion," In *Proc. ICCV'03*: 1079-1085.
24. Z. Zhu and A. R. Hanson, LAMP: 3D Layered, Adaptive-resolution and Multi-perspective Panorama - a New Scene Representation, *Computer Vision and Image Understanding*, 96(3), Dec 2004, pp 294-326.
25. Z. Zhu, E. M. Riseman, and A. R. Hanson, 2001. Parallel-perspective stereo mosaics. In *Proc. ICCV'01*, vol I: 345-352.
26. Z. Zhu, E. M. Riseman, and A. R. Hanson, Generalized Parallel-Perspective Stereo Mosaics from Airborne Videos, *IEEE Trans PAMI*, 26(2), Feb 2004, pp. 226-237.
27. Z. Zhu, L. Zhao, and J. Lei, 3D Measurements in Cargo Inspection with a Gamma-Ray Linear Pushbroom Stereo System, IEEE Workshop on Advanced 3D Imaging for Safety and Security, June 25, 2005, San Diego, CA, USA.
28. Z. Zhu, H. Tang, B. Shen, and G. Wolberg, 3D and Moving Target Extraction from Dynamic Pushbroom Stereo Mosaics, *IEEE Workshop on Advanced 3D Imaging for Safety and Security*, June 25, 2005, San Diego, CA, USA.

Chapter 9

3D MODELING OF INDOOR ENVIRONMENTS
for a Robotic Security Guard

P. Biber[1], S. Fleck[1], T. Duckett[2], and M. Wand[1]
[1]*Graphical-Interactive Systems, Wilhelm Schickard Institute for Computer Science, University of Tübingen, Sand 14, 72076 Tübingen, Germany*
[2]*AASS Research Center, Department of Technology, Örebro University, SE-70182 Örebro, Sweden*

Abstract: Autonomous mobile robots will play a major role in future security and surveillance tasks for large scale environments such as shopping malls, airports, hospitals and museums. Robotic security guards will autonomously survey such environments, unless a remote human operator takes over control. In this context a 3D model can convey much more useful information than the typical 2D maps used in many robotic applications today, both for visualization of information and as human machine interface for remote control. This paper addresses the challenge of building such a model of a large environment (50x60m²) using data from the robot's own sensors: a 2D laser scanner and a panoramic camera. The data are processed in a pipeline that comprises automatic, semiautomatic and manual stages. The user can interact with the reconstruction process where necessary to ensure robustness and completeness of the model. A hybrid representation, tailored to the application, has been chosen: floors and walls are represented efficiently by textured planes. Non-planar structures like stairs and tables, which are represented by point clouds, can be added if desired. Our methods to extract these structures include: simultaneous localization and mapping in 2D and wall extraction based on laser scanner range data, building textures from multiple omnidirectional images using multiresolution blending, and calculation of 3D geometry by a graph cut stereo technique. Various renderings illustrate the usability of the model for visualizing the security guard's position and environment.

Key words: 3D modeling, robotic security guard, simultaneous localization and mapping, graph cut stereo.

A. Koschan et al. (eds.), 3D Imaging for Safety and Security, 201–223.

1. INTRODUCTION

Robotic research is now in a mature state and ready to focus on complete mobile robotics applications. The research in the AASS Learning Systems Lab, for example, is aimed at building a Robotic Security Guard for remote surveillance of indoor environments[1]. This robot will learn how to patrol a given environment, acquire and update maps[5], keep watch over valuable objects, recognize known persons, discriminate intruders from known persons, and provide remote human operators with a detailed sensory analysis. The system should enable automation of many security operations and reduce the risk of injury to human workers. The design philosophy is based on augmenting remote human perception with super-human sensory capabilities, including (see also Figure *9-1*):

- omni-directional vision
- hi-resolution pan-tilt-zoom camera
- laser and ultrasonic range-finder sensors
- thermal infrared camera for human detection and tracking
- metal-oxide gas sensors for chemical monitoring.

By combining vision and 2D laser range-finder data in a single representation, a textured 3D model can provide the remote human observer with a rapid overview of the scene, enabling visualization of structures such as windows and stairs that cannot be seen in a 2D model.

In this paper we present our easy to use method to acquire such a model. The laser range scanner and the panoramic camera collect the data needed to generate a realistic, visually convincing 3D model of large indoor environments. Our geometric 3D model consists of planes that model the floor and walls (there is no ceiling yet, as the model is constructed from a set of bird's eye views). The geometry of the planes is extracted from the 2D laser range scanner data.

Textures for the floor and the walls are generated from the images captured by the panoramic camera. Multi-resolution blending is used to hide seams in the generated textures stemming, e.g., from intensity differences in the input images.

Then, the scene is further enriched by 3D-geometry calculated from a graph cut stereo technique to include non-wall structures such as stairs, tables, etc. An interactive editor allows fast post-processing of the automatically generated stereo data to remove outliers or moving objects.

After a brief summary of relevant techniques for generation of 3D models of real scenes, especially 3D indoor models, our method is outlined in the next section.

Figure 9-1. Robotic platform for security guard with sensors marked. The laser range scanner and the omni-directional camera are used to build a 3D model of the robot's operation environment.

2. TECHNIQUES FOR MODELING 3D SCENES

Relevant techniques for modeling real 3D Scenes and displaying them can be divided into geometric, image-based and hybrid approaches.

2.1 Geometric approaches

Geometric representations of scenes include triangle meshes, curve representations or simply point clouds to model surfaces. Material properties, light sources, and physical models provide the basis for rendering them. While it is possible to build mobile platforms that are able to acquire surface models of real world scenes by range scan techniques[14, 21, 22] even in real-time, estimation of material properties or light sources is a hard problem in general. Thus, to render visual information convincingly without

reconstructing or simulating physical properties it has been proposed to represent real scenes directly by images.

2.2 Image-based approaches

Image-based rendering is a now well established alternative to rendering methods based on geometric representations. The main promise is that it is able to generate photorealistic graphics and animations of scenes in real-time[19]. Nowadays, panoramic views are the most well known variant of image-based rendering and can be discovered everywhere in the web. A user can rotate his/her view freely and can zoom in real-time (but only with a constant position). To allow all degrees of freedom, the so-called plenoptic function has to be sampled. For a static scene, this is a six-dimensional function, and is thus hard to sample and to keep in memory. Aliaga et al.[2] presented a system that allows photo-realistic walk-throughs in indoor environments. A panoramic camera mounted on a mobile platform captures a dense "sea of images", that is, the distance between two camera positions is only around 5 cm. Advanced compression and caching techniques allow walk-throughs at interactive speed. For the calculation of the camera positions, battery powered light bulbs were placed at approximately known positions. The largest area covered was 81 m^2, requiring around 10.000 images. The disadvantage of such a model is that despite its high memory requirements, only walk-throughs are possible: the user is not permitted to move too far away from a position where an image has been recorded.

It is now common to attempt to combine the best of both worlds in so-called *hybrid* approaches.

2.3 Hybrid approaches

Debevec et al. combined still photographs and geometric models in a hybrid approach[8]. In their work, the user had to interactively fit parametrized primitives such as boxes to the photographs to build a basic model. This model in turn was the basis of a model-based stereo algorithm, which enriched the basic model with depth maps. Finally, *view-dependent texture mapping* was used to simulate geometric details not recovered by the model. This system allows generation of photo-realistic renderings from new viewpoints, as long as there exists a still photograph taken from a position close to that new viewpoint. *Texture mapping* per se, that is, mapping the color information of an image onto a plane, belongs to the oldest class of hybrid techniques, and is still the most commonly used method in computer graphics, so acquisition of textures from real world scenes is an important topic. A representative study was done by Früh and Zakhor[11]. They

presented a system that is able to generate 3D models of a city by combining textured facades with airborne views. Their model of downtown Berkeley, which is really worth a glance at, allows walk-throughs as well as bird's eye views.

Our method can be seen as a similar approach for indoor office environments, since we use a basic geometric model together with advanced texture creation and mapping methods. We emphasize especially blending methods to hide the seams when textures are generated from multiple images. In contrast to the "sea of images" approach, we recover also camera positions automatically by applying a simultaneous localization and mapping (SLAM) algorithm to the laser range-finder data.

However, our goal is not to produce photo-realistic results. Using a mobile robot driving on the floor as an image acquisition system, current techniques would allow only for walk-throughs (or drive-throughs) at a constant view height using view-dependent texture mapping. As we want our model to be viewable also from distant bird's eye views, our goal is to create visually convincing models. The acquired indoor model presented here is much larger than other indoor models reported, yet it is possible to view it as VRML model in a standard web-browser. In essence, our approach is much more similar to that of Früh and Zakhor than to Aliaga et al.

3. OVERVIEW

This section gives an overview of our method to build a 3D model of an office environment after remotely steering the mobile robot through it. At regular intervals, the robot records a laser scan, an odometry reading and an image from the panoramic camera. The robot platform is described in Section 4. From this data, the 3D model is constructed. Figure *9-2* gives an overview of the method and shows the data flow between the different modules. Five major steps can be identified as follows (the second step, data collection, is omitted from Figure *9-2* for clarity):

- Calibration of the robot's sensors.
- Data collection.
- Map generation.
- Texture generation.
- Stereo processing.

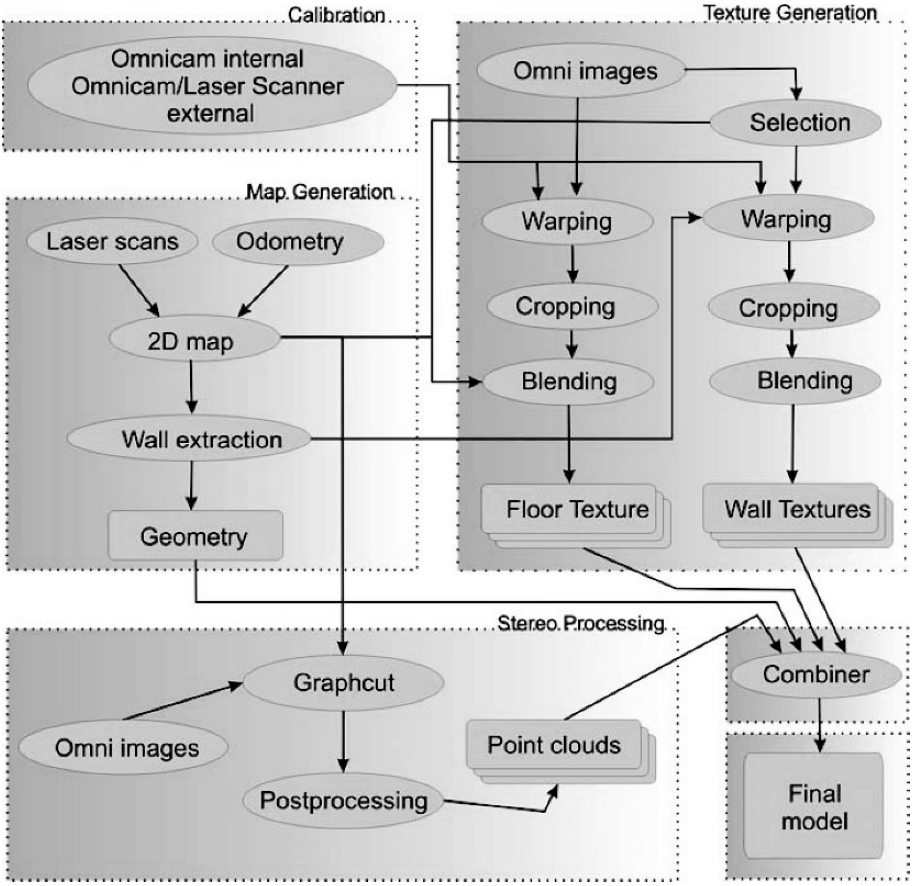


Figure 9-2. An overview of our method to build a 3D model of an indoor environment. Shown is the data flow between the different modules.
(See also Plate 28 in the Colour Plate Section)

Our method consists of manual, semi-automatic and automatic parts. Recording the data and calibration is done manually by teleoperation, and extraction of the walls is done semi-automatically with a user interface. Stereo matching is automatic, but selection of extracted 3D geometry and post-processing includes semi-automatic and manual parts. Thus the user can interact with the reconstruction process where it is necessary to ensure robustness (which plays a key role for large real world environments) and completeness of the model (there should be no holes, etc.).

After describing the hardware platform of our security guard, the remaining sections cover the mentioned steps. The paper ends with concluding remarks and, of course, various renderings of the resulting model.

4. HARDWARE PLATFORM AND SENSORS

The robot platform used in these experiments is an ActivMedia Peoplebot (see Figure *9-3*).

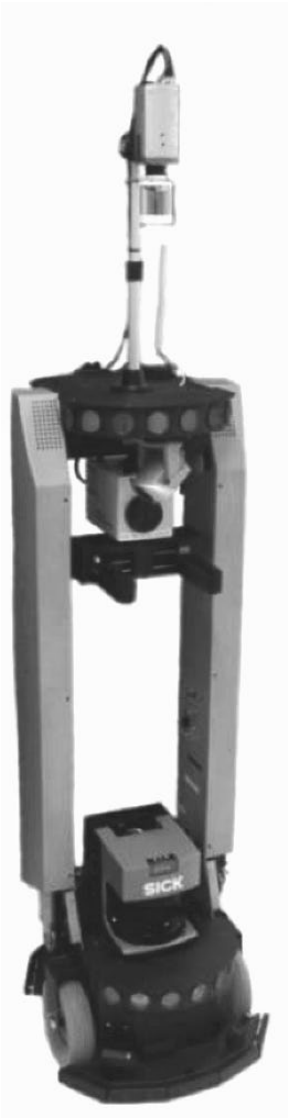


Figure 9-3. ActivMedia Peoplebot. It is equipped with a SICK LMS 200 laser scanner and panoramic camera (NetVision360 from Remote Reality.

It is equipped with a SICK LMS 200 laser scanner and a panoramic camera consisting of an ordinary CCD camera with an omni-directional lens attachment (NetVision360 from Remote Reality). The panoramic camera has a viewing angle of almost 360 degrees (a small part of the image is occluded by the camera support) and is mounted on top of the robot looking downwards, at a height of approximately 1.6 meters above the ground plane. Prior to recording a run, internal and external sensor parameters have to be calibrated.

4.1 Calibration of the panoramic camera

Since the geometrical shape of the mirror inside the omni-directional lens attachment is not known, a calibration procedure was applied to map metric coordinates p onto pixel coordinates p_p (see Figure *9-4*). We assume that the shape of the mirror is symmetrical in all directions θ, hence it is only necessary to perform calibration in one direction, i.e., to map 2D world coordinates onto 1D pixel coordinates.

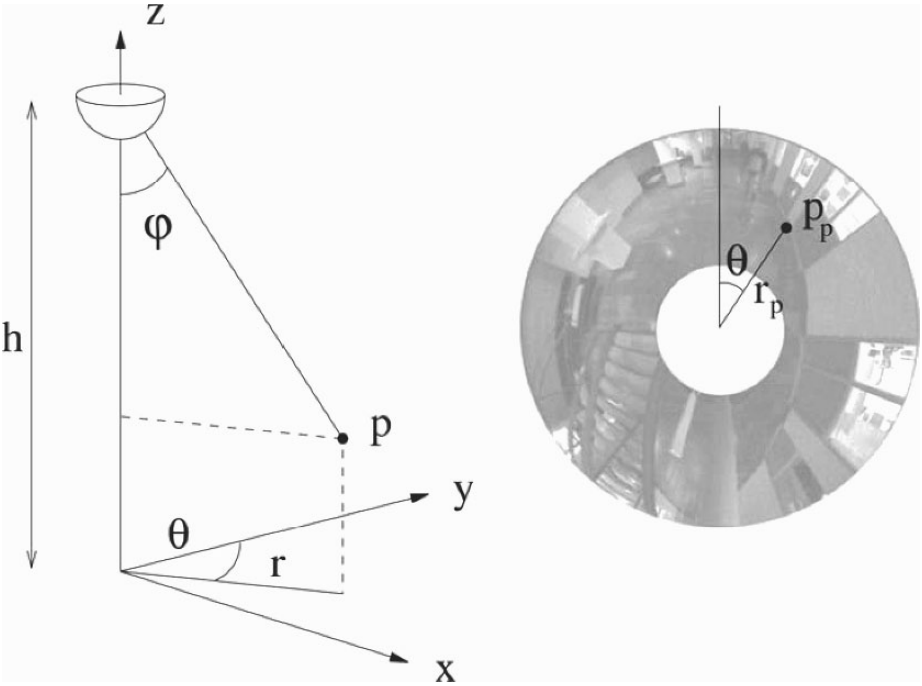


Figure 9-4. Left: Geometry of the panoramic camera calibration (the half-sphere represents the surface of the mirror inside the lens attachment). Right: Geometry of the panoramic images.

Several images with known positions (r, z) and measured corresponding pixels r_p were collected. From this data the parameter h, the camera height, was estimated using $\tan(\varphi) = r/(h - z)$. To handle conversions from φ to r_p a polynomial of degree 3 was fitted to the data. The polynomial function was then used to interpolate on the calibration measurements for 3D modeling.

4.2 Calibration of external sensor parameters

All methods in the rest of the paper assume that the laser scanner and the panoramic camera are mounted parallel to the ground plane. It is difficult to achieve this in practice with sufficient precision. While a small slant of the laser scanner has less effect on the measured range values in indoor environments, a slant of the panoramic camera has considerably more effect. Figure *9-5* (a) shows one panoramic image along with the corresponding laser scan mapped onto the ground plane under the above assumption. The alignment error can be considerable, especially for distant walls. Since a mapping like this is used to extract textures for walls, we need to correct this error.

A model for the joint relation between panoramic camera, laser scanner and ground plane using three parameters for the rotation of the panoramic camera turned out to be accurate enough. The parameters can be recovered automatically using full search (as the parameters' value range is small). To obtain a measure for the calibration, an edge image is calculated from the panoramic image. It is assumed that the edge between floor and wall also produces an edge on the edge image, and therefore we count the number of laser scan samples that are mapped to edges according to the calibration parameter. Figure *9-5* (b) and (c) show the result of the calibration: the laser scan is mapped correctly onto the edges of the floor.

5. BUILDING THE 2D MAP BY SCAN MATCHING

The basis of our algorithm is an accurate 2D map. This map is not only used to extract walls later, it is also important to get the pose of the robot at each time step. This pose is used to generate textures of the walls and floor and provides the external camera parameters for the stereo processing.

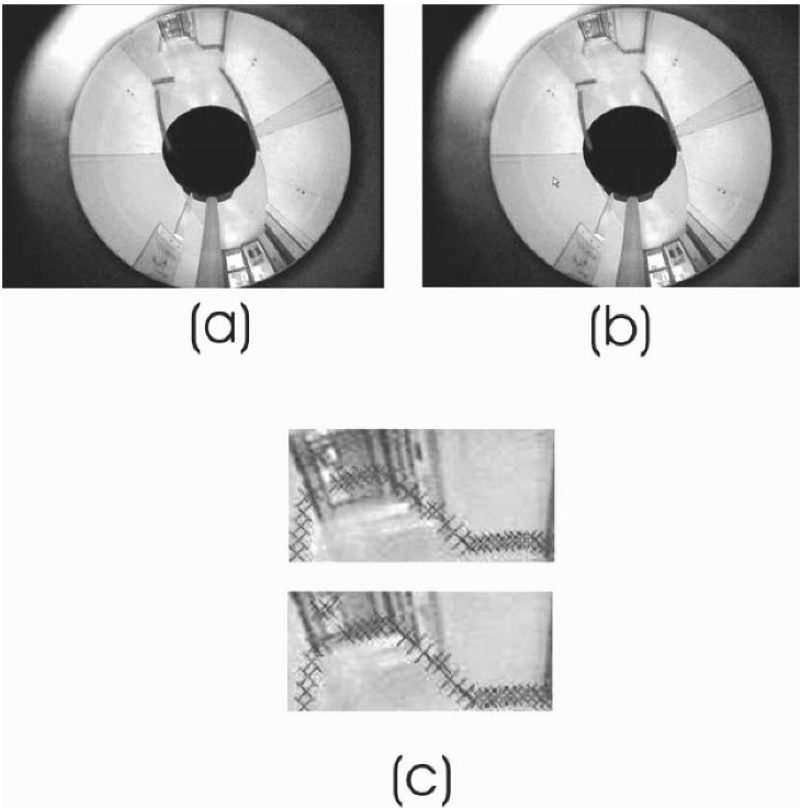


Figure 9-5. Joint external calibration of laser, panoramic camera and ground plane tries to accurately map a laser scan to the edge between floor and wall on the panoramic image; (a) without calibration (b) with calibration (c) zoom.

Our approach belongs to a family of techniques where the environment is represented by a graph of spatial relations obtained by scan matching[10,13,18]. The nodes of the graph represent the poses where the laser scans were recorded. The edges represent pair wise registrations of two scans. Such a registration is calculated by a scan matching algorithm, using the odometry as initial estimate. The scan matcher calculates a relative pose estimate where the scan match score is maximal, along with a quadratic function approximating this score around the optimal pose. The quadratic approximations are used to build an error function over the graph, which is optimized over all poses simultaneously (i.e., we have $3 \times nrScans$) free parameters). A more detailed description of our method is available[4]. Figure *9-6* shows a part of the underlying relation graph and the final map used in this paper.

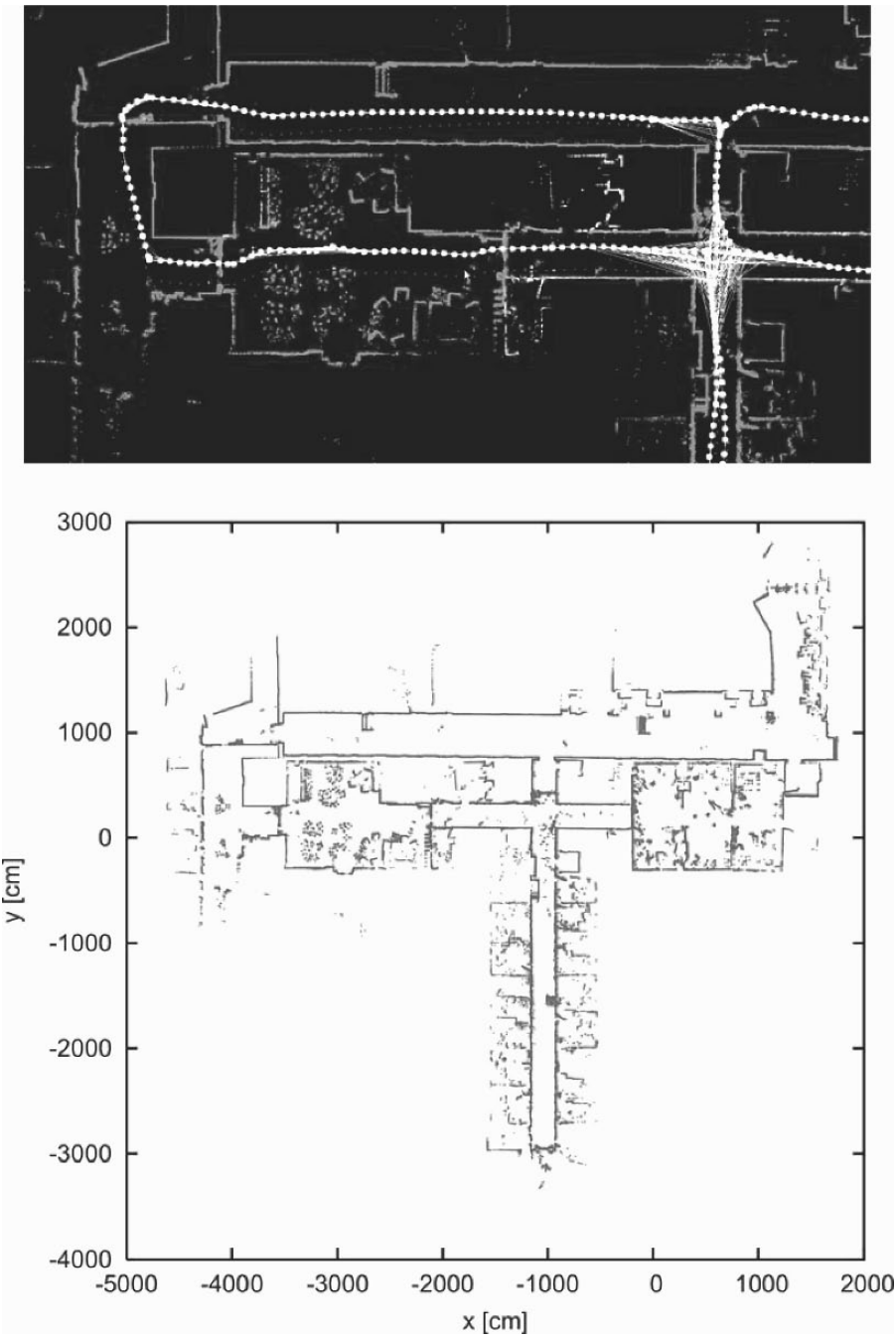


Figure 9-6. Part of the graph that the map consists of (top) and final map (bottom).

6. GENERATION OF GEOMETRY

The geometry of our 3D model consists of two parts: the floor and the walls. The floor is modeled by a single plane. Together with the texture generated in the next section, this is sufficient: the floor's texture is only generated where the laser scans indicate free space.

The walls form the central part of the model. Their generation is a semi-automatic step, for reasons described here. The automatic part of this process assumes that walls can be identified by finding lines formed by the samples of the laser scans. So in a first step, lines are detected in each single laser scan using standard techniques. The detected lines are projected into the global coordinate frame, where lines that seem to correspond are fused to form longer lines. Also, the endpoints of two lines that seem to form a corner are adjusted to have the same position. In this way, we try to prevent holes in the generated walls.

This automatic process gives a good initial set of possible walls. However, the results of the automatic process are not satisfying in some situations. These include temporarily changing objects and linear features, which do not correspond to walls. Doors might open and close while recording data, and especially for doors separating corridors, it is more desirable not to classify them as walls. Otherwise, the way would be blocked for walk-throughs. Furthermore, several detected lines were caused by sofas or tables. Such objects may not only cause the generation of false walls, they also occlude real walls, which are then not detected.

So we added a manual post-processing step, which allows the user to delete, edit and add new lines. Nearby endpoints of walls are again adjusted to have the same position. In a final step, the orientation of each wall is determined. This is done by checking the laser scan points that correspond to a wall. The wall is determined to be facing in the direction of the robot poses where the majority of the points were measured.

7. GENERATION OF TEXTURES

The generation of textures for walls and for the floor is similar. First, the input images are warped onto the planes assigned to walls and floor. A floor image is then cropped according to the laser scan data. Finally, corresponding generated textures from single images are fused using multi-resolution blending.

The calibration of the panoramic camera, the joint calibration of robot sensors and ground plane, and the pose at each time step allows for a simple basic acquisition of textures for the floor and walls from a single image.

Both floor and walls are given by known planes in 3D: the floor is simply the ground plane, and a wall's plane is given by assigning the respective wall of the 2D map a height, following the assumption that walls rise orthogonally from the ground plane. Then textures can be generated from a single image by backward mapping (*warping*) with bilinear interpolation.

The construction of the final texture for a single wall requires the following steps. First, the input images used to extract the textures are selected. Candidate images must be taken from a position such that the wall is facing towards this position. Otherwise, the image would be taken from the other side of the wall and would supply an incorrect texture. A score is calculated for each remaining image that measures the maximum resolution of the wall in this image. The resolution is given by the size in pixels that corresponds to a real world distance on the wall, measured at the closest point on the wall. This closest point additionally must not be occluded according to the laser scan taken at that position. A maximum of ten images is selected for each wall; these are selected in a greedy manner, such that the minimum score along the wall is at a maximum. If some position along the wall is occluded on all images, the non-occlusion constraint is ignored. This constraint entails also that image information is only extracted from the half of the image where laser scan data are available (the SICK laser scanner covers only 180°. Finally, a wall texture is created from each selected image, then these are fused using the blending method described as follows.

The generation of a floor texture from a single image is demonstrated in Figure *9-7*. The image is warped onto the ground plane. Then it is cropped according to the laser scanner range readings at this position, yielding a single floor image. This entails again that one half of the image is not used. Such a floor image is generated from each input image. Then, these images are mapped onto the global 2D coordinate frame.

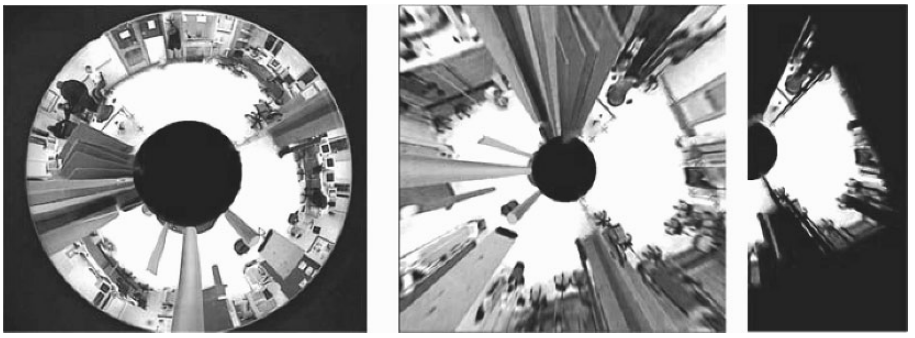


Figure 9-7. Generation of floor texture from a single image.

Both floor and wall textures are fused from multiple input images (Figure *9-8* shows an example). The fusion is faced with several challenges, among them:

- image brightness is not constant,
- parts of the input image may be occluded by the robot or the support of the panoramic camera, and
- walls may be occluded by objects in front of them and thus effects of parallax play a role.

Additionally, the quality along a wall texture degrades with the distance from the closest point to the robot position (this effect is due to scaling and can be seen clearly in Figure *9-8*. Similar effects can be observed for floor textures. These problems also exist in other contexts[3,20].

We use an adaption of Burt and Adelson multiresolution blending[7]. The goal of the algorithm is that visible seams between the images should be avoided by blending different frequency bands using different transition zones.

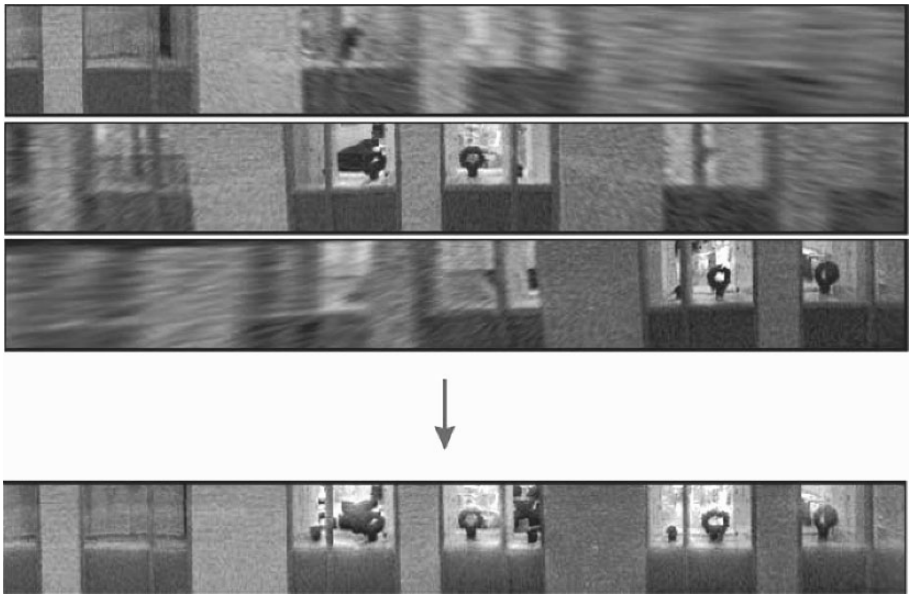


Figure 9-8. Final textures of walls are generated by blending multiple textures generated from single panoramic images. Shown here are three of ten textures which are fused into a single texture.

The outline is as follows: a Laplacian pyramid is calculated for each image to be blended. Each layer of this pyramid is blended separately with a constant transition zone. The result is obtained by reversing the actions that are needed to build the pyramid on the single blended layers. Typically, the distance from an image center is used to determine where the transition zones between different images should be placed. The motivation for this is that the image quality should be best in the center (consider, e.g., radial distortion) and that the transition zones can get large (needed to blend low frequencies). To adapt to the situation here, we calculate a distance field for each texture to be blended, which simulates this "distance to the image center". For the walls, this image center is placed at an x-position that corresponds to the closest point to the robot's position (where the scaling factor is smallest). Using such a distance field, we can also mask out image parts (needed on the floor textures, as in Figure *9-9*, to mask both the region occluded by the robot and regions not classified as floor according to the laser scanner).

8. ACQUISITION OF ADDITIONAL 3D GEOMETRY USING STEREO

Thanks to the available camera positions and the calibrated camera we are in an ideal setting to apply stereo algorithms to the input images. A high quality, state-of-the-art stereo algorithm - namely the *graph cut* algorithm by Kolmogorov and Zabih[16,17] - is used to calculate a disparity map for each panoramic image. Our implementation[9] is based upon the graph cut implementation of Per-Jonny Käck[15] that extends the work of Kolmogorov and Zabih and is adapted to omnidirectional imaging.

8.1 SSD matching

Our stereo matching pipeline starts with the following stage: first, for each pixel in the first image the epipolar curve in the second image is created according to the epipolar geometry of our panoramic camera.

This epipolar curve is represented by a set of points in image space where each point denotes a different disparity. These points are used to construct a rectified window taking zoom into account. Then, an SSD error value for each disparity on this epipolar curve is computed and saved.

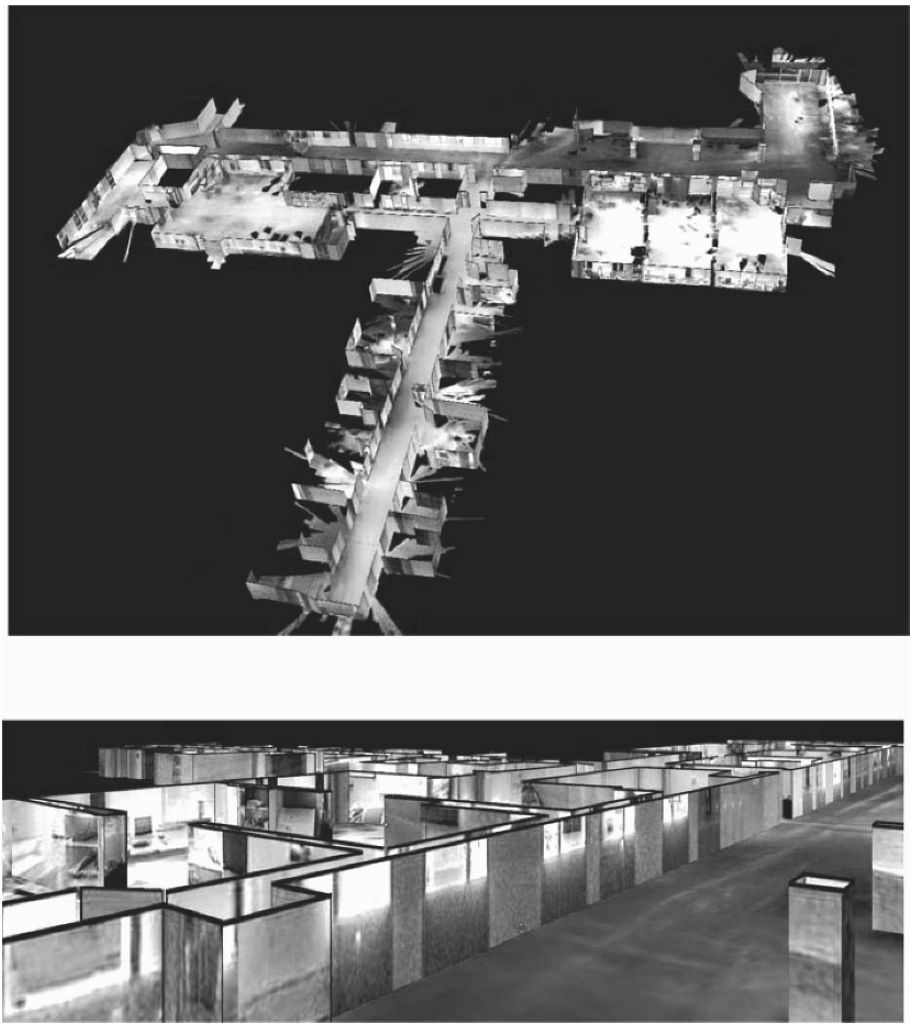


Figure 9-9. Two views of the resulting VRML model - yet without results from stereo matching.

The image that is being processed is compared both to the next and to the previous image. The matching costs are then mixed into one big array containing all matching costs for each pixel, except for those parts of the image where one curve contains more points than the other - here only the matching values of the longer curve are used. These steps provide the data needed by the graph cut algorithm.

8.2 Graph Cut

The graph cut algorithm used here follows the work of Kolmogorov and Zabih[16,17] and is adapted for omnidirectional imaging. The key is formulating the correspondence problem as an energy minimization problem. This is solved by an algorithm based on α -expansion moves[6]. The minimization is done iteratively by transforming the energy minimization problem into several minimum cut problems.

These lead to a strong local minimum of the energy function by computing the best α -expansion of lowest energy for different values of α, until convergence is reached.

To ensure that each α -expansion succeeds, which is key to solving the above correspondence problem, this is implemented via graph cuts.

Kolmogorov and Zabih investigated the necessary characteristics for an energy function of binary values to be optimized by graph cuts[17].

We use an appropriate energy function E of the form (following the notation of Kolmogorov and Zabih[17]:

$$E(f) = E_{data}(f) + E_{occ}(f) + E_{smooth}(f).$$

$E_{data}(f)$ embodies the SSD-based matching cost of corresponding pixels, i.e.

$$E_{data}(f) = \sum_{<p,q>\in A(f)} \left| I_{k-1}(p) - I_k(q) \right|^2 .$$

The occlusion term $E_{occ}(f)$ adds an additional cost C_p for each occluded pixel:

$$E_{occ}(f) = \sum_{p\in P} C_p T(\left| N_{p(f)} \right| = 0).$$

$E_{smooth}(f)$ imposes a penalty $V_{a1,a2}$ for neighbouring pixels having different disparity values:

$$E_{smooth}(f) = \sum_{\{a1,a2\}\in N_1} V_{a1,a2} T(f(a1) \neq f(a2)).$$

The resulting disparity map is converted into a point cloud and post-processed. Disparity values are refined to subpixel accuracy by finding the optimum of a local quadratic model built using the original matching cost at the integer disparity value and its adjacent disparity values. Regions around

the epipoles (there are two epipoles in omnidirectional images[12]) are removed because these typically provide too few constraints to extract reliable depth information.

In a further step, depth values that belong to the floor with high probability are corrected to be exactly on the floor. The epipole removal and the graph cut algorithm both mark some pixels as unknown or occluded. The distance values for these pixels are interpolated from the surrounding, known distances using linear interpolation along concentric circles.

Figure *9-10* shows one source image, the winner takes all solution based on the SSD score, the result of the graph cut algorithm and the final disparity map after post-processing. The point cloud from this figure (fused with the walls and floor model) is rendered. Another point cloud is shown in Figure *9-11*. Obviously, this point cloud contains outliers. These can be filtered out automatically in the next step, where several point clouds from different input images are combined, or manually using an interactive editor. These steps are described next.

8.3 Point cloud post-processing

Several data sets (point clouds) can then be combined into one big point cloud. Each 3D point in each point cloud (one for each image) is compared to all points from other images within a search radius. If the color is similar enough, a confidence counter for that point is increased. If the counter is above a threshold for a certain amount of images the point has been compared to, it is marked as good, otherwise it is discarded. This further improves the quality, because wrong matches are very unlikely to find similar points near them in other images and are thus removed, hopefully leaving only the correct points in the cloud. The resulting point cloud is then filtered to remove unnecessary points by calculating point densities and removing points from areas with high densities.

Also, points that are already represented by walls or by the floor are omitted. Finally the point clouds are combined with the rest of the model (i.e. the walls and the floor). An interactive point cloud editor and renderer allows the user to select the objects supposed to be part of the final model and to delete outliers. This editor uses features of modern graphics hardware (vertex and pixel shaders) to allow fast rendering and editing of large point clouds (several million points). Future versions will also allow point cloud filtering and application of hole filling algorithms. A screenshot of this tool is shown in Figure *9-12* (while editing the staircase of Figure *9-11*).

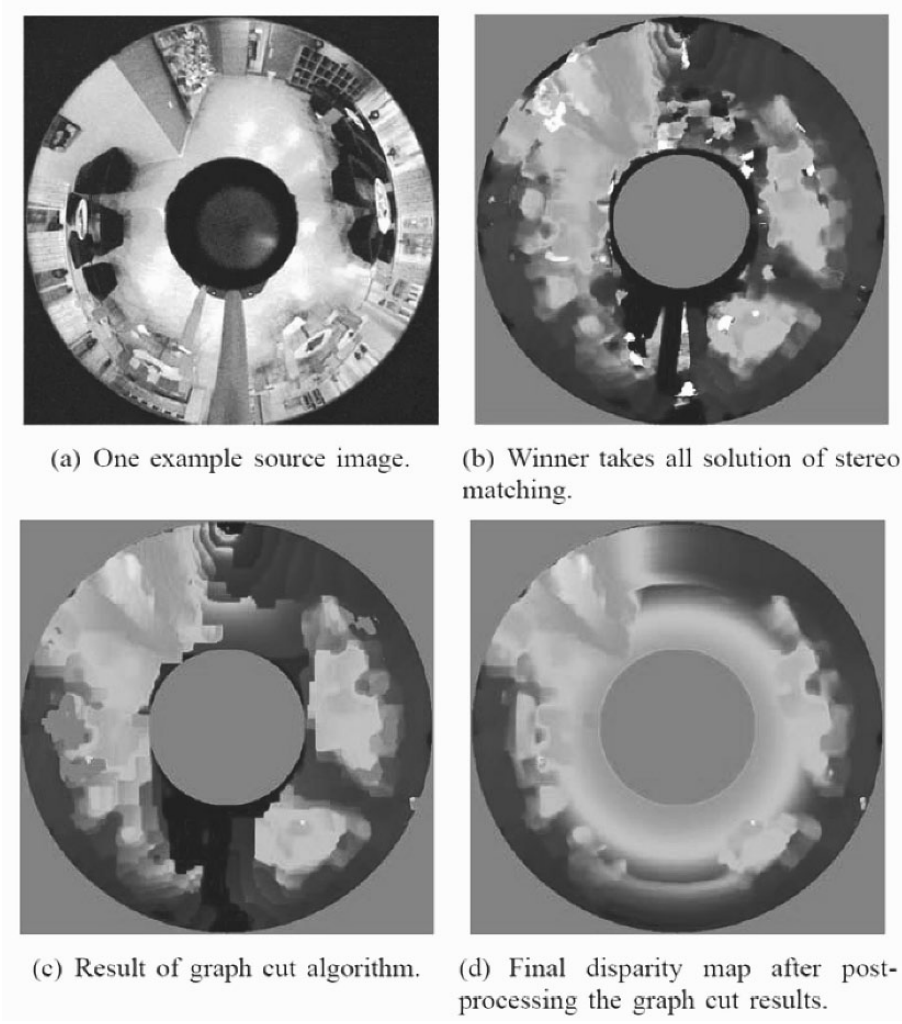


Figure 9-10. Stereo processing using graph cut algorithm and post-processing steps (subpixel refinement, epipole removal, floor correction and hole filling).
(See also Plate 29 in the Colour Plate Section)

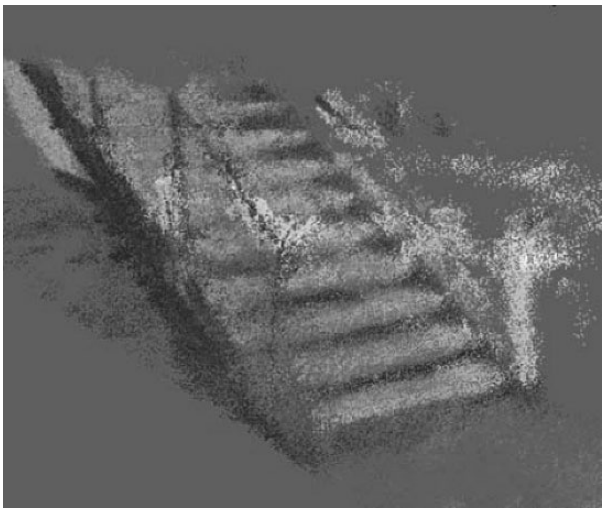


Figure 9-11. A staircase: output of graph cut-algorithm after removing walls and floor, but before removing outliers manually.

Figure 9-12. Screen shot of the tool that can be used to edit point clouds comfortably.

9. RESULTS AND CONCLUSION

A data set of 602 images and laser scans was recorded at Örebro University by teleoperation, covering parts of a region of about 60×50 meters. The built 2D-map was shown in Figure *9-6*. Screen shots of the resulting 3D model without stereo results can be seen in Figures *9-8* and *9-9*. This model can be exported as a VRML model, so that it can be viewed in a web browser with a VRML plugin. It is also possible to build a VRML model with point clouds (Figures *9-13* and *9-14*) but there are tight limits on the number of points such that the frame rate allows real time walk-throughs. For larger models it is suggested to use a native visualization environment based upon our point clouds editor (which makes heavy use of modern graphics hardware features like vertex and pixel shaders). Figure *9-14* also shows how the 3D model can be used to visualize information about the state of the robot. In the figure the robot's position is visualized in a natural way using a hand-crafted 3D model of the robot.

We see our technique as a successful easy-to-use method to acquire a 3D model that is highly useful for its intended application, as part of the human interface to a robotic security guard. Considerable work has been done also on other components and, with ongoing work to integrate these technologies, we are confident to reach a state where autonomous mobile robots are able to leave the laboratory and do useful work in the real world, based on their own sensor data and in cooperation with humans.

Figure 9-13. A view of the cafeteria with results from stereo matching included.

Figure 9-14. The map can be used to visualize information in 3D by augmenting with virtual content, here for example the position of the robot.

ACKNOWLEDGMENTS

The authors would like to thank Henrik Andreasson for his help with the data collection and calibration of the panoramic camera, Per Larsson for creating the client/server application used to teleoperate the robot and Florian Busch for implementation of the stereo algorithm.

REFERENCES

1. Robotic security guard webpage. `www.aass.oru.se/Research/Learning/proj rsg.html`.
2. D. Aliaga, D. Yanovsky, and I. Carlbom. Sea of images: A dense sampling approach for rendering large indoor environments. *Computer Graphics & Applications, Special Issue on 3D Reconstruction and Visualization*, pages 22–30, Nov/Dec 2003.
3. A. Baumberg. Blending images for texturing 3d models. In *Proceedings of the British Machine Vision Conference*, 2002.
4. P. Biber and W. Straßer. The normal distributions transform: A new approach to laser scan matching. In *International Conference on Intelligent Robots and Systems (IROS)*, 2003.
5. P. Biber and T. Duckett. Dynamic Maps for Long-Term Operation of Mobile Service Robots, in: *Robotics: Science and Systems*, 2005
6. Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1124–1137, 2004.
7. P. J. Burt and E. H. Adelson. A multiresolution spline with application to image mosaics. *ACM Transactions on Graphics*, 2(4):217–236, 1983.

8. P. E. Debevec, C. J. Taylor, and J. Malik. Modeling and rendering architecture from photographs: A hybrid geometry- and image-based approach. *SIGGRAPH 96*, 1996.
9. S. Fleck, F. Busch, P. Biber, H. Andreasson, and W. Straßer. Omnidirectional 3d modeling on a mobile robot using graph cuts. In *IEEE International Conference on Robotics and Automation (ICRA 2005)*, April 18-22 2005.
10. U. Frese and T. Duckett. A multigrid approach for accelerating relaxation-based SLAM. In *Proc. IJCAI Workshop on Reasoning with Uncertainty in Robotics (RUR 2003)*, 2003.
11. C. Früh and A. Zakhor. Constructing 3d city models by merging ground-based and airborne views. *Computer Graphics and Applications*, November/December 2003.
12. C. Geyer and K. Daniilidis. Conformal rectification of omnidirectional stereo pairs. In *Omnivis 2003: Workshop on Omnidirectional Vision and Camera Networks*, 2003.
13. J.-S. Gutmann and K. Konolige. Incremental mapping of large cyclic environments. In *Computational Intelligence in Robotics and Automation, 1999.*
14. D. Hähnel, W. Burgard, and S. Thrun. Learning compact 3d models of indoor and outdoor environments with a mobile robot. *Robotics and Autonomous Systems*, 44(1), 2003.
15. P.J. Käck. Robust stereo correspondence using graph cuts (master thesis), *www.nada.kth.se/utbildning/grukth/exjobb/rapportlistor/-2004/rapporter04/kack per-jonny 04019.pdf*, 2004.
16. V. Kolmogorov and R. Zabih. Computing visual correspondence with occlusions using graph cuts. In *International Conference on Computer Vision (ICCV'01)*, 2001.
17. V. Kolmogorov and R. Zabih. What energy functions can be minimized via graph cuts? In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2004.
18. F. Lu and E.E. Milios. Globally consistent range scan alignment for environment mapping. *Autonomous Robots*, 4:333–349, 1997.
19. L. McMillan and G. Bishop. Plenoptic modeling: An image-based rendering system. *SIGGRAPH*, 1995.
20. C. Rocchini, P. Cignomi, C. Montani, and R. Scopigno. Multiple textures stitching and blending on 3D objects. In *Eurographics Rendering Workshop 1999*, pages 119–130.
21. H. Surmann, A. Nüchter, and J. Hertzberg. An autonomous mobile robot with a 3d laser range finder for 3d exploration and digitalization of indoor environments. *Robotics and Autonomous Systems*, 45(3-4), 2003.
22. S. Thrun, W. Burgard, and D. Fox. A real-time algorithm for mobile robot mapping with applications to multi-robot and 3d mapping. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, 2000.
23. A. Treptow, G. Cielniak, and T. Duckett. Active people recognition using thermal and grey images on a mobile security robot. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2005)*, 2005.

Chapter 10

3D SITE MODELLING AND VERIFICATION

Usage of 3D Laser Techniques for Verification of Plant Design for Nuclear Security Applications

V. Sequeira, G. Boström, and J.G.M. Gonçalves
European Commission - Joint Research Centre, TP210, I-21020 Ispra, Italy

Abstract: It is important in Nuclear Security to detect changes made in a given installation or track the progression of the construction work in a new plant. This chapter describes a system accepting multi-sensory, variable scale data as input. Scalability allows for different acquisition systems and algorithms according to the size of the objects/buildings/sites to be modeled. Semi-automated tools support the comparison between the acquired "as-built" and the approved design models. The chapter presents examples of the use in indoor and outdoor environments. It shows that it is possible at any time to redo a data acquisition from the same area without any assumptions of exact position, orientation or used scanner. Based on this new information it is possible to detect millimeter to decimeter changes in the scene.

Key words: 3D reconstruction, change analysis, data fusion

1. INTRODUCTION

To ensure adherence to the Nuclear Non-Proliferation Treaty (NPT) obligations, Countries are required to declare design information on all new and modified facilities, which are under safeguards, and to ensure that the accuracy and completeness of the declaration is maintained for the life of the facility. It is the obligation of the United Nations' International Atomic Energy Agency (IAEA) to verify that the design and purpose of the "as-built" facility is as declared and that it continues to be correct. These activities are referred to as Design Information Examination and Verification (DIE/DIV) or in more general terms Scene Change Detection. The procedure can be divided into three steps: (1) examination of the declared design documents; (2) collection of information on the "as-built" facility using

A. Koschan et al. (eds.), 3D Imaging for Safety and Security, 225–247.

various methodologies; and (3) comparison of the "as built" facility with the declared information. Although methodologies have been available for DIV, they have not provided the level of continuity of knowledge needed for the lifetime of the facility.

Recently, advanced 3D measurement devices based on laser-light has emerged[1] enabling a rapid and accurate modeling of substantial volumes. When scanning a volume with such a device, one get an accurate virtual 3D model useful for inspection or verification tasks.

In recent years, techniques have been developed making it possible to perform a change detection both for areas inside buildings[2, 3] and for outside areas[4, 5].

The chapter describes the complexities associated to Scene Change Detection of large plant areas, both indoors and outdoors, including detailed discussion of acquisition techniques; registration techniques and inspection tools. In the end, results are presented showing some models generated with the system and change analysis performed for both indoor and outdoor environments

2. TASK COMPLEXITY

The DIV task is one of the main challenges that International Nuclear Safeguards organizations have to face. The size of some facilities as well as the complexity of their design and process poses an insurmountable challenge when considering 100% verification before the facility comes into operation.

As an in-depth verification of all areas is beyond the Inspectorates' resources, a structured, methodical approach is taken prioritizing equipment, structures and activities and randomizing the lower priority items.

Even with prioritized tasks and with the application of a random approach, the verification activities, especially for cell and piping verification remains a tedious and costly activity. Also the fact that DIV activities must take place over several years represents additional problems, the issue of maintaining continuity of knowledge of the previously verified equipment and structures being with no doubt the most important one, not only during the construction phase but also for the entire life of the plant.

3. SYSTEM DESCRIPTION

The availability of fast and accurate tools for 3D distance measurements encouraged the development of systems facilitating inspections and safeguarding

activities. Smaller areas such as isolated processing cells and limited areas have successfully been modeled with laser techniques. For instance, initial field tests have been reported in the reprocessing plant Rokkasho, Japan[3]. The success of this indoor project led to an increasing interest in extending the technique to cover large outdoor areas such as entire plant sites in which the technique is scaled and adopted to cover large distances and the usage of different scanning techniques.

The concept and the steps required to perform a successful Scene Change Detection are discussed below. This is followed by the description of the added complexities in performing Scene Change Detection for larger outdoor environments.

The main components of the system are: (a) 3D Data acquisition systems, and the tools to (b) create realistic "as built" 3D Reference Models, (c) detect and verify the changes between the current 3D reconstructed model and a reference model, and (d) track and document changes in successive inspections. The procedure for Scene Change Detection is divided into three distinctive phases:

- *Building of a 3D Reference Model.*
  The Reference Model is constructed consisting of the surface description in 3D. Any source of data containing a 3D representation of the reference area can be used. Available CAD models can be used as a reference of the ideal construction. The reference model provides a good description of the area, or, more exactly, volume, under observation. The quality of the DIV activities is highly dependent on how accurately and realistically the 3D model documents the "as-built" plant. As such, the reference model should be acquired with the best possible conditions including a) higher spatial resolution, b) lower measurement noise and c) multiple views to cover possible occlusions. The number of required scans depends on the complexity of the scene.

- *Initial verification of the 3D models.*
  This verification is performed with the 3D Model cells versus the CAD models or the engineering drawing provided by the plant operator. In the case of unavailability of CAD models, surveying operations such as 3D distance measurements, accurate calculation of container diameters, heights, pipe diameters and other complex distances impossible by traditional measurement tools can be made in the virtual 3D environment.

- *Re-verification.*
  At that time, new scans of the selected area are taken and compared with the reference model in order to detect any differences with respect to

initial verification. Each part of this scan, denoted Verification Scan is analyzed and the closest Euclidian distance to the reference model is computed. The result of the change-detection is a displacement-map for each element in the verification scan. The automatic detected changes can be further analyzed by an operator by using visualization aids such as color look-up table scheme. The re-verification phase can occur at any time after the reference model is constructed.

4. DATA COLLECTION

The acquisition of a large area can be a tedious task. In selecting the correct scanning devices for different areas, the time can be reduced still keeping some higher level requirements such as accuracy and density of measurement points. Also, the number of scans to be acquired for a given scene depends highly on the complexity. Several scans may be needed to overcome problems with occlusions and shadows.

Taking the Scene Change Detection to a large outdoor area implies new aspects of planning, acquiring data with multiple resolutions as well as data processing and storage. It is seldom required or necessary to maintain the highest possible reference model resolution (both spatial and depth) for the entire area under observation. Selective scanning might be preferred.

This section describes general requirements for scan acquisition as well as in more detail describing three different scanner techniques which have different approaches, application areas and resolutions.

4.1 General considerations of scanning architectures in various environments

When only a tripod-mounted standard terrestrial scanner is used, many scans are required to cover a complete area, considering the maximum range for these scanners varies from 50 to 200 meters. Performing several of these scans implies the need to manage and store large amounts of data representing highly detailed representations of the area. Moreover, one can say that a highly detailed description may not be necessary for all parts of the area under consideration. For the sake of efficiency, it is possible to assign different levels of accuracies to different areas. As an example, scenes including complex pipe-constructions should be modeled with the best spatial resolution available producing the best-possible millimeter resolution of the objects. To this effect, a high-resolution 3D scanner allows for

automated Scene Change Detection where objects as small as a centimeter can be tracked regarding displacement[3].

For other areas including the outside of office complexes and warehouses as well as open areas such as parking spaces and utility ground, more efficient scanning techniques can be devised. Vehicle-mounted 3D scanning equipment[6] can be used for such areas. This solution maintains a fairly good spatial resolution in the range of centimeters, and has the advantage of enabling fast scanning over large areas while driving. 3D Scene Change Detection based on data from vehicle-mounted scanners is appropriate for detecting object changes in the order of decimeters (see further examples below).

For open areas, it is possible to use the latest available commercial equipment for airborne laser (normally LIDAR) scanning, carried by either an airplane or helicopter. These types of 3D scanning equipment, once airborne, scan a large area very rapidly. The depth resolution acquired with such scanning equipment lies in the range of a couple of centimeters. The spatial resolution depends on the flight-speed, maintained altitude and the scanner equipment capacity. Reasonable values lay in the range of 0.5 meters. Apart the speed, one advantage of airborne scanners is that it easily acquires views, such as elevated positions or rooftops, which are normally difficult or tedious to acquire with terrestrial scanners. For airborne data, 3D Scene Change Detection detects changes of the order of a meter. This includes the detection of objects being moved such as containers and/or building changes.

All together, the scanning techniques discussed above, can be efficiently used together and are well suited for various and numerous tasks. Table *10-1* and Figure *10-1* sketch the best use for large open areas. For an extensive list of available commercial scanners see Stone et al.[7].

4.2 3D Laser scanner for indoor and detailed areas

For detailed areas where a scan accuracy in the range of millimeters are required, a tripod mounted 3D Laser Range Finder is suitable (Figure *10-2*).

The system makes use of a portable commercial off-the-shelf laser range scanner and a portable computer. These can be mounted on a tripod with a dolly and are battery operated for flexible usage. The laser range scanner is composed of a laser beam, a rotating mirror and a rotating base. The laser beam is deflected in two dimensions: vertically by the rotating deflection mirror and horizontally by the rotating movement of the entire measuring device. The maximum range of the scanner shown in Figure *10-2* is 53.5 meters[8]. It can measure distances with accuracy better than 5 mm.

Table 10-1. Usage of Scanner Architectures for different scanning architectures.

Scanning technique (E~Error in samples) / Constructions to be scanned	Tripod mounted Laser Scanner (E ~0.01-0.05 m)	Vehicle mounted Laser Scanner (E ~0.1-0.5 m)	Airborne LIDAR Scanner (E ~0.3-1.0 m)
Detailed scenarios outside	++	–	– –
House façades	+	++	– –
Structures internally	++	– –	– –
Elevated positions such as rooftops and terraces	+	– –	++
Not important utility areas	–	++	++
Extended constructions	++	++	–

(++: Highly appropriate; +: Appropriate; –: Not appropriate; – –: Not recommended)

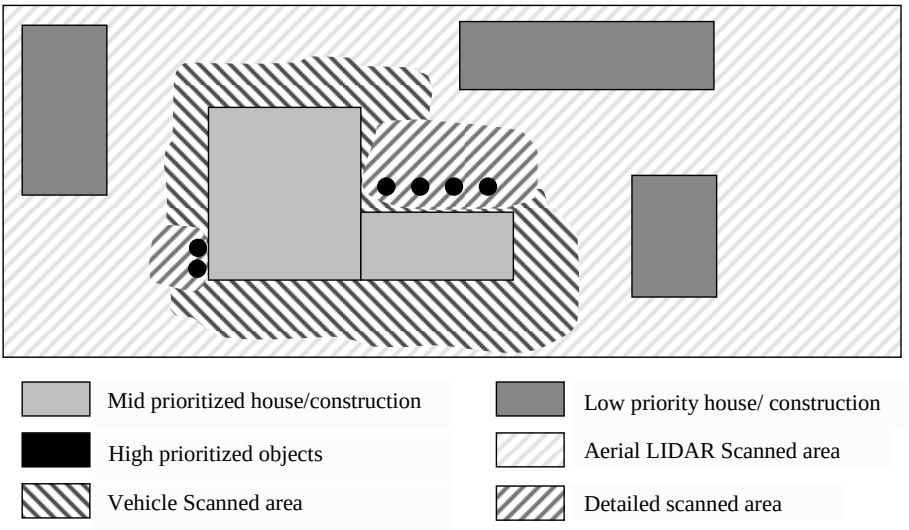


Figure 10-1. Schematic view of an outdoor area to be modeled.

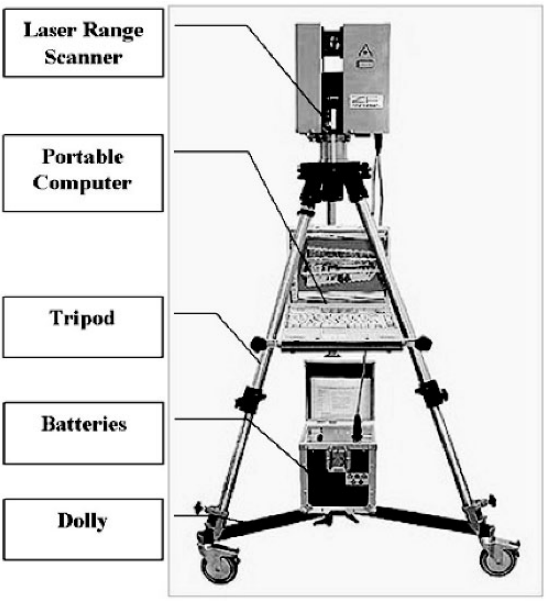


Figure 10-2. 3D data acquisition equipment.

4.3 Scanning outdoor areas

For a complete description of the outdoor scene the system combines ground acquired 3D data with airborne 3D laser data. Ground data is acquired with the help of a vehicle and integrates 3D laser scanners, high-resolution digital photography, video, GPS and inertial navigation systems (see conceptual drawing in Figure *10-3*).

4.3.1 3D Laser scanner for outdoor areas

For covering larger areas still keeping a good accuracy we have developed a laser scanning system to be mounted on a vehicle. The benefit of having the system mounted on a vehicle is that large areas can be covered in a short period.

The system as shown in Figure *10-4* is composed of one main inter-changeable laser range scanner that gives direct 3D measurements, a calibrated color CCD Camera for instant coloring of the model and a GPS receiver and inertial system for global positioning, vehicle trajectory reconstruction and internal synchronization. The system is controlled by a touch-screen tablet-PC.

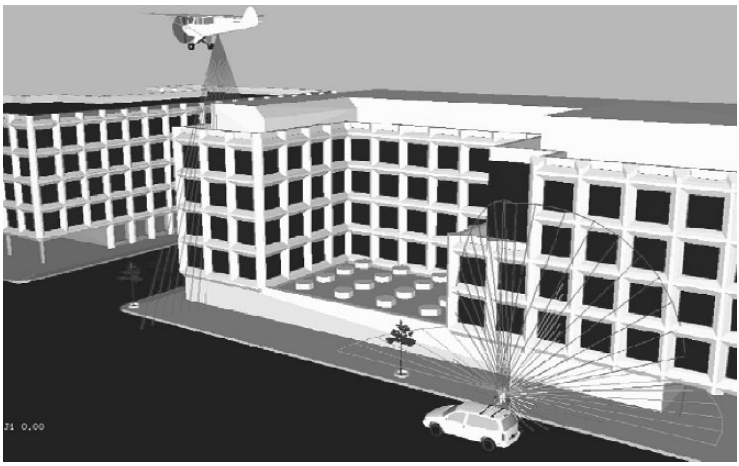


Figure 10-3. Different scanning techniques for outdoor areas.
(See also Plate 30 in the Colour Plate Section)

Figure 10-4. Vehicle with mounted acquisition systems.

The data acquisition is achieved by driving the vehicle equipped with the system along the road and/or from different static scanning positions. The data acquired by the sensor is further treated based on registered vehicle-trajectory and acquired color information; see Figure *10-5* for a sketch of the reconstruction paradigm. Further information and more in-depth discussion on the treatment of point samples can be found in previous work[6]. Figure *10-6* shows a sample model acquired.

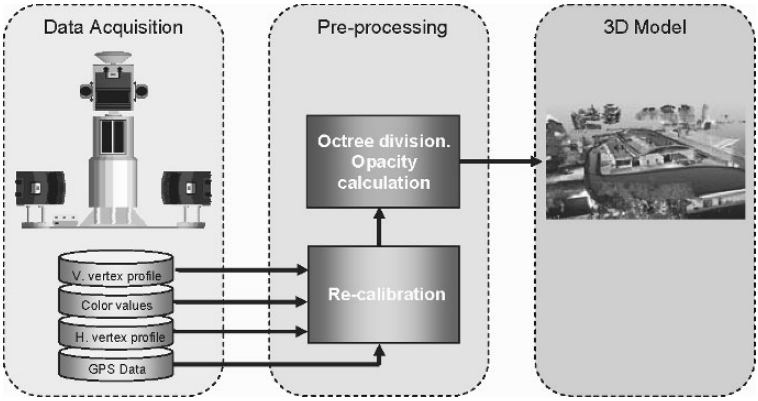


Figure 10-5. On-site data acquisition and processing for vehicle based scanners. (See also Plate 31 in the Colour Plate Section)

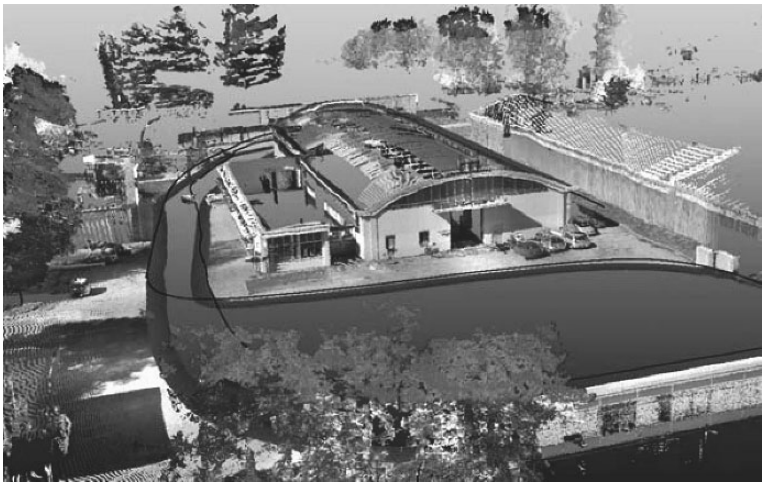


Figure 10-6. Sample model from a vehicle-mounted scanner. (See also Plate 32 in the Colour Plate Section)

4.3.2 Aerial 3D Data

For complementary 3D information it is also possible to acquire a bird's eye perspective of an area using a laser scanning system onboard an airplane with embedded GPS and inertial system to reconstruct the airplane trajectory, roll, yaw and pitch. This reveals height information unavailable from the previously mentioned scanning techniques. At the current level of the technology it is comparably sparse but gives a significant contribution to the resulting model because the aerial scan provides vital information about

roofs and other parts which the terrestrial and vehicle mounted scanners misses. The aerial information is acquired by an airplane or helicopter equipped with a Scanning system (see Figure *10-7*). Generally, an accurate visual image is acquired together with the height information.

4.4 Scan Acquisition

In making a model from an area, there are a number of things that must be kept in mind. Depending on the scanner technique used these matter vary.

4.4.1 Considerations for tripod- mounted 3D Laser scanners

To acquire information from an area with the tripod version of a scanner, the workflow mainly follows the procedure as shown in Figure *10-8*. The main points to consider during scan acquisition are:

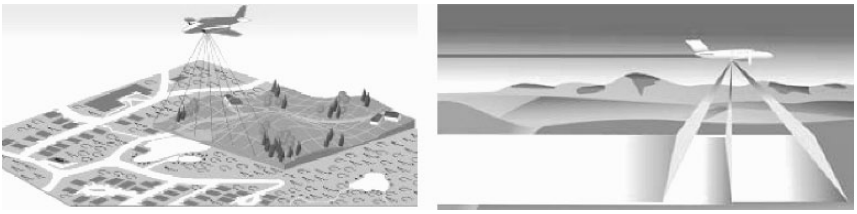


Figure 10-7. Aerial scanning techniques: Tri-Line-Scanning device (left) and LIDAR device (right). (Courtesy of Toposys GmbH[9] and Leica Geosystems Geospatial Imaging, LLC[10]). (See also Plate 33 in the Colour Plate Section)

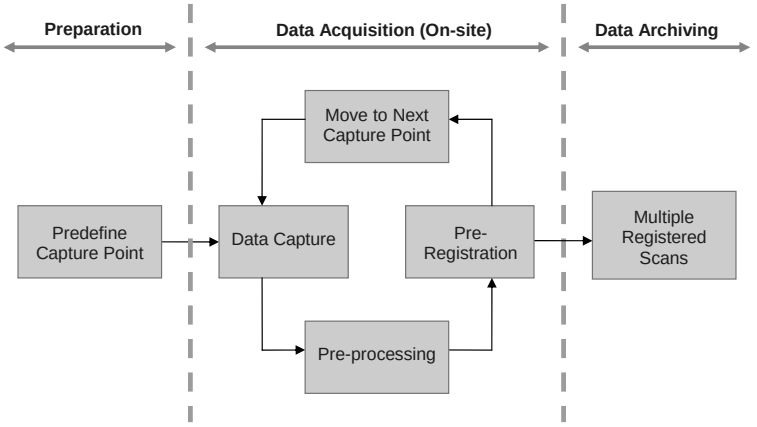


Figure 10-8. On-Site Data Acquisition and Pre-Processing for tripod based scanners.

Occlusion and shadows: The scanning position should be planned carefully to avoid missing data. Holes and occlusions become particularly evident when the data is to be triangulated at a later stage. DIV software allows the registration of the acquired scans in situ so that holes and occlusion can immediately be detected and resolved.

Angle of acquisition: Very shallow (i.e., acute) angles between the laser beam and the surfaces of the objects affect the quality of the final model both in what concerns distance accuracy and the spatial resolution (i.e., spatial detail).

Scan overlap: Registration requires sufficient overlap between scans. It is recommended to acquire one or more reference scans at low resolution covering as much of the scene as possible. These scans are later used to register the more detailed (i.e., higher resolution) scans. Also mixing techniques are suitable in which different scanner technologies are mixed having different coverage areas which automatically aids in the overall reconstruction phase.

Uniform scan resolution: scanner viewpoints and resolution settings should be selected to yield a fairly uniform resolution on the object in order to obtain a final 3D model with homogeneous geometrical properties.

Documenting the acquisition: If scans are not immediately registered, one should note the data acquisition parameters. This saves time at a later stage and avoids trial and error registrations.

4.4.2 Considerations for vehicle-mounted 3D Laser scanners

Vehicle Speed: Because of the scanning technique, where the scanning direction is perpendicular to the vehicle movement, the speed of the vehicle directly affects the density of the acquired model. A vehicle speed of about 7 km/h gives a scan resolution in the driving direction of roughly 10 cm for our scanner system having an acquisition rate of 18 Hz per vertical scan line.

Vehicle movements: The final resolution depends on the vehicle-trajectory. During step vehicle turns, parts of the scanned objects located near the rotation center may be scanned two times. This can be corrected for by simply removing overlapping lines or parts of lines. On the contrary, there are also parts of the scanned area which are scanned with a sparse grid on the opposite side. These effects must be considered when planning the vehicle trajectory and the acquisition.

4.4.3 Considerations for Aerial 3D Laser scanners

The aerial scans are acquired with off-the-shelf devices mounted in an airplane. The result from an acquisition is a height-map and optionally a

color image over the scanned area where plane movements and the trajectory have been taken care of. The result is normally re-sampled to a grid of sub-meter spacing.

5. REFERENCE MODEL CONSTRUCTION

A reference model is a set of data which fuses the individual scans keeping the most suited data and constructing a best possible model. The model finds its use in safeguards for measurements and direct inspections. Also for change detection, the reference model states the raw model to which other scans or data are compared. The reference model is constructed by transforming all scans into a common coordinate frame. Different techniques apply for different scan techniques as described later.

5.1 Registration of terrestrial scans

The registration of ordinary 3D scans is managed as being discussed extensively in the literature, see Rusinkiewicz and Levoy[11] for a summary of different approaches. By identifying corresponding point-pairs in a model to be registered and an existing reference scan, a continuous improvement in the model transform can be acquired. After a number of iterations the model has reached a best possible transform minimizing the average error between the scans. At this stage the model scan is grouped with the previous scans and the procedure is repeated for all scans. Figure *10-14* shows a model being built from four terrestrial scans and aerial data.

5.2 Registration of aerial data

The aerial scan data is mainly contributing to the overall model on roofs and ground areas. In order for a normal 3D ICP to work there needs to be a substantial amount of overlapping data. By the nature of aerial data, this is not generally the case. To solve the problem we make use of a 2D version of the registration where the façades of the reference, being the terrestrial scans, are projected onto a horizontal plane giving the first 2 out of 6 degrees of freedom. The aerial data is processed in which the roof-edges are extracted and projected to the same horizontal plane. Now it is possible to perform a 2D version of the ICP solving the problem by 3 degrees of freedom. For the last degree of freedom being the height change this is performed by simple mapping identified ground areas from the aerial data onto the ground of the terrestrial scans. Figure *10-9* shows the extracted edges as well as the result

of a 2D registration. The final 3D model where both the vehicle based scan and the complementary aerial data is shown can be seen in Figure *10-17*. The technique of registering aerial scans on to terrestrial scans is further described in Fiocco et al.[12].

6. VERIFICATION

6.1 Verification methodology overview

The selection of the area (or volume) to be verified, i.e., where changes will be searched for, should depend mainly on application considerations. The decision on the type of verification scan is based on specified/required inspection accuracy. Needless to say, the selection of the acquisition equipment (including its accuracy, spatial and depth resolutions) is strongly influenced by the final specified inspection accuracy.

If it is enough to detect geometric changes in the range of meters (e.g., major construction changes, including refurbishment), a good result might be obtained with a verification model based on an aerial scan. Further, this type of verification scan is easily generated for square kilometers of area during a one-hour flight. For increasing spatial requirements, i.e., increasing sensitivity in change detection, vehicle and tripod based scans should be used (in this order).

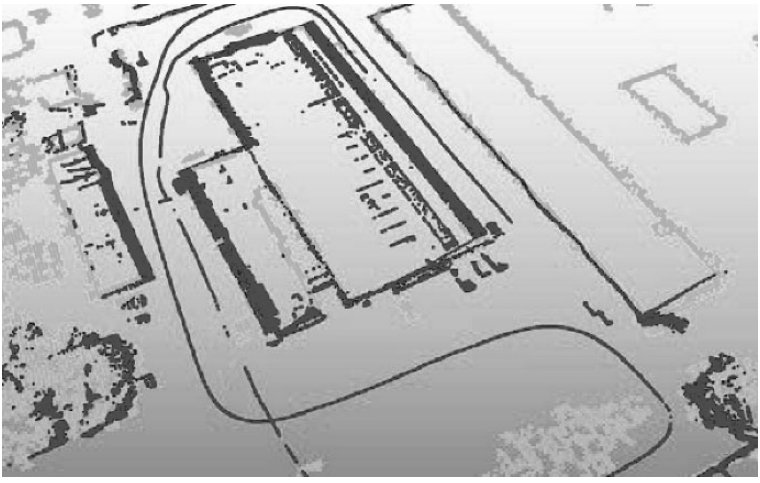


Figure 10-9. Aerial scan vertical edges (green) and vehicle-scan façade edges (red). (See also Plate 34 in the Colour Plate Section)

After acquisition, the verification scan needs to be registered onto the reference model by applying the same techniques as used in the reference model registration.

Figure *10-10* outlines the verification process into three sections: input, inspection and report. The input is a previously created reference model, where the reference model can be externally imported data or deriving from previously acquired scans, and newly acquired 3D data.

6.2 Triangle Search-tree construction

The inspection is based on searching the closest triangle of the reference model from each point of the scan. The search in the reference model has generally linear time complexity in number of triangles. To speed up the process, all triangles in a reference model are added to a spatial search tree (e.g. octree). This allows to pre-compute regions where triangles are localized in a hierarchical tree with which the closest triangle search becomes of logarithmic time complexity.

A leaf of the tree contain actually the reference to the cluster of triangles that are intersected by the leaf volume, so copies of some triangles may exist when they share the boundaries of more leaf volumes.

There are two parameters that control the tree creation, the maximum number of triangles per leaf (density of the leaf) and the maximum distance that one wants to measure (spatial size of the leaf). More triangles per leaf imply a shallower tree (less recursion) but more distance computations per leaf needs to be done.

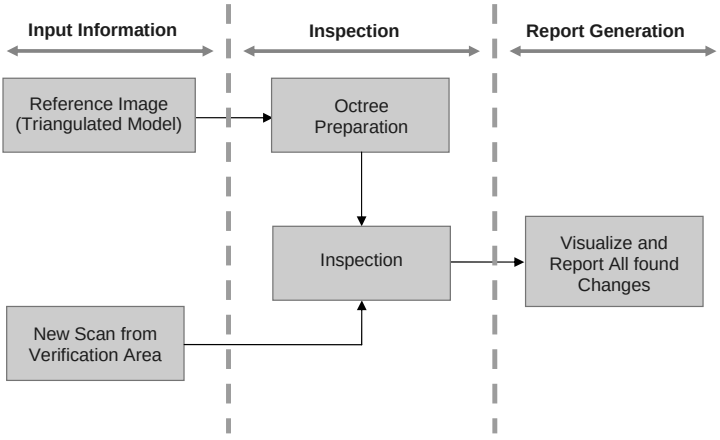


Figure 10-10. Inspection and verification process

The lower the maximum distance the smaller the size of the leaves, this improves the culling of leaves that are farther than the maximum distance from the surface, also less triangles per leaf is probable, but with a deeper tree though. The best performance is a trade off between these two parameters and they should be adapted each time a reference model is created.

6.3 Inspection step

During Inspection, for each 3D point for the Model under inspection, the distance to the closest triangle of the reference model is computed (see Figure *10-11*). This may imply comparisons against several models at the same time. An internal list keeps track of all changed areas.

All the leaf octants that intersect the maximum distance bounding box of the scan point are considered. The triangles inside are used to compute the distance to the scan point. The closest triangle among all of them is found; closest point and shortest distance, *d*, is computed.

The result is a file that stores the shorter distance that has been found for each point so it is not necessary to re-compute the inspection each time. These values can be encoded quickly with any kind of graduated pseudo-colors to help the user to grasp the difference of distance. In this viewing stage it is also possible to filter points based on their pre-processed confidence and lower the maximum distance of the scale in order to better discriminate smaller objects. Optionally a report file is created, where for each point the closest point and its distance are listed.

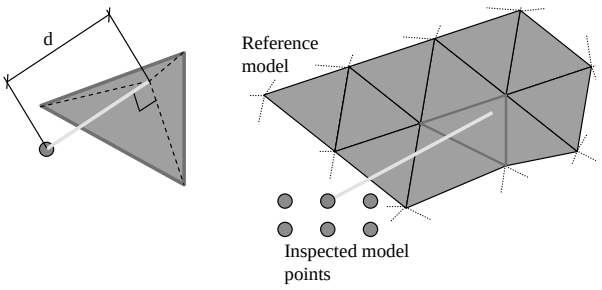


Figure 10-11. Triangle search during inspection.

6.4 Treatment of inspection results

For Change-detection, the result can be visualized in different ways to aid an operator.

- *Pseudo coloring* based on distance: The pseudo coloring gives a visibly correct way of showing detailed differences in highlighting colors. To further increase the focus on changes, the pseudo coloring can be applied to a certain region of change. Figures *10-15* and *10-19* show this presentation technique.
- *Color-coding* with an alarm level: This technique gives an operator means to identify and focus on objects which have been moved more than a specified amount. Figure *10-20* shows an inspection result with this color coding technique.

The result can be treated in such a way that areas or volumes which have distance changes above a threshold are saved to an inspection log which contains a global unique location reference (*xyz* location and extension).

7. EXPERIMENTS AND RESULTS

We have applied the software and scanning techniques on three different prototype scenarios. For each scenario scaled from millimeter changes to decimeter changes, the appropriate scanning devices were utilized, i.e. for a indoor case we used the highly accurate tripod mounted 3D Laser scanner and consequently for the decimeter change experiments we used vehicle-scan and aerial scan data for the reference and verification models. The different scenarios are:

1. Indoor environment searching for millimeter changes
2. Outdoor environment searching for centimeter changes
3. Large outdoor environment searching for decimeter changes

7.1 Indoor environment, analysis of millimeter changes

For this test we moved a fire-extinguisher and added a metallic rod and a several thin pipes (see Figure *10-12* for extracts of the reference model and the verification model respectively). The result of the Change detection can be seen in Figure *10-13*. The system has successfully detects the movement of the item-changes. The smallest change is a removed 8-mm rod as can be seen in the upper right part of Figure *10-13*.

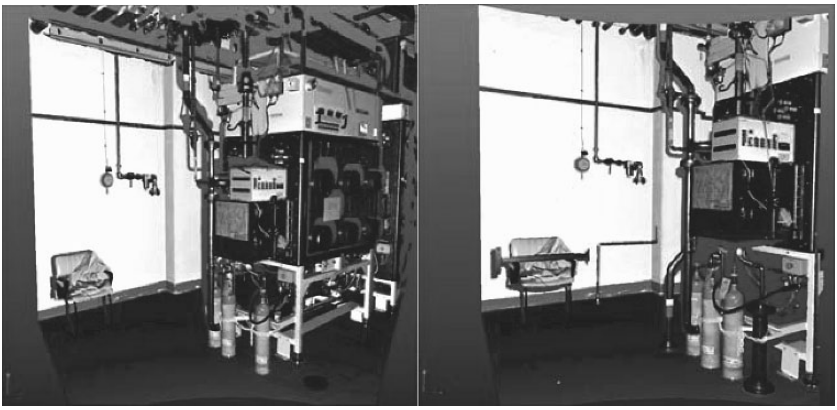


Figure 10-12. Reference model (left) and verification scan (right).

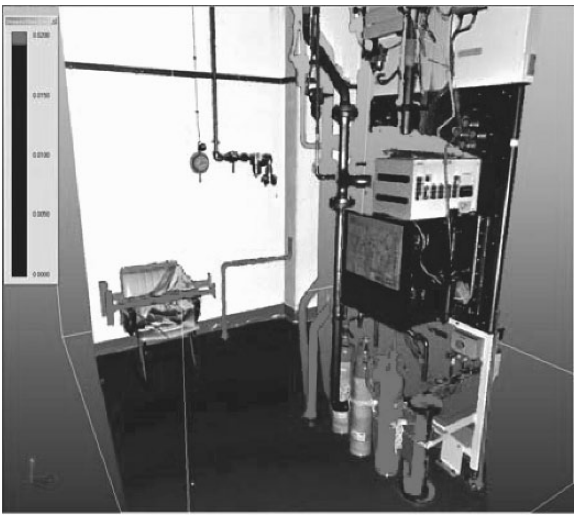


Figure 10-13. Automatically detected differences in red.

7.2 Outdoor environment with centimeter changes

The second test is performed in an outside environment. The reference model as can be seen in Figure *10-14* shows the façade of a ware-house building.

Figure 10-14. Reference model for outside analysis.

The reference model is constructed of four terrestrial scans and one aerial scan. The aerial scan was acquired by Compagnia Generale Ripreseaeree S.p.A. with a LIDAR scanner ALTM 3033 from Optech[13] and the color images by a Leica ADS40[10].

A verification scan was acquired from an independent position different than any of the reference scan positions. In the experiments two card-board boxes were placed in the scene. The bigger card-board box of 50x30x15 cm was placed against the building in such a way that the 50x30 cm side was facing the wall. The other box, significantly smaller, having a dimension of 15x10x20 was squeezed close to the wall under a concrete edge hiding the object as much as possible.

The inspection reveals the objects well. The result is shown in Figure *10-15*. As can be seen in the rainbow-colored result image, the larger box was placed to the lower left along the wall.

The smaller object is placed low to the right on the same wall a couple of meters away. In interactive rendering using the colorized displacement-map as texture, these objects can easily be identified.

7.3 Outdoor environment analyzing for large changes

This section describes the results obtained from the analysis of larger areas. The area covered is based on a warehouse area with some trees (see Figure *10-16*). The different scanners used are for these experiments one aerial scan and vehicle-scans.

Figure 10-15. Result of change detection of an outside environment. (See also Plate 35 in the Colour Plate Section)

Figure 10-16. Overview of the office and warehouse area for the experiments. In the centre of the figure, the warehouse can be seen. The whole area is approximately 300 meter by 530 meter.

The reference model is based on an aerial scan originally covering 200 km^2. An area of interest covering 300 by 530 meters was used. For improved spatial resolution, a scan from a vehicle mounted laser scanner was acquired for a part of this area. The verification scan was acquired with a vehicle borne scanner during a normal working day. This resulted in some distinct differences, basically parked cars. A zoom-in of the warehouse entrance for both the final reference model and verification scan is shown in Figure *10-17* and Figure *10-18*.

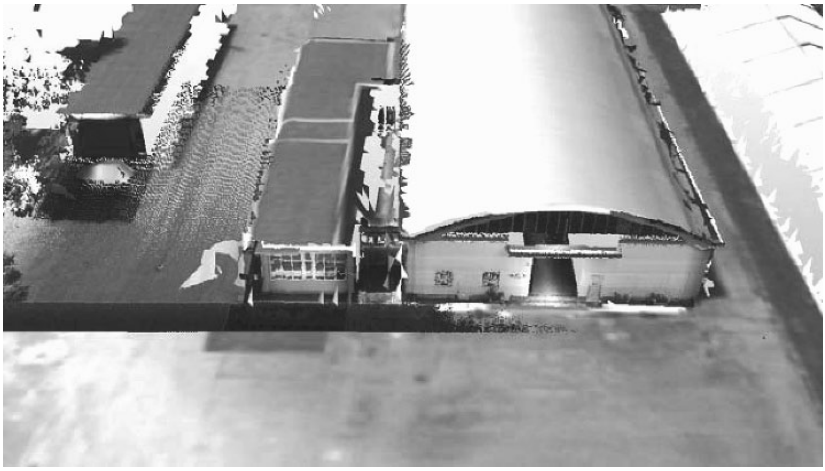


Figure 10-17. Detailed view of the reference model.

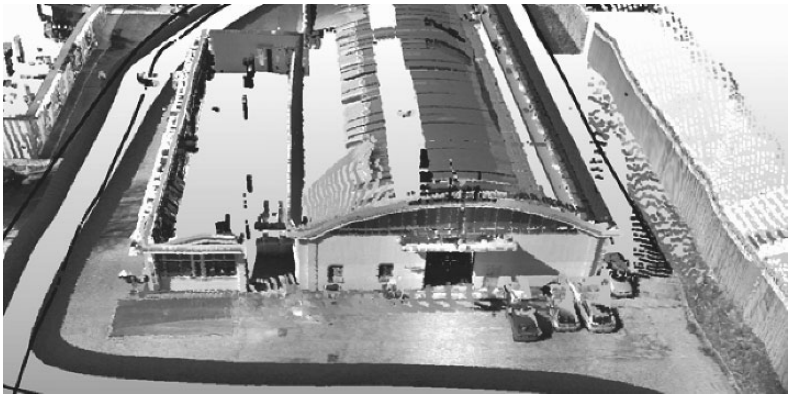


Figure 10-18. Detailed view of the verification model.

A change detection of the limited area outside the warehouse was performed with a maximum search distance of 1.0 meters, thus object movements of less than 1 meter can be resolved. Larger changes detected will be leveled to this maximum value. The result from the change detection is visualized in Figure *10-19* and Figure *10-20* with two different coloring schemes.

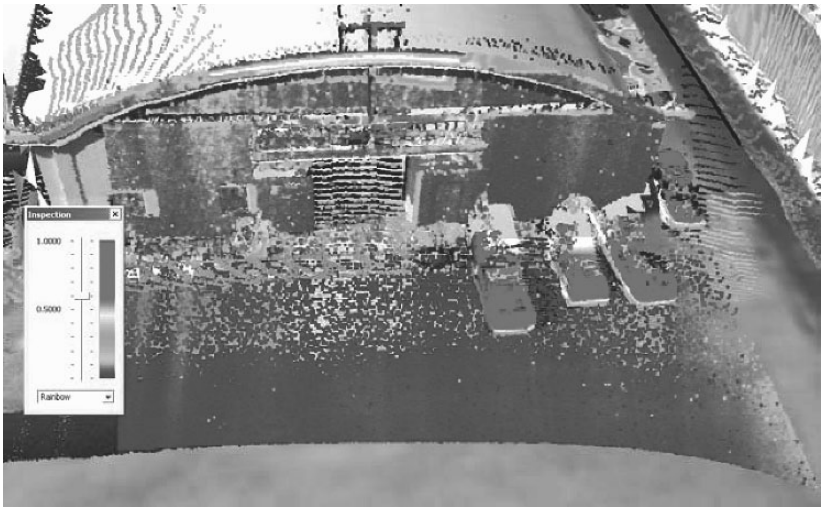


Figure 10-19. Results of scene change detection with rainbow coloring (blue = 0 m diff, red = 1.0 m diff). (See also Plate 36 in the Colour Plate Section)

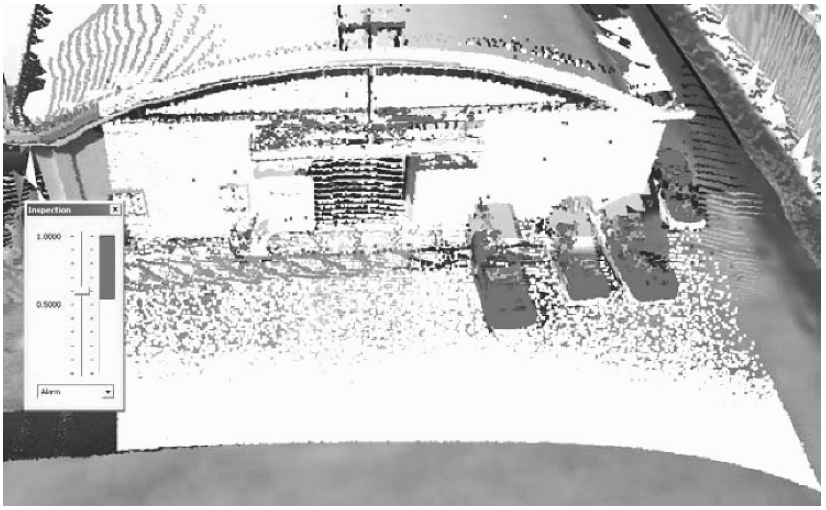


Figure 10-20. Results of scene change detection with alarm coloring (white = 0-0.5m diff, red >0.5 m diff). (See also Plate 37 in the Colour Plate Section)

8. CONCLUSIONS

The task of performing Scene Change Detection and Analysis of a large outdoor area involves a large amount of data acquisition and data management. The chapter discussed the techniques for performing an efficient overall Scene Change Detection, from the initial step of selecting the most suitable scanning techniques, through the steps needed for building a reference model, to the final steps of acquiring a verification scan and performing the final change detection. The technique presented is suitable for carrying out observations of large plant-areas. Prior selection and study of the area of interest, including the assessment of the spatial detail required for the detection of changes, definitely helps the selection of the scanning equipment to be used.

The results of our developments and experiments show that it is possible to merge data from multiple sources, including (a) various spatial resolutions, (b) various depth accuracy and resolution, (c) various capture points and orientations, (d) various types of equipment and (e) datasets acquired at different instants of time.

The technique provides a fast and easy way of modeling and documenting wide areas. These models can be further used as reference model or simply as a training tool for preparing inspections. Considering that all 3D data is based on accurate distance measurements, it is also possible to use the models to give some quantitative figures (e.g., precise area or volume measurement, mass measurements) that can be correlated with data from other sensory (e.g., radiation) devices.

Current research includes the extension of the 3D modeling and verifications tools for security planning including vulnerability assessment of public areas, preparation and training of rescue operations, and disaster assessment and remediation. 3D Realistic models of large areas, combined with a priori information for a given area (e.g., major infrastructures for utilities, vulnerability indications, etc.) can be most useful for assessing the consequences of a major disaster and plan for immediate rescue and remediation. The very same models, perhaps with adapted Human-Computer Interfaces, can be used for training of security and civil protection personnel. This applies both in public, urban environments as well as to major hazardous installations, e.g., chemical plants.

REFERENCES

1. C. Fröhlich and M. Mettenleiter, "Terrestrial Laser Scanning – New Prespective in 3D Surveying", in Proceedings of the ISPRS working group VIII/2, Volume XXXVI, Part 8/W2, Germany, 2004.
2. J. G.M. Gonçalves, V. Sequeira, B. Chesnay, C. Creusot, S. Johnson C., and J. Whichello, "3D Laser Range Scanner for Design Verification", *Proc. INMM 44th Annual Meeting*, Phoenix, Arizona, USA, 13-17 July, 2003.
3. C. Creusot, B. Chesnay, S. Johnson, S. Nakano, Y. Yamauchi, Y. Yanagisawa, J.G.M Goncalves, and V. Sequeira, "Innovative Approaches to DIE/DIV Activities at the Rokkasho Reprocessing Plant", *7th International Conference on Facility Operations – Safeguards Interface*, Charleston, SC, USA, February 29 – March 5, 2004.
4. V. Sequeira, J.G.M Gonçalves., G. Boström, M. Fiocco, and D. Puig, "Automatic scene change analysis of large areas", *Proc. INMM 45th Annual Meeting*, Orlando Florida, USA, July 18-22, 2004.
5. D. Girardeau-Montaut, M. Roux, R. Marc and G. Thibault, "Change Detection on Point Cloud Data acquired with a Ground Laser Scanner", in Proceedings of the ISPRS working group VIII/2, Volume XXXVI, Part 8/W2, Germany, 2004.
6. G. Boström, M. Fiocco, D. Puig, A. Rossini, J. G.M. Gonçalves, and V. Sequeira, "Acquisition, Modelling and Rendering of Very Large Urban Environments", *Proc. 2nd International Symposium on 3D Data Processing Visualization & Transmission*, Thessaloniki, Greece, 6-9 September, 2004.
7. W. C. Stone, M. Juberts, N. Dagalakis, J. Stone, and J. Gorman, "Performance Analysis of Next-Generation LADAR for Manufacturing, Construction, and Mobility", NISTIR 7117, National Institute of Standards and Technology, Gaithersburg, MD, May 2004.
8. http://www.zf-laser.com/
9. http://www.toposys.com/
10. http://gis.leica-geosystems.com/
11. S. Rusinkiewicz and M. Levoy, "Efficient Variants of the ICP Algorithm", *Proc. Third International Conference on 3D Digital Imaging and Modeling*, Quebec City, Canada, 28 May - 1 June 2001.
12. M. Fiocco, G. Boström, J.G.M. Gonçalves, and V.Sequeira, "Multisensor fusion for Volumetric Reconstruction of Large Outdoor Areas", *Proc. Fifth International Conference on 3D Digital Imaging and Modeling,* Ottawa, Canada, June 13-17, 2005.
13. http://www.optech.ca/

Chapter 11

UNDER VEHICLE INSPECTION WITH 3D IMAGING

Safety and Security for Check-Point and Gate-Entry Inspections

S. R. Sukumar, D. L. Page, A. F. Koschan, and M. A. Abidi
Imaging, Robotics, and Intelligent Systems Laboratory, The University of Tennessee, Knoxville, TN 37996-2100, USA, {ssrangan,dpage,akoschan,abidi}@utk.edu

Abstract: This research is motivated towards the deployment of intelligent robots for under vehicle inspection at check-points, gate-entry terminals and parking lots. Using multi-modality measurements of temperature, range, color, radioactivity and with future potential for chemical and biological sensors, our approach is based on a modular robotic "sensor brick" architecture that integrates multi-sensor data into scene intelligence in 3D virtual reality environments. The remote 3D scene visualization capability reduces the risk on close-range inspection personnel, transforming the inspection task into an unmanned robotic mission. Our goal in this chapter is to focus on the 3D range "sensor brick" as a vital component in this multi-sensor robotics framework and demonstrate the potential of automatic threat detection using the geometric information from the 3D sensors. With the 3D data alone, we propose two different approaches for the detection of anomalous objects as potential threats. The first approach is to perform scene verification using a 3D registration algorithm for quickly and efficiently finding potential changes to the undercarriage by comparing previously archived scans of the same vehicle. The second 3D shape analysis approach assumes availability of the CAD models of the undercarriage that can be matched with the scanned real data using a novel perceptual curvature variation measure (CVM). The definition of the CVM, that can be understood as the entropy of surface curvature, describes the under vehicle scene as a graph network of smooth surface patches that readily lends to matching with the graph description of the *apriori* CAD data. By presenting results of real-time acquisition, visualization, scene verification and description, we emphasize the scope of 3D imaging over several drawbacks with present day inspection systems using mirrors and 2D cameras.

Key words: under vehicle inspection, laser range scanning, surface shape description, 3D surface feature.

A. Koschan et al. (eds.), 3D Imaging for Safety and Security, 249–278.

1. INTRODUCTION

This chapter documents the research efforts of the Imaging, Robotics and Intelligence Systems Laboratory (IRIS) at the University of Tennessee, Knoxville, towards the construction and application of modular robotic systems for under vehicle inspection. In particular, we will focus on our contributions by demonstrating the convergence of 3D sensing technology, modular robotics and automation using 3D computer vision algorithms as a significant capability for deployment at gate-entry terminals, check points and parking lots of buildings with military, commercial and civilian interests. Towards that end, we will begin this chapter with a brief discussion of the state-of-the-art systems; document their drawbacks and hence list the expectations on a reliable under vehicle inspection system. These expectations serve as our motivation in proposing a multi-sensor robotic solution for the under vehicle inspection problem.

1.1 State-of-the-art systems

The first idea implemented and marketed for under vehicle inspection was to use a mirror at the end of a stick as illustrated in Figure *11-1*. The mirror-on-a-stick inspection though deters potential attacks, is primitive and exhibits some drawbacks. First, if the security personnel are slow in detecting the bomb, then they are still vulnerable to detonation before the inspection is complete. This weakness is of major concern as the security personnel must be within physical proximity of the vehicle. Second, the physical constraints of the mirror only allow about 40-50% coverage of the vehicle undercarriage. The center line of the vehicle in particular is difficult to reach, and the subsequent viewing angles are oblique. Moreover, the mirror-on-the-stick system though inexpensive, is only a visual inspection scheme and does not readily lend to archival and automation.

With the mirror-on-the-stick approach proving to be not so efficient with many limitations, the next logical evolution was the “buried sensors” approach. The idea was to embed cameras under ground and acquire images as the target vehicle drives over the sensor suite. The image data is further processed and visualized as high resolution mosaics[1]. With such a “buried sensors” approach, the motion of the target vehicle determines the resolution, coverage and completeness of the data required for inspection. Also, the embedded sensor approach consumes more time for operation and maintenance, leading towards the development of mobile robots such as the omni-directional inspection system ODIS[2] and Spector[3]. These low-profile robots mounted with cameras and capable of sneaking under cars and trucks recording images as video frames have proven to be a significant

improvement over the mirror-based systems[4]. Due to the low clearance of most cars, the field of view using a single camera, as seen in Figure *11-2*, becomes too restrictive that some components cannot be discerned from the live video stream even by a human inspector.

Both the robotic approach and the "buried sensors" approach lend to remote inspection and flexibility for archival and automation. But the robotic solution is favored over "buried sensors" approach because the acquired sensor data is only dependent on the robot's motion (which is measurable or can be controlled by the remote inspector) instead of the target vehicle's motion. The robotic solution also lends easily to multiple inspections of the same vehicle and enables the inspector to focus on a particular area of interest on a case-by-case basis. Several robots have hence been designed and among several enhancements on the preferred robotic solution, mobile robots with intensity cameras were made independent of illumination and hence operable even during the night by mounting light sources or using night vision cameras or thermal sensors. We show some examples from the data that we collected using such a robotic platform with visual and thermal sensors[5] in Figure *11-3*. We observe that the mosaic generated using the images acquired in a single pass provides more coverage compared to the mirror-on-the-stick approach, at the same time, suggesting the need for a better system to maximize visual information from a single pass using as few sensors as possible. Our proposed 3D approach is one such enhancement that also fills most of the other gaps observed with the contemporary systems.

Figure 11-1. The traditional method for inspecting a vehicle undercarriage is to use a mirror attached to the end of stick. This mirror-on-a-stick approach enables security personnel to search wheel wells and other vehicle cavities. On the right, we show a picture of the scene the inspection personnel views on the mirror.

Figure 11-2. An alternative to a mirror-based inspection is a low-profile mobile robotics platform (called SafeBot) that can navigate under a vehicle while inspection personnel remain at a safe distance. On the right, we see a single frame from the camera on the robot. Due to low ground clearance of the automobile and restricted field of view of the camera on the robot, a single frame only shows a small section of the undercarriage.

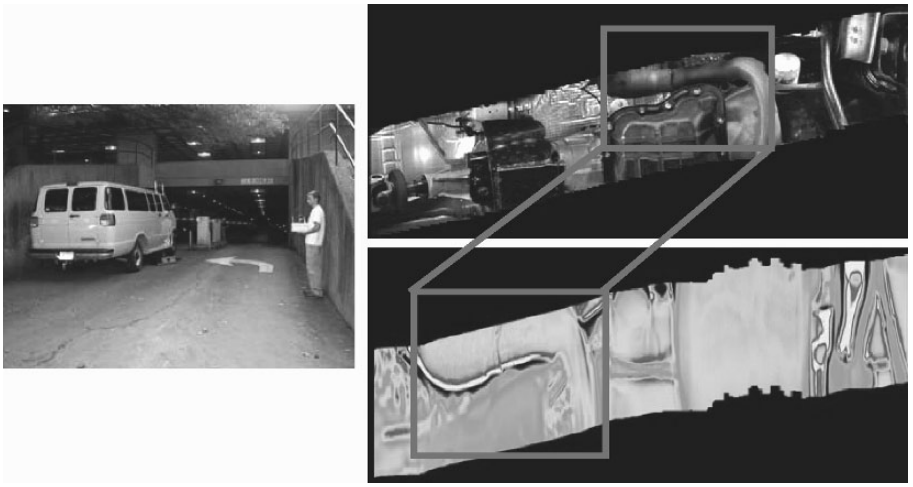


Figure 11-3. The images on the right are high resolution mosaics of visual and thermal data from the robot scanning the carriage. The thermal mosaic at the bottom is color coded, with the hot regions appearing in red and the colder regions in blue. The red boxes show the thermal scene characteristics corresponding to the visual scene. The mosaics were generated using a phase-correlation based method [5]. (See also Plate 38 in the Colour Plate Section)

1.2 Reliable under vehicle inspection

The state-of-the-art systems emphasize that the visual cameras alone may not be sufficient for a robust under vehicle inspection system. In addition to visual information, we see that a thermal sensor can help detect components that are not part of the exhaust system, chemical sensors can help sense explosives in the scene and nuclear sensors can detect radioactivity. The multi-modality sensor data aids a human inspector and also contributes towards potential automation for the detection of bombs, explosives and contraband. However, the major contribution with our research is in enhancing existing inspection robots with 3D sensors to acquire geometric information of the under vehicle scene and hence using the 3D data for automatic detection of anomalous objects in the scene as potential threats. In trying to provide such a robotic system as a reliable solution for under vehicle inspection, we enlist the following characteristics expected of an under vehicle inspection system:

- Ability to acquire and communicate data from a safe stand-off distance.
- Maximal coverage with minimal effort reducing the number of inspection iterations.
- Independence from environmental conditions like illumination, temperature, humidity etc.
- Rendering data in a visual and spatially understandable form for a remote inspector to make decisions.
- Flexibility for digital archival and automatic threat detection using simple and efficient algorithms.
- Inexpensive and less cumbersome maintenance and operation.

Our proposed framework and the automation procedures consider each one of the listed characteristics to improve upon the state of the art. In Section 2, we will elaborate on the “sensor brick” architecture that takes care of the acquisition, communication and pre-processing for visualization of visual, thermal, nuclear and range data. The proposed robotic architecture in addition to establishing the paradigm of communication between interchangeable sensors has processing software for multi-sensor integration and visualization. Then, in Section 3, we focus on the use of 3D sensors as an illumination independent approach to spatial scene mapping and show the acquired 3D geometry acting as a visualization bed for multi-sensor information. In Section 4, we discuss potential automation algorithms on the acquired 3D data before making conclusions for future direction in Section 5.

2. THE "SENSOR BRICK" ARCHITECTURE FOR ROBOTIC UNDER VEHICLE INSPECTION

We will begin this section by emphasizing the need for a robotic architecture towards extracting scene intelligence using multiple modality sensors on mobile robots. Then in Section 2.2, we will provide the definition of a "sensor brick" architecture that showcases the potential of our robots as being robust, compact, modular and also independent units with high degree of interoperability. The deployment of multiple modality visual, thermal, range, and nuclear sensors on robots, that can both act independently and in combination with one another, is a significant step in achieving the goals that we had listed towards a reliable under vehicle inspection system.

2.1 The need for a robotic architecture

Based on our study of the state-of-the-art robotic systems for under vehicle inspection, we observe that vision-based sensors provide useful information for inspection with a human required for making a decision about a threat in the scene. In removing the susceptibility to human carelessness or error in detecting bombs that could cause destruction to property and life, we require "perception intelligence" from the mobile robot. By perceptual scene intelligence, we are referring to aspects of sensing abnormal "activity" in the scene.

The activity in the scene could be in the form of an additional object that is not supposed to be part of the undercarriage but attached to the vehicle, or could be neutron radiations from a nuclear source, or free radicals from a chemical explosive. Such scene information can only be inferred using specific sensors. For example, an anomalous component in the undercarriage could be detected by visualizing the geometry from a range sensor, or by searching for abnormal temperatures of the components in the undercarriage using a thermal imager. Nuclear radiations, chemical explosives and biological agents can be also detected using specific sensor packages. With such advanced sensing technologies readily available, we are now confronted by the challenge of how to use these sensors in a real-world environment for automatically localizing potential threat objects in the under vehicle scene. Such automation calls for interaction and fusion of spatial data from vision-based 2D and 3D sensors with 1D measurements from the nuclear, chemical and biological detectors. Thus the necessity to define a robotic architecture for sensor interaction, data fusion and communication arises.

Our "sensor brick" architecture is an effort towards the definition of such a collaborative, multi-sensor communication interface to integrate visually

understandable maps of activity on spatial (visual) information. In addition, to visualization, integration and communication of multi-sensor data our proposed architecture introduces different levels of modularity in choosing specific sensors for particular environments without having to reconfigure any of the hardware or software components within the robotics framework. We will emphasize on this modular feature with our mobility platforms, that we call "SafeBots" later in this section after we briefly explain the robotics architecture for this under vehicle inspection application.

2.2 Definition of the sensor brick architecture

A "sensor brick" is an autonomous platform promoting the notion of a three-module concept with mobility, sensing, communication capabilities. The sense-fuse-communicate (SFC) modules of each "sensor brick" have well defined functionalities. That is, the sensor module contains one or more sensors to collect data about the target environment, the fusion module processes this data and incorporates reasoning and analysis and the communication module transmits this information to appropriate end users. Thus, each "sensor brick" built on this SFC structure "sees", "thinks" and "reports" as an independent, self-contained robot to a remote end user. The construction of such sensor bricks begins by first making the sensor component a "plug and play" device that when powered will automatically load in the information about the individual sensors in the brick and determine its functionality. By such a construction, the task of improving functionality by combining the bricks is extremely simplified because each sensor module is conscious about its composition and functionality, readily lending to interaction among sensor bricks within the architecture. We have pictorially depicted this concept in Figure *11-4*.

Each "sensor brick" utilizes primary subsystems to acquire data, to provide locomotion, as well as the artificial intelligence and computing resources that are necessary for navigation, data processing, and communication. Additionally, each brick also has an independent power supply system which consists of any batteries, converters, or other devices pertaining to the operation of other components in the brick. The specific choice of each component within this brick architecture is driven by an important design consideration to use readily available, reliable, time-tested, rugged and inexpensive technology that will facilitate maintenance of the system components and also minimize adverse effects due to a component failure. In our implementation, this design consideration explains the use of off-the-shelf components from commercial vendors allowing integration and improvement of our bricks with advanced technologies in the future.

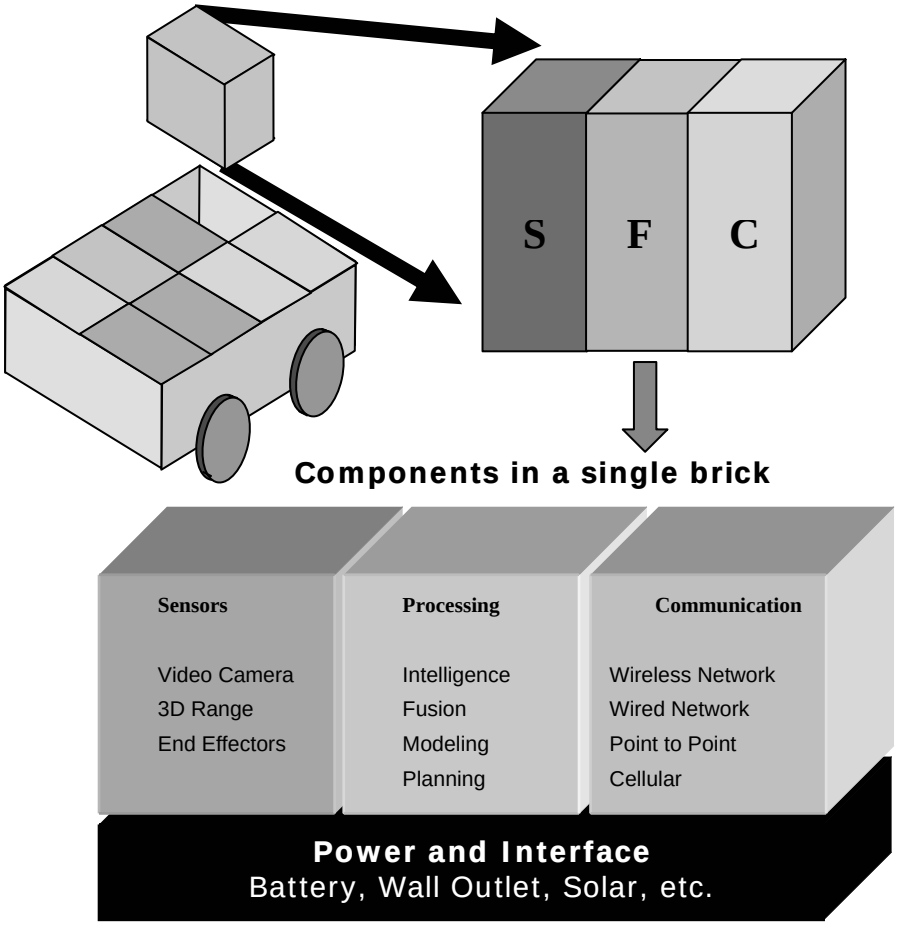


Figure 11-4. The sensor bricks fit into mobility platforms without having to reconfigure hardware or software in communicating information. They are able to do because each one of the bricks has its own power, communication, processing and sensing module with delineated tasks and task definitions.

We would also like to emphasize that the SFC design strategy achieves a different level of modularity, scalability and reconfigurability, in adding to existing robust architectures like JAUS[6]. In Figure *11-5*, we show the different levels of modularity and hence functionality achievable using the "sensor brick" architecture. We begin with the basic mobility bricks at the lowest level and bring together sensor modules with sensors such as cameras, thermal sensors, range sensors and nuclear sensors along with the hardware for definition, communication and power. Each sensor module with the mobility brick can act as an independent robot to acquire and communicate data wirelessly. We denote this as Level 2 Modularity in the

picture. Bringing together two or more Level 2 "sensor bricks" in building a multi-functional robot constitutes the next level of modularity, Level 3. Implementing such a modular approach to the design and construction of a robotic system for under vehicle inspection provides a method that overcomes the deficiencies of application-specific uniquely built state-of-the-art systems.

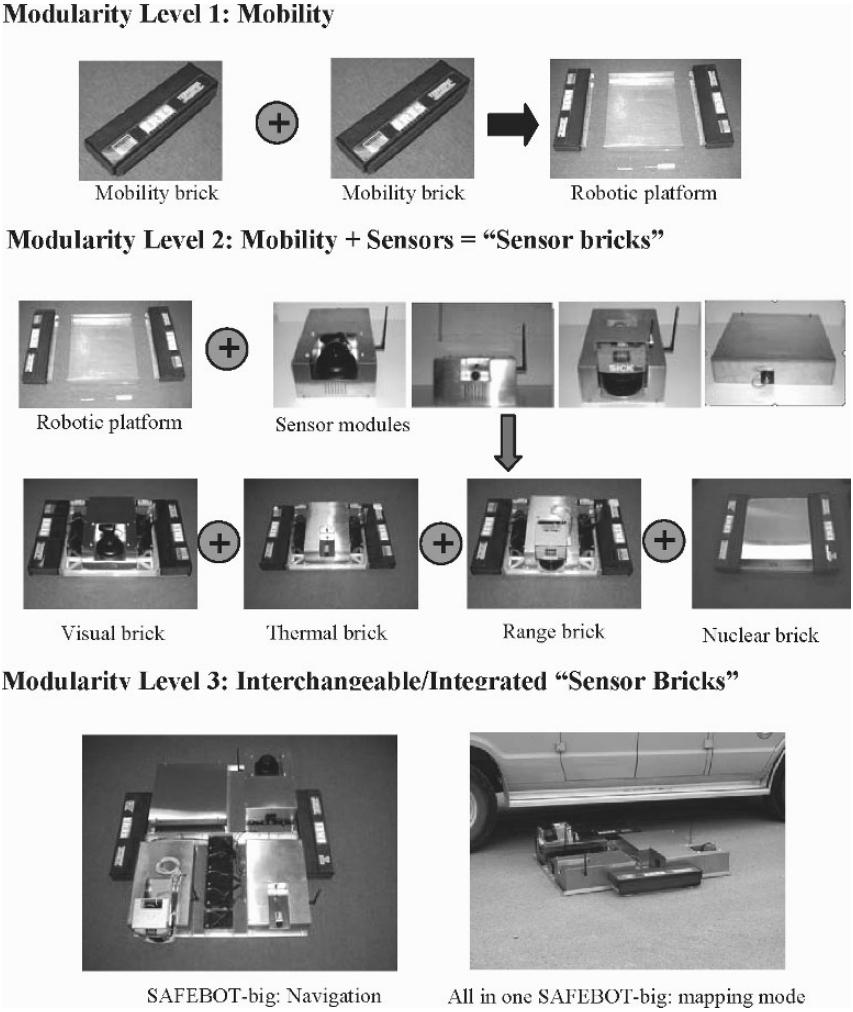


Figure 11-5. The "sensor brick" architecture lends to different levels of modularity. The base level begins with mobility platforms capable of housing self-contained wireless enabled sensor modules that can both function independently on its own as a mobile robot and also as in combination with other sensors also.

So far, we have provided a framework for the construction of a SafeBot that can navigate under the vehicle of interest carrying a suite of sensors and communicate via a wireless remote connection enabling the inspection to be carried out at a safe stand-off distance[7] of 300 feet from the target vehicle. The SafeBot in the sensor brick architecture is capable of transmitting data from its sensors to a base station and receiving control commands. Additionally, the low-profile stature of the SafeBot is designed towards getting under most cars and also providing a decent coverage of the undercarriage with storage and archival options. The robust implementation of such a robotic architecture based on well established hardware interaction protocols using RS232C serial interface standard for control signals and 802.11g wireless protocol for sensory data[8], leads to visualizable results in 3D virtual reality environments that we will demonstrate in the following section.

3. 3D IMAGING AND VISUALIZATION FOR UNDER VEHICLE INSPECTION

This section will primarily focus on the design considerations, processing and visualization of the undercarriage scene data from the range sensor brick. After summarizing the state-of-the-art in optical 3D sensing systems, we will outline the issues towards the deployment of 3D sensors on unmanned robotic platforms and make recommendations based on our experience. Later in Section 3.3, we present results of two such 3D sensing methods and demonstrate the visualization of multi-sensor data in a 3D virtual environment. We show how the 3D geometry that is transmitted for visualization to a remote computer could be very useful to a remote inspector in recognizing components in the undercarriage and also making the identification of threat objects easier.

3.1 3D Sensing methodologies

In this subsection, we would like to very briefly discuss several 3D sensing methods that we considered implementing on the robotic platform. The discussion helps in understanding why we chose to use commercially available 3D scanners on our robots. We begin by presenting a classification of different techniques in Figure *11-6*. Our classification is based on a study similar to Blais's review[9] on 3D range sensing. Different methods of 3D sensing can be broadly classified into two as passive and active. Let us begin our discussion with passive techniques. Passive triangulation (stereo) is the

way humans perceive depth and involves two cameras taking a picture of the same scene from two different locations at the same time. Primarily, passive 3D reconstruction methods take images as projective inputs of the 3D world and recover depth cues using specific algorithms. For example, depth information can be extracted by matching correspondences in the two images and using epipolar geometry. Passive triangulation algorithms are challenged by the ill-posed problem of correspondence in stereo matching.

Another idea to extract 3D shape is by using the principle of focusing and defocusing. The method infers range from two or more images of the same scene, acquired under varying focus settings. By varying the focus of a motorized lens continuously and estimating the amount of blur for each focus value; the best focused image is determined. A model linking focus values and distance is then used to approximate distance. The decision model makes use of the law of thin lenses and computes range based on the focal length of the camera and the image plane distance from the lens' center. However, this method has its limitation on the blur estimation that influences the focal length and hence the derived range. Also, the system required for the imaging process is not readily suited for mobile robots. Other methods such as the shape from texture and shape from shading also do not suit our application, because of the assumptions about certain scene characteristics that dictate these methods.

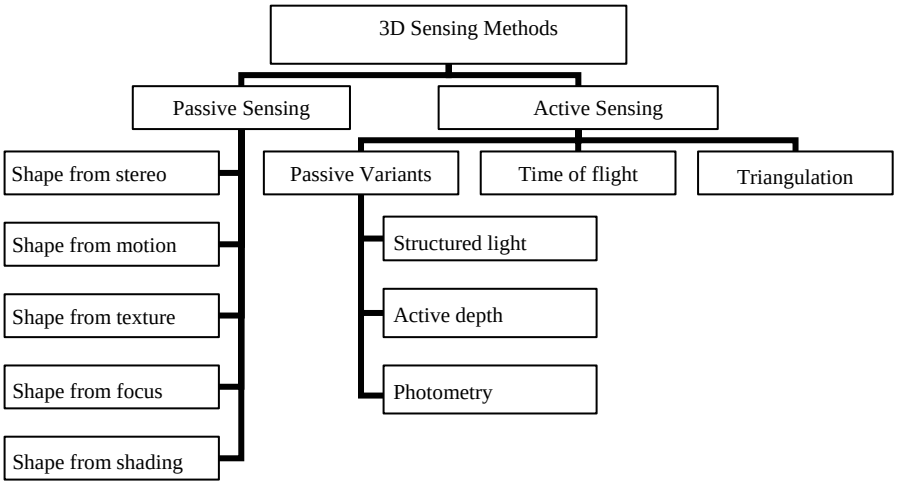


Figure 11-6. The classification of popular range acquisition systems based on the physical principle of range computation. Of these methods, we find the time-of-flight approach and the triangulation approach meeting accuracy, resolution, and real-time acquisition requirements for our under vehicle inspection application.

The shape from motion approach that recovers 3D structure by tracking specific features in an image sequence appears to be a promising algorithm for our robots with a single camera. However, the 3D depth accuracy that we expect from such a system might not preserve the scene fidelity for our interest in automatic scene inference. This drawback relates to the inherent limitations of the feature correspondence problem in stereo vision. Also, with the limited field of view with a single camera, multiple passes under the vehicle are required to reconstruct the entire undercarriage in 3D. Most of the passive methods discussed thus far, can be extended for better accuracy using an active source as an additional component with the camera. However, the only problem is that the passive variants of shape from images, may not serve well as a real-time acquisition approach for our application. For example, the depth estimation using structured light requires a stationary scene that a camera will have to image using different patterns of coded light. Imaging the same stationary scene with different patterns of structured illumination takes about 20 seconds for a small scene of interest. Given the low clearance and limited field of view, in our application, mapping the entire undercarriage would become a tedious time-consuming task.

So, we are left with two other methods from the classification chart namely the active triangulation and time-of flight systems. Both these systems are laser-based. With the active triangulation scheme, a laser in the visible spectrum (usually a line laser) illuminates the scene. The laser line profiles a line of the surface in the scene that is imaged using a high speed camera. By using a special calibration procedure to estimate depth, the surface profiles can be accumulated into a metric 3D structure by moving the camera and laser arrangement over the scene of interest. This approach can be configured to a high degree of accuracy and readily lends to our application, where the scene is static and the mobile robot with the sensor can map the static scene real-time. Again, being camera-based, the system has the same field-of-view restrictions as the passive methods on vehicles that have low clearance. On the other hand, the time-of-flight systems are based on physical principles of estimating distance from a scene by shooting out a laser and sensing the reflection. With the knowledge of the speed of the laser, the observed time taken for the laser to travel, reflect and return is then used to compute the distance from the laser source. Though this approach does not provide high accuracy as the laser triangulation methods, these sensors are capable of mapping the entire undercarriage in a single pass.

With each acquisition method having its own advantages and disadvantages, we had to make several considerations in finally building a 3D imaging system for our application. Out of the several challenges in the robotic data collection, we identified the two most significant ones to be the field of view and accuracy. We realized that the field of view is limited by

the ground clearance and the large variation in the size of the components that make up the scene. Hence, we require that our system accommodate the variance in the ground clearance of a variety of cars and automobiles from different manufacturers (typically varying from a minimum of 10 centimeters in compact cars to up to 90 centimeters in large trucks). We also require that the 3D data provide us high fidelity shape information from the scene for us to be able to perform shape analysis. Considering these factors in addition to the list in Section 1.2, we narrow down our choice to the time-of-flight and laser-triangulation based scanners over the other methods.

3.2 3D Scanning for under vehicle inspection

We constructed two prototypes using both these 3D scanners as two different sensor configurations for the SafeBot. We mounted a laser range finder (SICK LMS 200) and a laser-profile scanner (IVP Ranger SC-386) on the two mobility platforms to acquire the under vehicle scene data. Both the mobile robots, with two different 3D sensor configurations, were capable of scanning most automobiles with enough ground clearance for our robotic imaging system to slip under. We have tested our robot under several cars inside parking lots, collecting and visualizing data almost in real time.

Before we present the results from our system, we document the processing algorithms that we have used for visualization purposes. The SICK system provides us with 3D point samples of the surface. For visualization purposes, we triangulate the point cloud. We use the implementation of the surface reconstruction method proposed by Hoppe[10]. With the IVP system; the 3D modeling requires a different approach. We have used the model reconstruction pipeline discussed by Page et al.[11] for automotive reverse engineering. The process described in the paper includes data collection, surface registration[12] and mesh integration[13]. We present the results from both the sensor configurations with a detailed discussion in the following subsection.

Let us begin our discussion with the time-of-flight system that shoots out a laser pulse that gets reflected from objects within 8 meters radius in the 180 degree field of view of the sensor. The built-in electronics of the sensor detects the returned pulse and computes the distance based on the time taken for the pulse to reach the object and return to the sensor. We show the results of using this time-of-flight sensor configuration to inspect the underside of a van (Dodge RAM) in Figure *11-7* and a car (Dodge Stratus) in Figure *11-8*. We have shown the photograph of the scene of interest under the car, Figure *11-8 (a),* along with the 3D wire frame model after processing the range data, Figure *11-8* (b). We show the model with the range coded as color in Figure *11-8 (c)* and the shaded 3D rendered model in Figure *11-8 (d)*.

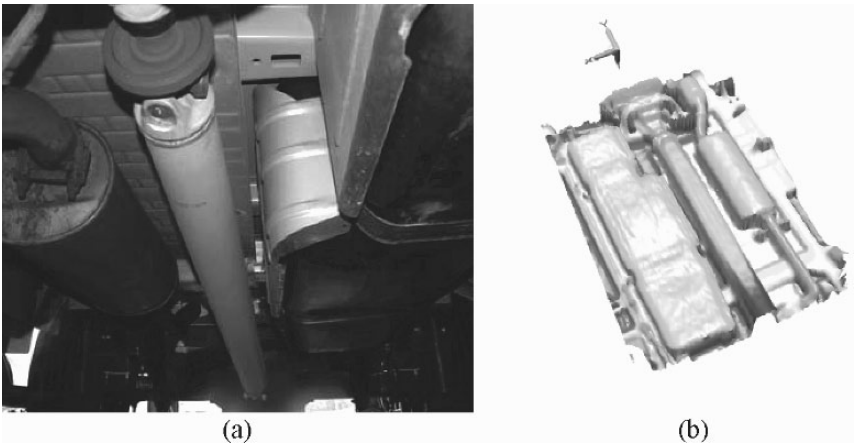


Figure 11-7. Under carriage of the Dodge RAM van. (a) The photograph of the scene of interest. (b) The rendered 3D scene using the time-of-flight system.
(See also Plate 39 in the Colour Plate Section)

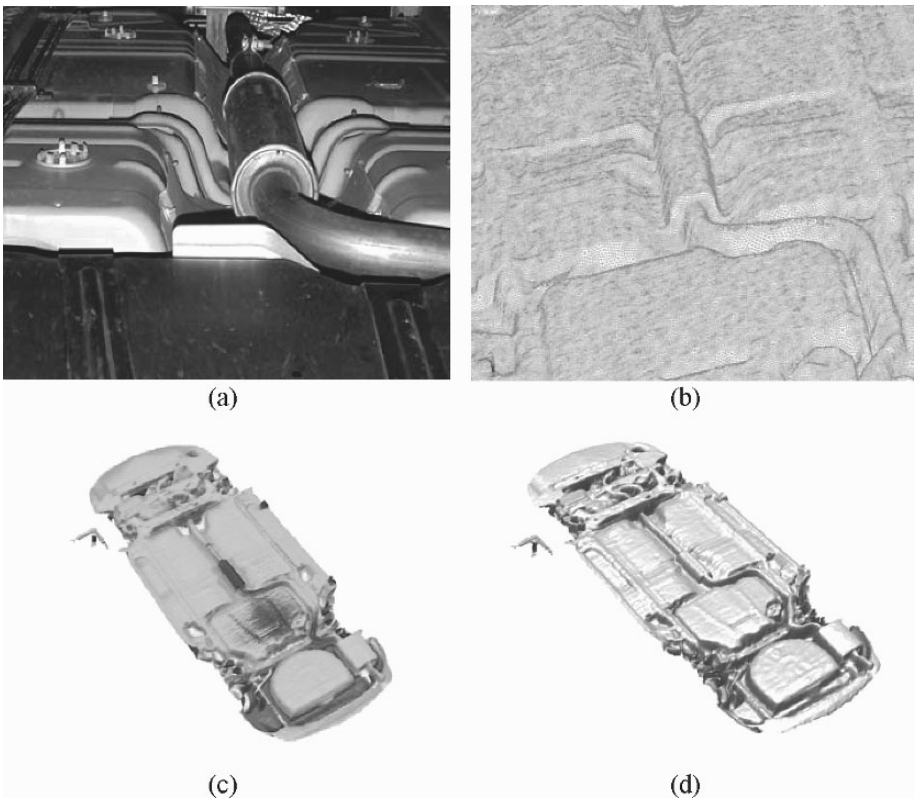


Figure 11-8. The 3D data of the undercarriage of a Dodge stratus car acquired using the range sensor brick – time-of-flight sensor configuration. (a) Photograph of the scene. (b) Wire frame emphasizing depth. (c) Range-coded 3D scene. (d) Complete scan rendered using OpenGL.
(See also Plate 40 in the Colour Plate Section)

Using the time-of-flight prototype we are able to model the complete structure information of the undercarriage with a single pass with the geometric accuracy in the order of a few (1-5) centimeters. A single pass along the center line of the vehicle takes about 30 seconds to map the entire undercarriage and the observed accuracy varies with the ground clearance of the automobile. We attribute the reason for the below par accuracy to the limitation in the timing electronics of the scanner. The time-of-flight systems also require a minimum stand-off of 25 centimeters at which we get 5-10 millimeters of accuracy.

Next, we present our experience with using the IVP range scanner. The IVP RANGER system consists of two parts: the range sensing camera and a low power laser. The sheet-of-light laser shoots on the target scene and the camera detects the laser as a single profile of the object surface shape parallel to the imaging plane. Such profiles are accumulated by moving the hardware setup on the robot relative to the scene traversing the area of interest. We use a pre-calibrated setup of the scanner. Although a powerful laser was used to counter ambient lighting, we could not compensate for spectral reflections since the metallic surfaces under a vehicle exhibit strong spectral reflection properties. A laser further complicates this problem as internal reflections lead to significant errors in range estimation. A promising solution for this problem involves the use of an optical filter tuned to the frequency of the powerful laser. The filter in front of the camera allows the data collection to isolate the initial reflection of the laser and thus improve the range sensing.

We present the data from this sensor configuration in Figure *11-9* where we have attempted to image the exhaust system of the Dodge van. We again show the photograph of the scene of interest and the range coded 3D data. With the IVP range sensor, we are able to achieve depth accuracy of 2-5 millimeters and the data provides promise for shape analysis algorithms.

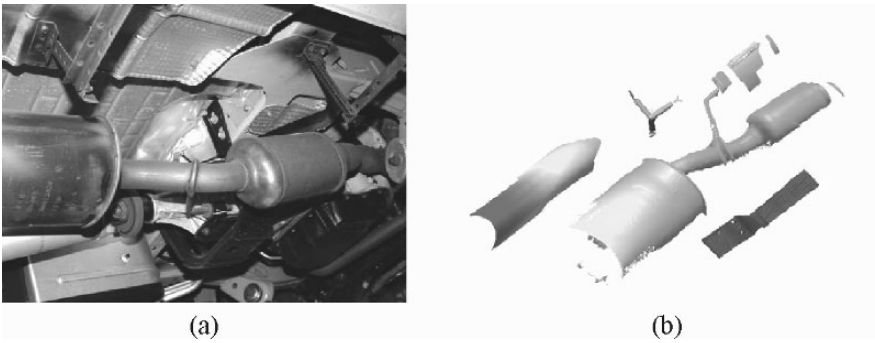


Figure 11-9. Laser-profile scanning for vehicle inspection using the IVP RANGER system. (a) The scene of interest. (b) Range-coded 3D model. (See also Plate 41 in the Colour Plate Section)

However, an issue of concern using this system is that of view occlusions. Objects underneath a vehicle have different shapes and scales located at different depths. With a pre-calibrated setup, we have indirectly fixed a particular field of view that can create problems at the time of scanning. The sensor cannot view objects that are occluded by other components resulting in partial and incomplete data. Our solution to fill up occlusions using multiple scans is a laborious one involving multiple passes and redundant information. The IVP sensor configuration allows high fidelity geometry that the other sensors do not offer, but at the price of potential data overload and redundancy. The data that we show in Figure *11-9* of such a small region of interest has been integrated using four passes under the vehicle. Based on the design and extensive testing of both these sensors for our application, we recommend the use of the time-of-flight system for remote visualization and inspection while the laser-triangulation system can be used when we require high accuracy on a particular area of interest.

3.3 Multi-sensor visualization of the under vehicle scene as virtual environments

So far, we have collected 3D data of the under vehicle scene. Now we will show how the 3D data can be used as a visualization bed for visualizing multi-sensor data from sensor bricks within the architecture and also localizing potential threat information from nuclear, chemical or biological detectors.

The results that we will discuss in this section are based on a simple experiment that we conducted on the Dodge Stratus car. We attached two radioactive Cesium sources under the car, and then used our SafeBot with multiple sensors, to scan the vehicle. We measured the radioactivity in 16 locations around the car seen as blue squares in the Figure *11-10*. Using a source localization algorithm[14], we were able to find the location of the radioactive sources in the 3D scene of the undercarriage. Such a capability is of significant assistance to a remote inspector to localize a threat in a physical and spatial sense and act on it accordingly.

Furthermore, we are also able to provide a remote inspector the ability to focus on a particular region and visualize the visual image data and thermal data of the scene on the 3D model. We have shown them separately on the image for clarity reasons though these datasets can be visualized as texture as shown in Figure *11-11*. The images from the color cameras are textured on the geometry from the laser scanners.

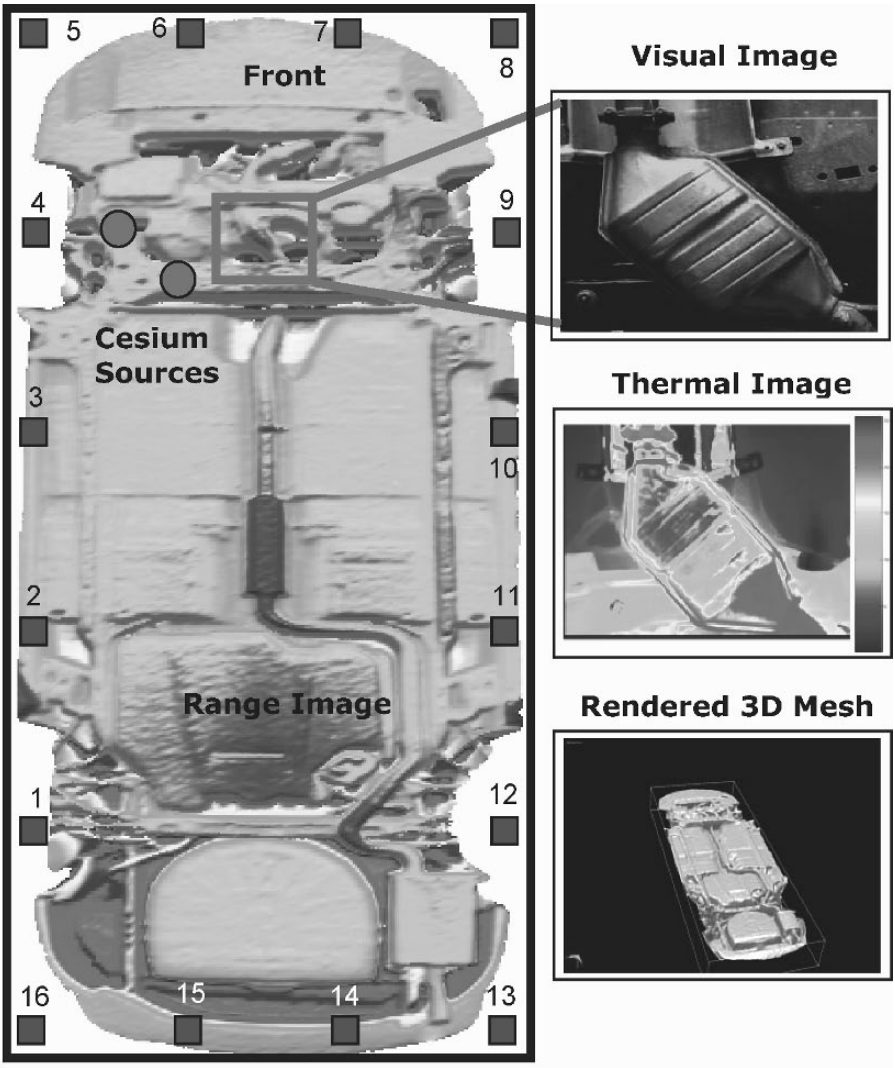


Figure 11-10. Visualization of multi-sensor data on the 3D geometry helps isolate radioactive sources in the scene, using our robotic solution. Also, we are also able to spatially relate the under vehicle scene to visual and thermal data. These images show that the visualization in a 3D environment brings together the functionality of each of the visual, thermal and range bricks into one single interface for the remote inspector for easy manipulation and interpretation with the multi-sensor data adding to extra scene intelligence.
(See also Plate 42 in the Colour Plate Section)

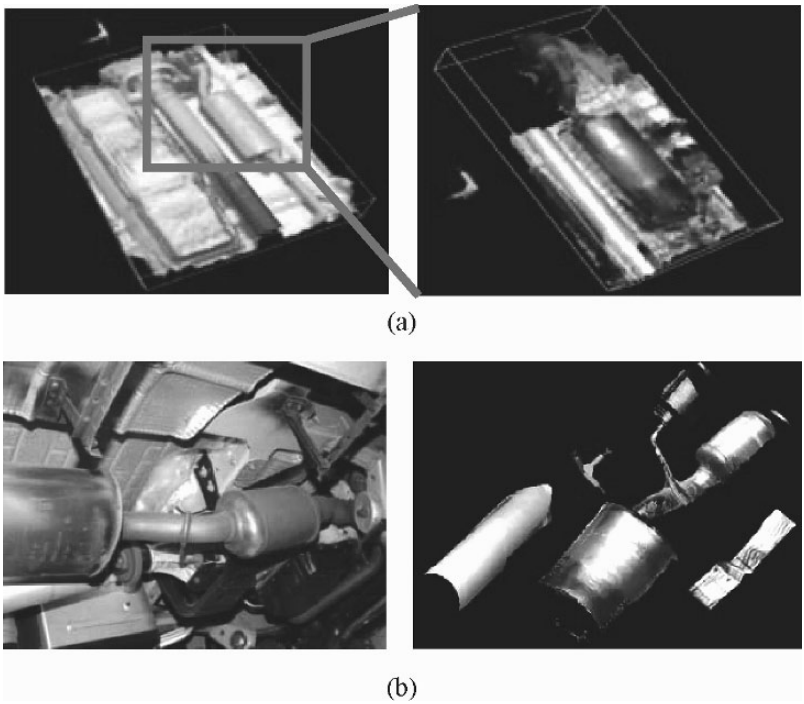


Figure 11-11. Both the 3D scanners lend to easy texture mapping. (a) Texture mapped 3D data from the SICK scanner. Though the entire under carriage is mapped in 3D, the field of view of the camera on the robot does not image the entire scene. Hence, from a single pass, only a part of the scene can be visualized as textured 3D models. (b) Texture mapped on the IVP data. The IVP laser scanning system is a camera based system that can also be used to compute range and color maps simultaneously. (See also Plate 43 in the Colour Plate Section)

4. AUTOMATION FOR THREAT DETECTION

We have thus far discussed the construction of the robotic platforms with interchangeable sensors capable of real-time acquisition, processing, integration and visualization of multi-sensor data in 3D virtual reality environments. In this section of the chapter, we will present potential methods of using the 3D information towards automatic scene inference. The first approach that we will explain is the automatic scene verification using the SICK scanner data using a popular 3D scene registration algorithm. The second approach is based on a shape analysis algorithm that defines a novel shape measure based on 3D surface curvature that can describe CAD surfaces and scanned 3D mesh models alike as symbolic structural graphs.

4.1 Scene verification approach

Let us consider the following scenario with John Q. Citizen who works as an analyst at a secure facility for some three-letter agency (TLA) within the U.S. government. John drives to work each day and passes through a security checkpoint to enter the facility. The TLA sticker that John has placed in the lower corner of his windshield validates his access. As he approaches the gate, the TLA security personnel observe the appropriate sticker and wave John into the secure area. This procedure is typical when terrorist threat levels are low, but when threat levels rise, TLA policy requires that security personnel check for the sticker and additionally inspect John's vehicle a little more thoroughly. The assumption is that John is a trusted employee and that his sticker is valid, but the inspectors are looking for bombs or other contraband that may have been hidden on the vehicle without John's knowledge. Essentially, John and—more precisely—his vehicle is a target of opportunity when the vehicle is outside the TLA facility. When John drives to a restaurant for lunch, his vehicle sits in the parking lot where the sticker advertises his association with TLA. A terrorist might exploit this opportunity by planting a bomb under the vehicle and then waiting for John to return to the TLA checkpoint. At the appropriate moment, the terrorist could remotely detonate the bomb and thereby kill John and the security personnel. The loss of life and the associated destruction would compromise the perimeter security of the entire TLA facility.

The 3D scene verification approach applies directly to such a scenario that assumes that we already have a scan of the underside of the automobile, and we are looking for potential modifications made to the undercarriage from the previously archived 3D scans. We demonstrate the difference shell idea applied to this scenario in Figure *11-12*. A black electronic board that simulates a threat object was attached to the undercarriage of the Dodge van. The vehicle underside was then scanned using our robotic platform. The previously archived 3D dataset was then aligned with the freshly scanned scene with the simulated threat using our implementation of the Iterative Closest Point (ICP) algorithm[12] and computed the difference between the two aligned shells. We show the difference shell in Figure *11-12 (d)* that highlights the areas of the largest difference pointing to the location of the simulated threat object. This approach can be used to detect arbitrary modifications made to the car, like a missing muffler for example, which would have been extremely challenging for a human inspector. This ability to quickly, reliably and automatically identify modifications of the scene is a significant additional feature with 3D sensors compared to the traditional camera approach.

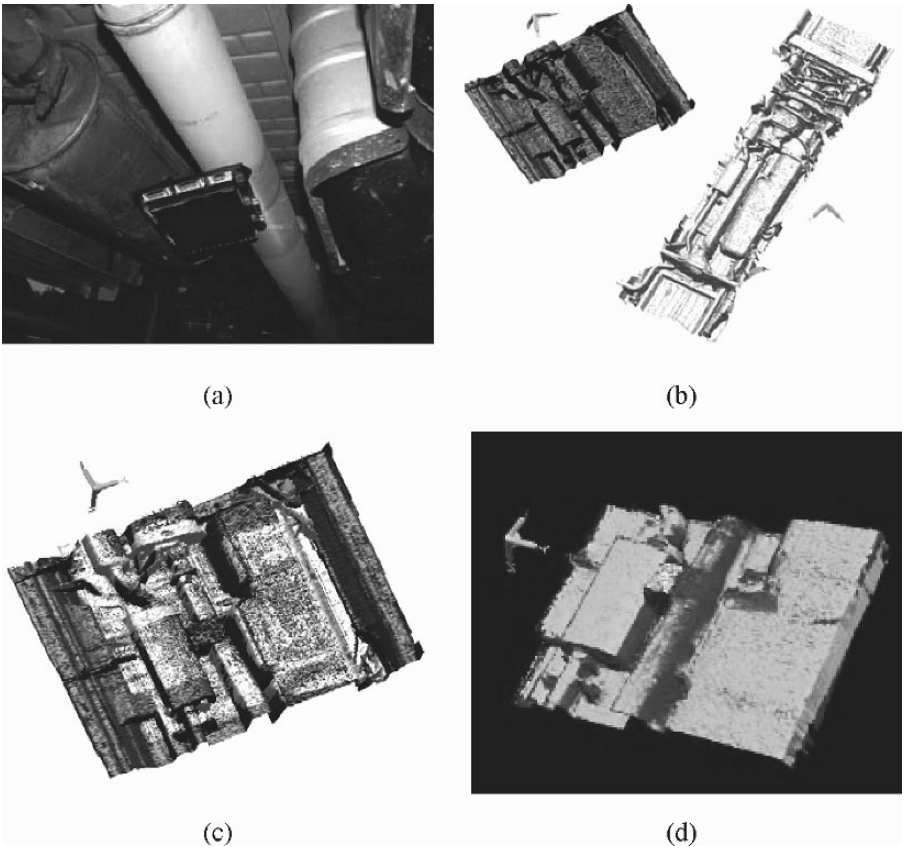


Figure 11-12. Scene verification approach for detecting changes made to the vehicle compared to previous scans. (a) An electronic board attached along the center line shaft of the Dodge van. (b) Scan of the under vehicle scene with and without the electronic board. (c) Registration result of the two shells in (b). (d) The difference shell highlights the anomalous object found. Detecting such a threat would have been very difficult using the mirror on the stick approach.

4.2 Shape analysis approach

The shape analysis approach tackles another scenario that could involve terrorists planting explosives in a car and parking the car in public access points like airports, bus and railway stations. The compromise in security from such an attack can cost many human lives in addition to the damage caused to property and functionality. With the huge volume of cars at such public access points, we can not expect to have apriori archive of the car's undercarriage to implement the scene verification approach. Instead, suppose

we have access to manufacturers CAD model of the car's undercarriage, the task of identifying components in the scene becomes a model-based object recognition problem requiring a 3D surface feature that can be used to describe CAD surfaces and scanned 3D surfaces alike. We will hence look into several features that can be used towards the goal of object description that leads to the definition of a simple yet novel curvature variation measure.

To be able to infer useful information from the scene, we are now trying to match CAD description of automotive components with partial, laser-scanned 3D geometry. We will assume that most of the automotive components are man-made and hence smooth and symmetric. We also expect the shape content in the surface description of the CAD models to be comparable to the content in the acquired high resolution geometry. This CAD-based recognition of 3D objects is a familiar problem in computer vision summarized by Arman and Aggarwal[15]. Building on Aggarwal's review several features specifically for 3D data have been proposed and demonstrated for object description.

Two such 3D features that would readily apply to our case is the shape index and curvedness measures proposed by Dorai and Jain[16] in their COSMOS framework for object description in range images. They represent un-occluded range images as maximal surface hyperbolic patches of constant shape index and formulate a shape spectrum on these patches. The definition of shape index and curvedness assumes the underlying topology of the range image. They also propose a graph-based system based on the connectivity information of the hyperbolic surface patch primitives towards object recognition. Our object description scheme that we propose is also similar to the COSMOS framework, the difference being, we have a different measure to replace the shape index. We also looked into several other features for our system that include 3D Fourier descriptors[17] local feature histograms[18], alpha shapes[19], 3D shape contexts[20] and 3D moments[21] as shift and rotation invariant point-based descriptors.

A different category of methods in the literature involve intermediate surface description like spin images[22] and harmonic shape images[23]. While spin images are data level descriptors that encode the global characteristics of the surface as a collection of points in an object centered co-ordinate system, the harmonic shape images decompose an object into its harmonics using a spherical raster. Both these methods involve high computational matching. Histogram based methods reduce the cost of complex matching but sacrifice robustness since such methods generate features based on the statistical properties of the object. Besl[24] has considered the crease angle at each edge between two triangles in a mesh as a surface feature. Osada et al.[25] demonstrate shape classification and database search capabilities of their simple shape functions using the shape distributions. Most of these methods

assume un-occluded and ideal 3D data and there are not many invariant region descriptive features that can be used for our application.

We hence identify the need for a surface feature on 3D meshes for description of CAD surfaces and scanned real data. To this end, we propose the curvature variation measure as a surface descriptor for a part-based shape representation scheme. Our CVM algorithm is based on a curvature heuristic that smoothly varying curvature conveys very little shape information, while unpredictable variations in curvature attribute to the shape's feature availability towards description. We have chosen curvature because curvature is an invariant surface property that is not affected by the choice of the coordinate system, the position of the viewer, and the particular parameterization of the surface. In an attempt to define a measure for the variation we use information theory. The more likely the occurrence of an event, the lesser the information the event conveys. For example, consider a flat surface that has uniform curvature. The density of its curvature hence is a Kronecker delta function of strength one. The entropy of a Kronecker delta function is zero. This result implies that flat surfaces and spherical objects convey little or no significant shape information. We also note that spheres of different radii will also have zero shape information and argue that change in scale (radius) adds no extra shape to the object. Furthermore, as the variation in curvature increases, broader the density of curvature gets, higher the entropy and greater the complexity for shape description. The most complex shape hence would be a hypothetical shape that has randomly varying curvature at every point.

We formulate our CVM as shown in the block diagram in Figure *11-13* and discuss in detail the implementation issues on each of these blocks in the next few paragraphs. We begin with the curvature estimation. For a given surface, the knowledge of curvature is sufficient information to define the shape. But surface curvature being an infinitesimal local feature lacks generalization to serve as a 3D feature in the computer vision context. Triangle mesh datasets are a piecewise smooth approximation to an underlying smooth surface. Several approaches to curvature estimation are summarized by Surazhsky et al.[26].

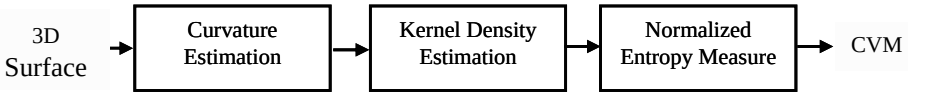


Figure 11-13. Block diagram of the CVM algorithm.

As a tradeoff between computation speed and estimation accuracy, we have chosen the Gauss-Bonnet approach proposed by Lin and Perry[27] to curvature estimation. This method uses the umbrella neighborhood of triangles immediately adjacent to a vertex to estimate the Gaussian curvature at that vertex. This method is also known as the loss of angle approach since we use the angles subtended by the edges emanating from the vertex of interest. If we consider a vertex v, then we can define the angle α_i for the corner of each triangle adjacent to v. From Gauss-Bonnet, we can estimate the Gaussian curvature, κ, of the underlying smooth surface at v as shown in the equation below where the summation is over the umbrella neighborhood and A is the accumulated area of the triangles around v.

$$\kappa = \frac{3}{A}\left(2\pi - \sum \alpha_i\right) \tag{1}$$

Now that we have computed curvature at each vertex of a mesh, our next step is to estimate the density function of curvature over the entire mesh. The simplest and perhaps the most common density estimation technique is the histogram, which Dorai and Jain[16] use for their shape spectrum. Although histograms are straightforward to implement, they do not provide accurate estimates. Additionally, histograms require selection of an origin and an appropriate bin width. Dependency on these user parameters reduces the confidence of our results when we later attempt to compute entropy from the histograms. To achieve a more robust solution, we make use of kernel density estimators (KDE) discussed by Silverman[28]. We use KDE as a tool to compute the density function p of the curvature values over the entire mesh. Consider Eq. (2) where $\hat{p}$ is the estimate of p, n is the number of vertices in the mesh, h is the bandwidth of interest, G is the kernel function and κ_i is the curvature at vertex v_i. We visualize KDE as a series of 'bumps' placed at each of the n estimates of curvature in the density space. The kernel function G determines the shape of these bumps while the bandwidth h determines their extent. With large data sets (n is large), the choice for G does not have a strong influence on the estimate. However, we recommend the Gaussian kernel although meshes provide large number of sample points.

$$\hat{p}(x) = \frac{1}{nh}\sum_{i=1}^{n} G\left(\frac{x-\kappa_i}{h}\right) \tag{2}$$

$$G(u) = \frac{1}{\sqrt{2\pi}} e^{-\frac{u^2}{2}} \quad \text{such that} \quad \int_{-\infty}^{\infty} G(x)dx = 1 \tag{3}$$

The more significant parameter for accurate and stable estimation is not the kernel but the bandwidth *h*. Data driven bandwidth optimization approaches discussed by Wand and Jones[29] such as the distribution scale methods, cross validation, L-stage, plug-in and advanced bootstrap methods theoretically aim to minimize the mean integrated square error between the actual density and the computed density. For our algorithm, we use the plug-in method for optimal bandwidth selection as this method provides useful results without the selection of user parameters. The plug-in method that uses Eq.(4) for bandwidth selection reduces the computational overhead compared with commonly used cross validation methods

$$h_{opt} = \left[\frac{243R(G)}{35\mu_2(G)^2 n} \right]^{\frac{1}{5}} \hat{\sigma} \tag{4}$$

where $R(G) = \int G(t)^2 \, dt, \mu_2(G) = \int t^2 G(t) \, dt$ and $\hat{\sigma}$ is the absolute deviation of the curvature data κ_i.

We have used curvature estimates at each vertex to generate probability distribution curves. With these curves, we now formulate an information theoretic approach based on entropy to define the CVM. We argue that the amount of information that the curvature conveys about the surface can be quantified as a measure using Shannon's framework[30]. His definition of entropy for communication systems as a measure of uncertainty is the minimum number of bits required to encode a message. We do not apply Shannon's definition of entropy in that context but use the logarithmic relationship to measure the predictability of curvature considered as a random variable. Furthermore, we also have to deal with resolution and normalized measure space. The resolution of the mesh (the number of vertices *n*) decides the quantization of curvature. More the number of vertices, better the approximation to the infinitesimal curvature. We counter the asymptotic exponential interaction between the resolution and curvature by normalizing the Shannon's entropy measure as shown below.

$$CVM = -\sum \hat{p}_i \log_n \hat{p}_i \tag{5}$$

The CVM of a 3D surface is the resolution normalized entropy of curvature that is indicative of the visual complexity associated with the surface. A high magnitude of the CVM specifies details of interest (feature availability) towards describing the object while small values of the CVM correspond to smooth flat surfaces. We use this definition of the CVM as a surface feature for object description.

The object description pipeline that we propose first segments an object into reasonably sized patches that the CVM measure characterizes. Our heuristic approach is to segment patches along crease discontinuities as these features should be excluded from the density estimates. Also, with man-made automotive components we note that the assumption about the existence of sharp edges can be justified. We use a simple region-growing method similar to Mangan and Whitaker[31] to segment the object along boundaries of high curvature that correspond to creases in the triangle mesh representation. During segmentation, we maintain patch adjacency information to create a graph where the nodes of the graph represent each segmented patch and the edges represent the connectedness of patches. The patches along with their orientation describe the global shape of the object while the CVM lends to describing the local variation within each of these patches. The graph description of the 3D object serves as the basis of object detection in a scene acquired from the robot. We demonstrate the pipeline pictorially in Figure *11-14*. Similar objects will have similar graph description. Hence, objects with similar surfaces can be easily identified. In addition, the graph description serves as common platform for identifying objects in the partial data from the robot to the *a priori* manufacturer's CAD model.

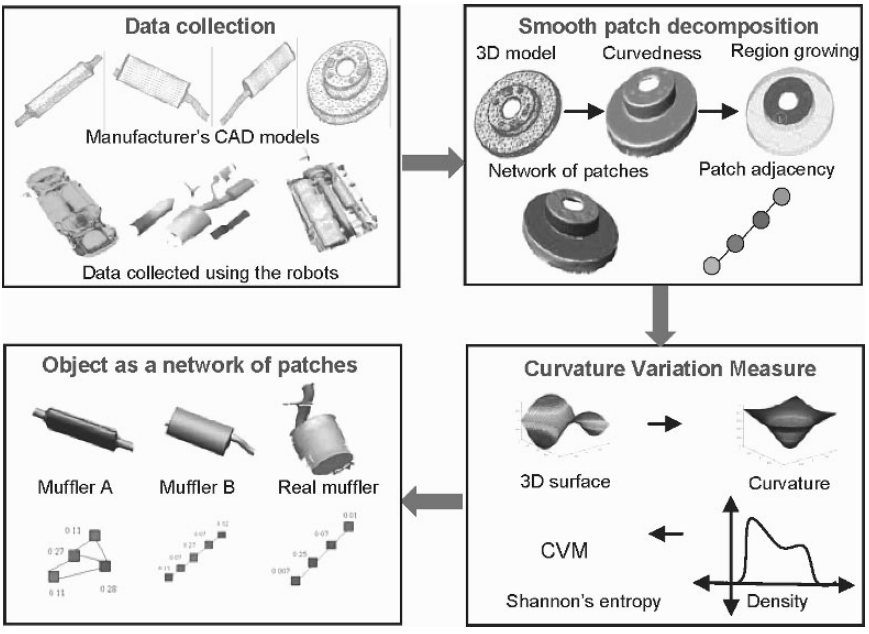


Figure 11-14. The object description pipeline using the CVM.
(See also Plate 44 in the Colour Plate Section)

We begin our experiments with 3D CAD models of automotive components such as the muffler, and the catalytic converter. In Figure *11-15*, we show the mesh model as an assembly of segmented patches and its graph definition with the CVM describing each of the segmented patches. The graph that is the assembly of smooth patches also indirectly encodes the information about the topology of the patches. We note the descriptiveness of the CVM with a negligible value on almost flat surfaces and substantial shape information in certain descriptive unique patches. We also note that the deformed cylindrical surfaces of the disc brake and the muffler and its connecting pipes have close CVM's indicating the common underlying shape. On the other hand, we also observe the difference in the main body of the Toyota muffler and the Dodge muffler. Though both of them appear cylindrical, we are able to distinguish between the deformed cylinder in the Dodge muffler and the Toyota muffler.

With these results, and the real data from the laser scanner, we now present results on the data from the IVP range scanner in Figure *11-16*. We see that the muffler from the sensed data has components that match closely with the Dodge muffler's CAD model. At this point we disclose that the segmentation of the muffler from the real scene is not fully automatic because of the incomplete nature of the 3D data. Our future research efforts will target automating the scene analysis.

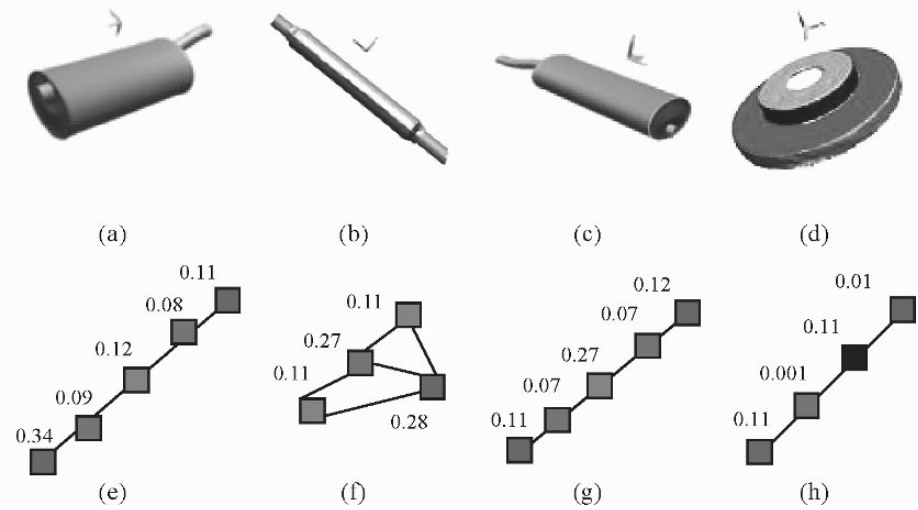


Figure 11-15. Describing CAD models using the CVM. (a) Segmented Toyota muffler model. (b) Segmented catalytic converter model. (c) Segmented Dodge muffler model. (d) Segmented disc brake model. (e) Graph description of the Toyota muffler. (f) Graph description of the catalytic converter. (g) Graph description of Dodge muffler. (h) Graph description of the disc brake. (See also Plate 45 in the Colour Plate Section)

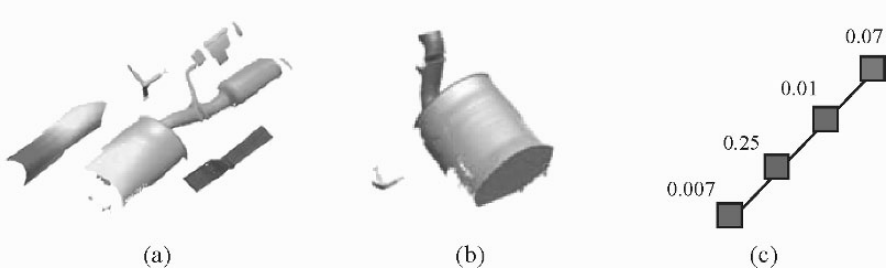


Figure 11-16. CVM graph results on an under vehicle scene. (a) 3D under vehicle scene. (b) Segmented muffler model from the sensed data. (c) Graph description of the muffler from the sensed data. (See also Plate 46 in the Colour Plate Section)

5. CONCLUSIONS AND FUTURE DIRECTIONS

With this chapter, we have demonstrated a robotic multi-sensor solution for the under vehicle inspection scenario to acquire and communicate data from a safe stand-off distance, rendering data in a visual and spatially understandable form for a remote inspector to make decisions. The multiple sensor approach that includes visual cameras, thermal imagers and radioactivity sensors with future potential for biological and chemical detectors in a robust modular architecture, enables visualization of scene characteristics that we call perception intelligence in virtual reality environments. In addition, with the 3D sensing enhancement on our SafeBot, we have shown how the flexibility for archival and automatic threat detection using specific algorithms as technology ready for deployment at places of military and civilian interest. Both the scene verification approach and the shape analysis using our novel definition of the curvature variation measure appear promising for automatic threat detection.

Furthermore, we have identified a few areas of improvement towards the next generation SafeBot. First, the real time data collection involves a 3D sensing robot that has three degrees of freedom in its motion. The data needs to be corrected for the motion based on the robot's trajectory. We have made assumptions about the trajectory being smooth to simplify the data processing and visualization of the data shown in this article. Though such assumptions can be made on fairly uniform trajectories of the robot, integration of inertial measurement units with the range sensing system can correct the data for vibrations and keep track of the robot's motion. We hope to include inertial sensors in our future systems to make our robots suitable in dynamic environments. Next, with the CVM, we have demonstrated the

ability to describe objects from the laser scanned scenes and are encouraged to test the recognition pipeline. Our model-based strategy benefits the creation of a symbolic graph representation from a mesh representation. Our method follows the part (region) relationship paradigm suggested by Shapiro and Stockman[32] in moving from a geometric object representation to a symbolic and intelligent one. To claim robustness with the object recognition towards a threat detection system, we plan to incorporate important scale information to support the scale-invariant CVM.

ACKNOWLEDGEMENTS

The authors would also like to thank Nikhil Naik, Santosh Katwal, Brad Grinstead and all other students and staff of the robotics team that have contributed towards the design, implementation and maintenance of the hardware and software sensor systems that are part of this research. The authors are also grateful to the financial support through the University Research Program in Robotics under grant DOE-DE-FG02-86NE37968, the DOD/RDECOM/NAC/ARC Program through grant R01-1344-18 and the US Army through grant Army-W56HC2V-04-C-0044.

REFERENCES

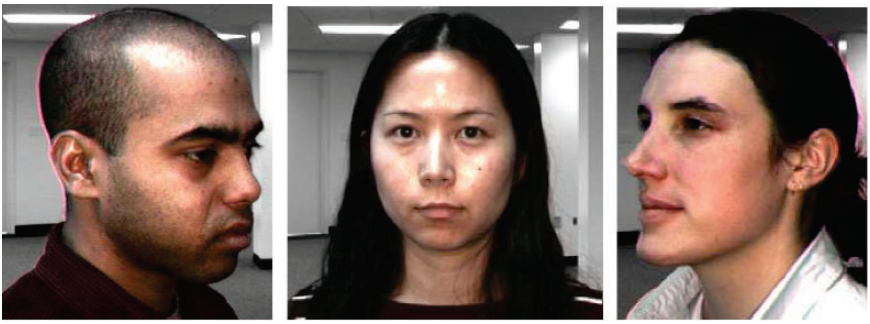
1. P. Dickson, J. Li, Z. Zhu, A. Hanson, , E. Riseman, H. Sabrin, H. Schultz and G. Whitten, “Mosaic Generation for Under-Vehicle Inspection,” *IEEE Workshop on Applications of Computer Vision*, 251-256 (2002).
2. L.A. Freiburger, W. Smuda, R.E. Karlsen, S. Lakshmanan and B. Ma, “ODIS the under-vehicle inspection robot: development status update”, *Proc. of SPIE Unmanned Ground Vehicle Technology V*, Vol. 5083, 322-335 (2003).
3. Autonomous Solutions Inc., “Spector: Under vehicle inspection system”, Product Brochure (2005).
4. B. Smuda, E. Schoenherr, H. Andrusz, and G.Gerhart, “Deploying the ODIS robot in Iraq and Afghanistan”, in the Proceedings of SPIE Unmanned Ground Vehicle Technology VII, Vol. 5804, 119-129, 2005.
5. A. Koschan, D. Page, J.-C. Ng, M. Abidi, D. Gorsich, and G. Gerhart, “SAFER Under Vehicle Inspection Through Video Mosaic Building,” *International Journal of Industrial Robot*, Vol. 31, 435-442 (2004)
6. J. S. Albus, “A reference model architecture for intelligent systems design”, in *An Introduction to Intelligent Autonomous Control*, Kluwer Publishers, 27-56 (1993).
7. WMD Response Guide Book, *U. S Department of Justice and Louisiana State University*, Academy of Counter-Terrorism Education.
8. C. Qian, D. Page, A. Koschan, and M. Abidi, “A 'Brick'-Architecture-Based Mobile Under-Vehicle Inspection System,” *Proc. of the SPIE Unmanned Ground Vehicle Technology VII*, Vol. 5804, 182-190 (2005).

9. F. Blais, "Review of 20 years of Range Sensor Development," *Journal of Electronic Imaging*, Vol. 13(1), 231-240 (2004).
10. H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle, "Surface Reconstruction from Unorganized Points," *ACM SIGGRAPH Computer Graphics*, Vol. 26(2), 71-78 (1992).
11. D. Page, A. Koschan, Y. Sun, and M. Abidi, "Laser-based Imaging for Reverse Engineering," *Sensor Review, Special issue on Machine Vision and Laser Scanners*, Vol. 23(3),223-229 (2003).
12. P.J. Besl and N.D McKay, "A Method for Registration of 3D Shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 14(2), 239-56 (1992).
13. G. Turk and M. Levoy, "Zippered Polygon Meshes from Range Images", *Proc. of ACM SIGGRAPH*, 311-318 (1994).
14. A. T. Hayes, A. Martinoli, and R. M. Goodman, "Distributed odor source localization", IEEE Sensors, Vol. 2(3), 260-271 (2002).
15. C. Dorai and A. K. Jain, "COSMOS – A Representation Scheme for 3D Free-Form Objects," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 19, 1115-1130 (1997).
16. F. Arman and J. K. Aggarwal, "Model-based object recognition in dense range images," *ACM Computing Surveys*, Vol. 25(1), 5-43, (1993).
17. D. V. Vranic, "An Improvement of Rotation Invariant 3D Shape Descriptor Based on Functions on Concentric Spheres," *Proc. of the IEEE International Conference on Image Processing*, Vol. 3, 757-760 (2003).
18. G. Hetzel, B. Leibe , P. Levi, and B. Schiele, "3D Object Recognition from Range Images using Local Feature Histograms," *Proc. of IEEE International Conference on Computer Vision and Pattern Recognition*, Vol. 2, 394-399 (2001).
19. R. Ohbuchi, T. Minamitani, and T. Takei, "Shape Similarity Search of 3D Models by using Enhanced Shape Functions", *Proc. of Theory and Practice in Computer Graphics*, 97-104 (2003).
20. M. Körtgen, G. J. Park, M. Novotni, and R. Klein, "3D Shape Matching with 3D Shape Contexts," *Proc. of the 7th Central European Seminar on Computer Graphics*, (2003).
21. A. Khotanzad and Y. H. Hong, "Invariant image recognition by Zernike moments," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 12(5), 489-497 (1990).
22. A. Johnson and M. Hebert, "Using spin images for efficient object recognition in cluttered 3D scenes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 21(5), 433-444 (1999).
23. D. Zhang and M. Hebert, "Harmonic Maps and Their Applications in Surface Matching," *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, Vol. 2, 524-530 (1999).
24. P. Besl, "Triangles as a primary representation: Object recognition in computer vision," *Lecture Notes in Computer Science*, Vol. 994, 191-206 (1995).
25. R. Osada, T. Funkhouser, B. Chazelle, and D. Dobkin, "Shape Distributions," *ACM Transactions on Graphics*, Vol. 21(4), 807-832 (2002).
26. T. Surazhsky, E. Magid, O. Soldea, G. Elber, and E. Rivlin, "A comparison of gaussian and mean curvature estimation methods on triangle meshes," *Proc. of International Conference on Robotics and Automation*, Vol. 1, 1021-1026 (2003).
27. C. Lin and M. J. Perry, "Shape description using surface triangulation," *Proc. of the IEEE Workshop on Computer Vision: Representation and Control*, 38-43 (1982).
28. B. W. Silverman, *Density Estimation for Statistics and Analysis*, Chapman and Hall, London, UK (1986).

29. M. P. Wand and M. C. Jones, *Kernel Smoothing*, Chapman and Hall, London (1995).
30. C. E. Shannon, "A mathematical theory of communication," *The Bell System Technical Journal*, Vol. 27, 379-423 (1948).
31. A. P. Mangan and R. T. Whitaker, "Partitioning 3D surface meshes using watershed segmentation," *IEEE Transactions on Visual Computer Graphics*, Vol. 5(4), 308-321 (1999).
32. L. G. Shapiro and G. C. Stockman, *Computer Vision*, Prentice Hall, Upper Saddle River, New Jersey (2001).

Colour Plate Section

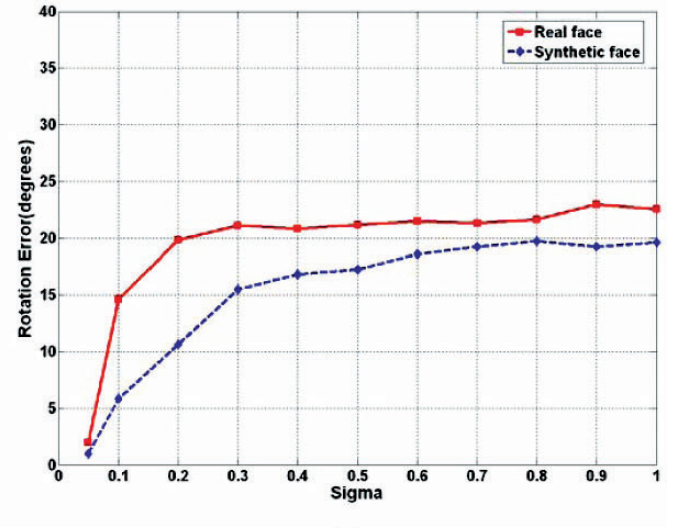
Plate 1. See also Figure 3-2 on page 79



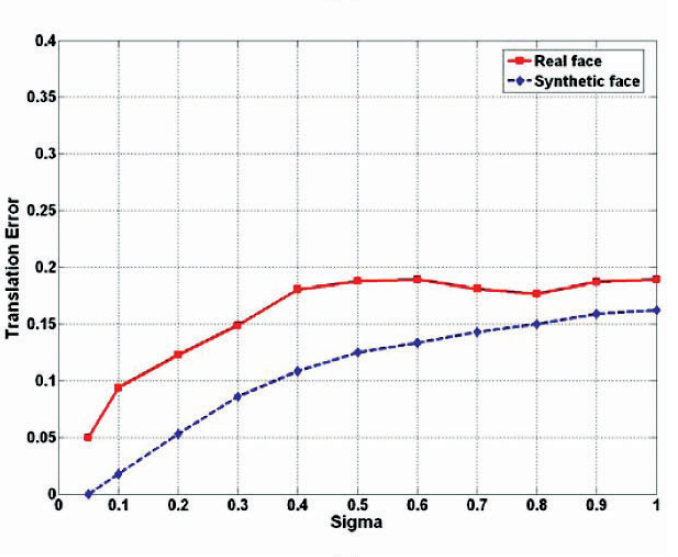
(a)

(b)

Plate 2. See also Figure 3-3 on page 79

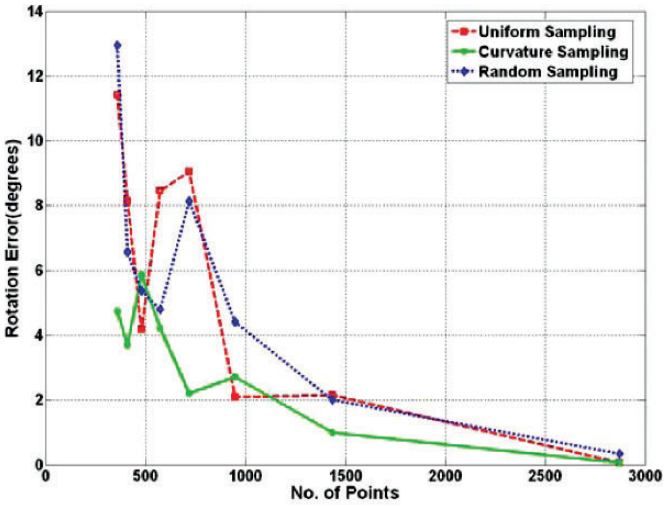


(a)

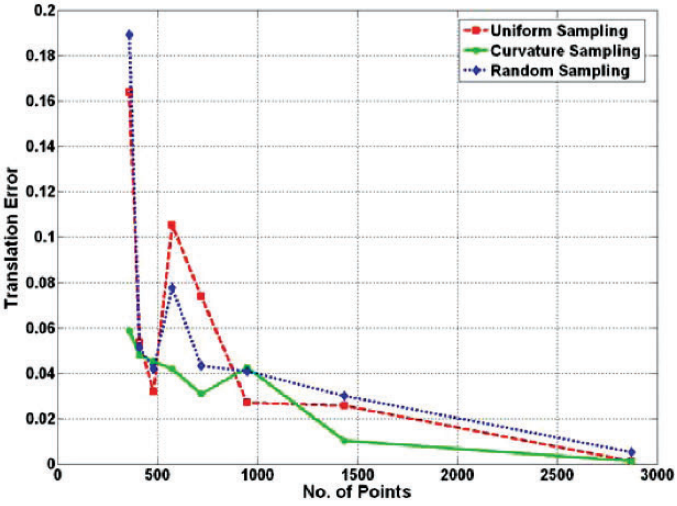


(b)

Plate 3. See also Figure 3-4 on page 81



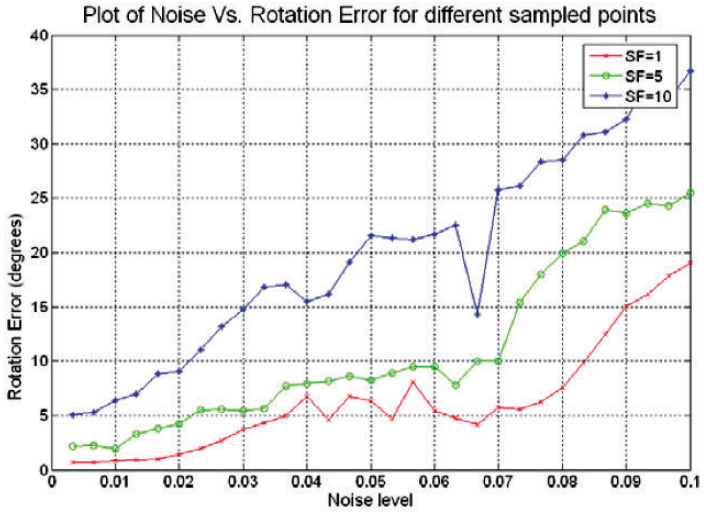
(a)



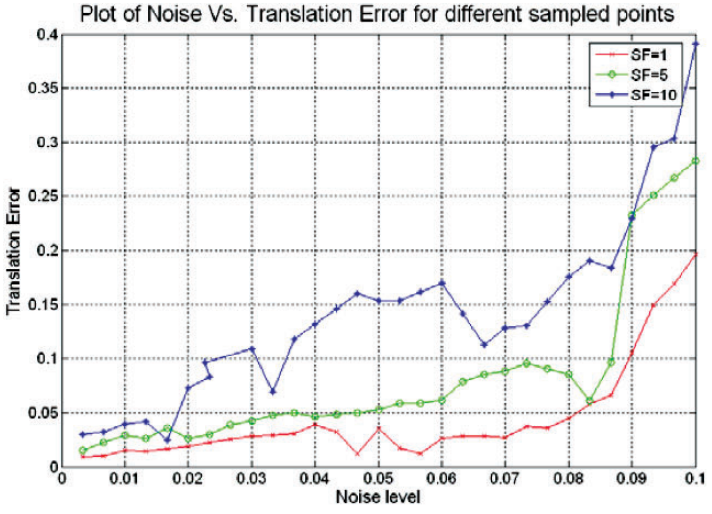
(b)

Plate 4. See also Figure 3-5 on page 83

.

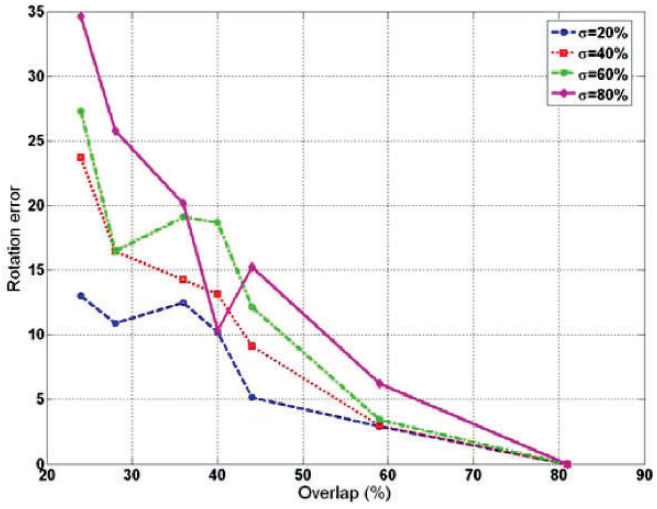


(a)

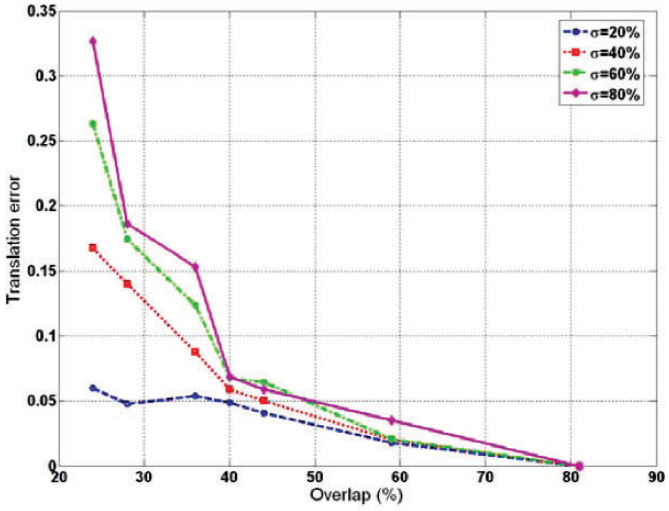


(b)

Plate 5. See also Figure 3-7 on page 85

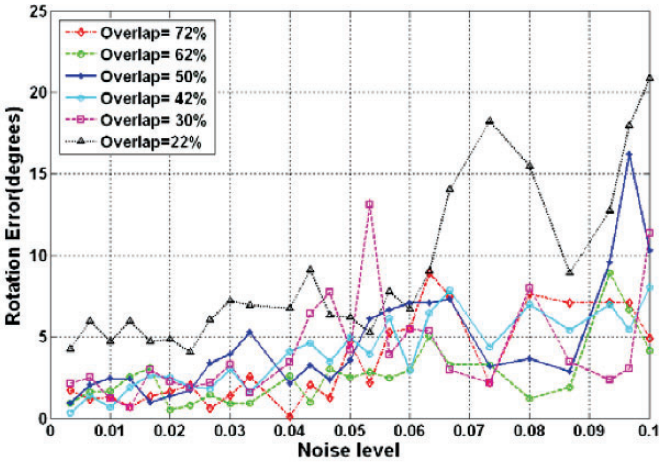


(a)

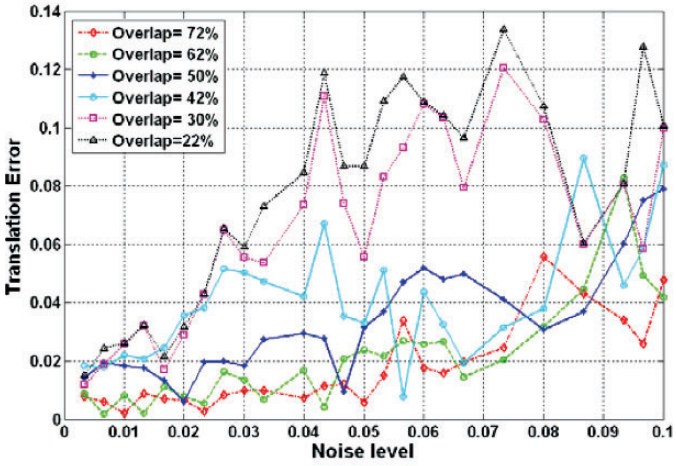


(b)

Plate 6. See also Figure 3-8 on page 87

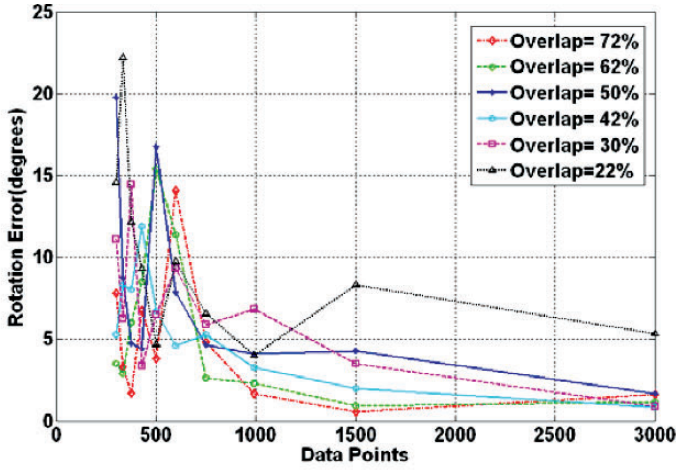


(a)

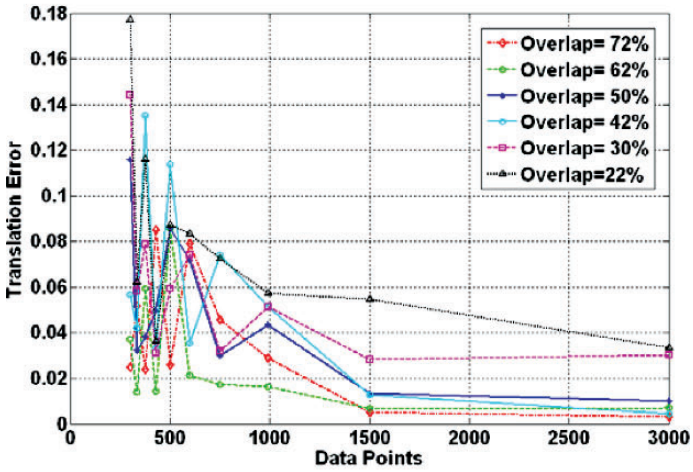


(b)

Plate 7. See also Figure 3-9 on page 88

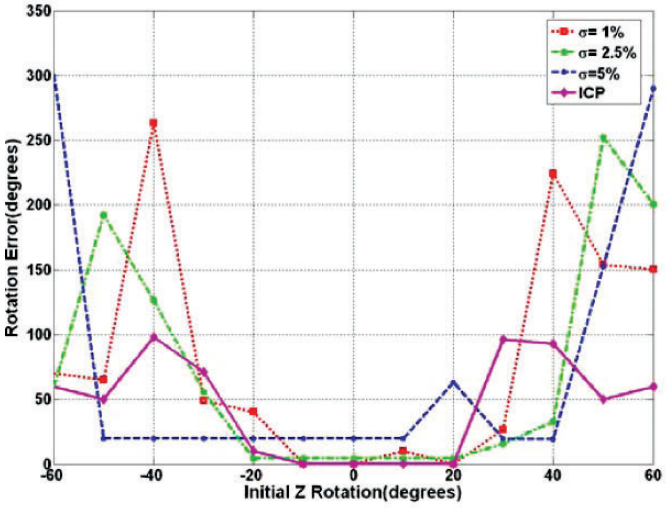


(a)

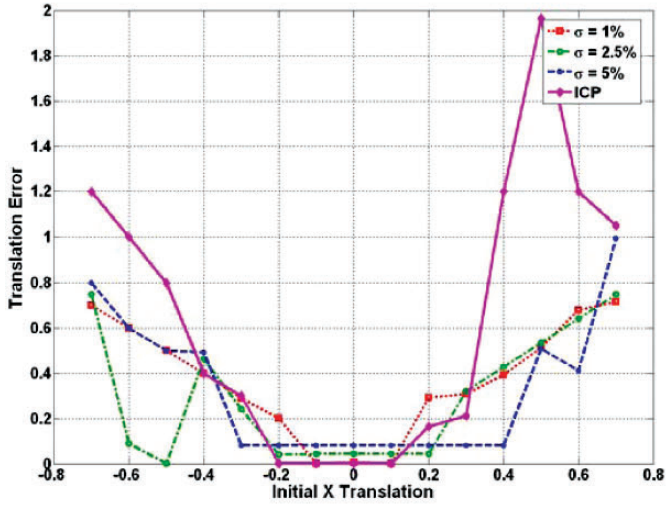


(b)

Plate 8. See also Figure 3-10 on page 89



(a)



(b)

Plate 9. See also Figure 3-11 on page 90

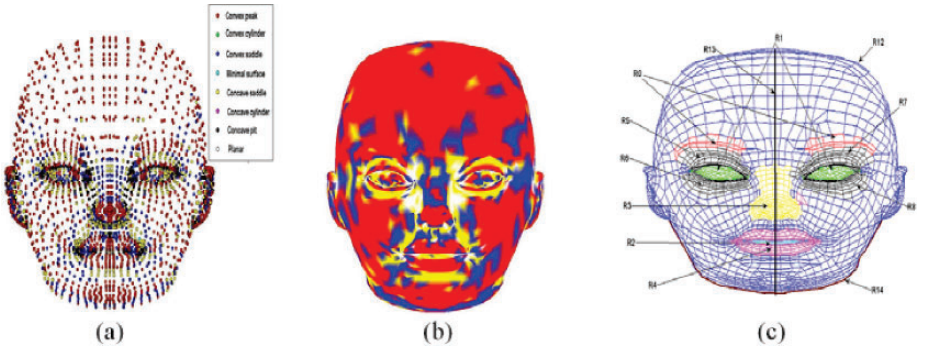


Plate 10. See also Figure 4-5 on page 104

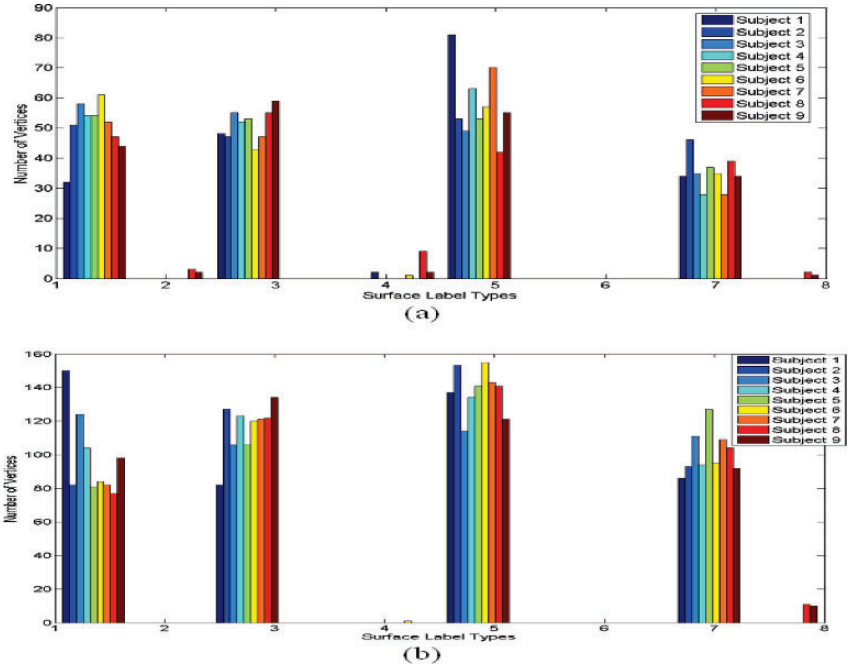


Plate 11. See also Figure 4-6 on page 105

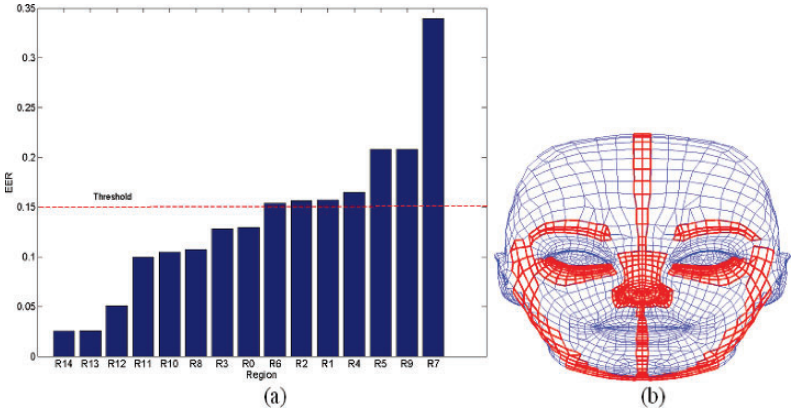


Plate 12. See also Figure 4-8 on page 108

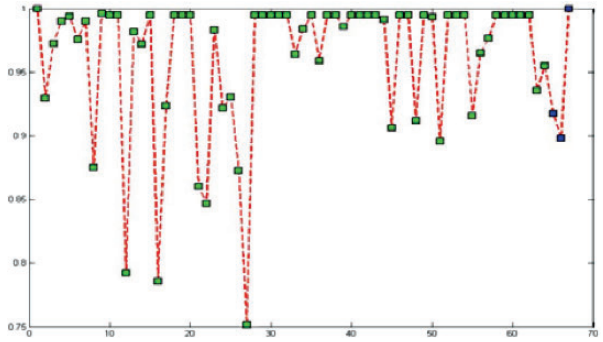


Plate 13. See also Figure 4-9 on page 109

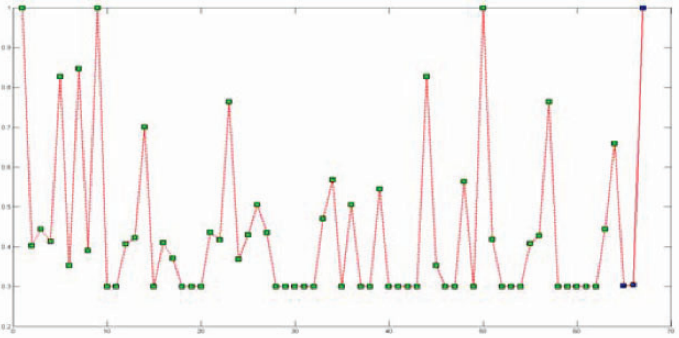


Plate 14. See also Figure 4-10 on page 111

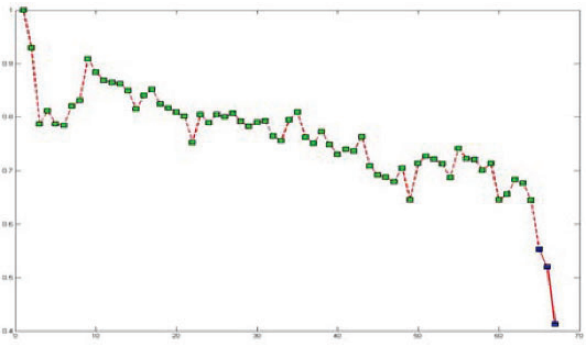


Plate 15. See also Figure 4-11 on page 111

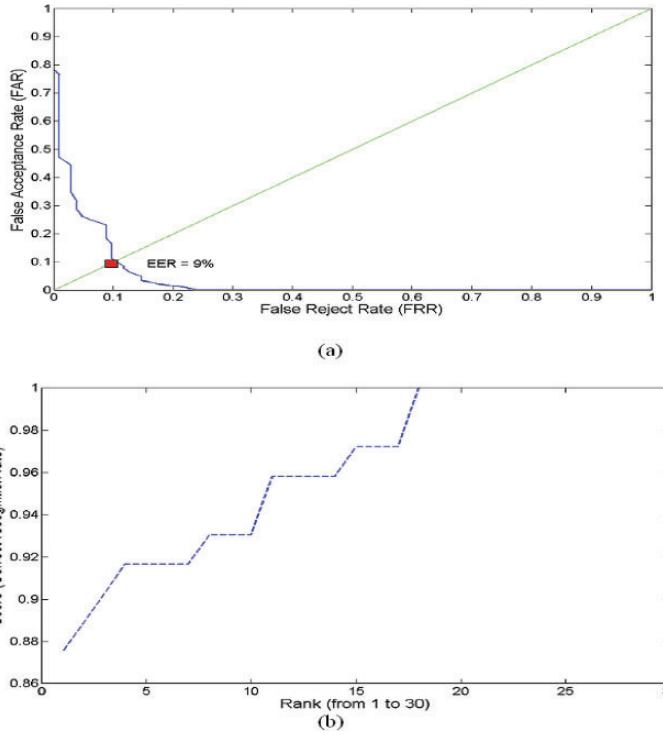


Plate 16. See also Figure 4-12 on page 113

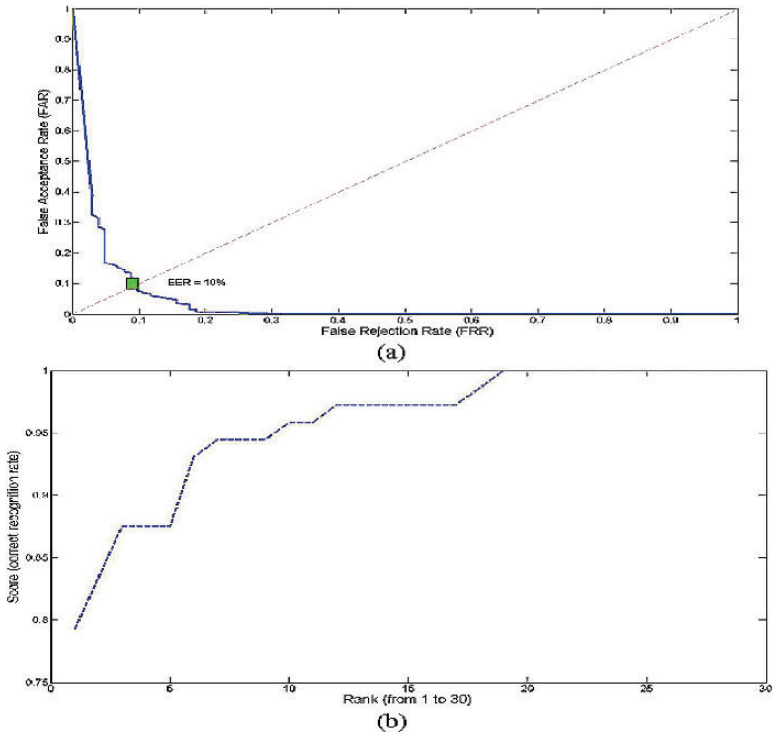


Plate 17. See also Figure 4-13 on page 114

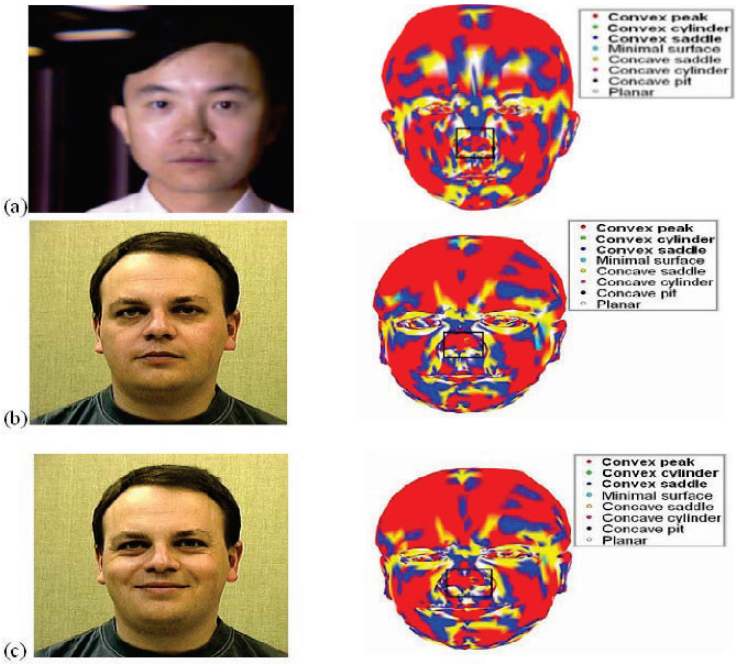


Plate 18. See also Figure 4-14 on page 115

Plate 19. See also Figure 4-15 on page 115

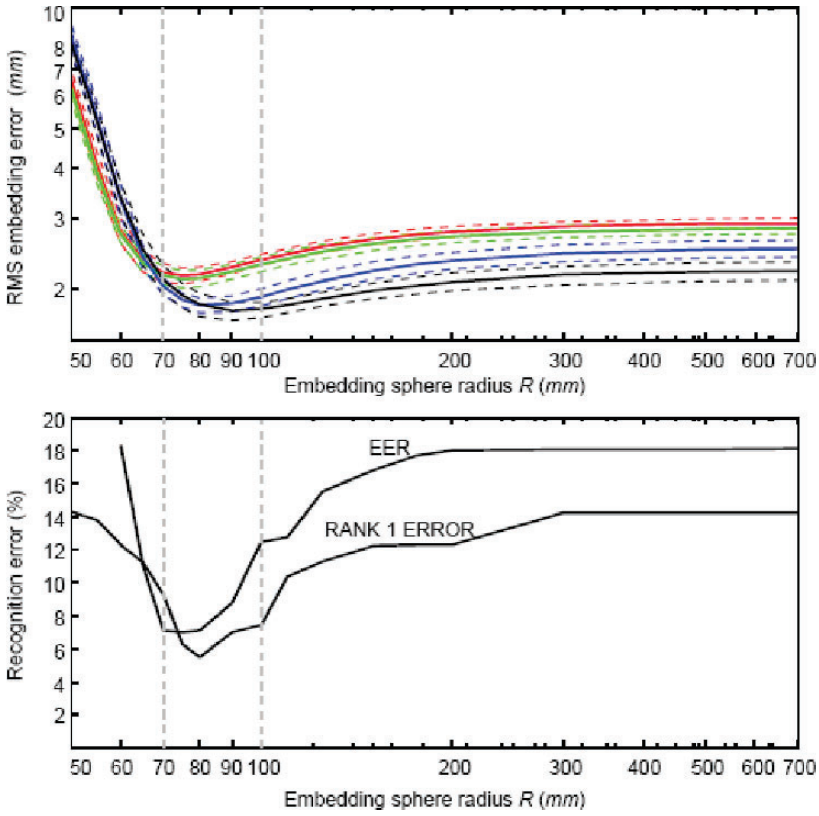


Plate 20. See also Figure 5-4 on page 125

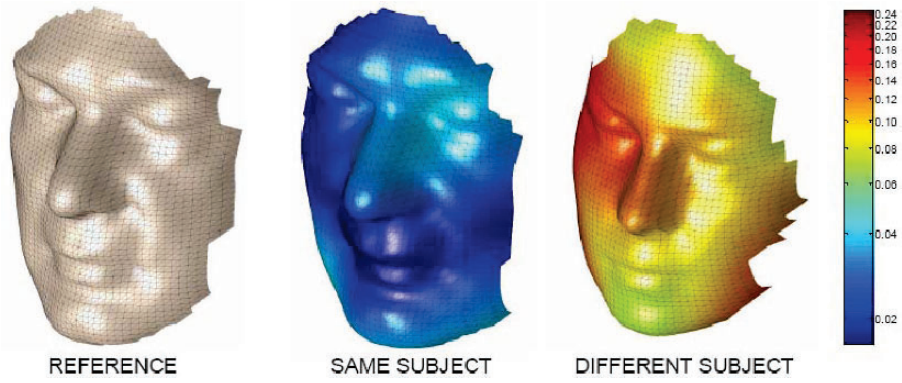


Plate 21. See also Figure 5-5 on page 127

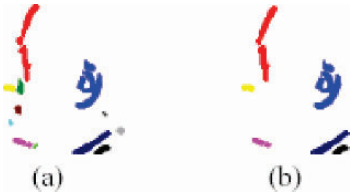


Plate 22. See also Figure 6-5 on page 141

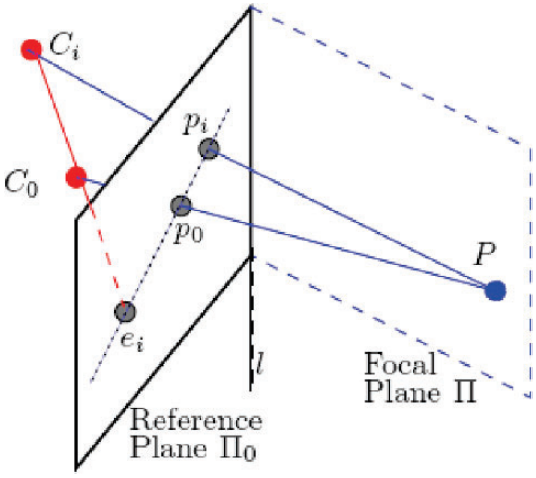


Plate 23. See also Figure 7-2 on page 162

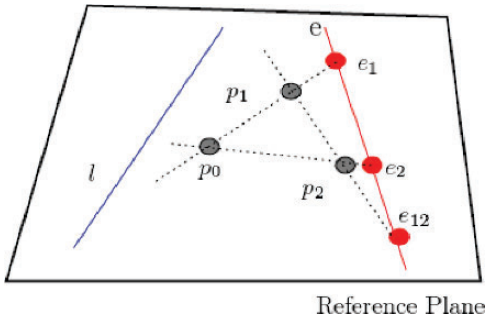


Plate 24. See also Figure 7-7 on page 171

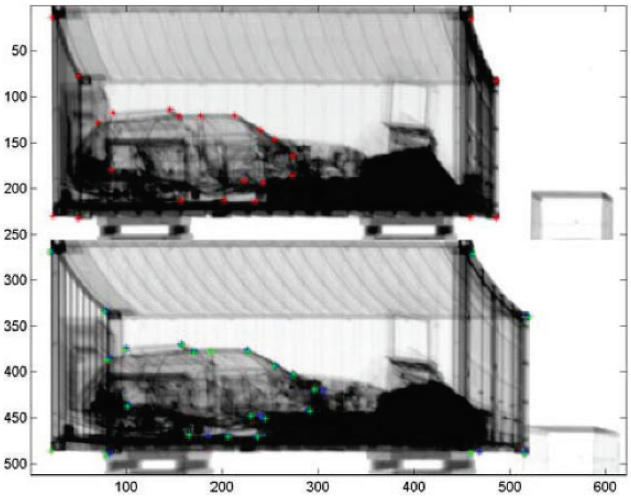


Plate 25. See also Figure 8-4 on page 182

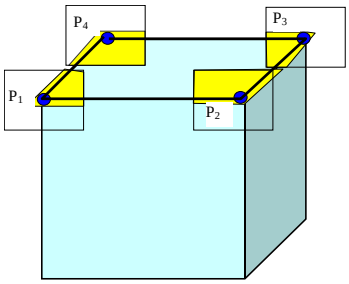


Plate 26. See also Figure 8-8 on page 192

Plate 27. See also Figure 8-11 on page 195

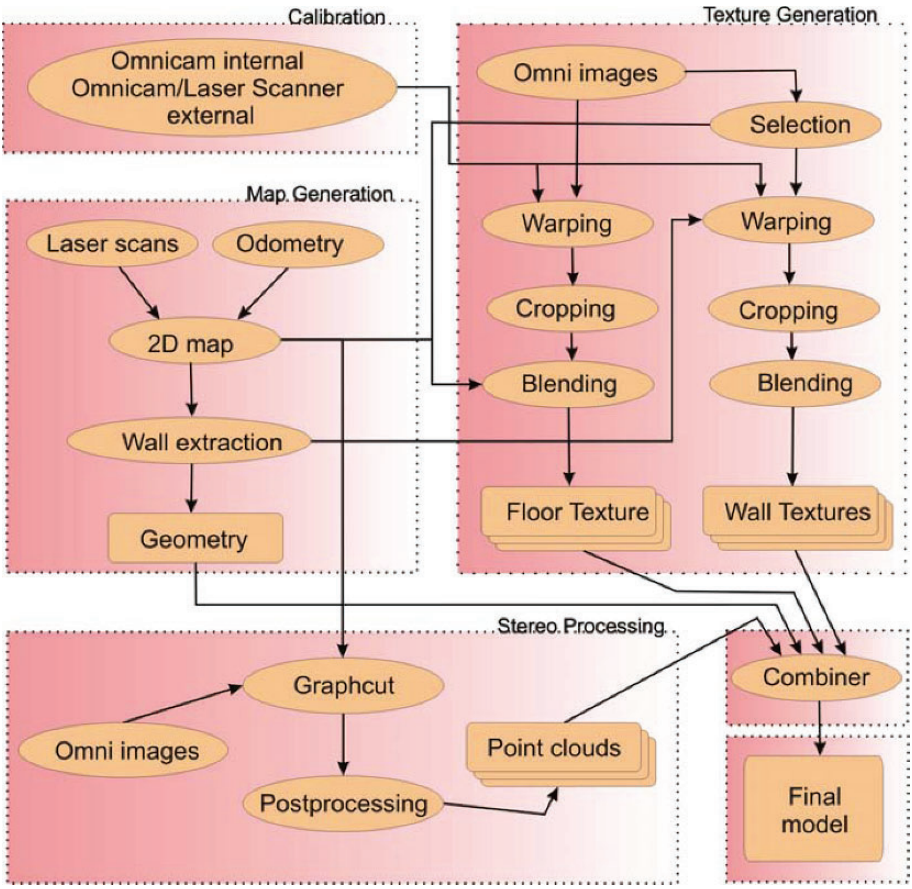
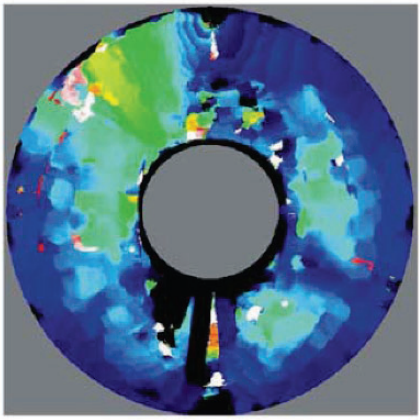
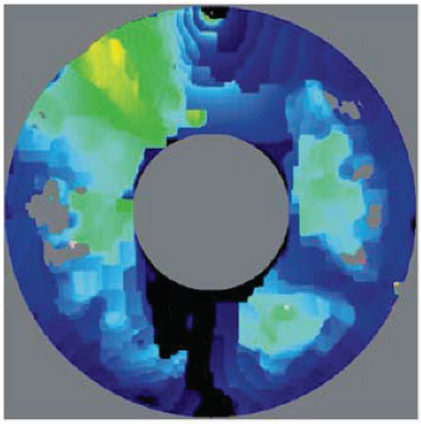


Plate 28. See also Figure 9-2 on page 206

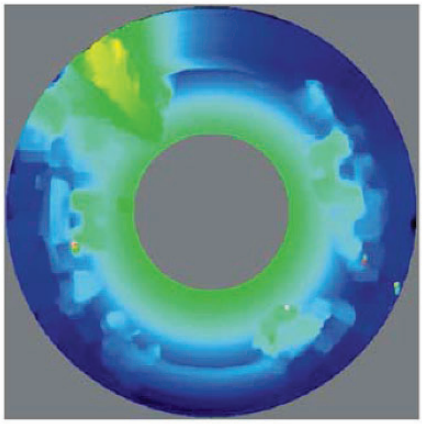
(a) One example source image.



(b) Winner takes all solution of stereo matching.



(c) Result of graph cut algorithm.



(d) Final disparity map after post-processing the graph cut results.

Plate 29. See also Figure 9-10 on page 219

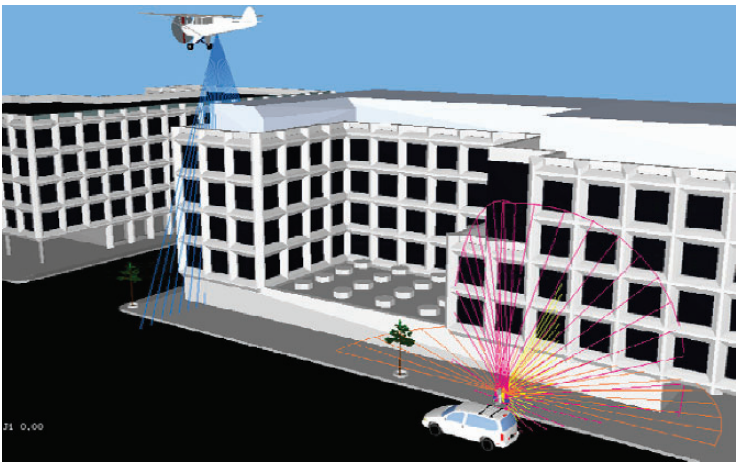


Plate 30. See also Figure 10-3 on page 232

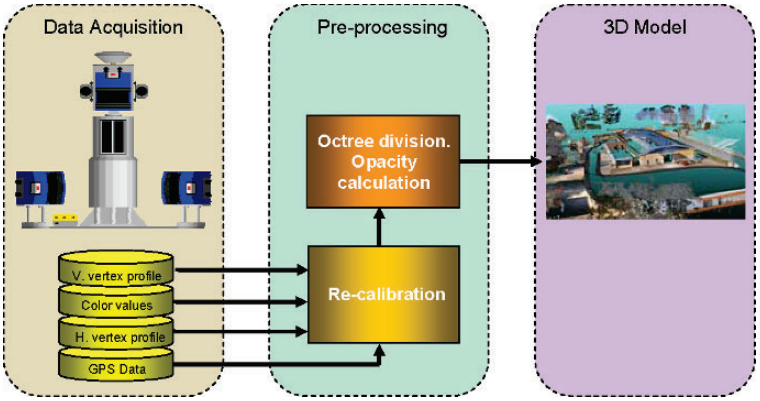


Plate 31. See also Figure 10-5 on page 233

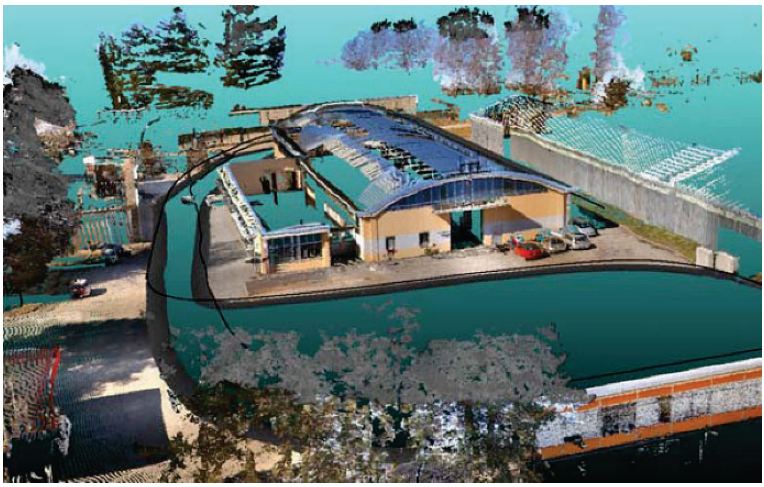


Plate 32. See also Figure 10-6 on page 233

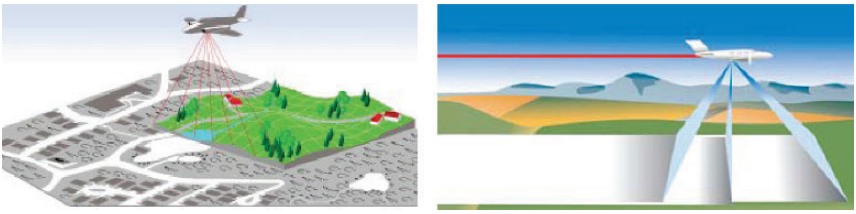


Plate 33. See also Figure 10-7 on page 234

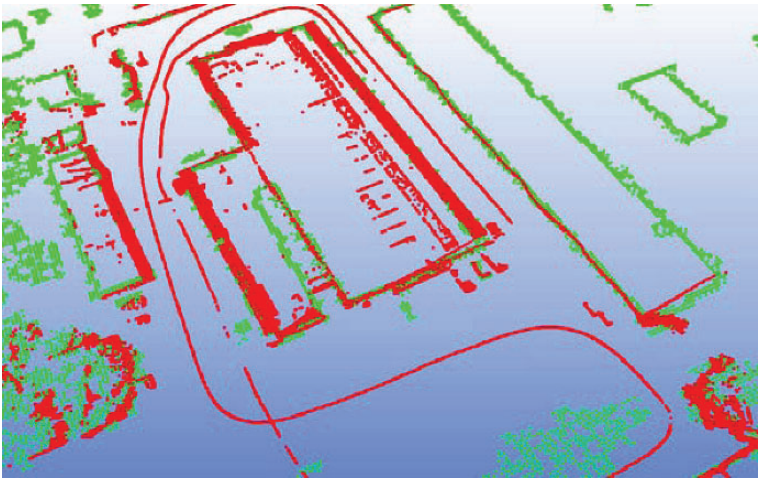


Plate 34. See also Figure 10-9 on page 237

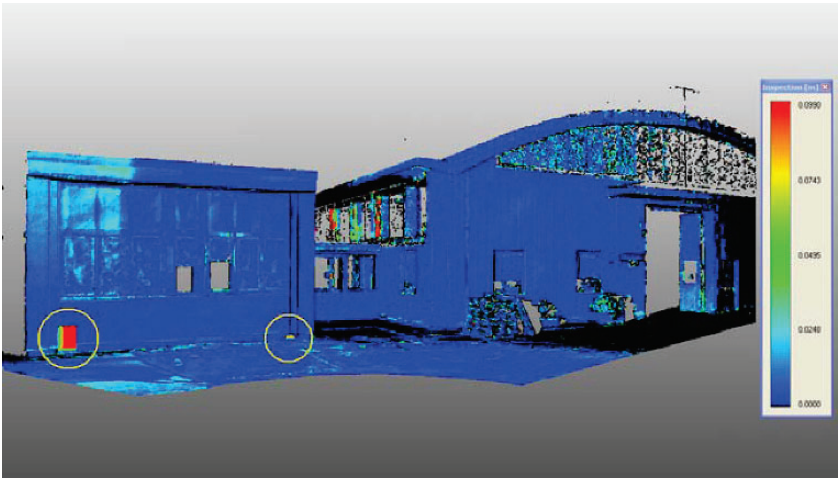


Plate 35. See also Figure 10-15 on page 243

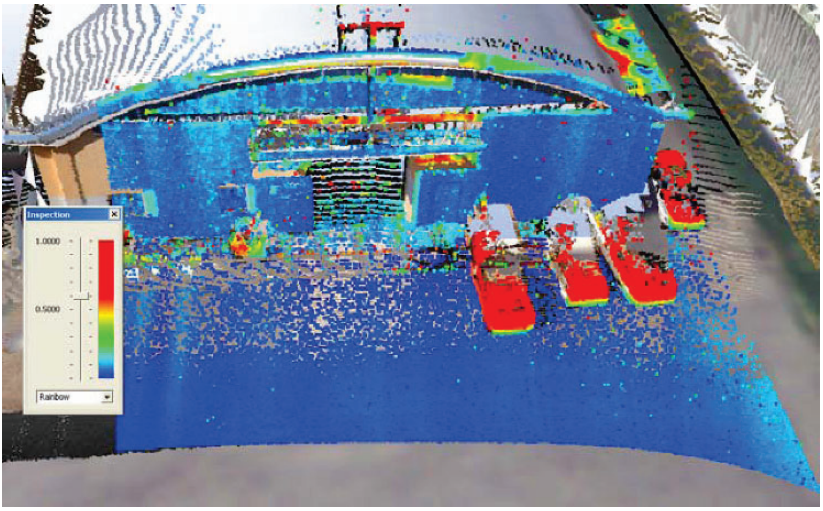


Plate 36. See also Figure 10-19 on page 245

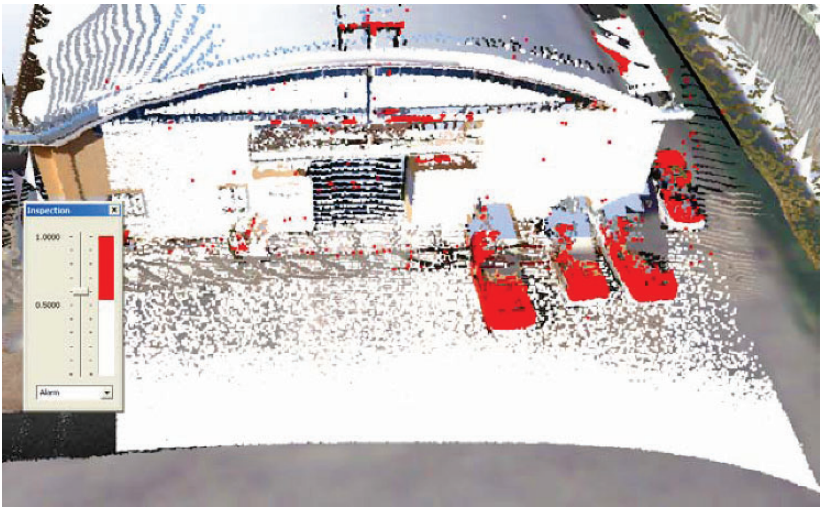


Plate 37. See also Figure 10-20 on page 245

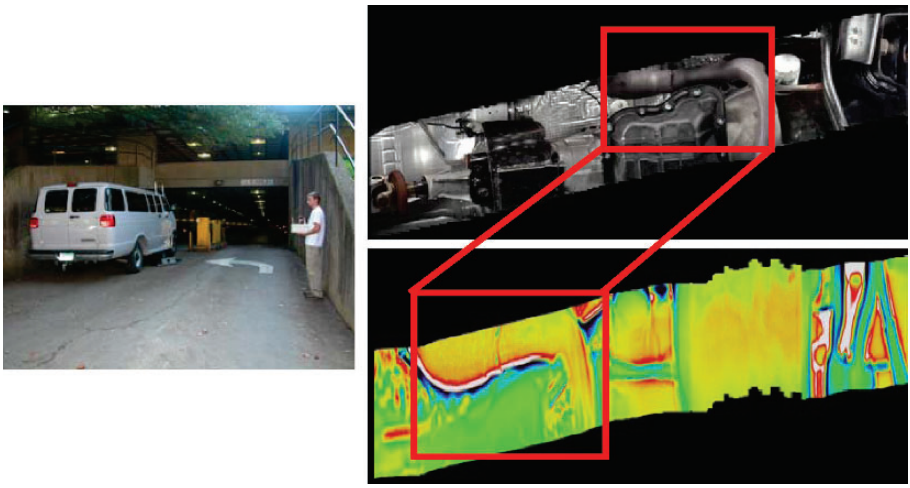


Plate 38. See also Figure 11-3 on page 252

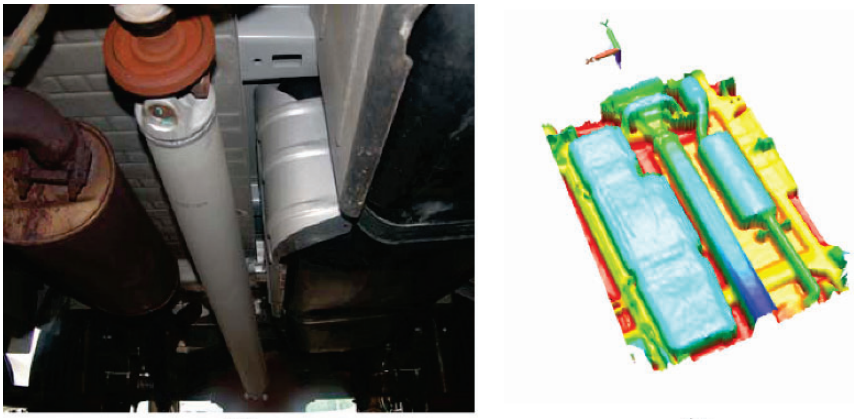


Plate 39. See also Figure 11-7 on page 262

Plate 40. See also Figure 11-8 on page 262

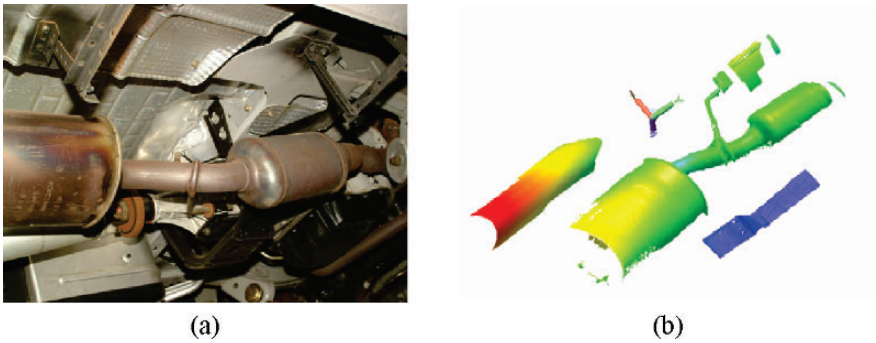


Plate 41. See also Figure 11-9 on page 263

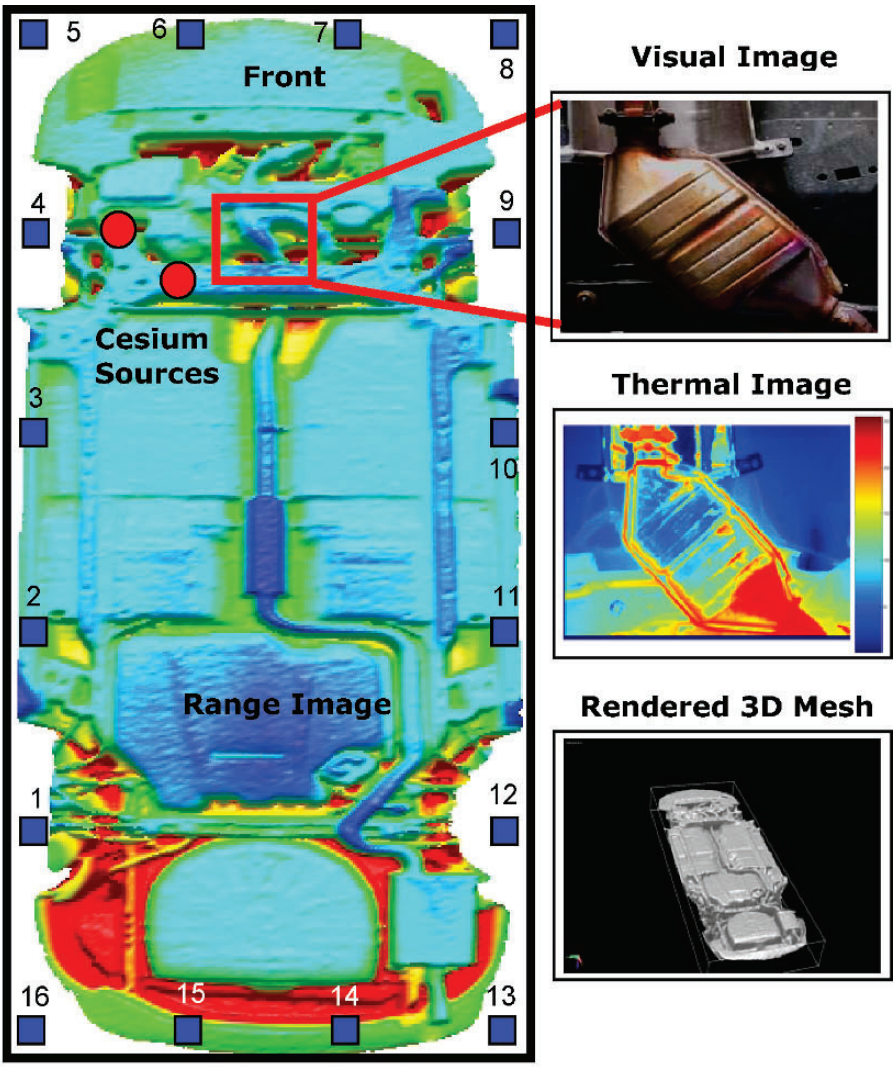


Plate 42. See also Figure 11-10 on page 265

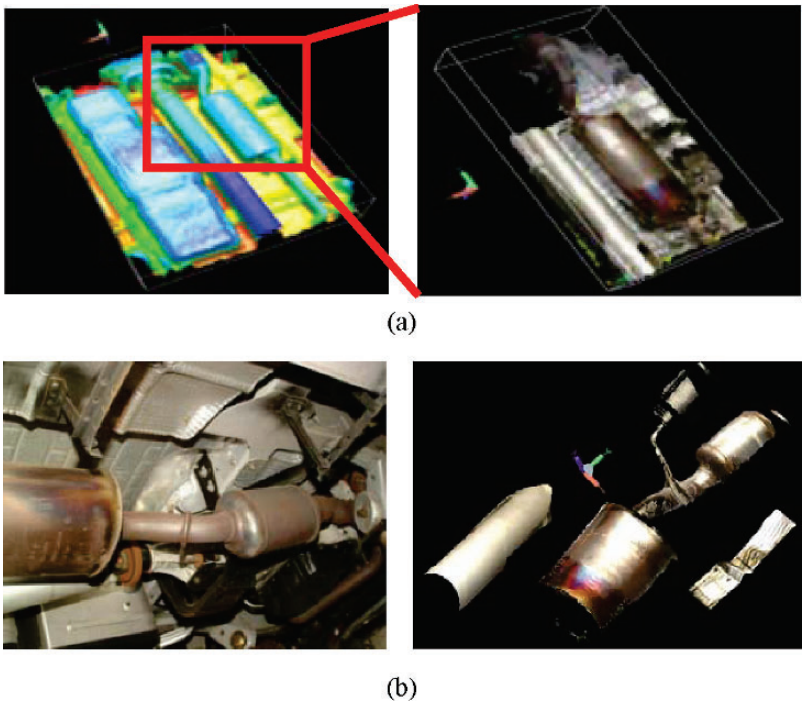


Plate 43. See also Figure 11-11 on page 266

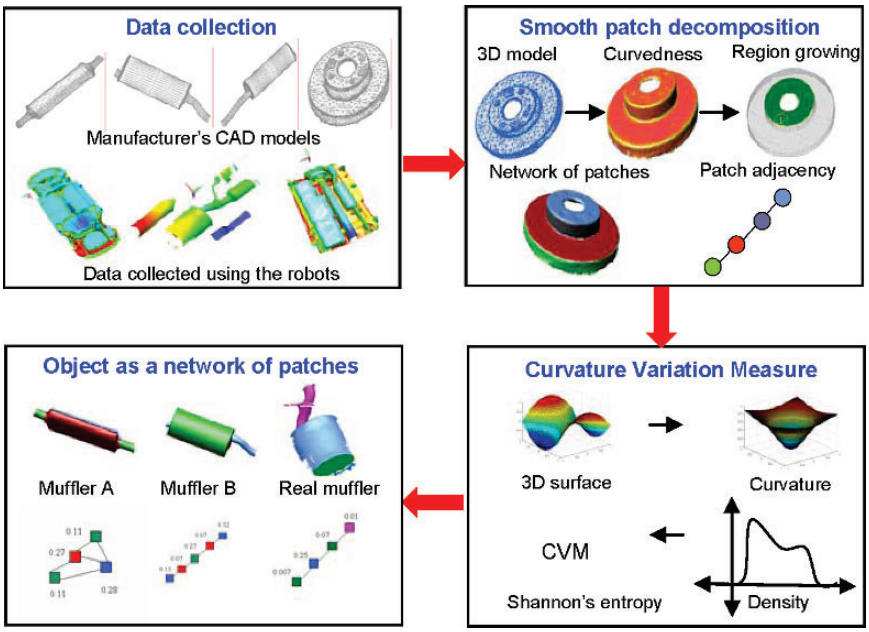


Plate 44. See also Figure 11-14 on page 273

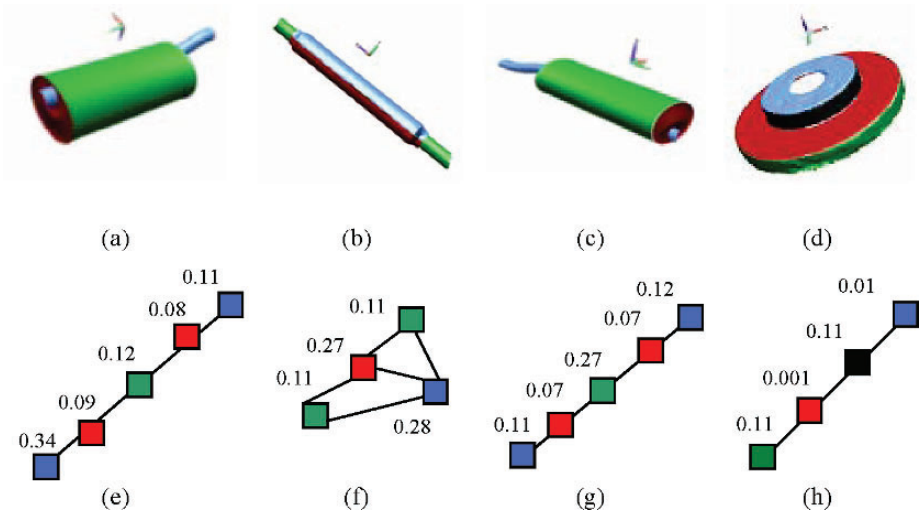


Plate 45. See also Figure 11-15 on page 274

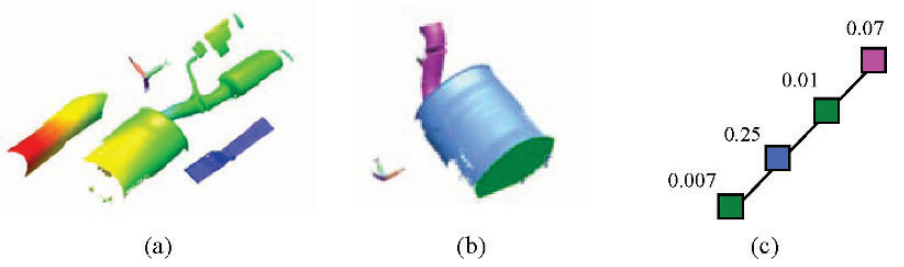


Plate 46. See also Figure 11-16 on page 275

Index

3D face models, 3
3D face recognition, 3, 95, 119
3D face reconstruction, 25, 69
3D face registration, 3
3D modeling, 202
3D reconstruction, 173, 225
3D registration, 69
3D sensors, 3
3D surface feature, 249

Biometrics, 119

Change analysis, 225

Data fusion, 225

Ear biometrics, 133
Ear detection, 133

Face recognition, 25, 119
Facial expression, 119
Feature selection, 95

Gaussian fields, 69
Generic model, 95
Genetic algorithm, 95
Geometric modeling, 95
Graph cut stereo, 202

Isometry, 119
Intrinsic geometry, 119

Laser range scanner, 25, 249
Light fields, 159

Morphing, 25
Motion analysis, 173
Moving target extraction, 173
Multidimensional scaling, 119

Pushbroom stereo, 173
Projective geometry, 159

Range images, 133
Real-time system, 159
Robotic security guard, 202

Shape from motion, 25
Shape from shading, 25
Shape index, 133
Shape model, 133
SLAM, 202
Stereo vision, 25, 173
Structured light, 25
Surface shape description, 249
Synthetic aperture, 159

Under vehicle inspection, 249

Video mosaicing, 173