Lecture Notes in Computer Science 2539
Edited by G. Goos, J. Hartmanis, and J. van Leeuwen

Springer
Berlin
Heidelberg
New York
Barcelona
Hong Kong
London
Milan
Paris
Tokyo

Katy Börner   Chaomei Chen (Eds.)

Visual Interfaces to Digital Libraries

Springer

Series Editors

Gerhard Goos, Karlsruhe University, Germany
Juris Hartmanis, Cornell University, NY, USA
Jan van Leeuwen, Utrecht University, The Netherlands

Volume Editors

Katy Börner
Indiana University, School of Library and Information Science
10th Street and Jordan Avenue, Main Library 019
Bloomington, IN 47405, USA
E-mail: katy@indiana.edu

Chaomei Chen
Drexel University, College of Information Science and Technology
3141 Chestnut Street, Philadelphia, PA 19104-2875, USA
E-mail: Chaomei.Chen@cis.drexel.edu

Cataloging-in-Publication Data applied for

A catalog record for this book is available from the Library of Congress.

Bibliographic information published by Die Deutsche Bibliothek
Die Deutsche Bibliothek lists this publication in the Deutsche Nationalbibliografie; detailed bibliographic data is available in the Internet at <http://dnb.ddb.de>.

CR Subject Classification (1998): H.3.7, H.2, H.3, H.4, H.5, I.7

ISSN 0302-9743
ISBN 3-540-00247-2 Springer-Verlag Berlin Heidelberg New York

Springer-Verlag Berlin Heidelberg New York
a member of BertelsmannSpringer Science+Business Media GmbH

http://www.springer.de

Typesetting: Camera-ready by author, data conversion by Steingräber Satztechnik GmbH, Heidelberg
Printed on acid-free paper SPIN: 10871584 06/3142 5 4 3 2 1 0

Preface

This edited book contains a selected set of extended papers that were presented at the first and second international workshops on "Visual Interfaces to Digital Libraries," held at the Joint Conference on Digital Libraries (JCDL) in 2001 and 2002.

Visual interfaces to digital libraries exploit the power of human vision and spatial cognition to help individuals mentally organize and electronically access and manage large, complex information spaces. They draw on progress in the new field of Information Visualization and seek to shift the user's mental load from slow reading to faster perceptual processes such as visual pattern recognition.

The workshop series aimed to raise awareness of several interconnected fields of research related to the design and use of visual interfaces to digital libraries, especially information visualization, human-computer interaction, and cognitive psychology. It also sought to encourage participants to reflect on the state of the art in their own fields by identifying challenging issues regarding visual interfaces, and thereby foster a multidisciplinary research agenda for future research and development.

While the first workshop (http://vw.indiana.edu/visual01) meant to broadly identify the past, present, and future of visual interfaces to digital libraries, the second workshop (http://vw.indiana.edu/visual02/) invited papers on specific topics such as web-based visual interfaces, mobile access to information, collaborative document spaces, usability and formal methods, as well as real-world needs, products, and applications.

The workshops brought together researchers, practitioners, software developers/vendors, and graduate students in the areas of information visualization, digital libraries, human-computer interaction, library and information science, and computer science from around the world, and sparked exciting discussions and collaborations.

This book provides comprehensive coverage of the topics presented and discussed at the workshops. We hope it will familiarize a wider audience with current research on visual interfaces to digital libraries as well as inspire new explorations.

As workshop organizers, we would like to recognize the program committee for their expert input as well as very detailed and insightful reviews that helped to improve the quality of papers enormously. We also thank Steven Eick (Visual Insights) and Tim Bray (Antarti.ca Systems), both of whom gave stimulating and thought-provoking invited talks from an industry perspective in 2001 and 2002, respectively.

Special thanks go to the general workshop and local JCDL organizers Marianne Afifi, Richard Furuta, and Lynetta Sacherek for their support and cooperation. We also thank the team at Springer-Verlag for their assistance in the production of this book. Last, but not least, we would like to acknowledge the financial support the workshop received from VRCO systems, ACM, and IEEE.

The electronic version of this book containing color figures can be found at http://link/service/0558/tocs/t2539.htm.

September 2002

Katy Börner
Chaomei Chen

Program Committee

Katy Börner (Chair)	Indiana University, USA	
Chaomei Chen (Chair)	Drexel University, USA	
Ann Blandford	UCL Interaction Centre, University College London, UK	
Kevin W. Boyack	Sandia National Laboratories, USA	
Martin Dodge	University College London, UK	
Xia Lin	Drexel University, USA	
John MacColl	University of Edinburgh, UK	
Sougata Mukherjea	Verity, USA	
Sue O'Hare	Consignia, UK	
Scott Robertson	Drexel University, USA	(2002)
André Skupin	University of New Orleans, USA	(2002)
Henry Small	Institute for Scientific Information, USA	(2001)

Table of Contents

Part III: Visualization of Knowledge Domains

Part IV: Cartographic Interfaces to Digital Libraries

Part V: Towards a General Framework

Visual Interfaces to Digital Libraries: Motivation, Utilization, and Socio-technical Challenges

Katy Börner[1] and Chaomei Chen[2]

[1] School of Library and Information Science, Indiana University, Bloomington, IN, 47405, USA,
katy@indiana.edu
[2] College of Information Science and Technology, Drexel University
Philadelphia, PA 19104-2875, USA
chaomei.chen@cis.drexel.edu

Abstract. The accelerating rate of scientific and technical discovery, typified by the ever-shortening time period for the doubling of information – currently estimated at 18 months [1] – causes new topics to emerge at increasing speed. Libraries have a hard time just cataloguing the large amount of produced documents. Scientists and practitioners who must read and process relevant documents are in need of new tools that can help them to identify and manage this flood of information. Visual Interfaces to digital libraries apply powerful data analysis and information visualization techniques to generate visualizations of large document sets. The visualizations are intended to help humans mentally organize, electronically access, and manage large, complex information spaces and can be seen as a value-adding service to digital libraries. This introductory chapter motivates the design and usage of visual interfaces to digital libraries, reviews diverse commercially successful systems, discusses major challenges, and provides an overview of the chapters in this book.

1 Motivation

Digital Libraries (DLs) [2] are content-rich, multimedia, multilingual collections of documents that are distributed and accessed worldwide. Given that they are becoming the main repository of mankind's knowledge, the design of useful interfaces to access, understand, and manage DL content has become an active and challenging field of study.

Today, our primary means of accessing DLs are search engines that typically retrieve very large amounts of more or less relevant documents. Search interfaces lack the ability to support information exploration, making it increasingly difficult for scientists and practitioners to gain a "big picture" view of DLs, to locate germane resources, to monitor the evolution of their own and other knowledge domains, to track the influence of theories within and across domains, etc.

K. Börner and C. Chen (Eds.): Visual Interfaces to Digital Libraries, LNCS 2539, pp. 1–9, 2002.

Visual interfaces to DLs exploit powerful human vision and spatial cognition to help humans mentally organize and electronically access and manage large, complex information spaces. The aim is to shift the user's mental load from slow reading to faster perceptual processes such as visual pattern recognition.

Visual interface design draws on progress in Information Visualization (IV) [3], a field rooted in geography and scientific visualization. The IV field is not even 10 years old, but is growing quickly. It is both far-reaching and of interdisciplinary nature - spanning areas such as IR, WWW, DL, and HCI - and has tremendous potential to improve/change information access, processing, and management.

IV research as well as research on visual interfaces to DLs is facilitated by several factors: the explosion of information available digitally; the decreasing cost of storage and computing power; larger hard disk sizes which support faster information access; accelerated graphics processors and high resolution color monitors; alternative user interfaces; and the rapidly expanding connectivity between systems. Current research is also spurred by existing mismatches between computer displays and the human perceptual system as well as discrepancies between computer controls and human motor functions.

Well-designed visual interfaces reduce visual search time (e.g., by exploiting low-level visual perception); provide a better understanding of a complex data set (e.g., by exploiting data landscape metaphors); reveal relations otherwise unnoticed (e.g., by exploiting the mind's ability to see relationships in physical structures); enable a data set to be viewed from several perspectives simultaneously; and offer effective sources of communication.

Visual interfaces to DL's aim to provide rapid and efficient access to enormous amounts of multimedia knowledge and information; provide new ways to analyze document collections; help leverage information from previous users (e.g., annotations, digital dog ears, footsteps); and facilitate information sharing and collaborations.

2 Utilization

Three common usage scenarios for visual interfaces to DLs are to: (1) support the identification of the composition of a retrieval result, understand the interrelation of retrieved documents to one another, and refine a search [4]; (2) gain an overview of the coverage of a digital library and to facilitate browsing; and (3) visualize user interaction data in relation to available documents in order to evaluate and improve DL usage. These three scenarios are exemplified below using commercially successful products that can be interactively explored online.

Visual Interfaces for Search and Browsing. Today, most search engines display matching documents in long scrolling lists. KartOO[1], developed in France by cousins Laurent and Nicolas Baleydier, organizes search results retrieved from relevant web

[1] http://kartoo.com/

search engines by topics and displays them on a 2-dimensional map, cf. Fig. 1. Each document (here, relevant Web page) is represented by a ball. The size of the ball corresponds to the relevance of the document to the query. Color-coded links suggest how the documents interrelate. Resting the mouse pointer over a "ball" causes a brief description of the contents to appear.

Alternatively, the user can move the pointer over a topic in the list on the left and two buttons, plus (+) and minus (-), appear. The plus button adds the topic to the search. The minus button eliminates the sites relative to this topic. The green bar on the right of the screen indicates how many Websites are related to a request.

By showing the semantic links between retrieval results, KartOO gives the user perspectives about things that one could not see in an ordered list. KartOO processes about 150,000 requests each day, and log data has shown that users do use the links to navigate the information space.

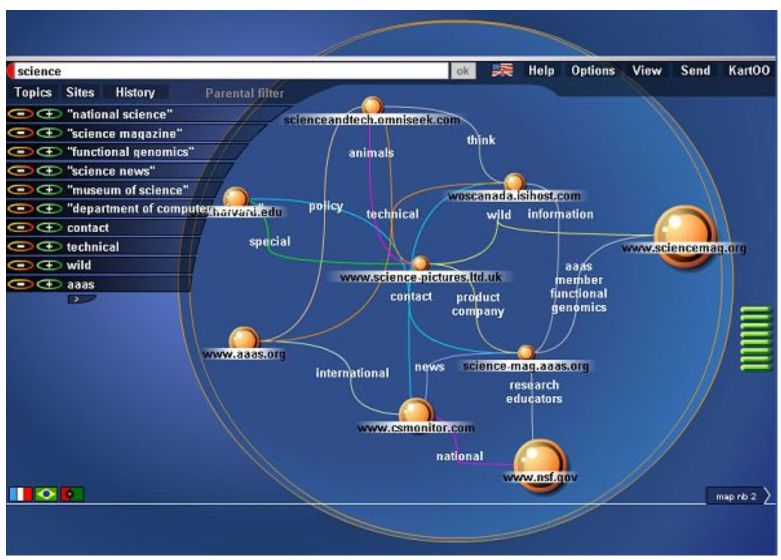


Fig. 1. The KartOO visual search interface

Visual Interfaces for DL Overview. Antarcti.ca System Inc.'s Visual Net™ product was applied to create a sample visual interface to the National Library of Medicine's PubMed database, in particular the Anatomy/Body Regions section[2]. The initial data map, Fig. 2, shows the top-level Medical Subject Headings (MeSH) categories arranged alphabetically in rows from left to right, top to bottom, e.g., Anatomy, Anthropology, etc. The colored areas represent the documents filed under each top-level category, with the size indicating the number of citations. The shape of the areas is arbitrary but supports recognition of different areas.

[2] http://pubmed.antarcti.ca/start

Users of this interface gain an immediate overview of available categories and the number of documents these categories contain. They can click on an area of interest to zoom into the corresponding area, causing an enlarged version of the area to appear further subdivided into subcategories, if there are any. The subcategories are listed in the legend on the left, and labeled in bold on the map (see Fig. 3).

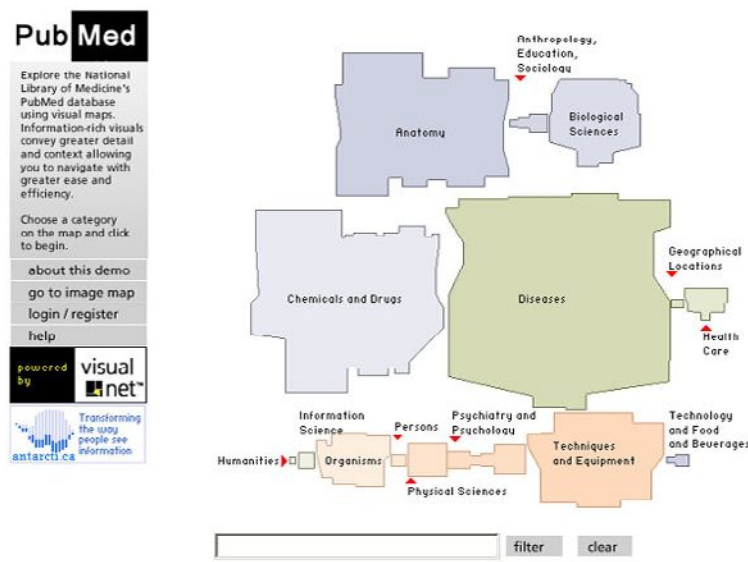


Fig. 2. Antarcti.ca System Inc.'s visual interface to the PubMed database

Alternatively, users can filter out documents of interest by entering a keyword in the search window. Matching documents will be marked on the map to facilitate visual browsing based on the Boolean search result.

On the map, individual citations are represented using target graphics and non-bold titles. The target graphics indicates how new the citation is; if the article is written in English; if the citation involves human or non-human subjects; if it is a review article; or if the citation has been published within the last three months.

Mousing over a target displays the article title, author, date of publication, and PubMed ID number. Clicking on a citation name or target retrieves the citation summary from the PubMed database. The number of citations that exist in a particular part of the database is indicated at the bottom left of the map. Visual Net implements a customizable ranking system called visibility so that only the most visible citations are displayed on any map.

Visualizing User Interactive Activities. Tools developed by Visual Insights[3] can be employed to analyze huge volumes of online customer behavior data, e.g., every visit, click, page view, purchasing decision, and other fine-grained details.

---

[3] http://www.visualinsights.com/

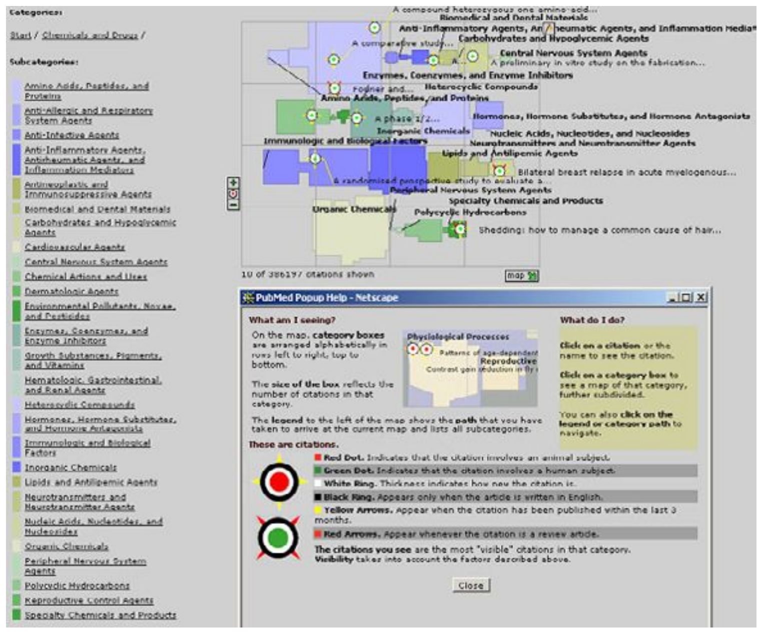


Fig. 3. Antarcti.ca System Inc.'s visual interface to the PubMed database

eBiz Live is a real-time visual path analysis tool that uses spatial metaphors to visualize the website structure, paths and flow through the site, popular entry and exit points, page flows and referrals, common click streams, etc. (see Fig. 4). It can be used to assess the effectiveness of the website design and merchandising strategies, among other concerns.

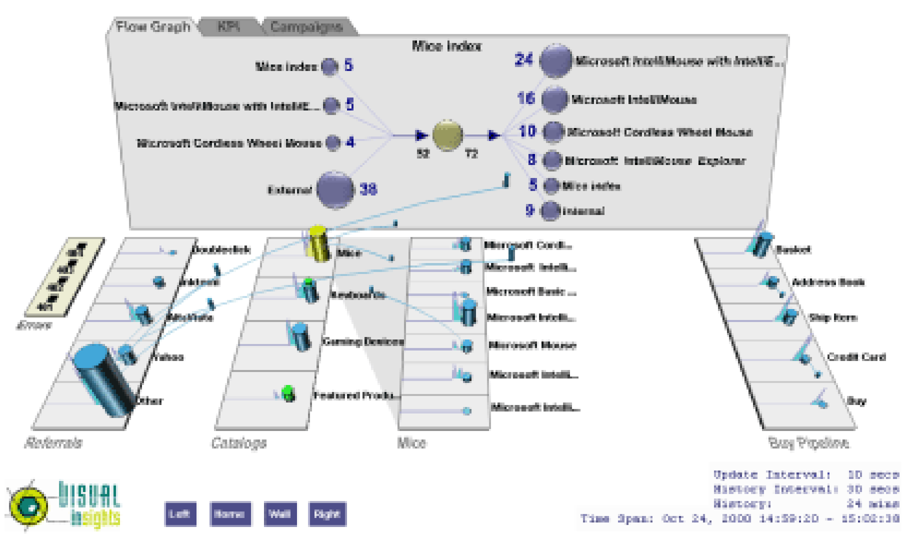


Fig. 4. Visual Insights eBiz Live real time user data visualization

Stephen G. Eick's chapter in this book further describes the underlying techniques and details inherent in different visualizations, and discusses common usage scenarios.

3 Socio-technical Challenges

The discussions at the two workshops on Visual Interfaces to Digital Libraries and the many follow-up suggestions and feedback provided by the workshop participants resulted in the identification of diverse socio-technical challenges associated with the design of visual interfaces to digital libraries. The following list represents a core set of challenges.

- Research on any (visual) interface to DLs should be based on a detailed analysis of users, their information needs, and their tasks.
- Visual interfaces provide new means to interact with data – for example, they provide an overview of what is covered by a DL, enable users to filter out relevant documents and to examine relationships among those documents. However, today's users are used to retrieval interfaces exclusively and their first reaction is to use them for the retrieval of specific documents, thereby neglecting their more robust potential value.
- Ideally, visual interfaces to DLs connect users not only to data, but also to expertise, i.e., they are collaborative.
- Simple but rich visualizations with constant information density are desirable.
- Faster, incremental, scalable ordination, mapping, and layout algorithms are needed.
- Research on how to extend 2D geographic representations to n-dimensional representations while preserving neighborhood relationships is needed.
- Good labeling, i.e., the selection of meaningful words and their display without over-plotting, is a key issue.
- It is often desirable to provide multiple perspectives to one data source.
- Federated searching of text, image, video, spatial data needs to be supported.
- Users want personal baskets to store previously selected document sets.
- Modularize visual interfaces and agree upon interface protocols to connect them to diverse DLs.Usability & usefulness studies are needed to improve interfaces and to specify what does and does not work.
- Strong collaborations among librarians and programmers help to improve the design usability of interfaces considerably.

Many of these challenges are further analyzed in the final chapter of this book, in which we propose a top-ten problem list in order to define a research agenda for visual interfaces to digital libraries.

In addition, a data and software repository is desirable for IV education and research and for facilitating the implementation and comparison of new (commercial) applications (which in turn challenge the development and improvement of the algorithms), the exploration of new ideas, the consultation with others working on related topics, and the dissemination of results.

4 The Organization of the Book

This book contains five parts. The first part introduces visual interfaces to documents, document parts, document variants, and document usage data. George Buchanan, Ann Blandford, Matt Jones and Harold Thimbleby illustrate how spatial hypertext can be applied for user-centered data access and interaction in digital libraries. Michael Christoffel and Bethina Schmitt describe a desktop virtual reality interface that uses the Quake game engine to make accessing traditional digital libraries "as easy as a game." Carlos Monroy, Rajiv Kochumman, Richard Furuta and Eduardo Urbina introduce the Interactive Timeline Viewer (ItLv), a tool that enables the user to explore the difference between various versions of the same document. Mischa Weiss-Lijn, Janet McDonnell and Leslie James report an empirical evaluation of GridVis, a system designed to help the user navigate through relevant paragraphs within a document using interactively visualized metadata. Stephen Eick, our invited speaker in 2001, reports his work at Visual Insights on the visual analysis of Website browsing patterns. Since the Web has become the ultimate 'digital library', effective methods for analyzing the usability of Web pages are urgently needed to better utilize the vast amount of data available online.

Part two is devoted to visual interfaces for image and video documents – a very timely research topic given the recent widespread adoption of digital photography and home video production. Research in this field provides additional challenges in terms of handling diverse and large-scale multimedia data. Adrian Graham, Hector Garcia-Molina, Andreas Paepcke and Terry Winograd's paper describes a system that exploits "bursts of activity patterns" in order to organize thousands of images visually for "extreme temporal photo browsing." Mike Christel's paper reports the findings of the Informedia project, in particular the access of news video libraries through dynamic information extraction, summarization, and visualization. Anselm Spoerri presents Souvenir, a system that uses handwritten notes as index and visual interface to access audio and video documents.

The third part of this book focuses on the visualization of knowledge domains. This line of research applies advanced data mining and visualization techniques to generate interactive visualizations. These visualizations exploit the power of human perception and spatial cognition and help users organize and access large, complex information spaces. Jan Buzydlowski, Howard White and Xia Lin describe how term co-occurrence analysis can be used to generate maps of author and publication networks in real-time as a visual interface to DLs. Kevin Boyack, Brian Wylie and George Davidson demonstrate the use of VxInsight, a commercial tool developed at Sandia National Laboratories, for analyzing and visualizing interrelated fields such as information visualization, human-computer interaction, and cognitive psychology.

Part four highlights cartographic interfaces to digital libraries and the role of cartographic principles in designing easy-to-understand and easy-to-use visual interfaces. Andre Skupin introduces cartographic methods, their advantages, challenges, and practical issues concerning production of information visualization maps. Guoray Cai describes a visual interface to a geographic digital library, called GeoVIBE, which combines a traditional information retrieval approach – the vector-space model – with geographic maps that visualize spatially referenced portions of documents for users to access documents. TeongJoo Ong and John Leggett describe an online interface to the Digital Flora of Texas library collection that is actively used by botanists and botanically-interested non-specialists. Dan Ancona, Mike Freeston, Sara Fabrikant and Terry Smith present work in progress on visual explorations for the Alexandria Digital Earth Prototype. They discuss the significance of using geographical location as an index to information.

The final part is entitled "Towards a General Framework". Rao Shen, Jun Wang and Edward Fox describe a lightweight protocol between digital libraries and visualization systems. The protocol can be seen as a first step towards modularization and standardization of DLs, knowledge discovery tools, and IV services. In the final chapter, Chaomei Chen and Katy Börner identify a list of top-ten problems for future research on visual interfaces to digital libraries.

Both workshops featured a large number of systems demonstrations. Many demos are linked from the workshops' webpages[4]. Systems demonstrated in 2001 include:

- VGeo Virtual Reality by Virtual Reality Software & Consulting
- LVis - Digital Library Visualizer by Katy Börner, Andrew Dillon & Margaret Dolinsky
- Starwalker by Chaomei Chen
- VxInsight™ by Kevin W. Boyack
- AuthorLink and ConceptLink by Xia Lin et al.
- Gridvis by Mischa Weiss-Lijn
- Integrated Data and Metadata Browsing by Mark Derthick
- GeoVIBE by Guoray Cai
- Librarea a 3-D Library Project by Jack Colbert
- FullView by Lorraine Normore

Systems demonstrated in 2002 include:

- PubMed, Map.Net, and VCDeal map by Tim Bray, Antarti.ca Systems
- Interactive Information Visualization in the Digital Flora of Texas by John Leggett, Texas A&M University
- 2D and 3D Visualization of Large Information Spaces by Carlos Proal, Universidad de las Americas-Puebla, Mexico
- Ted Nelson's ZigZag by Nicholas Carroll, Hastings Research
- James Burke's Knowledge Web by Patrick McKercher
- Collaborative Visual Interfaces to Digital Libraries by Katy Börner
- Visualizing Knowledge Domains by Katy Börner, Chaomei Chen, and Kevin W. Boyack

[4] http://vw.indiana.edu/visual01 and http://vw.indiana.edu/visual02/

We hope that the two workshops and this book crystallize the state of the art in visual interfaces to digital libraries. We also hope that this collection of the latest works will inspire further research and applications. Enjoy.

References

1. Scholtz, J., *DARPA/ITO Information Management Program Background.* 2000.
2. Fox, E.A. and S.R. Urs, *Digital libraries.* Annual Review of Information Science & Technology, 2002. **36**: p. 503-589.
3. Card, S., J. Mackinlay, and B. Shneiderman, eds. *Readings in Information Visualization: Using Vision to Think*. 1999, Morgan Kaufmann.
4. Hearst, M., *User Interfaces and Visualization*, in *Modern Information Retrieval*, R. Baeza-Yates and B. Ribeiro-Neto, Editors. 1999, Addison-Wesley Longman Publishing Company. p. 257-224.

Spatial Hypertext as a Reader Tool in Digital Libraries

George Buchanan[1], Ann Blandford[2], Matt Jones[3], Harold Thimbleby[2]

[1]Middlesex University, Bramley Road, London, N14 4YZ, UK
g.buchanan@mdx.ac.uk
[2]UCL Interaction Centre, 26 Bedford Way, London
{a.Blandford, h.Thimbleby}@ucl.ac.uk
[3]Waikato University, Hamilton, New Zealand
always@acm.org

Abstract. Visual interfaces may facilitate human to computer interaction as well as computer to human communication. In this paper, we introduce Garnet, a novel visual interface for interaction between humans and Digital Libraries. Garnet provides a visual workspace in which the user can structure and organize documents of interest. This structure is then used to organize and filter further documents which may be of interest, such as search results. Spatial hypertexts are introduced as a framework for creating DL interfaces, and Garnet is compared to existing DL and Spatial Hypertext systems.

1. Foreword

Visual interfaces for Digital Libraries mediate communication between the user and the system. Many sophisticated visual interfaces exist for complex information resources. Most of these exploit the visual cognitive powers of humans to make the structure of all or part of the library more accessible to its users, borrowing particularly from the field of information visualization [2].

However, focusing on the visual system as a powerful means of communicating to the user is to lose half the story. Visual interaction is used to powerful effect on the desktop to express instructions from the user to the system, and visual spaces – physical and virtual – can also be powerful resources for collaboration between users, and for tasks such as organizing information [11].

This paper presents one approach, instantiated in a prototype system which exploits a visual workspace in this second manner, centered on the user's task rather than the data and structures of the digital library itself.

2. Introduction

Information seeking [10] is a demanding and complex task which forms the driving motivation behind the adoption of digital libraries. There is typically unequal support for information seeking and the subsequent information structuring process [11, 12]. Information seeking is well supported through searching and browsing facilities, for example, whereas there is little or no support for organizing and collating the

K. Börner and C. Chen (Eds.): Visual Interfaces to Digital Libraries, LNCS 2539, pp. 13–24, 2002.

discovered documents into user-generated structures – a process which improves comprehension and reflection. (Note that we use 'document' to refer to a discrete item of information recorded in any medium: a journal paper; video excerpt etc.)

Our system, Garnet, is intended to assist users in these latter phases of document collation and organization. This task is the target of a family of tools known as Spatial Hypertexts. The concept of spatial hypertext was introduced by Marshall and Shipman – a virtual workspace in which each document is represented by a shape, and cues such as its position, color or form are used to indicate document relationships, purposes, etc. Spatial hypertexts demonstrate some superficial similarities to both concept maps and the 'concept spaces' found in some visualization systems – e.g. those in GeoVIBE and the Visual Explorations in the Alexandria Digital Library found elsewhere in this volume. However unlike such visualizations, the organization is created by the user – rather than by the system from its data – and the positioning of documents determines conceptual areas, rather than vice-versa.

Marshall and Shipman's systems, VIKI [8, 9] and the later VKB [13], have both informal and formal structuring features. Informal structuring facilities include the use of color and proximity of document shapes. Formal structuring facilities are represented by the ability to create a formal hierarchical structure of document collections, just like folders or directories in a filing system, for example. Garnet's structuring facilities and interactions are similar to those of VKB. However, unlike VKB, Garnet is deeply interconnected with a number of information systems.

Marshall and Shipman intended, and expected, a number of benefits in the use of spatial hypertexts. One was that, by using the hypertext, information workers could articulate their findings, expectations and conclusions to others. Another was that it provided the opportunity for users to clarify their own thinking through the process of organizing, selecting and rating documents. Their studies into the use of VIKI demonstrated good benefits, so providing this functionality in a DL seemed attractive.

VIKI has itself not been connected directly to an information discovery system such as a digital library, so its own support for information seeking as a whole has a complementary role to existing digital library systems. This, therefore, suggested that an integrated system would bring benefits not found in two, disconnected, digital library and spatial hypertext systems.

In addition, for many information workers information seeking and information structuring are long-term tasks. If documents are organized and re-organized in a workspace over a period of time, they become an evolving reflection of the information work the worker is engaged in.

We wondered whether the implicit information in the workspace of a spatial hypertext could be extracted and used to amplify the effectiveness of the user at a later date through, say, improved precision or recall in retrieval, or assistance in the organization of documents. This is similar to the economic concept of the positive externality, where one person's actions indirectly benefit another, but in our case a user's past activity benefits their present work.

Therefore, in Garnet we are introducing features which exploit the implicit knowledge which can be discovered in the contents of a workspace, combining that knowledge with the advantages of a direct connection between the workspace and digital libraries. The first of these is a feature we call 'scattering', in which the organization of the user's workspace is used as a filter over search results, both improving the identification of documents relevant to the user's interests and facilitating consistent placement for documents into the existing organization of the workspace. We will illustrate the use of this feature and its benefits later.

3. Sample Scenario

A pilot version of Garnet has been created, which is connected to the New Zealand Digital Library Project's Greenstone software [16]. Greenstone is a comprehensive Digital Library software system, supporting common actions such as full-text and index searching, and browsing in category hierarchies.

We will now walk through the system in use, starting from a 'bare' hypertext. The library material we will use is the Humanity Development Library of the United Nations, one of the widely available examples of a Greenstone library collection, which consists of several thousand pages.

3.1 Overview

In Figure 1, we see a Garnet user session in progress; a *window* appears inside the main application window. This is a *collection* of materials which the user has recorded in the current, or a previous, session. Each document is represented by a rectangle containing some text, as indicated in the diagram, which we term a *label* for simplicity. The user can create a hierachy of as many collections as they wish.

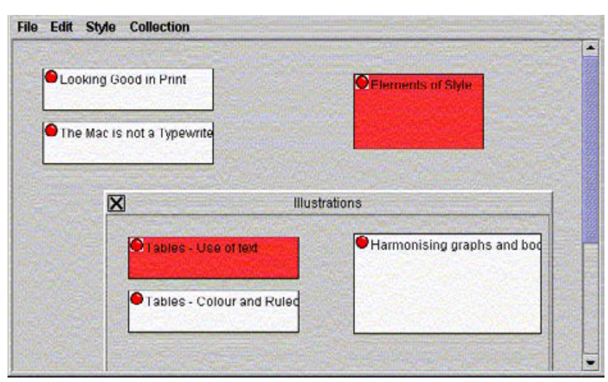


Fig. 1. A Garnet Client in use

Within a collection, the user is free to place, size and color each document label as they see fit – the space is entirely freeform. Labels can be moved and/or copied between collections in the usual way for similar direct manipulation environments. Document labels can be added explicitly by the user or through interaction with a digital library's facilities, e.g. search.

Therefore, the user is free to use the document labels both in freeform structures of their own making inside collections, and in a more formal organization by using the explicit hierarchical forms of a set of document collections. Taking the example above, we have a collection called "Illustrations", which has a column of documents on the left-hand side, and a single document on the right. The column is a structure created by the user's exploitation of space – it is not a feature enforced by the system. The column idiom can also be seen at the left of the root collection. Some use of color can be seen here, but the relationship is not clear to a reader 'from the outside'.

Larger examples, including examples of other structures such as tables (grid-like arrangements), piles (documents placed apparently randomly) and composites (groups of documents with repeated internal structures) can be seen in Marshall and Shipman's papers on the subject [8, 12].

3.2 Example Search

Let us now follow a simple sequence of interactions, starting with an example search. For our purposes, we are going to investigate snail farming, in an attempt to discover whether we have the appropriate resources to consider that form of agriculture. With Garnet loaded, we start a new search in the Greenstone system (Garnet also supports searching via Google), and we enter the simple query "snail". As shown in Figure 2, a simple collection window appears with a number of document labels appearing one beneath the other, similar to a typical search result list.

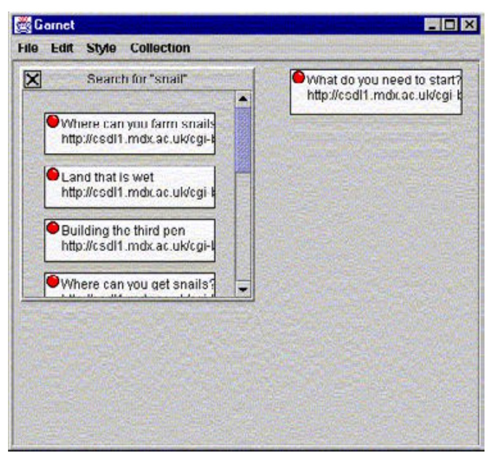


Fig. 2. A simple search

On reading the first two documents (achieved by a simple double-click on the appropriate documents), we decide that we'd like to keep the second document ("What do you need to start?"), and we move it to our root workspace window – simply dragging the document onto the main Garnet window.

The first document, however, seems a bit advanced, and we can delete it from the list, clicking on the small red circle on its top left corner (Figure 2). As a result of this, the later documents move upwards (Figure 3). Should we wish to return to the search results at a later date, by default these changes would be retained.

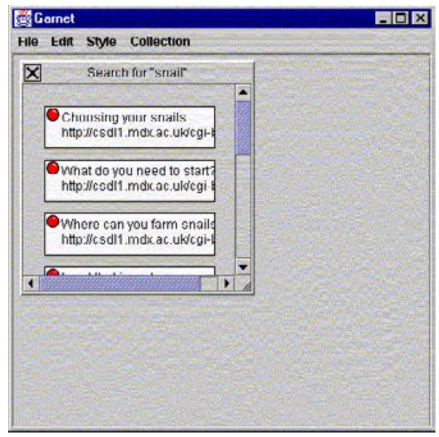


Fig. 3. The workspace after initial alterations – note changes in search list c.f. Fig. 2

3.3 Demonstration of Scatter Results'

In the previous search, we performed a plain search. Garnet, however, can exploit the organization done by the user in a novel manner. We can 'scatter' a set of documents (including search results) over the existing layout of documents in the workspace. 'Scattering' places the search documents near to groups of existing documents with which they have a strong similarity.

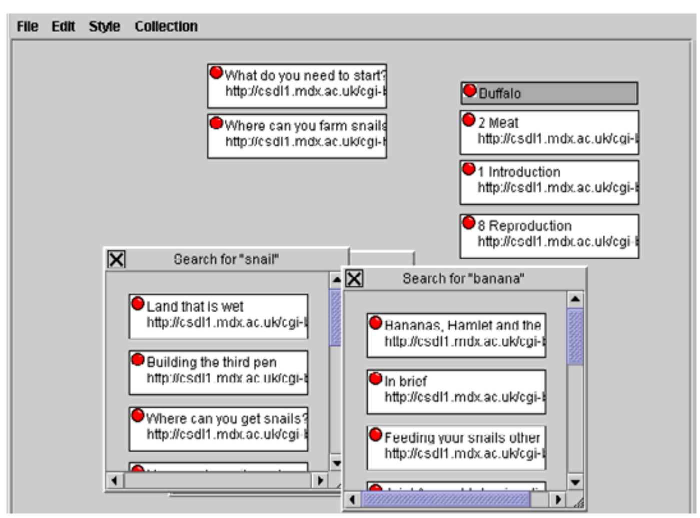


Fig. 4. Situation before 'scatter'

Continuing our previous example, we have now selected a variety of documents of interest, but let us suppose that a couple of questions remain unanswered.

Suppose we have a plentiful supply of bananas which we would like to use, but we are not sure whether this food would be appropriate. If we were to do a naïve search, on "banana", the initial results do not well match our particular interest (Figure 4).

In fact, documents which relate to our interest can be found in both the 'snail' and 'banana' searches. However, these documents of interest may not appear at the very top of either list. Normally, we would have to try and re-work our query manually to make it more targeted. In the case of Garnet, we could use the 'scattering' feature to discover any material similar to documents we have already selected for storage. Or, in other words, Garnet can generate existing search terms or filtering to represent our user's interests, based on the workspace layout they have already created.

Viewing Figure 4 again, note the third item from the top of the list "Feeding your snails …" (for clarity in this example we've chosen something that is visible). If we now do a 'scatter', (Figure 5), a subset of the 'banana' search results appear on the main collection. This small subset, which appears in a light gray below, has been found by Garnet to be a close match to the existing pair of documents, which appear in white. Suggestions are always displayed in this gray color, and below and to the right of the group of documents which they are believed to be similar to.

We can now investigate the two suggested documents which are similar to the previously selected pair. As it happens, these documents would confirm that ripe bananas can indeed be used to feed snails. If we wanted to permanently add one or other suggestion to the workspace, we can click on the 'circle' which appears on the top right corner of each of the suggestions.

If we no longer wish to see the existing suggestions, or when another set of documents is scattered, the current suggestions are cleared.

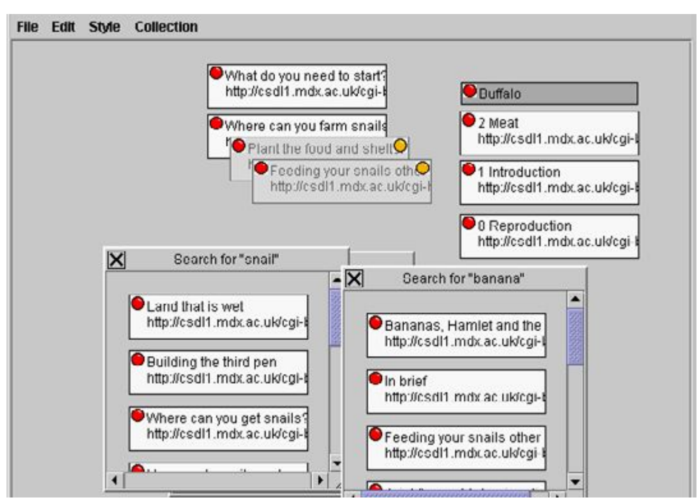


Fig. 5. After 'scatter' – note the shaded document labels added in comparison to Fig. 4

We will now discuss the underlying technologies that support this behavior, and the general operation of Garnet.

4. Supporting Technologies

At the heart of Garnet is a spatial parser [7,12], which identifies visual patterns within the arrangement of shapes (i.e. based on shape/position similarity rather than textual similarity). Our spatial parser simply identifies groups of closely placed documents; it does not distinguish visual patterns within groups of documents such as stacks or rows. Though proximity is a simple measure, it permits a wide variety of complex shapes to be successfully identified without recourse to a special recognizer for each pattern. For each identified group, the contents of the corresponding documents are used to generate lexical classifiers to represent the common topic of the documents.

The lexical classifiers are used in matching documents to the existing document groups. So, returning to our example above, the two documents on the root collection in Fig. 4 are seen as one 'group' by the spatial parser. The common topic of the group is calculated by the lexical analyzer, and a corresponding classifier produced. When the 'scatter' action was executed, Fig. 5, the classifier found a match for the two documents which were suggested. The spatial parser was again used to determine where the new, matching document labels were placed.

Importantly, a document does not yet have to appear in the workspace to match a 'scatter'. For instance, we only display the first ten matches for a search in the corresponding collection in a workspace. When the search is 'scattered', matching is done against the full returned search results, not just the top ten matches. Therefore, scattering can filter large result sets which would otherwise be laborious to browse.

To build the lexical classifiers, Garnet needs to profile each document in its workspace, and then agglomerate these individual document profiles into those used to represent each group of documents. For web documents, found through Google, Garnet builds this profile itself. In the case of digital library documents from Greenstone, this information is passed through an extension of Greenstone's digital library protocol. For other digital library protocols, such as Z39.50, Garnet could be altered to again build its own profile of documents. The profile of a document is a list of the most common words in a document (excluding stop-words such as 'and', 'the' etc.), and the profile of a group is a selection of the words a group has in common using a variant of Zamir and Etzioni's clustering method [17].

Many other capabilities can be generated from this combination of spatial and lexical analysis, such as mapping between different workspaces, or consolidating visually isolated yet topically related groups.

One significant and predictable problem may arise from groups of documents which are highly heterogeneous (e.g. those yet to be sorted and reviewed by the user) – in such circumstances, the corresponding classifier would be of little use, often matching many disparate documents. These 'miscellaneous' sets were often observed by Marshall and Shipman, and occur due to the provisional nature of the organization and collation task. As our classifiers are based on clustering techniques [4] which include the identification of incoherent clusters, we use this common clustering function to avoid building active classifiers for heterogeneous groups. Clustering techniques can also permit Garnet to assist in the organization of disorganized groups by providing a 'first-cut' at partitioning them. Our use of Zamir and Etzioni's clustering algorithm [17], which does not use word weighting, is also a distinct advantage when working across collections or when details of word weights are not available (which is often the case).

When the workspace is changed, the spatial parser must re-generate the corresponding classifiers. At present, this evaluation is done eagerly, but this could cause Garnet's response to slow down when the workspace is changing rapidly; therefore, we are moving towards lazy evaluation. Users often temporarily place individual documents arbitrarily during the ongoing flow of their work, and detecting this is problematic. The matching algorithm again helps, being highly robust to noisy data, but delaying classifier construction will improve accuracy as well as speed.

5. Previous Work

Having described some features of Garnet, we will now review a small sample of systems similar to Garnet and compare them to each other and Garnet. Firstly, we examine a number of existing visual interfaces for Digital libraries, and then we will study some spatial hypertext systems. Finally, we will discuss some issues of interest which have arisen from implementing Garnet.

5.1 Visual Interfaces for Digital Libraries

A number of visual interfaces specifically designed to support extended work in Digital Libraries already exist, e.g. DLITE [2], NaviQue [5] and SketchTrieve [6]. All three are intended to give coherent access to a number of information services (e.g. search) and sources (different collections and libraries), and represent separate searches as discrete objects in a 2-d workspace. The workspace supplies a persistent context for storing and recalling previous activity (e.g. searches) across a long period of time, the intention being to reduce unnecessary repetition of work, and to assist in the co-ordination of repeated searches across numerous information sources.

From the perspective of Spatial Hypertext, a major question would be the range of expressiveness which these systems give in the organization and annotation, implicit and explicit, of documents. Therefore, we will now consider how much control the user has over the appearance of objects in their workspace, the relative significance of different objects in the workspace, and (considering our immediate interest) in what manner they can use the objects in their workspace to perform further work.

The representation of individual documents varies considerably. In DLITE documents appear as small text-less icons, and only color can be set by the user. SketchTrieve's representations are larger, include text, and the user has complete control of their appearance. NaviQue uses a similar representation but it emphasizes zooming as a means of browsing large areas (a property inherited from Bedersen et al's Pad++ [1]). A consequence of zooming is that when a wide area is visible, the impact of individual documents is very small – often just as single points of color.

Structuring facilities are generally weak. In DLITE the ordering of documents within sets is system-controlled, and sets cannot be organized hierarchically, only as a set of peers. NaviQue lacks formal document groups – the user identifying sets of documents ad-hoc by explicitly selecting each document. SketchTrieve falls between these two positions: document sets do exist, but only as the results of a search – sets therefore cannot be used for organizing document groups. In comparison, Garnet has formal sets which can be organized hierarchically. Documents are added to a set at the user's discretion, and the internal organization of each set is fully within the user's control, except when the system has created the set (e.g. search results).

The focus of these interfaces has been on connecting the information work-space to an information system. Here, DLITE is very extensible, though most of its standard facilities are traditional (e.g. browsing), and the core remains search. NaviQue is more developed, possessing a 'similarity engine' which permits the comparison of documents or sets of documents. Though similar to the underpinnings of Garnet's 'scatter' facility, in NaviQue the principal benefit is navigational assistance, rather than support for organization and structuring.

Overall, therefore, these interfaces facilitate traditional digital library actions and exploit a few spatial hypertext idioms to assist long-term work. However, compared to Garnet and Spatial Hypertext systems, the scope for emphasizing, organizing and structuring documents is weak. Consequently, such features have not been leveraged further, as we have done with Garnet.

Thus, in its inheritance from spatial hypertexts, Garnet contains strong structuring and representational facilities compared to 'traditional' visual interfaces. In addition to this, Garnet's document comparison and matching methods, and its evaluation of the user's workspace have no direct equivalent in existing interfaces.

5.2 VIKI, VKB and Spatial Hypertexts

As already stated, VIKI and other spatial hypertexts have generally had a limited connection to information systems such as digital libraries. This limitation has consequences, no matter the benefits obtained when organizing documents.

VIKI, like Garnet, performs spatial parsing – identifying document groups by their visual properties (though the implementations differ). In VIKI this analysis is used to support visual interaction and direct manipulation in the workspace. As VIKI's connection to digital libraries is at best limited, no related textual analysis can be performed. This feature is found, however, in NaviQue and Garnet.

Though VIKI has been extended to obtain search results from the Web, when compared to the range of facilities accessible through D-LITE or Garnet, the connection to the digital library is basic, lacking browsing, metadata information, etc.

A recent paper on VIKI's successor, VKB [15], introduces a 'suggestions' facility, similar to Garnet's 'scatter'. However, the main role of suggestions in VKB is proposing visual properties for documents already in the workspace, whereas scattering in Garnet can introduce documents not yet displayed in the workspace

Visually, Shipman et al's approach is also different to ours – we place the suggestions into the workspace near to matching documents, whereas in VKB suggestions appear in text in a pop-up modal dialogue. Thus, the ability of our spatial parser to place suggestions does not appear to occur in VKB. The text matching algorithms used are not described, so a detailed technical comparison is not possible.

5.3 Summary

When one reviews the comparison of Garnet to existing visual interfaces for DLs, and to existing Spatial Hypertext research, some common themes emerge. This may not be surprising, as the papers on SketchTrieve and NaviQue refer to spatial hypertext research (VIKI in the former case, Pad++ in the latter). So, spatial hypertexts certainly influence research on visual workspaces in digital libraries.

Frank Shipman recently identified seven outstanding issues in spatial hypertext research [14], which also provide an interesting focus for visual interfaces for DLs. For example, Shipman highlights the expressiveness of different representations for documents and groups and as we have seen this is also a substantial issue for visual interfaces. In the DL interfaces discussed here and in Garnet, both user and system can place objects into the workspace – raising issues of ownership and control which Shipman notes in multi-user spatial hypertexts, but which are also present when the system plays a more active role. Similarly, Shipman questions how to connect (technically and interactively) spatial hypertexts to the general information environment and this has been an enduring issue in DL interfaces.

The most novel facet of Garnet – the evaluation of user workspace, and the subsequent use of that interpretation to assist the user in document placement and interpretation – has not previously been seen in either field, and represents one development closely related to one of the most underdeveloped areas highlighted by Shipman – computation in and over spatial hypertext.

6. Future Work: User Interface Issues

Having successfully implemented a working version of Garnet, we have performed initial usability investigations with a small number of expert subjects using talk-aloud protocol and interviews, to elicit some of the impacts of combining spatial hypertext and digital library systems, and particularly to identify and remove any particularly impeding features before performing a formal evaluation. Subjects were observed interacting with Garnet after an initial introduction to the concepts of spatial hypertext and the particular features of Garnet such as scattering. The general spatial hypertext features were easily adopted, and subjects were able to perform scattering operations successfully. Nonetheless, a number of design questions have emerged regarding the user interface. Firstly, there is the question of user versus system control of certain features; secondly, of how to represent information to the user; and thirdly, there are potential problems of metaphor dissonance, as discussed below.

On the question of control, with the scatter feature, we chose to place the control of this feature fully into the hands of the user. If scattering were applied automatically, document labels representing the suggested position of the documents could appear and disappear continuously, removing the user's sense of control and continuity. A related question was also which search results should be scattered at one time. Again, placing this within the user's control should provide a more user-centered approach.

One representational problem is the placing and appearance of scattered documents in their suggested places. We currently use a very simple technique (using a shaded label placed at the bottom left of the group), but many other possibilities exist. This choice has been informed by the observed choices of human subjects in Marshall and Shipman's work when representing uncertainty and doubt regarding the role of a document – it remains to be seen how universally intelligible this is. One unresolved issue is how to demonstrate the strength of the correspondence; the current display gives no indication of this. A further difficulty is how to inform the user of the positioning of scattered documents which are in areas of the workspace that are not currently visible.

A second representational problem is how to provide browsing of the digital library. The integration of the simple structures found in search facilities is reasonably straightforward, but browsing structures are often much more complex.

Some common desktop/GUI program metaphors do not seem to hold well, creating metaphor dissonance. For instance, in the walk through above (Figures 2 and 3) we deleted a document from the search results. We don't normally destroy documents in a library, or remove items from a list 'owned' by the system, so the 'delete' metaphor is inappropriate. Furthermore, should we bar the document from appearing in the current result set only, or from appearing in later result sets (i.e. 'blacklist' it)?

These questions are only a sample of the many aspects which offer scope for further investigation. Initial responses to the suggestions facility and the general interaction of the system have been favorable, so our next steps will be to more systematically evaluate our novel features and the best means of resolving these implementation details.

7. Conclusions

At present we have a simple pilot system working; our initial system has shown the potential for making sensible suggestions within the context of our trial Digital Library sources. A number of the issues faced so far have been discussed above. Some parts of the system are as yet incomplete, but we will soon perform a formal evaluation of the system through a full user study, and will then develop Garnet further to discover the issues which arise within a co-operative, multi-user context.

Beyond our own project, we believe that the idioms of spatial hypertext and the understandings of the spatial hypertext community represent a useful input and framework to future research into DL interfaces. Similarly, much can be investigated in visual DL interfaces which will extend the understanding of spatial hypertext. Some of these synergies have been covered in this paper, but others have yet to be fully explored.

Acknowledgments

Our thanks to the University of Waikato's New Zealand Digital Library Project, especially David Bainbridge, Stefan Boddie and Ian Witten. This work was funded by Middlesex University.

References

1. Bederson, B. and Hollan, J. Pad++: A zooming graphical interface for exploring alternate interface physics. In Proceedings of the ACM Symposium on User Interface Software and Technology (UIST '94, Marina del Rey, CA, Nov.) ACM Press, New York, NY 17-26.

2. Card, S. K., Mackinlay, J. D. & Schneiderman, B. "Readings in Information Visualization – using vision to think." Morgan Kaufmann Publishers, Inc. 1999.
3. Cousins, S. B., Paepcke, A., Winograd, T., Bier, E., and Pier, K. The digital library integrated task environment (DLITE). In Procs ACM DL '97. ACM Press, 142-151.
4. Cutting, D. R., Karger D.R., Pederson, J.O. and Tukey, J.W. Scatter/Gather: a cluster-based approach to browsing large document collections. Procs. ACM/SIGIR 1992, 318-329.
5. Furnas, G., and Rauch, S., Considerations for information environments and the NaviQue workspace. In Procs. ACM DL '98. Pittsburgh,PA, June) ACM Press, New York, 79-88
6. Hendry, D. G., and Harper, D. J., An informal information-seeking environment. Journal of the American Society for Information Science, 48(11):1036-1048, 1997.
7. Lakin, F., Visual Grammars For Visual Languages, ProceedingsAAAI-87, Seattle, Washington, July 12-17, 1987, pp 683-688
8. Marshall, C., Shipman, F. and Coombs, J., Spatial Hypertext supporting emergent structure. Proceedings of the 1994 ACM European Conference on Hypermedia Technology (ECHT '94, Edinburgh, UK) ACM Press, NY, pp.13-23.
9 . Marshall, C. and Shipman, F., Spatial Hypertext and the practice of information triage. In Proceedings of the Eighth ACM Conference on Hypertext (Hypertext '97, Southampton, UK) ACM Press, New York, NY pp.124-133.
10. Marchionini, G. Information Seeking in Electronic Environments. Cambridge Series in Human Computer Interaction, Cambridge University Press, Cambridge, UK, 1995
11. O'Day, V. and Jeffries, R. Orienteering in an Information Landscape: How Information Seekers Get From Here to There, InterCHI'93, ACM Press, Amsterdam, 1993, pp. 438-445.
12. Shipman, F., Marshall, C., and Moran, T., "Finding and Using Implicit Structure in Human-Organized Spatial Layouts of Information", Proceedings of Human Factors in Computing Systems (CHI '95), 1995, pp. 346-353.
13. Shipman, F., Hsieh, H., Airhart, R., Maloor, P., and Moore, J.M., "The Visual Knowledge Builder: A Second Generation Spatial Hypertext", Procs. ACM Hypertext, 2001, 113-122.
14. Shipman, F, "Seven Questions on Spatial Hypertext", 2nd International Workshop on Spatial Hypertext, Aarhus, Denmark, 2001.
15. Shipman, F., Moore, J.M., Maloor, P., Hsieh, Akkepeddi, R., "Semantics Happen: Knowledge Building in Spatial Hypertext", Proceedings of the ACM Conference on Hypertext, 2002, 25-34.
16. Witten, I., McNab, R., Boddie, S., Bainbridge, D. "Greenstone: A Comprehensive Open-Source Digital Library Software System". Proceedings of the Fifth ACM Conference on Digital Libraries, June 2000, ACM Press, pp.113-121.
17. Zamir, O., Etzioni, O., Mandani, O. and Karp, R. M. "Fast and Intuitive Clustering of Web Documents". Third International Conference on Knowledge Discovery and Data Mining, August 14-17. AAAI Press, Menlo Park, California. 1997, pp..287-290.

Accessing Libraries as Easy as a Game

Michael Christoffel and Bethina Schmitt

Institute for Program Structures and Data Organization
Universität Karlsruhe
76128 Karlsruhe, Germany
+49 721 608-{4069, 3911}
{christof,schmitt}@ipd.uni-karlsruhe.de

Abstract. One main idea when developing user interfaces for digital and hybrid libraries is to make use of real-world metaphors. This gives library customers the advantage to access digital collections the same way as they would traditional collections. However, while existing "real world" library interfaces still miss the attraction of the wider public, game industry is very successful selling virtual reality games. In this paper we describe a study how to close the gap between these two worlds. We describe the development of a library interface that bases on a commercial computer game. The interface models the interior and exterior of an existing library building including the most important functions for literature search. Not only teenagers tested the developed prototype with big attention.

1 Introduction

Current research in the development of user interfaces for digital and hybrid libraries aims in finding new and appropriate ways to access and present large collections of digital and non-digital documents. One idea is to provide graphical user interfaces using real-world metaphors so that library customers can access digital collections the same way as they would traditional collections. Very sophisticated solutions already exist; however, they still lack a wider audience. On the other hand, computer game industry - following different aims - manages to attract people with virtual reality environments.

Game industry is especially successful among teenagers and young people. These are exactly the groups libraries have a special concern to educate and lead them to work with books and documents. Politicians and pedagogues complain that teenagers and young people prefer sitting in front of computer games than books; libraries treat entertainment industry as a threatening of their existence.

Maybe a better idea than to complain about the success of computer games (and other products of the entertainment industry) and the change in spare time culture is to accept this as a part of contemporary society and find own solutions how to offer real alternatives for young people. This way, a 'serious application' that is supposed to attract young people must keep pace with the pleasure level dictated by entertainment industry.

Existing 'real world' or 'virtual reality' interfaces for libraries are often accused to be slow, or difficult to operate, or to have an unattractive look. Although the authors

K. Börner and C. Chen (Eds.): Visual Interfaces to Digital Libraries, LNCS 2539, pp. 25–38, 2002.

of this paper do not agree with this statement, modeling virtual worlds is one point where developers of 'serious applications' can learn from developers of computer games. A very successful branch of computer games - 3D action games - allows the player to walk through virtual worlds - indoor and outdoor - and is able to display these worlds precisely from the player's point of view. The player has the illusion of walking through the world himself/herself and is faced with the same adventures the game figure has to overcome. This way, 3D action games already have found solutions to present those things developers of visual interfaces for application programs need: Buildings with several floors, elevators and stairs, doors, rooms with chairs, tables, and furniture, view screens, moving objects, transparent objects, and, very important, build-in communication protocols. A large number of games is available for different operating systems and hardware platforms.

Fig. 1. The entrance of the university library in Karlsruhe

Recognizing this fact, the question is whether it is possible to use the capacities of existing computer games for the development of 'serious applications'. In order to give a hint for the solution of this question, this paper will present a graphical user interface for library access based on an existing computer game. The prototype has been shown to different user groups. Their satisfaction and their comments can be used as a measure for the goodness of such a 'game interface'.

For our experiment, we have selected the game Quake II, produced by id software (http://www.idsoftware.com). Based of this game, we present a real-world model of the university library in Karlsruhe (figure 1).

The work presented in this paper is part of the Modern (M)art project (Modeling, Experiments, and Simulation in/with Information Markets). In this project, we want to investigate business models, business strategies, and customer behavior in the application domain of scientific literature markets. Modern (M)art is an interdisciplinary research project at the university of Karlsruhe, encompassing the department of computer science, the department of economics and business administration, and the university library. It is funded by German Research Foundation (DFG) as a part of the national strategic research offensive "Distributed

Processing and Delivery of Digital Documents (V^3D^2)". An overview on the works in this project in the field of computer science can be found in [2,3,4].

We continue as follows. In the following section, we will discuss shortly the objectives of our real world interface. Then we will take a deeper look in the implementation of our 'game level' and the modifications we have done on the game so that it can be used as a library interface. Then we will describe in which way our interface is suitable for library access. In the following section, we will shortly discuss preliminary results of demonstrations and experiments with different user groups. Thereafter we will give an overview on some related work in the field of real world interfaces for digital libraries. We finish this paper with a conclusion.

2 Objective

Practically all library interfaces that can be found all over the world are text-based. While first generation OPACs are being removed by Web interfaces, these new interfaces are often ergonomic and colorful, but queries still have to be formulated by filling our text forms, and results are presented in result lists. In practice, the way of result presentation is dominated by the capabilities of the HTML table construct.

Most people agree that in general figures and graphical information are more suitable to present information to human persons than pure text. Especially for young people it is important to have an attractive design, in order to fulfil their expectations in regard of a "modern look and feel". But although there are solutions for graphics-based user interfaces for libraries, text-based interfaces are prefered. The problems with graphics-based interfaces is that

- they are said to be slow;
- they are said to waste resources;
- they are said to be anything but intuitive to use;
- they are said to be rarely used;
- they are said to be unattractive and boring.

The real reason for the absent success of graphical interfaces is that people have learned to use text interfaces and now can use them very efficiently, and that they shy to use new ways when they cannot see the advantages at a glance.

One approach in interface design is to confront customers with things they already know, or at least which look familiar. This approach prevents customer to become deterred by an unknown technology. Interfaces following a real-world metaphor try to model existing worlds or worlds that look similar to existing structures. A world can be a room, a collection of rooms, a building, or a complete city. The idea is that the customer, who knows how to deal with objects in the real world, will be able to do this the same way in the virtual world.

For the designer of a library interface, the challenge is to model a library with possibly most of the services associated with such institution. The objective is to offer an access to the services and collections, which is not treated to be an exotic but a better alternative than the usual text-based interfaces, at least for some user groups.

3 Building a Library World

Quake II is a 3D action game, sometimes better known as egoshooter. The aim of practically all games of this kind is to find the way through a virtual world, kill all resistance, and, in the end, save earth. Some pedagogues worry about the high level of aggressiveness that is necessary to win this kind of games, and call into question the mission that can be learned in these games: solve all problems with the gun.

The authors of this paper see the danger that lies in this kind of games, too. Because of this, one of the first decisions made was to remove all traces of violence from the library interface. There are no enemies and no monster, and also there are no weapons and no armory. In order to retain the metaphor, the user now holds a laser pointer, which really looks fearful, but is completely harmless.

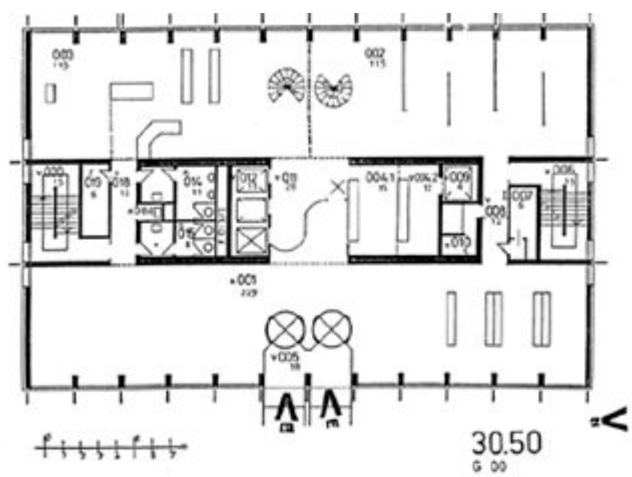


Fig. 2. Floor plan of the university library (here: ground floor)

3.1 Geometric Model

Quake II easily allows to add new virtual worlds by specifying a new map. This map contains the look of the new world, the inhabitants of this world, and also dynamic events that can happen in this world, e.g., a door opens or an earthquake starts. It is important to remark that maps in Quake II are always 3-dimensional. It is necessary to consider the height of an object. E.g., a player standing on a bridge has a different view than a player standing under the bridge. And, of course, one and the same building may look different from one floor to the other.

We began modeling the interior and exterior of the library building beginning with the original building plans (figure 2), using the map editor Worldcraft by Ben Morris. This level editor allows the construction of a map in all details (figure 3). Floors, ceilings, walls, windows, wardrobes, tables, doors can be grouped to rooms, and afterwards they can be equipped with more pieces of furniture. Several rooms put together become buildings. And, of course, it is possible to construct the world outside the buildings.

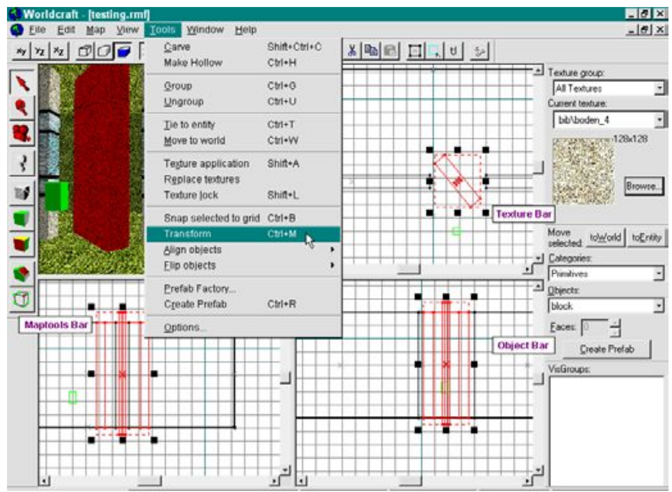


Fig. 3. Creating the map (here: designing the revolving doors)

composition of these basic geometric shapes. In most cases, brushes will be solid, but there can be also a use for non-solid brushes, e.g., when creating water or other liquids.

While brushes are always static, entities can have an active and dynamic behavior; they can be animated, they can be carried, or they can move through the world on their own. Entities can either be solid entities or point entities. Solid entities are similar to brushes with the one difference that they can have an active behavior. Doors and lifts are a good example of solid entities. In opposite to brushes and solid entities, which always have a volume, point entities are logically located at one point. Point entities can either be hidden entities or models. Hidden entities are used for light sources, hidden switches, and also markers for several purposes (e.g., for the position where a player enters the world). Models are more sophisticated visible objects than brushes. They are used for things that can be carried around, complicated pieces of furniture, and people. Figure 4 shows the complete object hierarchy.

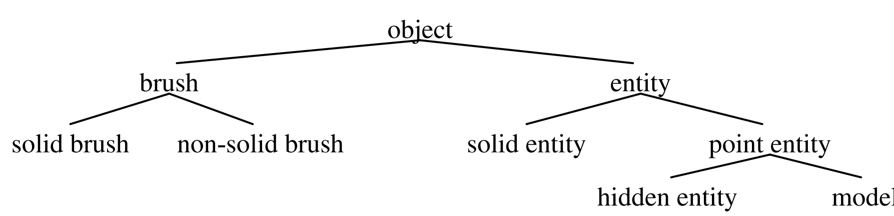


Fig. 4. Hierarchy of geometrical objects in Quake II

The definition of active entities needs special care, because it is necessary to specify the behavior in detail. E.g., for doors, the designer of the map must specify the answers to questions like these: Are the doors supposed to open automatically when a player comes near, or is it necessary to press a button to open? How fast will they open? Will they close automatically, and, if yes, when? What happens if they close and a player stands in between?

After completing this first step of the design, we had a correct model of the geometry of the library building. However, the result did not look very close to reality, because many things were missing which appear in reality, and all colors, material qualities, and light conditions were wrong (figure 5).

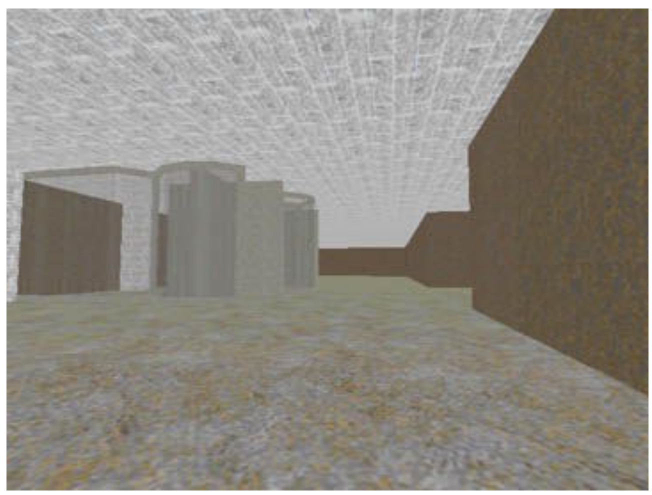


Fig. 5. The geometry of the university library without textures (here: entrance area)

3.2 Textures

The next step was to model the textures of the objects: wallpaper, floor covering, material and painting of the furniture. In order to get an image from the reality as close as possible, we made digital photos from the whole building. The textures extracted from the digital photos had to be re-sized and adapted to a color palette. For a realistic impression, each texture can be assigned various material properties (see figure 6).

3.3 Adding Entities

There is a wide range of predefined models for the game, but some necessary objects were not contained in the game, or at least not with the necessary functionality, e.g., coat racks, lockers, computers, and books. For these objects we had to create models on our own, using the tool 3D Studio Max by Discreet Software (figure 7). Each model consists of a wire model having a skin. For each model it was necessary to specify in detail how the object looks when viewed from different directions and from different distances.

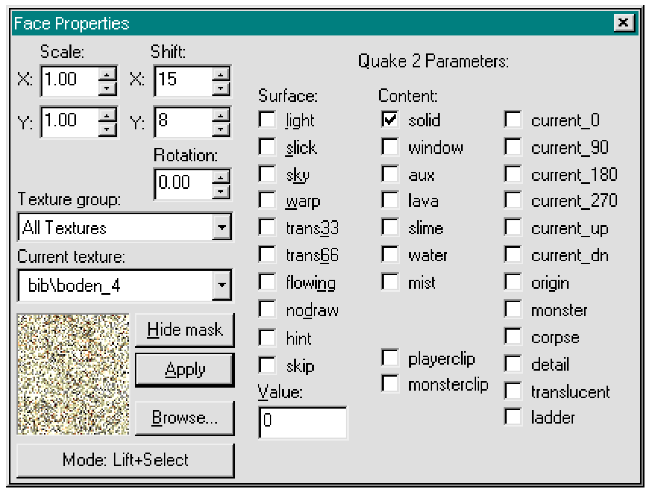


Fig. 6. Assigning material properties to textures (here: floor of the entrance area)

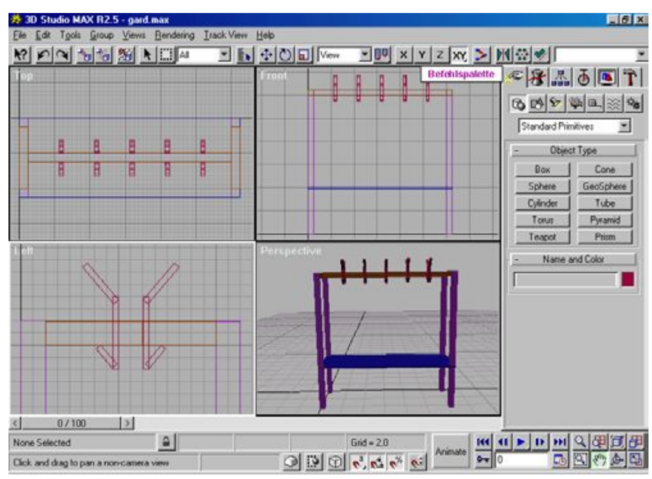


Fig. 7. Creating models (here: coat rack)

After finishing this step, the visualization came very close to reality. Still missing were shadows, because until now in the library map were no light sources, so in every room was the same diffuse light. We used hidden entities in order to place light sources and light effects. We added both daylight shining through the windows and different kinds of artificial light sources. The behavior of the light sources had to be described in detail. Similar to lights, we added sound and sound sources. It is possible to assign pre-defined sounds or import recorded sound files. A very important next step was to define the look and function of the laser pointer. The result of the complete modeling process can be seen in figure 8 and 9.

Fig. 8. Interior of the university library (here: bookshelves in the ground floor)

Fig. 9. Interior of the university library (here: computer room)

3.4 Extending the Game Logic

The steps described before were possible using the map editor without writing any single line of code. Unfortunately, for a more individual adaptation of the game, so that it can work as a library interface, it was necessary to touch the source code of the game. Id software allows to edit and re-compile the part of the software that contains the game logic. In the Microsoft Windows/Intel version of the game, this the game logic is contained in the file game.dll. The source code is written in C programming language.

For example, one of the things we had to do was to extend the functionality of the revolving doors. Quake II offers by default a manifold of possibilities how revolving doors can operate. What we had to add is maybe the most common way: A person pushes against the door, and the door turns. Similar with elevators. It is possible to make an elevator operate automatically, it is possible to have some switches or keys somewhere in the building, but the game developers did not cover one case: that there is an array of buttons inside the cabin, and when pressing a button, the elevator moves to the floor associated with the button. We added a new elevator control.

Most challenging, of course, was the communication with the library agent. Same as in the real library, it is possible to send a query by filling out a form using one of the computers standing in the library. The modified game logic will open a TCP/IP connection with the library agent and wait for the answer. As soon as the answer arrives, the user can watch the new bookshelves be filled with books that represent the documents contained in the result list.

There are some limitations of the game engine, which we could not resolve. E.g., it is not possible to define curved surfaces, only approximations by polygons. It is possible to define transparent objects, but these are not allowed to move. It is not possible to display more than 256 colors, which really spoils the impression of reality. Moreover, it is not possible to display more than 128 movable objects at the same time. Since every book must be movable, this is a strong limitation of the size of the bookshelves.

4 Features

The most important feature of our prototype implementation is literature search, covering different collections and libraries. Results can be ordered by collection, availability, publication year, and language. There are different ways to distinguish books: by different bookshelves, by the color of the book, and by different labels on the back of the book. The user can dynamically change the assignment of shelves, colors, and labels. The thickness of a book is permanently associated with the number of pages.

The user can select a book standing in a bookshelf according to the metaphor by “shooting” on it with the laser pointer. For a selected book, short information is displayed on the bottom side of the screen (author, title, publication year). A second shoot, and display changes to full information. A third shoot could automatically fill out the lending form, but this feature is not yet working. It is possible to browse a collection by simply move along the shelf with the laser pointer (figure 10).

One important feature of most 3D action games is the multi-player capacity. Quake II supports a client-server-architecture that allows several persons to share the same virtual world, connected by Internet or by a local network. The number of person that can be in the virtual library at the same time is limited by 64. Every user sees the virtual world from his/her own point of view and is able to talk with other people (talking is not necessarily limited by the people standing in one room, since Quake II supports a kind of radio transmission). Every user has his/her own inventory, but also the possibility to exchange objects with other people in the virtual library. Gender and look of the virtual library users can be selected before entering the library. It is also possible to write models for the individual users so that every user can specify his/her own look.

There is a large number of services in the real world that can also be found in the virtual world, although our prototype implementation does not include all of them:

- information and issue desks, where the user can receive assistance and information about the collections and his/her account;
- personal workplaces, where the user can deposit books and make notes and annotations that will be saved when the user leaves the library;
- group workplaces that can be used by more than one person;
- virtual librarians who can be played by human librarians or automated by the computer;
- access restrictions by intelligent doors, e.g., for the access to collections which are only available to registered users, such as special commercial databases;
- rooms prepared for the presentations of geographic documents (such as maps), video clips, or sound files;
- bulletin boards for information interchange.

It is also possible to add instructions to the interface that tell the user to go somewhere or do something in a predefined order, e.g., for a new user. This way the library interface could be used as a tutorial for the use of itself.

Fig. 10. Browsing the collection

5 Acceptance

In the introduction we have proclaimed that teenagers and young people need computer programs and user interfaces which are more attractive in their minds than the traditional interfaces. In order to find out what young people think of a game interface for accessing library collections, we visited schools and showed the prototypes to pupils of the upper classes, and we also used information days for pupils to invite school classes to Karlsruhe university. The test persons were not pre-selected in any way, and they had no previous experiences in the library interface.

The interest of the young people was higher than expected. One significant observation was that the young people followed a "play instinct". Without further assistance, they began to explore the virtual world, and after a while they also began to send sample queries to the library system. They learned very fast how to navigate through the library building. Most of them seemed to have previous experiences with 3D action games (and recognized the underlying game engine as Quake II), others learned the most important keyboard controls by themselves or with some tips from classmates. We could not notice any significant differences between the behavior of boys and girls. Nobody complained that there were no monsters in the library.

Adults (school teachers, library visitors) also showed interest, but their behavior was somehow different. They did not have problems with navigation and use of the library interface, but they used the interface more goal-directed and less exploring. All showed surprise that the world on the screen really resembles the library building. Some adult test persons had a negative opinion in advance because of prejudices against computer games.

One interesting target group we have hardly touched until now are people who have less experience with computers and Internet. We hope an interface showing the world they know will make it easier for them to use library services electronically.

We plan to repeat these acceptance test with the larger number of test persons and in comparison to other library interfaces. From more elaborated experiments, we do not only want to receive qualitative statements, but also quantitative measurements. Of special interest for us is the question whether there are dependencies between the acceptance of different interfaces to social parameters like age, gender, formal education, experience with computers in general, experience with computer games, and experience with library services.

6 Related Work

Quake II is far from being a standard tool for the development of virtual worlds. More common tools and languages are:

- Java 3D by Sun Microsystems (http://www.sun.com) is written in pure Java and this way platform-independent. Java technology makes it also possible to view Java 3D applets with a standard Web browser. Java 3D is a package of classes and interfaces that can be used in own Java programs. Developing worlds with Java 3D needs the ability to model a world in the abstract view of a scene graph.
- Open GL (http://www.opengl.org) has been published by an industry consortium and has become an industry standard for 2D and 3D graphics. It is available for various programming languages, operating systems, and hardware platforms. It allows the definition of graphical objects as composed of groups of vertices (points, lines, polygons), having different properties.
- VRML (Virtual Reality Modeling Language) by the Web3D Consortium (http://www.web3d.org) is a descriptive language for the creation of virtual worlds, which can be viewed with special VRML browsers and also standard Web browsers. VRML allows the creation of scenes out of hierarchy of nodes derived from of primitive objects by adding properties and transformations.

- Active Worlds by the Activeworlds Corporation (http://www.activeworlds.com) is a software tool especially designed for the creation of virtual worlds that can be shared by different people.

There is a great number of research project dealing with the representation of library services and document collections in a virtual library. There are approaches with very sophisticated means for document representations, but also approaches using real-world metaphors. Within the size of this paper, we can only mention three approaches.

CAVE-EDT at Virginia Tech is a 3D virtual world modeling a (fictive) library, based on VRML [7,8]. The user can navigate among several rooms which contain bookshelves for different subjects. User interaction is based on a special wand in the hand of the user, which can be used for both navigation and browsing collections of theses and dissertations. It is possible to receive title information of selected books as well as an abstract. The concern of the project was not to have a full-working library interface but a test bed for studies of the user behavior.

LibViewer at Vienna University of Technology [9,10] uses real-world metaphors for the representation of documents in bookshelves in order to assist the library user in browsing large collections. Not only location, size, color, label are considered, but also properties like the material and the condition of the book cove, or the existence of dust. LibViewer can be used by a Java Applet.

CNAM (Conservatoire National des Arts & Métiers) in Paris is working in the visualization of rare book collections [5,6]. A very interesting idea in this research project is not to use metaphors for the representatives of the underlying collection in the virtual bookshelves but textures showing digital images of the books. This way the user can visit the books exactly the same way he/she would in a real library. The interface is realized with VRML.

Finally, we want to mention the project PSDoom at the University of New Mexico [1]. This project is not a library project, but it is very interesting for us, because here a 3D action game (Doom by id software) has been used as the basis for an interface for operating system administration. In PSDoom processes on a Unix computer system are displayed as monsters. The user can survey the running processing and can wound processes in order to decrease priority or even kill processes by shooting on the monsters.

7 Conclusions

In this paper we presented an approach to close the gap between computer games and 'serious' applications like a graphical user interface for library services. We created a virtual world visualizing the university library of Karlsruhe, using the game engine of the 3D action game Quake II.

The resulting prototype surprises with very impressive graphical capacities, it allows a very fluent movement through the virtual world, and it is very easy to use. Very important, of course, there is a fun factor in the exploration of the various possibilities the user can navigate and act in this virtual library world.

We did not use all possibilities of the selected game engine. Neither we have exhausted the graphical capacities, so the impression of a real world is limited; and we have not converted all the ideas for library services in our virtual world. All this was not our aim.

One of our aims was to show that the use of a computer game as a basis for serious applications is possible, in the special regard for the design of visual interfaces. We have presented a technical solution for the adaptation of one game. We invite research groups from all over the world to develop applications on their own.

Our second aim was to investigate the acceptance of different, innovative library interfaces. The particular question here was whether a game interface is really attractive. Preliminary results confirm this assumption, but for a definite, verified result we need more investigations.

In section 2, we have listed some problems of graphics-based interfaces in comparison to text-based interfaces. Graphics-based interfaces are said to be slow, waste resources, are not intuitive to use, are rarely used, and, in general, are unattractive and boring. Some of the problems we have solved: The game engine is fast, is intuitive in use, and the virtual world looks exactly like the world where the user lives. And our preliminary user studies confirm that for most persons the game interface is attractive and fun. And Quake II and similar games are widely known. However, we must admit that the resources needed are not negligible. Moreover, it is necessary to have Quake II installed on each computer used as terminal for library access. Internet access is possible, but needs an own copy of Quake II and a fast network connection.

Meanwhile, id software has decided to disclose the entire source code of Quake II, in order to give other parties the chance to develop action games on their own. This opens the possibility to create a new version of the library interface where some limitations and restrictions could be removed. Another idea is to rewrite the interface for the improved game engine that id software has introduced with Quake III. This game engine supports 24 bit colors and curved surfaces. However, it will also have higher demands on hardware resources, so we are not sure whether it will run on the computers available in the library.

We do not want to conclude this paper without a warning. The content of 3D action games like Quake II is war, similar to chess. But unlike chess, the good graphical possibilities give the player the chance to take part on the action deeply affecting. Although our library interface is harmless, it is possible to use the same game engine with maps found in the Internet which may have an unexpected influence on some people. However, another fact is also true: playing games is fun. And what is wrong when people have fun while walking through a library or browsing in digital collections?

References

1. Chao, D.: Doom as an Interface for ProcessId Management. In: Proceedings of the Conference on Human Factors in Computing Systems, Seattle, USA, pp. 152-157 (2001)
2. Christoffel, M., Franke, T., and Kotkamp, S.: Trader-Supported Information Markets: a Simulation Study. In: Proceedings of the 2nd International Conference on Electronic Commerce and Web Technologies, München, Germany, pp. 101-110 (2001)
3. Christoffel, M., Nimis, J., Pulkowski, S., Schmitt, B., and Lockemann, P.: An Infrastructure for an Electronic Market of Scientific Literature. In: Proceedings of the 4th

IEEE International Baltic Workshop on Databases and Information Systems, Vilnius, Lithuania, pp. 155-166 (2000)
4. Christoffel, M., Pulkowski, S., Schmitt, B., and Lockemann, P.: Electronic Commerce: The Roadmap for University Libraries and their Members to Survive in the Information Jungle. In: ACM Sigmod Record 27 (4), pp. 68-73 (1998)
5. Cubaud, P., Thiria, C., and Topol, A.: Experimenting a 3D Interface for the Access to a Digital Library. In Proceedings of the 3rd ACM Conference on Digital Libraries, Pittsburgh, USA (1998)
6. Cubaud, P., and Topol, A.: A VRML-based user interface for an online digitalized antiquarian collection. In Proceedings of the Web3D Conference, Paderborn, Germany, (2001)
7. Das Neves, F., and Fox, E.: A Study of User Behavior in an Immense Virtual Environment for Digital Libraries. In Proceedings of the 5th ACM Conference on Digital Libraries, San Antonio, USA, pp. 103-111 (2000)
8. Kipp, N.: Case Study; Digital Libraries with a Spatial Metaphor. In Proceedings of the SGML/XML '97 Conference, Washington, USA (1997)
9. Rauber, A., and Bina H.: A Metaphor Graphics Based Representation of Digital Libraries on the World Wide Web: Using the libViewer to Make Metadata Visible. In Proceedings of the Workshop on Web-based Information Visualization (in conjunction with DEXA'99), Florence, Italy (1999)
10. Rauber, A., and Bina H.: Visualizing Electronic Document Repositories: Drawing Books and Papers in a Digital Library. In Proceedings of the 5th Working Conference on Visual Database Systems, Fukuoka, Japan, pp. 95-114 (2000)

Interactive Timeline Viewer (ItLv): A Tool to Visualize Variants among Documents

Carlos Monroy, Rajiv Kochumman, Richard Furuta, and Eduardo Urbina

TEES Center for the Study of Digital Libraries
Texas A&M University
College Station, TX 77843-3112, USA
(979) 845-3839
{cmonroy,rajiv,furuta,e-urbina}@csdl.tamu.edu

Abstract. In this paper we describe ItLv (Interactive Timeline Viewer), a visualization tool currently used to depict the variants obtained in a textual collation. A textual collation is a process in which a base text is compared against several comparison texts to identify differences (variants) among them. The interactive options of ItLv provide different abstractions of a dataset by enabling the presentation and exploration of the relationships that exist within the dataset. Applying ItLv to the dataset resulting from a collation therefore helps understand the relationships among the texts. The example dataset used in this paper is a collation of six early editions of Cervantes' *Don Quixote.*

1 Introduction

We are in the process of creating an Electronic Variorum Edition (EVE) of Miguel de Cervantes y Saavedra's (Spain, 1547-1616) *Don Quixote* [4,5,6]. The EVE will be included in the Cervantes Digital Library (CDL), part of the ongoing Cervantes Project [13]. Our EVE will contain multiple copies of all the early significant editions of this important work, in interlinked facsimile and textual forms. We also are creating critical editions of the work, reflecting the result of scholarly interpretation and emendation of the work, as well as associated commentary, explanation, and other forms of annotation.

An EVE is created with the MVED [5], a collation tool in which a selected base text is compared against several comparison texts to identify the differences (variants) among them. Much of the work of the scholar will involve understanding the variants. Consequently, we apply a visualization tool (ItLv) to depict those variants as well as to provide information about each of them—for example, where in the text it appears, its length in characters, its content, and an image of the original page in the book. Moreover, we can say that ItLv can also be applied to any other set of texts for which a variorum edition needs to be created.

In the context of digital libraries several approaches to provide visualization interfaces have been advanced to represent collections of books, journals, and magazines [7]. Furthermore, there are tools that enable users to search for specific attributes in the items of a collection [8]. Once an item has been found, the user then needs a browsing mechanism that enables the exploration of the item in further detail

K. Börner and C. Chen (Eds.): Visual Interfaces to Digital Libraries, LNCS 2539, pp. 39–49, 2002.

[9]. In addition to specific attributes, we are finding that it is useful to apply visualization mechanisms to depict the variants among different editions of the same book as well as among different copies of the same edition.

In terms of comparing different copies of the same text, the Digital Variants Browser (DV Browser) [3] is a system that enables users to "analyze several different versions or writing stages of the same text providing an overview of the entire text material." However, in our case we are not visualizing different stages of the same text, but several different "final versions" of the same text, that is, several editions of the same book.

The DV Browser is based on the Flip Zooming technique [2], which provides focus+context to visualize information. The information is divided into tiles, the tile with the focus is positioned in the center of the display and the rest of the tiles are placed around it. This is a very useful approach for comparing several texts. However, in terms of visualization, it depicts each version of the text as one variant.

Analyzing different versions of software can be seen as a similar process to what we are doing. In this context, Baker and Eick [1], propose a system that enables users to visualize not only the structure of a big software project, but also the evolution of the source code. However, in the case of different versions of software, variants are expected, since new functionality requires new code; whereas, in our analysis, the texts in different editions "should be" the same, since the author has not modified the text. However, due to printing errors, or compositor's preferences variants are introduced between two editions or even between two copies of the same edition; therefore, our focus is to visualize those variants.

Plaisant, et al. [11], describe the use of LifeLines, which is a tool that depicts personal histories using a timeline-based visualization. The use of LineLines helps users to analyze a dataset of discrete events, especially because "LifeLines reduce the chances of missing information, facilitate spotting anomalies and trends, streamline access to details…." These are some reasons why we find a timeline representation useful to analyze variants among several texts. In this paper, we describe our use of ItLv and its functionality in visualizing collation results. Interactive functions allow generation of different representations of a dataset, which allows users to develop a better understanding of the relationships among variances identified in a collation.

2 Visualizing the Results of a Textual Collation

ItLv is based on earlier work in our center by Kumar [7] and applies a timeline-like visualization metaphor for the purpose of representing datasets whose elements are ordered by at least some of their attributes. In the EVE application discussed in this paper, we are depicting variants among six editions of *Don Quixote*. *Don Quixote* was first published in Madrid in 1605. The Madrid 1605 edition is called the *princeps* and is used as base text in this collation. In addition to the princeps, the other five editions used are Valencia 1605, Madrid 1605 (Second Edition), Brussels 1607, Madrid 1608, and Madrid 1637. Each of these five editions is compared in turn to the base text, and figure 1 depicts the results of this collation for the first chapter of the novel. Here, the Y-axis represents the differences between each of the five editions and the princeps. The Y-axis is arranged in chronological order from bottom to top; for example, the bottom-most line represents the differences between the Madrid

1605 princeps and the Valencia 1605 edition, while the top line represents differences between the Madrid 1605 princeps and the Madrid 1637 edition. The X-axis represents the offset in the text; a value of zero represents the beginning of the text, and the larger the value the closer to the end of the text. Each variant is depicted as a rectangle; the height of the rectangle represents the length of the variant in characters.

By clicking on any rectangle a new window will be displayed (figure 2). This new window depicts all the attributes of the variant represented by that rectangle; (a) includes most of the string and numerical attributes, (b) depicts the text of the variant, and (c) provides an image of the page where that variant appears in the original book. This option provides a context and detail for any variant in the collation. (Note that the bold arrow, letters and square brackets in figure 2 are not part of ItLv; they were included to explain the figure.)

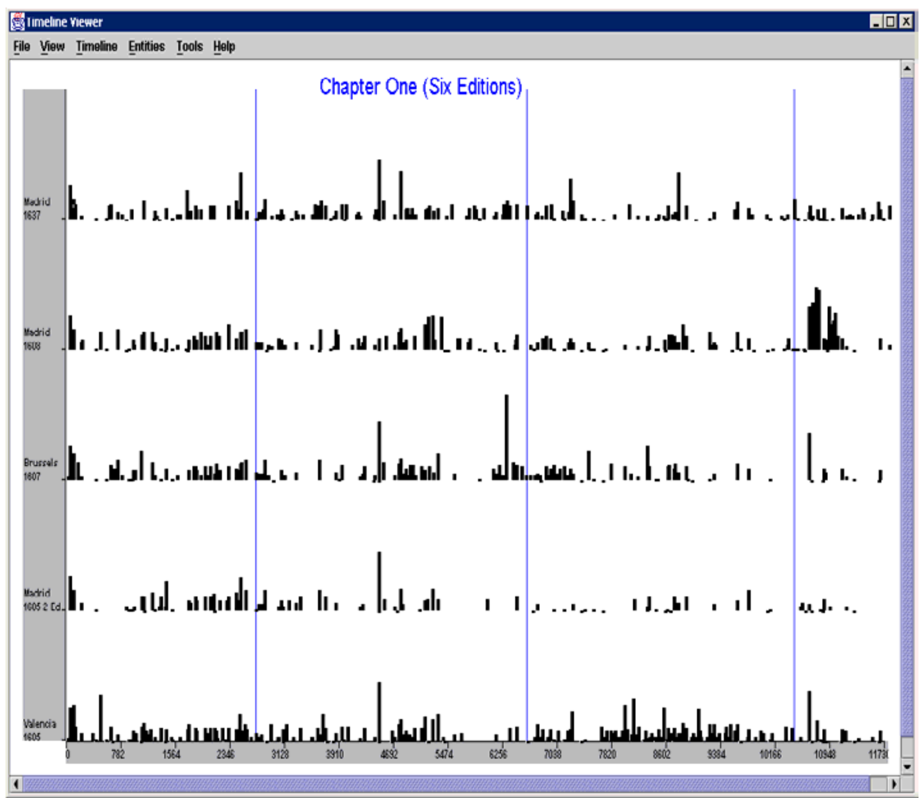


Fig. 1. Variants of the collation among six editions of *Don Quixote*

The results of a collation can be depicted in different ways based on the attribute selected for the Y-axis. This enables the user to analyze the same dataset from different perspectives. Following we discuss two examples. In figure 3, instead of presenting the editions on the Y-axis, as it is in figure 1, a string of five characters containing Y's and N's is displayed. This string represents in what texts a variant was found, for example the string “YYNNY” means that a particular variant was found between the base text (the princeps) and the first, second, and fifth comparison texts; i.e., that corresponds to the editions of Valencia 1605, Madrid 1605 (Second Edition), and Madrid 1637 respectively. Presentation of a variance summarization helps highlight anomalies (e.g., editions with unusual differences from the others) as well as “families” of editions (i.e., editions containing similar patterns of variances suggesting a relationship in their derivation).

Figure 4 shows the same results as in figure 3. On the left is the same display as figure 3’s with some clusters highlighted. It is also possible to observe certain clusters of variants for the following combinations of texts: “YNNNN” with a high concentration of variants at the end of page two and in two thirds of page three. For “NNNNY” there are two clusters of variants, one in pages one and the beginning of page two, and the other in page three. On the right, a sorted list shows the number of variants for all the 31 possible combinations of texts in the collation. As the number of texts increases, the number of combinations in which variants can appear increases also, therefore, using this attribute can be useful only for a small number of texts, e.g., less than seven. An alternative method to overcome this problem can be creating categories based on the number of texts in which the variants appear. On each screen, only those variants present in the same number of texts would be depicted regardless of the text they appear on. Thus, one screen will depict the first category, i.e., those variants that appear on only one text; another screen will depict those variants that appear on two texts (the second category), and so on.

Figure 5 shows a different presentation of the results of the collation. This time, the results are grouped by the length of the variant in characters (depicted on the Y-axis). In figure 1, the size of the rectangles is proportional to the length of the variant. Therefore shorter variants have smaller rectangles, whereas larger variants have bigger rectangles. Nevertheless, selecting the length of the variant as the Y-axis attribute and depicting the rectangles proportional to the length of the variant would be redundant, that is the reason all the rectangles have the same size. In this context, small variants are represented by those rectangles depicted at the bottom of figure 5, whereas large variants are represented by those rectangles depicted at the top of figure 5.

ItLv provides an option to represent the number of variants in each element of a category both as a table and a graph (figure 6), the user can sort any of the columns either in ascending or descending order. In this example, the table and graph shows the number of variants for each size. These values are taken from the results of the collation depicted in figure 5. In this particular case, variants of length 3, 1, and 0 characters are on the three top cases. Variances of length zero mean that a variant was present in the base text but not in the comparison text, therefore a value of length zero is given.

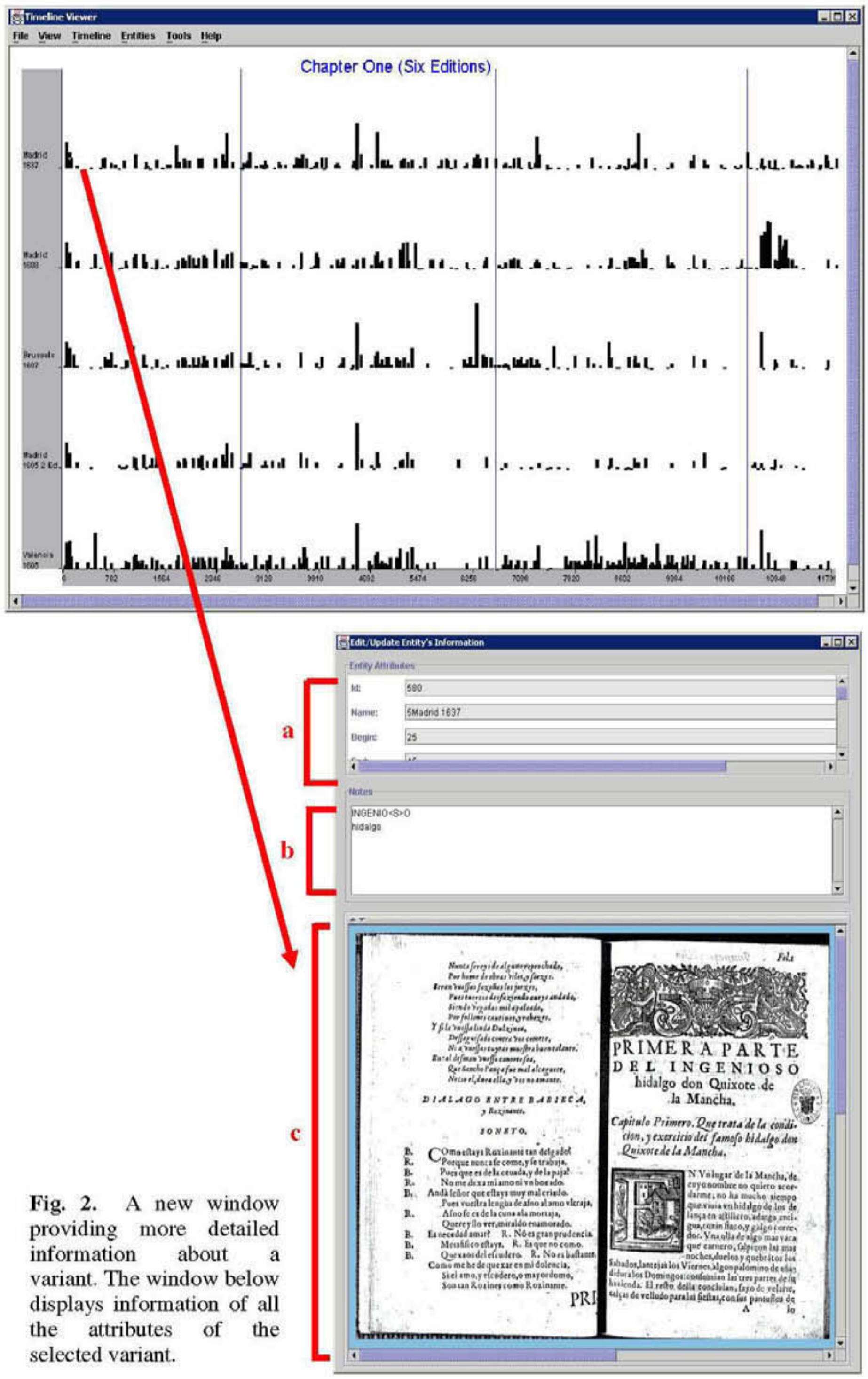


Fig. 2. A new window providing more detailed information about a variant. The window below displays information of all the attributes of the selected variant.

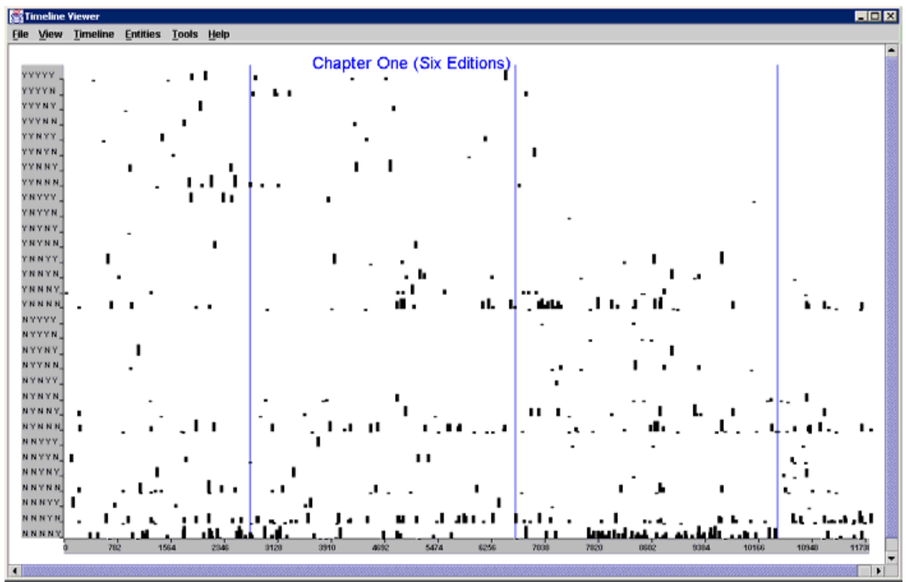


Fig. 3. A visualization of the variants categorized by the texts they appear on

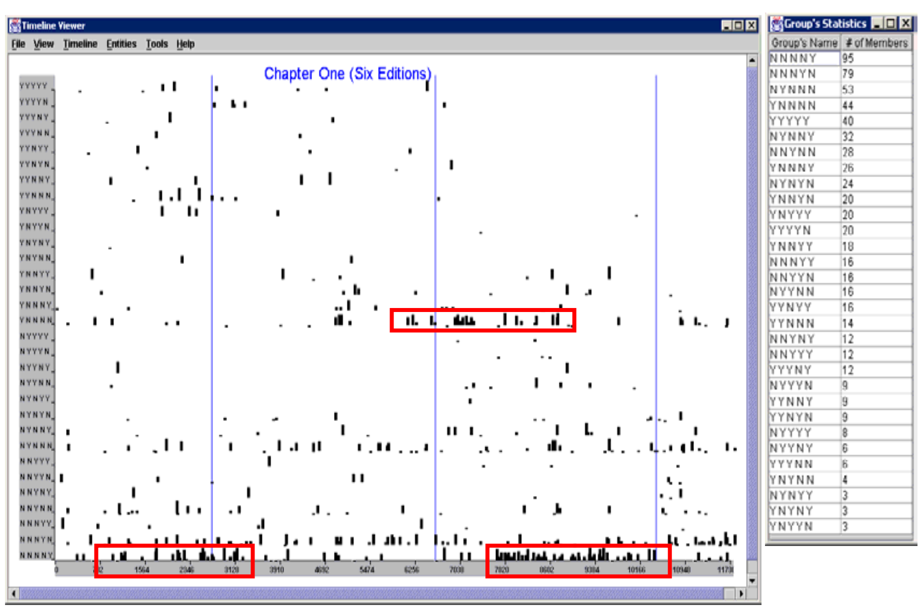


Fig. 4. A visualization of the variants categorized by the texts they appear on and some clusters of variants. The three horizontal rectangles were added manually to highlight three clusters of variants

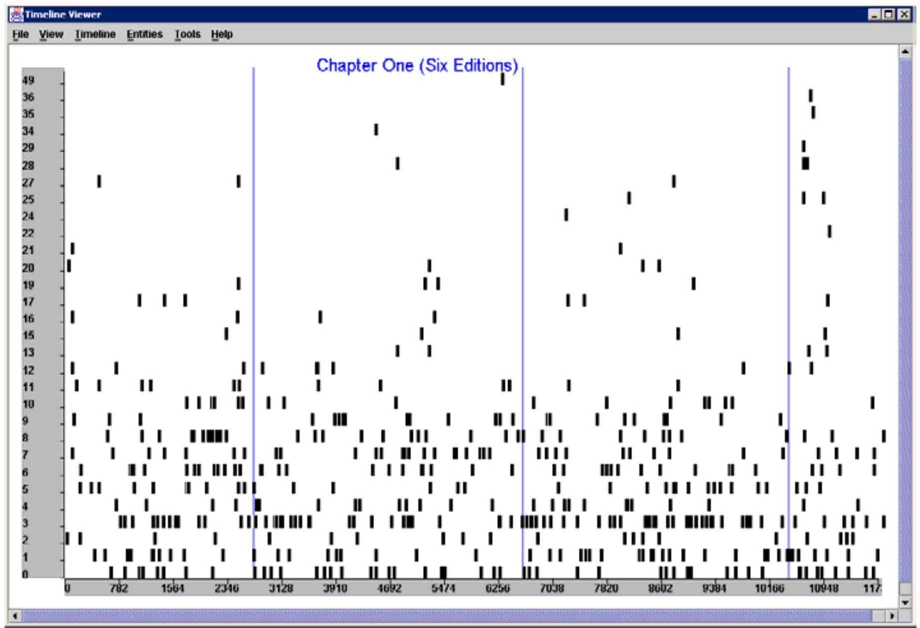


Fig. 5 Variants categorized by length in characters

Group's Statistics

Group's Name	# of Members
3	117
1	100
0	82
8	72
6	62
7	62
4	54
5	51
2	38
9	35
10	25

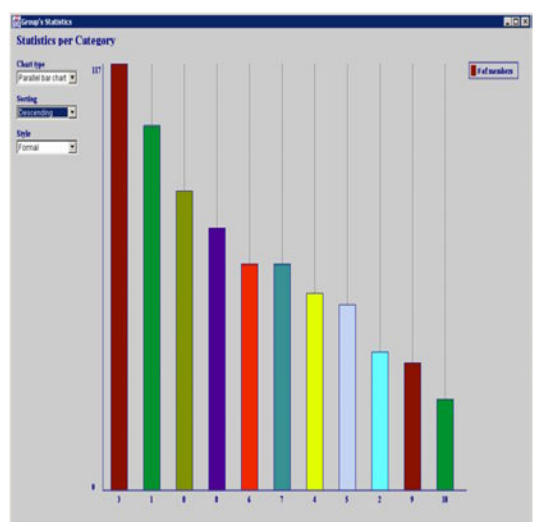


Fig. 6. A table and a graph displaying the number of variants of each size in the collation

In figure 7, the use of a pop-up window is presented. This pop-up window is activated when the mouse cursor is positioned over one of the variants. The information displayed in the pop-up window can be set in advance by the user, and modified at any time. In this example, the attributes selected are the offset where the variant begins in the text, the content of the variant, and the image of the page in the original book where the variant was found. Two lines, one horizontal and the other vertical show the context of the variant. All variants with some similarities are highlighted in a different color, in this particular case the variant is between the word "que" and the abbreviation q̃, which we encode in our transcriptions as "<q>".

Manipulating the dataset to explore different subsets can be achieved by using the filter option. This option enables users to filter variants based on a logical condition applied to any of the attributes. For example a user can specify that only entries containing a particular substring of characters should be displayed in order to observe whether that substring appears with some frequency across the texts. Figure 8 depicts those variants that include the abbreviation q̃. This shows that this abbreviation was used in page one and in part of page two. Therefore, we can raise the hypothesis that this abbreviation was the preference of the compositor in charge of these two pages.

Fig. 7. A pop up window displaying some attributes of a variant: a) offset, b) the variant, c) the length in characters, and d) the image of the text

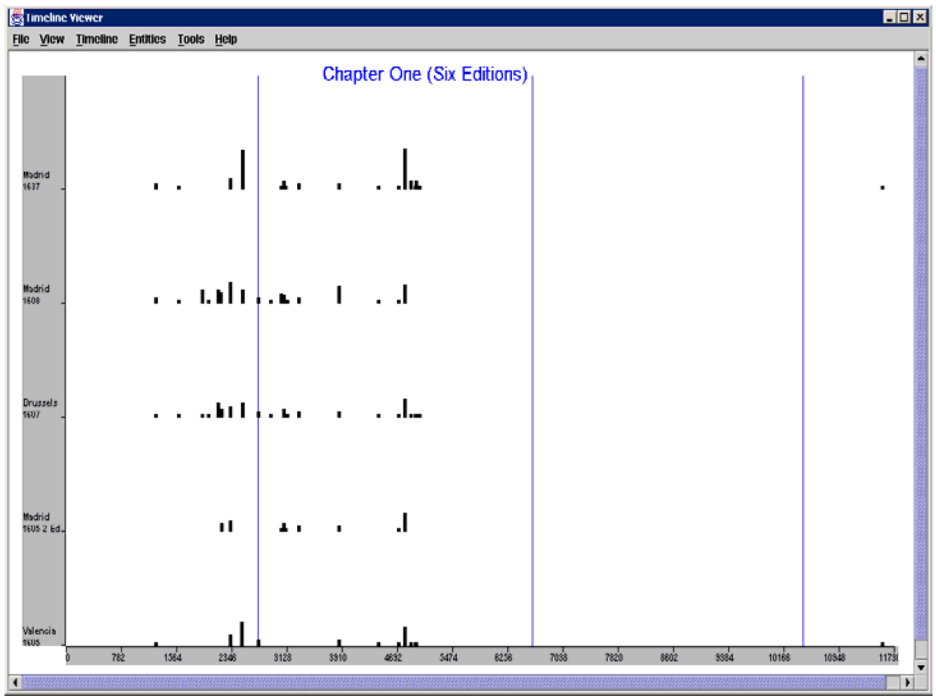


Fig 8. Depicting only those variants including the abbreviation q̃

As pointed out earlier, the ability to remove and add variants to the display is a very useful option provided by the ItLv. Another scenario could be for the user to remove those variants belonging to the category of punctuations, e.g., commas, semicolons, and so forth, prior to analyzing the results of a collation. By using the filter option, the user is able to hide those variants and focus on the rest. The variants can also be displayed again by removing the filter that has been applied.

When depicting the results of a collation, all variants are displayed in relation to the offset in the base text. Let us take texts A and B as an example, and let us assume that text A is the base text and text B is the comparison text. If a whole paragraph in text B appears in a different page than it appears in text A, those variants will be depicted relative to their offset in the base text (text A) not their offset in the comparison text (text B).

We have also used the ItLv to validate the results of the collation [12]. This enables users to analyze a dataset in which false variants have been either removed completely or minimized as much as possible from the display. In this context, we are using ItLv to investigate hypotheses about the origins of *Don Quixote's* early editions and their interrelationships.

3 Conclusions and Future Work

The main goal of this paper is to present the use of a visualization tool (ItLv) to depict the results of collating several copies of one text. Using ItLv's interactive features enables scholars to explore the similarities and differences among the texts in further detail. In the present case, ItLV is primarily for the use of domain experts in textual analysis, since it is applied to textual editing, and specifically to the analysis of textual variants in the creation of a variorum edition, as well as to present in visual format the results of comparing the texts included in the preparation of the edition.

ItLv as a visualization tool provides a flexible interface to visualize the differences among versions of a text, for example several copies of the same edition of a book or among different editions of a book. Interactive manipulations of the display provide both context and detail for the variants, for example by linking a variant to the image of the page in which it appears. Different visualizations of the same dataset provide different abstractions that can enable users to recognize and further understand the differences and similarities among texts.

We continue to explore the application of ItLv to visualization of the differences found within a collection of documents. Through this exploration, we hope to gain a better understanding of the changes that have been made among the editions, as well as to gather evidence of relationships between editions—for example, evidence about which earlier editions were consulted by the publishers of the later editions.

A preliminary use of ItLv showed good results as a visualization tool to explore the similarities and differences, and to identify patterns for a set of texts. However, when the number of chapters increases, we expect challenges such as how to identify patterns present in two non-contiguous chapters, especially if the chapters are far apart from one another. Along with the number of chapters is the need to provide a mechanism that enables the user to navigate through all the screens.

Presently, the image of the page where a variant appears is displayed, but there is no any indication of where in the image that variant appears. We are currently exploring the use of an additional visual aid to indicate the region in the image where the variant appears; this will help the user to find the variant in the image with more accuracy.

It is important to point out that in this paper we report the initial results of the use of ItLv. At this stage we have not performed extensive user testing or usability experiments. Presently, the preliminary experiments include small chapters only. We therefore, expect to start user testing soon in order to evaluate the user's experiences after incorporating longer chapters.

Acknowledgements

This material is based upon work supported by the National Science Foundation under grant no. IIS-0081420. Support for this study was also provided in part by the Interdisciplinary Research Initiative and the Telecommunications and Informatics Task Force Programs, administered by the Office of the Vice President for Research, Texas A&M University.

References

1. Baker, M., Eick, S., “Space-Filling Software Visualization.” Journal of Visual Languages and Computing. Vol. 6, No. 2, June 1995. pp. 119-133.
2. Bjork, S. “Hierarchical Flip Zooming: Enabling Parallel Exploration of Hierarchical Visualizations” Proceedings of the Working Conference on Advanced Visual Interfaces. Palermo, Italy 2000, pp. 232-237.
3. Bjork, S., Holmquist, L., “Exploring the Literary Web: The Digital Variants Browser.” Proceedings of Literature, Philology and Computers, Edinburgh, UK, 1998.
4. Furuta, R., Kalasapur, S., Kochumman, R., Urbina, E., Vivancos-Pérez, R., “The Cervantes Project: Steps to a Customizable and Interlinked On-line Electronic Variorum Edition Supporting Scholarship", Research and Advanced Technology for Digital Libraries: 5th European Conference, ECDL 2001, Darmstadt, Germany, September 2001, pp. 71-82.
5. Furuta, R., Hu, S., Kalasapur, S., Kochumman, R, Urbina, E., Vivancos-Pérez, R., “Towards an Electronic Variorum Edition of Don Quixote”, Proceedings of the first ACM/IEEE-CS joint conference on Digital Libraries, Roanoke, Virginia, 2001, pp. 444-445.
6. Kochumman, R., Monroy, C., Furuta, R., Goenka, A., Urbina, E., Melgoza, E. “Towards an Electronic Variorum Edition of Cervantes’ Don Quixote: Visualizations that support preparation”, Proceedings of the second ACM/IEEE-CS joint conference on Digital Libraries, Portland, Oregon, July 2002, pp. 199-200.
7. Kumar, V., Furuta, R., Allen, R., “Metadata Visualization for Digital Libraries: Interactive Timeline Editing and Review”, Proceedings of the third ACM conference on Digital Libraries, Pittsburgh, Pennsylvania, May 1998, pp. 126-123.
8. Maayan, G., Feitelson, D., “Hierarchical Indexing and Document Matching in BoW”, Proceedings of the first ACM/IEEE-CS joint conference on Digital Libraries, Roanoke, Virginia, 2001, pp. 259-267.
9. Marshall, C., Price, M., Golovchinsky, G., Schilit, B., “Designing e-Books for Legal Research”, Proceedings of the first ACM/IEEE-CS joint conference on Digital Libraries, Roanoke, Virginia, 2001, pp. 41-48.
10. Monroy, C., Kochumman, R., Furuta, R., Urbina, E., Melgoza, E., and Goenka, A., “Visualization of Variants in Textual Collations to Analyze the Evolution of Literary Works in The Cervantes Project”, Proceedings of the 6th European Conference on Research and Advanced Technology for Digital Libraries, Rome, Italy, September 2002, pp. 638-653.
11. Plaisant, C., Milash, B., Rose, A., Widoff, S., Shneiderman, B. “LifeLines: visualizing personal histories”, in Proceedings of CHI’96, Vancouver, BC, Canada, April 14-18, 1996, pp. 221-227.
12. “The Cervantes Project”, Center for the Study of Digital Libraries, Texas A&M University. http://www.csdl.tamu.edu/cervantes

An Empirical Evaluation of the Interactive Visualization of Metadata to Support Document Use

Mischa Weiss-Lijn[1], Janet T. McDonnell[1], and Leslie James[2]

[1]Department of Computer Science,
University College London,
London WC1E 6BT, UK
{m.weiss-lijn, j.mcdonnell}@cs.ucl.ac.uk
[2] J Sainsbury plc, London UK

Abstract. Digital Libraries currently focus on delivering documents. Since information needs are often satisfied at the sub-document level, digital libraries should explore ways to support document use as well as retrieval. This paper describes the design and initial evaluation of a technology being developed for document use. It uses interactive visualization of paragraph level metadata to allow rapid goal-directed search and navigation within documents. An experimental evaluation of a prototype's performance on representative work tasks is described. Quantitative analysis finds that the prototype does not increase performance. However, qualitative analysis of the data suggests that there is room for performance improvements and has inspired design changes to realize this potential.

By regarding a set of relevant documents as the final product of digital library use, digital libraries may be stopping short of their ultimate mission. As well as enabling people to get to relevant documents, digital libraries should support users in getting to the parts of documents they need. We describe a new technology developed to enable information seeking within documents on a corporate intranet. The resulting tool, named GridVis, uses interactive visualization of metadata to support goal-directed document use. An empirical evaluation of this tool is then reported; its implications for design improvements, further evaluation and the utility of GridVis are discussed.

Although digital libraries generally focus on providing documents, their goal is actually to support the information needs of communities of users [2]. In a study of the needs of digital library users Van House [12] states that: "Workplace users […] want to retrieve information rather than documents per se.". The work of Adler et al [1] looked at the reading tasks performed by people in a variety of work settings and found that rapid goal-directed types were very prominent. In goal-directed reading, users only make use of small portions of documents. Hence, it seems that in both document and digital library use, information needs seem often to be met at the sub-documents level. Yet, apart from a few interesting exceptions (e.g. [10], [6], [5], [7]), digital libraries and the visualization tools designed for them, do not provide support for information seeking within documents.

K. Börner and C. Chen (Eds.): Visual Interfaces to Digital Libraries, LNCS 2539, pp. 50–64, 2002.

System Description

The development of GridVis was motivated by Sainsbury's, a large UK retail organization that sponsors this research. As such, the ideas behind the technology and the resulting application have been developed for a specific corporate environment in an iterative fashion with the close involvement of potential users.

In this section, the documents visualized and the metadata created for them are discussed. The visualization's design is described, and its use in goal directed search is illustrated. The construction and design of GridVis are more fully described in [15].

The Documents and Their Metadata

GridVis was envisaged for use with a small but important portion of a diverse corporate document landscape, which can be characterized as being information rich and having a relatively long life. A set of documents with these characteristics was selected (as described in the evaluation section) to serve as a basis for developing a metadata taxonomy and conducting evaluation.

GridVis parses the document, which has been marked up with metadata, and a separate file specifying the metadata taxonomy. It uses the result to produce a visualization and customized HTML views of the document text. The metadata taxonomy is necessary for several reasons, only some of which can be fully described here. For the users of GridVis undertaking a goal-directed search, it is essential that the metadata has some meaningful organization. With a good taxonomy the user can rapidly locate, or confirm the absence of, the metadata they need by identifying the sub-set relevant to their information need, rather then undertaking an exhaustive linear search through all the metadata tags. Other benefits of the metadata taxonomy are that it reduces the impact of increasing the amount of metadata, it may aid metadata authoring [15], and permit the use of more powerful search algorithms.

Since GridVis is designed for use on a corporate intranet in a corporation where software installation for low priority applications is problematic it was given a client-server architecture: the visualization is a client-side Java applet, the queries are answered by a Java servlet using XLS-T to produce customized HTML documents. The documents and taxonomy are encoded as XML. The document structure is described with a standard hierarchical XML structure; the metadata is embedded within this alongside the text it describes.

The Visualization

The GridVis client-side application contains a visualization of a document's metadata. This visualization can be considered as three interlinked sections; the metadata tree on the left-hand side, the iconic document overview running along the top, and the grid sitting at the center (see Fig. 1). The metadata tree consists of the document's metadata tags and the metadata taxonomy. The taxonomy is used to create a tree structure and the metadata tags are added to this as leaf nodes. Each tag corresponds to one row in the central grid. The iconic document overview is a miniature image of the document laid out horizontally along the top of the visualization. Each paragraph in this overview lines up with one of the columns in the grid below. The central grid shows which metadata tags have been applied to which paragraphs in the document; each column represents a paragraph and each row represents a tag. Hence, to show that a tag has been applied to a particular paragraph the cell where the appropriate column and row meet is shaded. The color of the cell is determined by the value of the applicability attribute given to the tag when it was applied to this paragraph; the higher the applicability the darker the color (a legend for this scale is always visible in the top left hand corner). The metadata tags and taxonomy, iconic document overview and grid, are thus brought together to produce a visual overview of the document.

The user can query the visualization with their mouse through a combination of dynamic querying and brushing [3]. When the mouse is placed over a cell in the grid all the tags in the column are highlighted; hence the paragraph's content is described. By looking along the row of an interesting tag, the user can see where and to what extent that topic comes up in the document. So by moving the mouse in the central grid the user can explore what different parts of the document are about, and where in the document particular topics are covered.

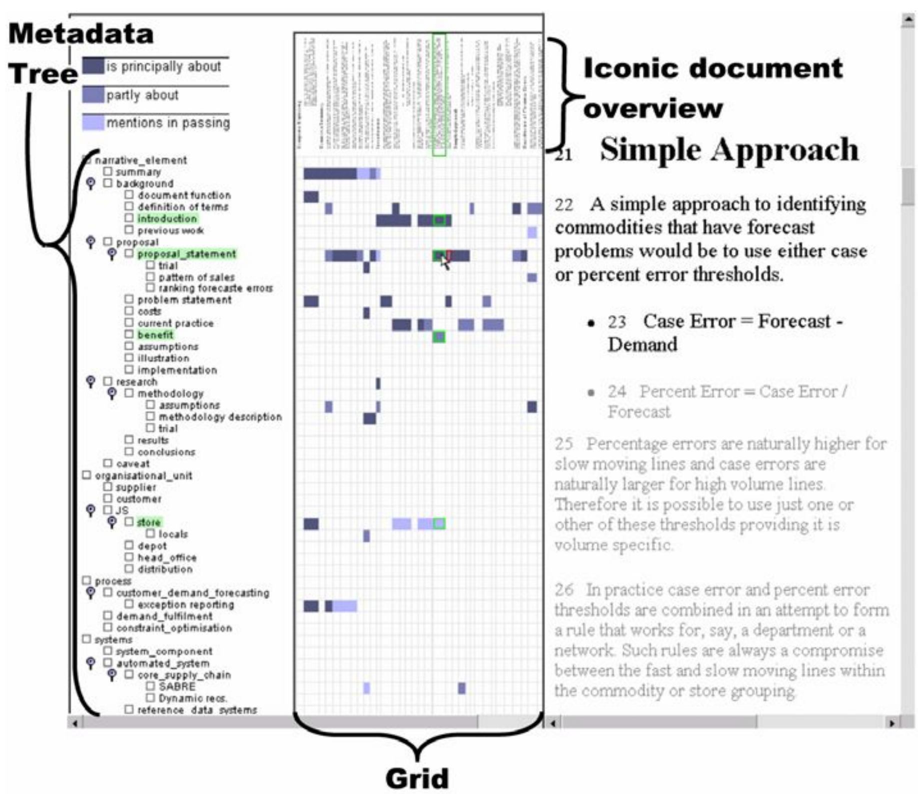


Fig. 1. Screenshot of GridVis

The user can also query the document by clicking on any cell in the grid. Clicking on the grid will produce a query that is sent to the servlet formulated with the metadata and paragraph corresponding to the selected cell. The servlet will produce an HTML version of the document in which every paragraph tagged with the selected metadata is highlighted using a bold font. The browser will display the paragraph selected; thereby offering details on demand [3].

An Illustrative Scenario

Scenarios [4] have been found to be effective design and communication tools. Consequently we have included the scenario below to give a better idea of how the functionality described above might be used in practice.

Clair, a Sainsbury's (JS) employee, sits down to look at this week’s 'industry update' newsletter, with a view to finding out whether the company plans to sell its US interests. She starts up Gridvis and looks through the metadata to see if any of JS's US subsidiaries are mentioned. She scans down the top-level entries in the metadata taxonomy until she reaches 'organizational unit'. Then she scans the next level of entries under 'organizational unit' until she reaches 'JS'. Under JS she sees an entry for 'Shaws' which is a US subsidiary of JS.

Having found a relevant tag, Clair proceeds to see if any of the paragraphs relevant to this tag will say anything about plans for a sell off. She looks along the tag's row in the grid and sees that two cells have been colored in. Passing her mouse over the first she sees the other colored cell in the same column and its tag become highlighted. The other cell’s tag label reads 'loyalty cards'. Seeing that therefore the paragraph is not relevant she moves her mouse on to the next colored in cell. In this case the 'acquisitions' and 'USA' tags light up. This surprises Clair, since she expected Shaws to be on sale, not making acquisitions; she decides to look at the paragraph and clicks on the cell. The right hand window is refreshed, with the paragraph she is interested in appearing at the top of the screen in bold.

Experimental Evaluation

This section describes an experiment which attempts to provide an initial empirical assessment of GridVis's utility for goal-directed search. This small controlled experiment looked at the effect of a fully functional interactive prototype of GridVis on goal-directed search performance and gathered data for use in improving the design.

The experiment required participants to perform goal-directed search with and without GridVis. Twelve participants were each presented with four documents of varying lengths. For each document, each participant was asked to rapidly find the paragraphs containing information relevant to three questions. For two of the documents each participant was asked to use GridVis, for the other two they simply used the scroll bar to navigate through the document. Performance was measured in terms of task time and analogues of precision and recall adapted to interactive retrieval.

Method

The Sample

The twelve participants in this experiment worked in the Sainsbury's logistics and planning department. This is the department in which, and for which, the documents used in this experiment were written. The participants were familiar with the basic concepts dealt with by the documents and had not previously read any of the documents used in the experiment, nor seen GridVis or the metadata taxonomy.

The Documents

Five analytical reports written in the company were selected for use in the experiment. Document length was varied to see whether the performance advantage or disadvantage conferred by GridVis would depend on document length. Two of the documents were long (8-9pgs) and two were short (5-4pgs). An earlier exploratory study found the length of the majority of documents used at Sainsbury's to be in this range. All five documents were tagged with metadata as described in [15].

Three questions for each document were elicited from the original readership for the practice document and each of four remaining documents, giving 12 non-practice questions. This ensured the questions used, and their wording, was representative of those occurring in real work situations.

Judge Relevance Ratings

Retrieval performance was measured by comparing the paragraph relevance ratings made by the participants to an aggregation of the ratings made by expert judges.

The relevance judgments were created using a procedure based on that used for the NTCIR information retrieval workshops [11]. A four-level relevance scale was used since this has already been used successfully at NTCIR, and can deliver high definition information about relevance judgments. It was also decided to use three judges for each document. In each case one of the judges was the document's author and the other two were participants in the experiment.

The procedure used to create a single set of relevance judgments used a simple algorithm based on an analysis of the variation between judge ratings. The analysis found that a 36% of paragraphs rated as relevant had no clear relation to the query. An aggregation algorithm was devised which used objective data (the judge ratings and presence of corroboration between judges) to exclude the majority of ratings to paragraphs which were actually irrelevant while only excluding a minority of the rating to actually relevant paragraphs.

Experimental Design

The goal-directed search task at the heart of the experiment requires the participant to locate, as rapidly as possible, all the paragraphs in a document that are relevant to a given question. They start by reading the document's title and management summary, and then launch into the document using either GridVis or just a browser window. Once the participant locates a paragraph they consider relevant they rate it for relevance using a four point scale, with unrated paragraphs being assigned a rating of to 0. They used a web form to enter the paragraph's ID number (which was listed alongside each paragraph) and its relevance rating. Once the participant felt that they had enough information to answer the question, or that there were no more relevant paragraphs left, they used a HTML form button to move onto the next question.

The experiment began with the participant being given a short list of written instructions describing the experiment and the goal-directed search task. The participant was then shown a training document visualized in GridVis. GridVis's functions were explained by the experimenter. The participant was shown how the metadata taxonomy could be used for locating relevant tags, and how paragraphs' cells could be chosen using co-occurrence and applicability. The participants familiarized themselves with GridVis and the goal-directed search task by using GridVis with the training document to find the paragraphs relevant to the three questions.

Once the participant had completed the familiarization run, they moved onto the experimental trials. In these, the participants were asked to repeat the goal-directed search task for three questions on each of the four other documents. For two of the documents the participant was asked to use GridVis, for the other two they simply used the scroll bar. When using GridVis, the participants were asked to only make minimal use of the document window's scroll bar and thus rely on GridVis for navigation and search. Order effects were minimized by counterbalancing for the different documents and the with and without GridVis conditions.

Upon completing the experimental condition, each participant was briefly interviewed about GridVis. They were asked about the problems or difficulties they encountered, and questioned on their views of the tool's usefulness.

Data Capture

Data from the experiment was captured using screen recording, interaction log files, and audio recordings. Detail of users' interactions were preserved through a continuous screen recording of the entire experimental session, using HyperCam [9]. Every action (i.e. when the participants clicked on a cell in the grid) for each document and question was automatically logged in a text file. Participants were asked to record the paragraph numbers of germane paragraphs and to rate these paragraphs for relevance. The participant entered this information into an online form, which transmitted the information to a Java servlet that recorded the information in a text file. The audio for the entire session was recorded.

Quantitative Results

A quantitative analysis looked for performance differences using the standard information retrieval metrics of recall, precision and task time. A qualitative analysis was then undertaken to examine the strategies used with GridVis and their impact on task performance.

Analysis

In order to measure retrieval performance interactive recall and precision metrics were employed. These are close analogues of the traditional recall and precision metrics [13] developed for use in the interactive track of the TREC conferences [14]. The traditional precision metric was also used in the qualitative analyses when the data were not available to calculate interactive precision. These metrics are defined as follows:

$$\text{Interactive Recall} = \frac{\text{Number of relevant paragraphs found by the user}}{\text{Total number of relevant paragraphs}}$$

$$\text{Interactive Precision} = \frac{\text{Number of relevant paragraphs found by the user}}{\text{Total number of paragraphs deemed relevant by the user}}$$

Differences between conditions were tested for statistical significance using the non-parametric Wilcoxon test [8] for related pairs.

Retrieval Performance Results

The time taken to perform the goal-directed search for a question was on average significantly longer in the 'with GridVis' condition. The mean time taken with GridVis was 168 seconds, while without GridVis the time was 131 seconds. The difference was significant to $p<0.05$, ($Z = -2.040$).

The mean interactive precision and recall calculated using expert judge relevance ratings show little difference between the 'with GridVis' and 'without GridVis' conditions (0.67 vs. 0.60 for precision $p>0.1$, $Z= -.756$, and 0.17 vs. 0.16 for recall $p>0.1$, $Z= -.471$).

Document length had a limited effect on performance. Essentially, document length only significantly affects interactive recall in the 'with GridVis' condition, where interactive recall for the short documents (0.26) is higher than for the long ones (0.1)($p<0.05$, $Z = -2.13$). It was hoped that GridVis would enjoy relative independence to document length. In reality, it seems, document length has a more deleterious effect on performance with GridVis than without it.

Qualitative Results

Here we discuss qualitative analyses; firstly we examine, the tag selection strategies used, secondly the paragraph cell selection strategies, and finally the utility of co-occurrence information.

Investigating Strategy Use

A qualitative analysis of log and video data was undertaken to determine why GridVis did not produce a large performance improvement over the 'without GridVis' condition and to discover what improvements might be possible.

Analysis Procedure

This analysis started by using a task analysis (described in [15]) to look at what strategies the participants might be using to do goal-directed search with GridVis. It divided the use of GridVis into two components. Firstly, the user has to find a relevant metadata tag, then they must select from amongst the paragraphs relevant to this tag. The task analysis also suggested the strategies that might be used to make these selections.

This initial set of strategies, taken from the task analysis, was used as a starting point for classifying each tag and paragraph selection action based on the video data and text log files. The classification was refined, using the first few subjects, until it could account for 96% of valid tag selections and 88% of paragraph selections. The final classification of strategies is described in Table 1 and Table 2.

Table 1. The tag selection strategies

Strategy name	Description of strategy
Keyword mapping	A tag is selected because it matches a keyword from the question
Synonym mapping	A tag is selected because it matches a synonym of a keyword from the question
Related concept keyword mapping	A tag is selected because it deals with concepts related to those in the question
Narrative element mapping	A tag is selected because it refers to part of the documents structure (i.e. ‘conclusion’) which is likely to contain content related to the question
Unrelated	Not relevant to the question in hand
Unexplained strategy	None of above strategies apply

Table 2. The paragraph cell selection strategies

Strategy name	Description of strategy
Left to right	Paragraph cells are selected in left to right order
Applicability only	Paragraph cells are selected in order of their applicability
Co-occurrence only	The relevance of the tags co-occurring on each cell is evaluated. The paragraph cells are selected in order of the relevance of their co-occurring tags.
Applicability and co-occurrence	The relevance of the tags co-occurring on each cell is evaluated. The paragraph cells are selected in order of the relevance of their co-occurring tags. Given cells with similarly relevant co-occurring tags, the cell with the highest applicability is selected.
Unexplained strategy	None of above strategies apply

For each tag selection, if a tag could have been chosen using a strategy then the strategy was given a rating of 1, otherwise it received a rating of 0.

For every paragraph cell selection, evidence from the log files and video data was used to assign scores to the selection strategies. The log files were used to compare the order of cell selections made on a tag, with that which would be predicted given the use of a particular cell selection strategy. For each strategy, a score of 1 was assigned if the order of selections recorded in the log file, conformed to the predicted order and a score of 0 was given if they did not. Separate scores were made using the video data. Use of the mouse to highlight and read co-occurring tags, resulted in a score of 1 for the co-occurrence strategies, absence of this resulted in a score of 0. If the mouse was brushed over several candidate cells, this indicated that attention was being given to applicability, so a score of 1 was given to the applicability strategies.

This procedure yielded separate sets of scores for the selection order and video evidence; these then had to be merged together. This was done with a simple algorithm that gave priority to the positive scores derived from the video evidence.

A final transformation was applied to the tag and paragraph cell selection strategy scores. This was necessary to avoid ambiguous selections being over represented because they contribute scores to several strategies. A transformation was used to ensure each selection made the same contribution to the overall scores for each strategy. Each score was divided by the sum of the scores given to all the alternative strategies for that selection.

Tag Selection Strategies

The overall prevalence of different tag selection strategies is shown by Fig. 2. The 'narrative element mapping' and 'keyword mapping' strategies are the most prevalent, with 'related concept keyword mapping' coming next and leaving 'synonym mapping' a poor fourth. Ten percent of the tags selected were deemed to be 'not related' to the question in hand. These are likely to have been due to users selecting cells on rows that were not actually the same as those of the tags they had identified as relevant. A further four percent of tag selections could not be assigned a strategy.

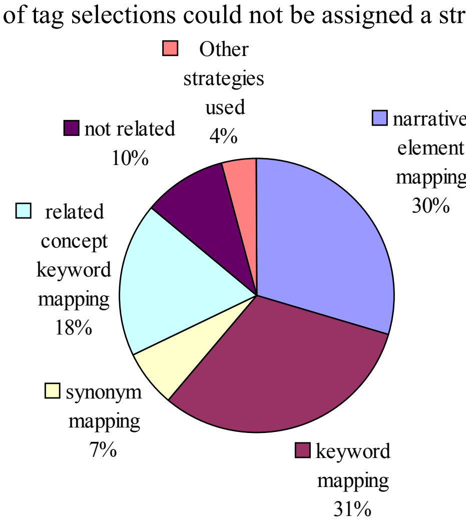


Fig. 2. The mean scores for each tag selection strategy across all participants, show as percentage of total score.

The results in Fig 2 suggest that metadata was worded appropriately. Synonyms only have to be used when the questions use a different vocabulary from that used in the metadata. Since the prevalence of the 'synonym mapping' strategy is low and the question wording came from Sainsbury's staff, the metadata must have generally used the same wording as the questions.

There were large individual differences so the pattern of preferences shown in Fig. 2 does not apply equally to all the participants. An analysis of these differences did not however yield any insights of relevance to evaluation of GridVis.

Paragraph Cell Selection

The overall prevalence of different paragraph cell selection strategies is shown by Fig. 3. It shows that the 'left to right' and 'applicability only' strategies are both twice as prevalent as the 'co-occurrence only' strategy, which was itself three times as popular as the ‘co-occurrence and applicability strategy'.

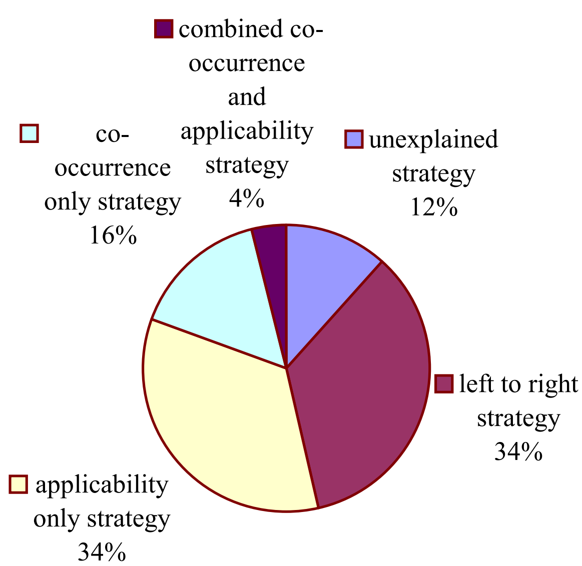


Fig. 3. The mean scores for each paragraph cell selection strategy across all participants, shown as percentage of total score.

As with the tag selection strategies there is much variation across participants. There were two main configurations of strategies; ‘no co-occurrence use’ and ‘some co-occurrence use’. The three participants with a ‘no co-occurrence use’ configuration relied primarily on the 'left to right' strategy while making some use of applicability. The six participants with the ‘some co-occurrence use’ configuration, however, make some use of the 'co-occurrence only' strategy although the 'left to right' strategy is just as popular and 'applicability only' strategy is the one most relied upon.

An interesting aspect of these results concerns what they say about the extent to which applicability and co-occurrence information was used. Overall, applicability was used about 38% of the time, and co-occurrence was used 20% of the time (see Fig. 3). Even in the 'some co-occurrence use' group the co-occurrence information was only used in 27% of selections, while applicability was used in 47%. Participants therefore, made relatively little use of co-occurrence information in choosing paragraph cells. Perhaps the poor search performance seen with GridVis is partly due to this tendency to not use applicability and more importantly co-occurrence information.

This hypothesis that the sparse use of co-occurrence explains the low performance was examined by comparing the performance of the 'no co-occurrence use' and 'some co-occurrence use' groups. If co-occurrence information leads to better performance one would expect the group that use it to have higher interactive recall and interactive precision, but lower task time. Contrary to this expectation, although precision and recall were slightly higher, task time was actually higher for the co-occurrence strategy. Nevertheless, none of these differences were statistically significant ($p>0.1$). Either the use of co-occurrence makes no difference to performance or it increases precision and recall at the cost of increasing task time. This suggests that the sparse use of co-occurrence might explain the low recall and precision, but not the bad task time performance.

This ambiguous result, along with the evidence that users made little of co-occurrence information prompted further analysis. Subjective experience suggests that co-occurrence is a powerful way of discriminating relevant from irrelevant material. It was therefore surprising that such little use was made of it, and that it might not result in any performance increase. Further investigation was needed to understand why the co-occurrence information was not being used.

Co-occurrence information could be problematic by being misleading, or by taking too much time to exploit; the data from this experiment only allows the former possibility to be examined. The co-occurrence information could be inherently useless and even misleading by not clearly distinguishing the relevant from irrelevant paragraph cells. The use of co-occurrence could also be a time consuming distraction from more fruitful activities. Unfortunately, it was not possible to examine the implications of co-occurrence's time cost here. However, it was possible to examine the effect of the co-occurrence information retrieval effectiveness in terms of traditional precision and interactive recall. This investigation and its results are described in the next section.

Investigating the Use of Co-occurrence Information in Cell Selection

This section presents an investigation into whether the use of co-occurrence in paragraph cell selection can increase retrieval effectiveness. This was done by exhaustively applying the 'co-occurrence only' and 'co-occurrence and applicability' strategies and comparing the interactive recall and traditional precision achieved with them with that achieved using the other two strategies.

Analysis Procedure

In this investigation, the strategies identified in the previous section were systematically applied each to of the 12 questions used in the experiment. For each question every relevant tag was identified. This was done by exhaustively applying each of the four tag selection strategies in order of their overall popularity thereby picking out the tags that each strategy deemed relevant.

For each of these relevant tags, the paragraph cell selection strategies were used to produce a separate ranking of the cells linked to the tag. These rankings were based solely on the strategy being used; so cells indistinguishable on the basis of that strategy were given the same ranking. Paragraph cells that had already been selected on a previous tag, were skipped.

Since the strategies could only be differentiated on their ranking of paragraphs the strategies needed to be compared by assuming the selection of a certain proportion of the top ranked cells. It was decided to compare the top ranked third of paragraphs. The interactive recall and traditional precision achieved with this group of paragraphs for each strategy was then calculated. Traditional precision is calculated as:

$$\text{Traditional Precision} = \frac{\text{Number of actually relevant paragraphs found by the user}}{\text{Total number of paragraphs deemed relevant by the system}}$$

Traditional precision can be applied here, if one considers that the tool being used makes paragraphs available via affordances for avoiding irrelevant text. Hence, traditional precision can be calculated by substituting the 'Total number of paragraphs inspected' for 'Total number of paragraphs deemed relevant by the system'.

Results - an Exploration of the Effect of Paragraph Cell Selection Strategy

This section reports the results of the exhaustive application of the tag and cell selection strategies. The key question examined is whether the use of co-occurrence information affords benefits in terms of interactive recall and traditional precision. Interactive precision was not used because this analysis did not provide data on user relevance judgments.

The use of co-occurrence offers a slight advantage over the use of applicability alone. This advantage is about 10% for both traditional precision and interactive recall. It suggests that co-occurrence information can indeed offer an improvement in information retrieval performance. Nevertheless, the difference is small. This raises the question of whether it is indeed a noteworthy difference. Unfortunately, since the result for each strategy is taken by pooling the data from all twelve queries and four documents, only one data point is produced per strategy and thus cannot be subjected to statistical analysis. Hence, although this analysis suggests that co-occurrence may offer an advantage, the advantage is small and may not be reliable.

Finally, this investigation permitted some light to be shed on the potential of the retrieval performance that a user of GridVis might achieve. If the user were to read all the paragraphs related to the tags selected in this analysis they would achieve a interactive recall of 0.65 and a traditional precision of 0.2. This gives an indication of the maximum interactive recall that could be achieved. The application of the cell selection strategies should then improve the precision while having a minimal impact on interactive recall. If the user were to use the paragraph selection strategies to choose a third of the paragraphs, then, at best, interactive recall drops to 0.52, and precision climbs to 0.32. This performance compares very favorably to the interactive recall of 0.16, found in the empirical experiment's 'without GridVis' condition. Overall, this analysis suggests that if used diligently, retrieval performance with GridVis would be substantially increased relative to the performance observed by participants in this experiment.

Discussion

The quantitative results suggest that performance with GridVis is broadly similar to that without it. It is only on task time where performance with GridVis is actually significantly worse than when simply using a browser window.

A qualitative analysis examined the strategies used with GridVis in order to locate any design problems that might be limiting performance. This analysis found that there was little use of co-occurrence information. A separate investigation showed that making use of this information could increase interactive recall and traditional precision scores by around 10%. Nevertheless, the data suggest that participants who made some use of co-occurrence might have higher task times. This could be a consequence of investing extra time in the use of co-occurrence information. Further work should be directed at examining this possibility.

The results of the investigation, into the potential of co-occurrence information in cell selection, suggest that GridVis has the potential to deliver high retrieval performance. This showed that the diligent application of the tag selection strategies in the use of GridVis could result in much better interactive recall performance than that achieved by participants in the experiment. Nevertheless, this investigation does not take into account the time cost incurred in pursuit of this performance. As such, the high interactive recall performance achieved can only be considered as an upper bound. However, this suggests that in principle GridVis should be able to offer better performance than demonstrated in this experiment. Further evaluation will more closely examine GridVis's potential taking both time spent and information gained into account. However, if GridVis can potentially offer high performance why was this not apparent in the quantitative results?

There are a number of possible explanations for the poor performance observed with GridVis. Firstly, there is the matter of the users' expertise with GridVis. Although participants claimed to be comfortable with the use of GridVis by the end of the training session, they were still essentially novice users of the interface. It is possible that performance would be much improved by greater familiarity with the structure of the ontology and visualization techniques used. Secondly, it is not clear to what extent performance was impaired by shortcomings of the prototype's design and implementation. For example, the substantial number of repeat visits to paragraphs could be blamed on the inadequacies history visualization feature. The failure to investigate relevant tags might partly be attributed to the relatively small font size, and the lack of visual features, other than indentation, used in bringing out the structure of the metadata taxonomy (see Fig. 1). The 10% of tag selections which seemed irrelevant could be explained by a difficulty in selecting the correct row in the grid for a particular tag. This problem could be tackled by encouraging the user to first select the tag of interest so that the corresponding cells are highlighted and by highlighting rows as well as cells.

Finally, the qualitative analysis suggested that co-occurrence information was underused. The analysis of the paragraph cell selection strategies used, showed that relatively little use was made of the co-occurrence strategy. However, the investigation, into the potential of co-occurrence information, showed that making use of co-occurrence could increase performance.

Why might co-occurrence information been underused? Lack of familiarly and expertise might be one reason. A more important problem is that the cost, in terms time and effort, of using the co-occurrence strategies might be too high. Moreover, the application design makes using co-occurrence information particularly difficult and time consuming in some circumstances. For example, if there is a set of interesting tags, the user might want to find paragraphs relevant to them all. This requires the user to scan along the row of each interesting tag to find colored cells. Upon arriving at each of these, the user must then scan vertically to the check if the cells relating to the other tags of interest are colored in.

Another possible factor inhibiting the use of both applicability and co-occurrence information is GridVis' voracious appetite for screen real estate. With longer documents, the users have to scroll horizontally to see the whole grid. This makes it impossible to see every paragraph relevant to a tag and therefore confounds attempts to select the most appropriate paragraph.

Further development work on GridVis has addressed these problems by improving details of the design and adding new functionality [15]. One new piece of functionality makes the co-occurrence strategy easier by offering partial automation; if the user selects tags of interest, GridVis indicates the number of tags that co-occur on each paragraph. Another addresses the screen real estate problem by dynamically redrawing the grid at varying levels of detail to only show in full the parts of the document relevant to the user's current activity.

So to conclude, GridVis demonstrates how interactive visualization of paragraph level metadata can support goal-directed search within single documents. In an empirical evaluation, GridVis was not found to afford a performance advantage over a scrolling window. However, a qualitative analysis suggests that GridVis has the potential to offer better performance. Further work is being undertaken to qualify the applications true potential using a 'best-case' method based on cognitive modeling.

Acknowledgements

This research was sponsored by J Sainsbury plc and undertaken within the Postgraduate Training Partnership established between Sira Ltd and University College London. Postgraduate training partnerships are a joint initiative of the DTI and the EPSRC. The first author would also like to thank Dr John Gilby for his wise counsel and Anne Poméry for emotional support.

References

1. Adler, A., Gujar, A., Harrison, B. L., O'Hara, K., and Sellen, A. A diary study of work-related reading: design implications for digital reading devices. 241-248 1998. Conf. Proc. On Human factors in computing systems.
2. Borgman, C. L. Final Report, UCLA-NSF, Social Aspects of Digital Libraries Workshop. 1996. Social Aspects of Digital Libraries Workshop. http://is.gseis.ucla.edu/research/dl/UCLA_DL_Report.html.
3. Card, S. K., Mackinlay, J. D., and Shneiderman, B. Information visualization: Using vision to think. 1999. Morgan Kaufmann.
4. Carroll, J. M. Five reasons for scenario based design. Interacting with Computers 13, 43-60 2000.

5. Egan, D. E., Remde, J. R., Gomez, L. M., Landauer, T. K., Eberhardt, J., and Lochbaum, C. C. Formative design evaluation of SuperBook. ACM Transactions on Information Systems 7(1), 30-57 1989.
6. Hearst, M. TileBars: Visualization of term distribution information in full text information access. 59-66 1995. Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems. Denver, CO, ACM Press.
7. Hornbæk, K. and Frøkjær, E. Reading Electronic Documents: The Usability of Linear, Fisheye, and Overview+Detail Interfaces. 293-300 2001. Proceedings of CHI'2001.
8. Howell, D. C. Statistical methods for psychology. 1997. Wadsworth publishing company.
9. Hyperpionics. HyperCam. 2001. http://www.hypercam.com/.
10. Janney, K. and Sledge, J. A user model for CIMI Z39.50 application profile. CHIO 9/1995. www.cimi.org/old_site/documents/Z3950_app_profile_0995.html.
11. Kando, N. NTCIR Workshop: Japanese- and Chinese-English cross-lingual information retrieval and multi-grade relevance judgments. 24-33 2001. Lecture Notes in Computer Science; 2069. Lisbon, Portugal, Springer.
12. Van House, N. A. User needs assessment and evaluation for the UC Berkeley electronic environmental library project. 1995. Digital Libraries '95: The 2nd Conference on the Theory and Practice of Digital Libraries. Austin, TX.
13. Van Rijsbergen, C. J. Information Retrieval. edition 2 1979. London, Butterworths.
14. Veerasamy, A. and Heikes, R. Effectiveness of a Graphical Display of retrieval results. 1997. SIGIR 97. ACM Press.
15. Weiss-Lijn, M., McDonnell, J. T., and James, L. Interactive visualization of paragraph-level metadata to support corporate document use. Visualising the Semantic Web. 11/2002. (Eds). Geroimenko, V. and Chaomei, C. Springer Verlag.

Visual Analysis of Website Browsing Patterns

Stephen G. Eick

Visintuit, 1413 Durness Court, Suite 100
Naperville, IL 60565
eick@acm.com

Abstract. The explosive growth on-line activity has established the e-channel as a critical component of how institutions interact with their customers. One of the unique aspects of this channel is the rich instrumentation where it is literally possible to capture every visit, click, page view, purchasing decision, and other fine-grained details describing visitor browsing patterns. The problem is that the huge volume of relevant data overwhelms conventional analysis tools. To overcome this problem, we have deve loped a sequence of novel metaphors for visualizing website structure, paths and flow through the site, and website activity. The useful aspect of our tools is that they provide a rich visual interactive workspace for performing ad hoc analysis, discovering patterns, and identifying correlations that are impossible to find with traditional non-visual tools.

1 Introduction

The web is the ultimate "digital library" as vast amounts of material are becoming available on-line. Better techniques to improve access and understand on-line behavior are desparately needed. Many sites are difficult to navigate, hard to use, and have confusing structure. Examples abound because the phenomenon is so widespread. Users may become lost, and they are forced to make "large leaps" within the site, e.g. returning to the home page, because they do not understand the site structure. The result is that they may be unable to find content and may prematurely abandon the site.

Obviously site authors do not frustrate their users intentionally. It is extremely difficult to create easy-to-use sites. A site structure that seems intuitive to the author may, in fact, be highly confusing to everyone else. Furthermore, there is no mechanism for site operators to understand how users are navigating through the site. Authors go blind after the site goes live!

As web technology has matured, websites have progressed from static brochureware to hugely complex portals providing rich, highly customized services. For many companies websites have become a significant, if not the primary distribution channel for a company's products and services. With this transition websites are becoming a focal point for customer interactions. Customer relationships are formed, personalized, and managed through this medium.

K. Börner and C. Chen (Eds.): Visual Interfaces to Digital Libraries, pp. 65–77, 2002.

By adding instrumentation to a site it is possible to obtain fine-grain data tracking site activity. This is quite different from the real world were such instrumentation does not exist. For example, in department store, we might know that a shopper purchased a tie in the Men's Department and, by matching credit card transactions, perhaps also purchased a portable drill in the tools department. In the equivalent web scenario, we would know where the browser came from, when he entered our store, what he searched for, which other departments he visited, what other tools he put in his shopping basket, and if he abandoned the purchase because the checkout was too complicated.

The challenge is then how best to exploit this exceedingly rich new information source to make sites easier to use. The first problem, capturing site activity, is conceptually simple but practically difficult. The next level of problem, performing effective analyses, involves collecting and coalescing this dispersed information set into a coherent schema, correlating site activity with related attributes, and linking it with enterprise information.

To address the second level problem, we have developed a suite of visual tools that focus on various website analysis subproblems. The subproblems we consider are data collection (Section 2), traffic analysis (Section 3), site structure (Section 4), site monitoring (Section 5), and path and flow analysis (Section 6), and related work (Section 7).

2 Data Collection

Broadly speaking, there are two ways to collect raw site activity data. The first, called server-side data collection, extracts website usage information – hits, page views, dwell time, transactions, etc., from web (or other application) server logs. These data are recorded as the web server processes visitor page requests. There are several formatting standards for web server log files, such as W3C, Common Log Format, and Extended Log Format. However, these share the characteristic that for each hit (page request, image download, FTP request, ...) an entry is appended to the log file. This means that a single page access can generate dozens or even hundreds of log-file entries. Table 1 lists the standard field definitions that most vendors support.

The second method for collecting site data, called client-side data collection, operates by means of tracking agents, such as Java or JavaScript page tags, embedded in each instrumented HTML page, and executed by the client's web browser. Every time an instrumented page is loaded, the page tag sends an HTTP request to a recording server that captures the information in real time. The information sent might include essentially the same variables that are recorded in the web server log file with the exception of the web server status. Collecting data using page tags avoids network caches: the HTTP request is sent every time the page is loaded, even if from a cache. The disadvantage is that it is administratively difficult to put tags on every page of a website and to operate the recording server. As with web log data, this data can be sessionized (see below) to obtain raw paths. B y cleverly using JavaScript, it is possible to insert

Table 1. Common fields recorded in web server log files.

Date	Date on which the activity occurred
Time	Time the activity occurred
Client IP Address	IP address of the client accessing your server
User Name	Name of the user who accessed your server
Service Name	Internet service that was running on the client computer
Server Name	Name of the server on which the log entry was generated
Server IP	IP address of the server on which the log entry was generated
Server Port	Port number the client connected to
Method	Action the client was trying to perform (e.g. Get, Post)
URI Stem	Resource accessed; for example and HTML page, CGI-bin program, etc
URI Query	Query, if any, the client was trying to perform
HTTP Status	Status of the action, in HTTP terms
Win32 status	Status of the action, in term used by Windows NT
Bytes Sent	Number of bytes sent by the server
Bytes Received	Number of bytes received by the server
Time Taken	Length of time the action took
Protocol Version	Protocol (HTTP, FTP, ...) version used by the client
User Agent	Browser used on the client
Cookie	Content of the session cookie, if any
Referrer	Site on which the user clicked on a link that brought the user to this site

sessionization information directly into the HTTP request headed toward the collector and thus sessionize in real time.

Data Organization and Transformation

To be most useful, raw hit data needs to be organized into sessions, the sequence of pages viewed by individual users. This process is complicated since the HTTP protocol is stateless: each page is an independent HTTP request, and page views from all users are co-mingled in the log file. Furthermore, if the website is implemented as a web farm, then the requests from individual users may be distributed across the log files of different machines in the farm. A technique to accurately track sessions is to add a session cookie that the browser automatically propagates as the user navigates through the site. Without session cookies, perfect sessionization is impossible. Instead, approximate sessionization algorithms function by matching time stamps, page URLs, referrers, and browser attributes recorded in the log file.

Two useful transformations are domain naming mapping and robot elimination. Domain name mapping translates IP addresses into meaningful domain names (such as 216.219.254.117 into www.visintuit.com). Robot elimination excludes machine-generate traffic from robots, spiders and crawlers. For small sites, as much as 50% of the traffic may be machine-generated.

While straightforward conceptually and well-documented in [8], [12], and [14], collecting path data using either of these techniques is difficult in practice, particularly in production environments. Scale is perhaps the most significant issue, as the largest sites generate tens of millions of hits per day. Corrupted entries in the log files, dropped page tag messages and other data quality issues complicate processing and must be handled. Data inaccuracies can arise from network caches, dropped requests, and other problems. For example, pages served from network caches do not cause web server hits, and thus are not recorded in server log files. Web farms are large numbers of servers that function in parallel as a single website. The servers in the farm may be geographically dispersed, run different operating systems, and use different server software, in order to increase reliability. Pages may be served up from any server in the farm. The log files from every server in the farm must be integrated, time-stamp aligned, and processed together to obtain a complete usage data set.

3 Traffic Flow Analysis

Traffic flow analysis involves understanding the overall patterns of activity on the site, correlating activity with visitor type, understanding which promotions are most effective, and gaining insight into site usage patterns. This problem has been well studied both in the data mining [1, 3] and scientific communities [7, 10]. It involves answering a series of unstructured questions, formulating hypothesis, and navigating through a complex information space.

To understand site activity, we developed an interactive *workspace* for visual discovery, analysis, and correlation (See Figure 1). As described in [5], there are three significant parts to our visual tool: a tree-structured workflow control along the left, interactive bar charts down the middle, and an analysis pane with three tabs labelled `Overview`, `Details`, and `Paths`. Other system components include a color legend, navigation controls, and selection control. Clicking on any entry in the tree-structured workflow control automatically populates the bar charts and sets the tabbed pane to the most useful data and display for the particular analysis. In its initial configuration there are seven high-level categories in the workflow:

1. `Top 20s` provides a quick view of the top page, path, promotion, visitor, and visit statistics.
2. `General Activity` reports on site activity by time period, either hours, days, weeks, months, or years.
3. `Hosted Ads` shows click through rates and other advertising effectiveness attributes for sites displaying banner adds.
4. `Promotions` show which sites, URLs, search engines, etc., are most effective at driving traffic to the site.
5. `Site Effectiveness` investigates site-oriented statistics such as errors, paths though the site, entry and exit pages, page stickiness, etc.
6. `Visitors` organizes e-commerce site visitors into *Browsers*, *Abandoners*, and *Buyers* through a configuration option. It then supports visitor analysis via organization, country, type, etc.

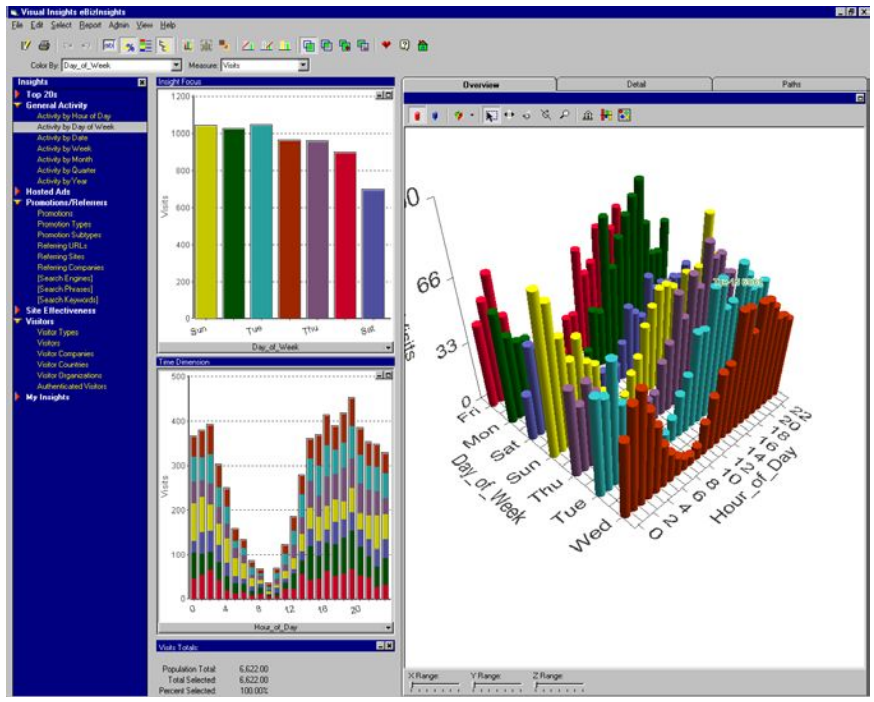


Fig. 1. Interactive Workspace for Website Traffic Analysis.

7. My Insights is a special tab for users to save custom analyses and, in the future, an integration point for analyses based on additional data sources.

The workspace visualization contains three barcharts named Focus, Correlation and Time[1], all of which function similarly, and a 3D plot called a Multiscape for showing two-way correlations. Users set the statistic in the bar chart either by clicking on an entry in the tree or by manipulating the bar chart selection control at the bottom. The bar charts both show results and serve as an input environment for user selections. Expanding the bar charts shows that they are richly parameterized and can be oriented, zoomed, panned, labeled, and sorted. See Figure 2.

4 Site Structure

An important information visualization problem involves visualizing the structure of a website. In a typical problem formulation the pages are treated as nodes in a graph and hyperlinks or directory structure as edges. For site visitors, visualizations of site structure (website maps) serve as a navigation aid.

[1] The correlation barchart has been minimized in Figure 1.

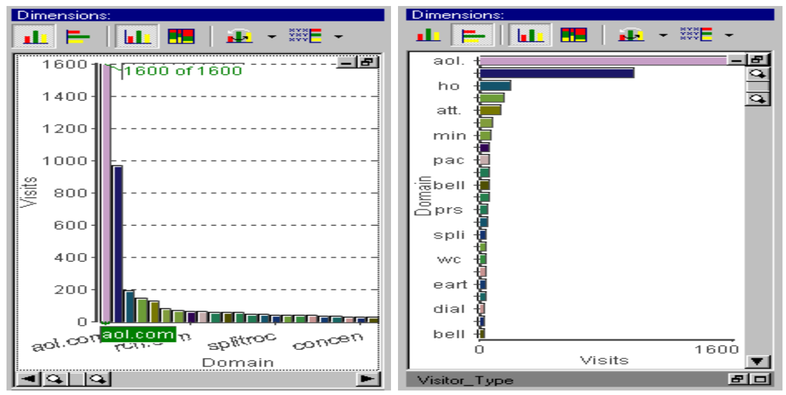


Fig. 2. Bar charts may be expanded, zoomed, panned, labeled, oriented, and sorted.

For site designers the maps can be used to ensure that the site has an easy-to-understand structure. For researchers, the visualizations can help gain insight into commonly used site designs.

An obvious way to visualize structure for sites organized hierarchically uses a tree metaphor. The trees are frequently laid out radially with the root page in the middle and pages at various depths positioned in circles around the root node. The reason for the radial layout is to utilize screen real estate more efficiently. As the number of nodes increases horizontal tree layouts use log(n) vertical and n-squared horizontal screen real estate. For larger trees the aspect ratio becomes very wide and does not match normal displays. Lines between the nodes show tree structure. The radial layout works well since many sites are designed hierarchically with progressively more detailed and finer levels of information. Sites with other, more complex, structure can be coalesced into a tree by dropping back-links and ignoring links that skip hierarchical levels. Ignoring symbolic and back-links is effective for understanding site structure, but ineffective for navigation tasks. Straightforward implementations of tree layout algorithms work well for small sites with perhaps tens to hundreds of pages. The information visualization research challenge involves displaying sites with hundreds to thousands or even tens of thousands of nodes. For these sites screen real estate limitations make it impossible to show every page simultaneously. A general strategy particularly suited for aiding users navigating a site is to distort the layout so that pages close to the visitor's current lo cation are shown in full detail and distant pages in the periphery are shown in less detail or not at all. Figure 3 illustrates these concept by showing four visualizations of site structure.

5 Live Site Monitoring

For many decision-making processes involving site activity, promotions, offers, inventory, etc., daily, weekly and monthly reporting lags are inadequate. Information technology organizations and providers have long appreciated the need

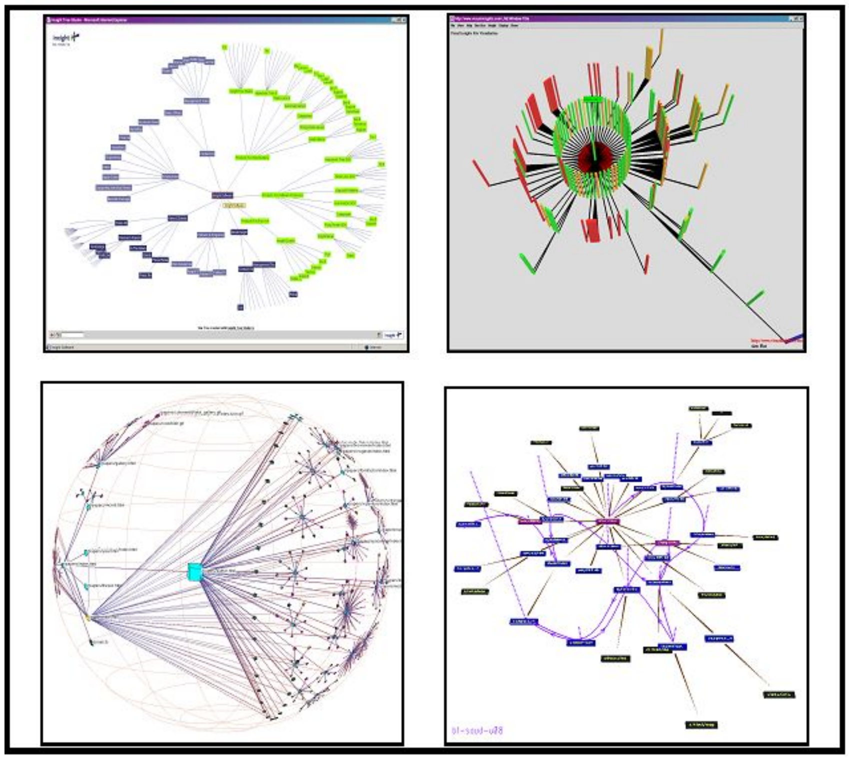


Fig. 3. Four visualizations of website structure. Top left: Inxite Software's Hyperbolic Tree showing site structure [9]. Top right: Visual Insights site map showing page usage. Lower left: Silicon Graphics's site manager. Lower Right: A visitor's path through a website [4].

for real-time monitoring and management of networks and other critical infrastructure. In Web-based marketing campaigns, companies can and do change content, adjust banner ads, modify e-mail messaging, and change site content literally throughout each day. Businesses must understand campaign productivity as it is happening and make adjustments on the fly. This involves measuring how different stimuli affect site traffic flows, site stickiness, entry and exit points and, for e-commerce sites, relating this activity directly to buying behavior. There is no point, for example, in stimulating more demand for a promotion if inventory is running low, if the site is experiencing technical problems, or if a weather pattern will delay product shipments. Figure 4 shows a prototype interface for monitoring site activity. This problem has also been studied by [11]. In contrast to network monitoring systems that focus on errors, low-level packet counts, and link availability, this interface targets understanding visitor and site activity. The version shown in Figure 4 focuses on commerce sites. Each cylinder represents a set of pages being monitored-referring URL, promotion type, a

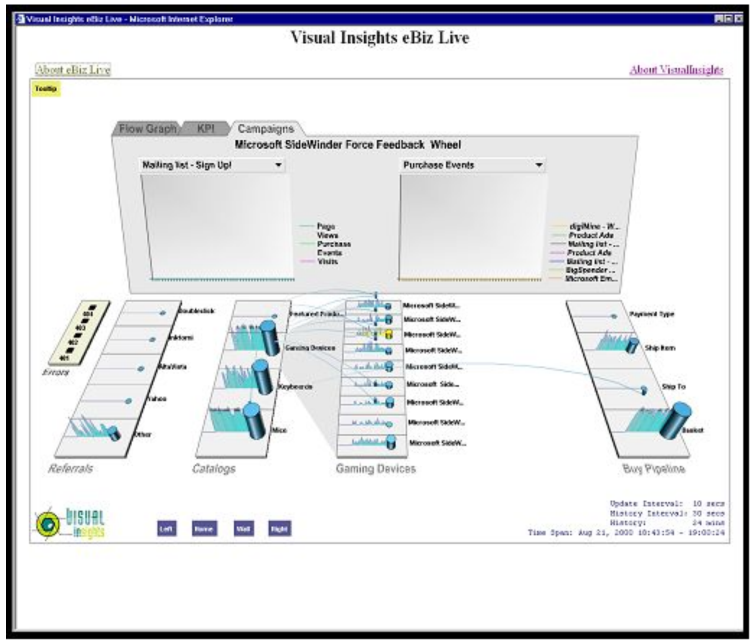


Fig. 4. Visitors moving through a commerce site.

catalog entry, or item. Server errors and purchases are shown on the far left and right, respectively. The height of each cylinder represents the number of page views during the last sample. The trailing graph associated with each cylinder shows historical trends. Visitors moving through the site are shown by animating a glyph that "jumps" between the pages. The size of the animating glyph encodes the number of visitors moving between the pages.

Floor and Back Wall. The display is organized into a "floor" and "back wall." On the floor, the cylinders corresponding to the referrals typically represent visitors entering the site stimulated by marketing campaigns. Catalog entries and products are organized hierarchically, as with aisles in a store. Clicking on any catalog entry expands it to show the products within that aisle. Users may expand entries to see activity at finer detail. This "drill-down" capability can be organized by category or actual product pages. Each cylinder in the "Buy Pipeline" shows activity by stage-"Shopping Basket," "Ship To," "Ship Item," and "Payment Type." Organizing data hierarchically increases the scalability of eBizLive.

The site overview provides instant value. It shows where visitors entering the site are coming from, what they are doing when they are on the site, and how long they are staying. Site traffic flows, errors and problems are instantly apparent. Problems can be fixed and opportunities addressed immediately rather than days later when reports are published.

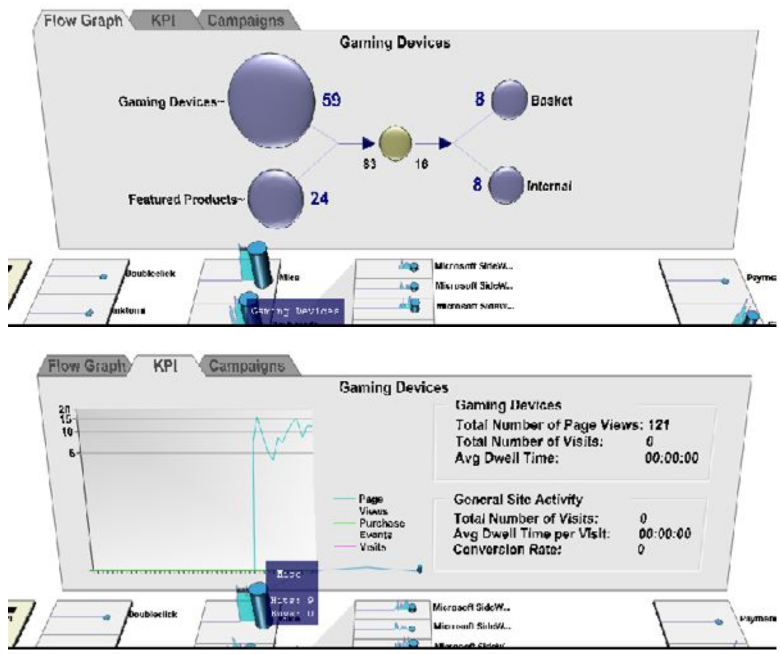


Fig. 5. Top: Flow through site. Bottom: Key Performance Indicators.

The tabbed pane on the back wall organizes three displays. The Flow Graph (Figure 5 top) shows traffic into and out of the selected set of pages. The sizes of the disks encode traffic flow to and from the selected pages and the numbers show the actual counts. The Key Performance Indicator (KPI) tab (Figure 5 bottom) displays KPIs using a time-series graph and textural displays. Common KPIs for eCommerce sites are page views, purchases, number of visitors, dwell time and conversion rates. The Campaigns tab presents the effectiveness of any ongoing marketing promotions.

Interaction and Navigation. Brushing over any item with the mouse causes detailed information about that item to be displayed using a transparent pop-up text field. The floor and walls are part of a 3D scene. Using the mouse and keyboard accelerators, a user can zoom in, out, and move around the scene. Clicking on the Left, Home, Wall, and Right buttons at the bottom moves the scene to fixed viewpoints. Catalog entries and items along the flow can be sorted and filtered.

Analogue of Store Activity. The site activity display (Figure 4) is the website analogue of store activity. The motivating idea is that, in a large department store, the manager frequently has a second floor office. From this bird's eye view, he or she can follow flow patterns among the departments, find high activity areas, identify aisle obstructions, and see under-utilized sections of the store. From this overview, the manager sees overall store activity that frequently correlates with transaction activity. In the same way, eBizLive's site activity display provides a broad "gestalt" showing crucial site activity.

6 Path and Flow Analysis

The goal of a path analysis is to understand the sequences of pages (urls) that a visitor traverses within a site. By understanding typical visitor browsing patterns, which hyperlinks are actually followed, and where visitors linger authors can create more usable sites. Flow analysis aggregates over individual users to understand the overall traffic patterns within a site.

Path Visualization. One way to show paths for an individual visitor superimposes a trace of the pages visited on top of a site map [13]. This idea is similar to showing how a visitor might progress through a road network and is implemented as part of the WebMetrics toolkit from NIST, implements this technique for path and timing data [4]. Figure 3 (lower right)[2], shows a visitor's path using a curved purple line connecting each page visited. Pages are positioned out as a directed graph on a 2D plane using a forced-directed graph layout algorithm. The dotted lines above certain pages indicate high dwell times, that is pages where the visitor lingered.

Flow Analysis. In contrast to path analysis, flow analysis aggregates over individual paths to find the flows within the site. Flow analysis can identify the most frequently clicked on links, entry and exit points, bad links, and other problems with the site.

What are common traffic flow patterns on the Visual Insights corporate website? Figure 6 shows a visual interface for tracking flow, identifying common entry and exit pages, most frequently followed paths, and pages with unusual dwell times. The image shows flow into and out the page designated in the center ("Default.asp"), our home page. The large pink arrows labeled "ENTRY" and "EXIT" show the number (percentage) of visitors entering the site at this page (92%) and exiting the site (60%) at this page. Thus 8% of the visitors entered through book-marked pages or other links pointing to internal pages on the site. The symbols down the left and right show other pages from which traffic flows into and out of "Default.asp" in decreasing order. The lower pane in the interface is an html viewer that displays the designated page. Clicking on any page re-centers the analysis to show the flow into and out of that page and displays its page in the html viewer. By progressively clicking on pages or selecting from a list (not shown), the interface enables users to track flows, navigate through the site by tracing interesting and important flow patterns,

[2] See also http://www.itl.nist.gov/iaui/vvrg/cugini/webmet/visvip/pix/fig3.gif

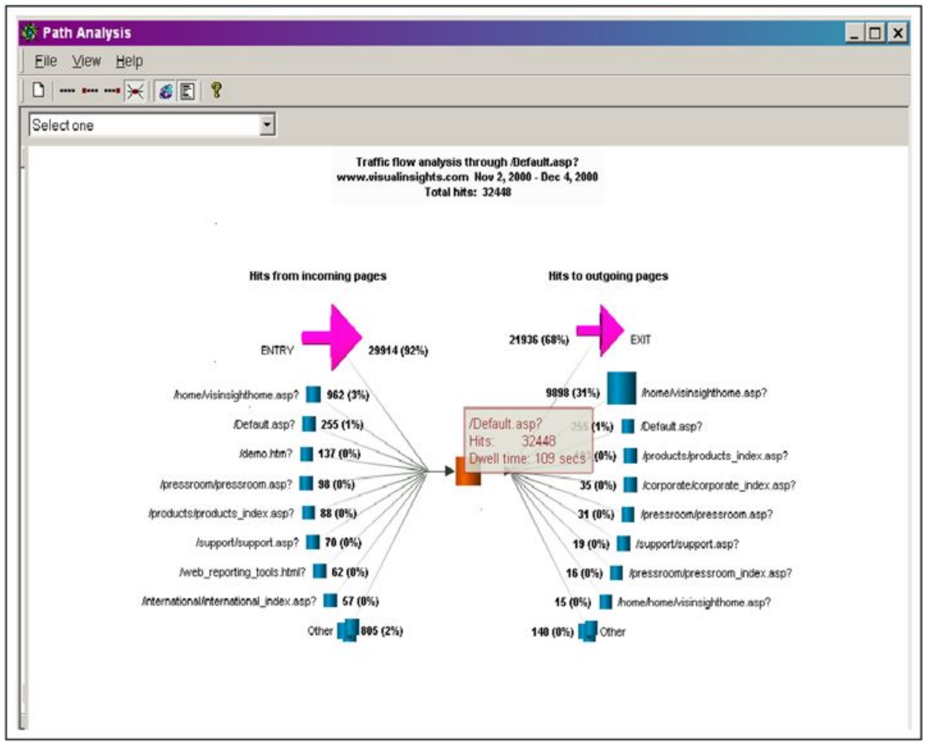


Fig. 6. Flow through a website.

A useful feature of the information visualization shown in Figure 6 is its ability to transition between flow and path analyses. By either clicking on the list icon activates a list selection control or mousing on any page a user can adjust the display to show lists of the most common paths from or to any page (not shown). The paths are sorted in decreasing frequency and show average dwell times.

7 Related Work

Since web analytics is such a rich area it is impossible to mention all related papers. Instead, we will highlight a few efforts that are directly relevant.

Fry's Anemone is an interactive Applet that shows live site activity and site structure using a node and link-based layout[3]. Deep Metrix[4], Webtrends[5],

[3] http://acg.media.mit.edu/people/fry/anemone/applet/
[4] http://www.deepmetrix.com/
[5] http://www.webtrends.com

Accrue[6], Visual Insights[7], and Digimine[8] are all commercial vendors offereing web analytics products. In another research effort Brner, et al. visual user activity in three-dimensional virtual worlds[2].

8 Summary

Improving website usability is a critical problem for both website authors and website users. In this paper we describe a sequence of techniques and visualizations for various web analytics problems. Our goal is to provide insite into web activity.

Acknowledgments

This research was complete while I was with Visual Insights. Many of the visualizations were created by the Visual Insights staff and I particularly acknowledge Alan Keahey for many insightful and creative dialogues and an anonymous referee. An early versions of this paper appeared in [6].

References

1. J. Borges and M. Levene. Data mining of user navigation patterns. In *Proceedings 1999 KDD Workshop on Web Mining*, San Diego, California, 1999. Springer-Verlag. In Press.
2. Katy Brner, Richie Hazlewood, and Sy-Miaw Lin. Visualizing the spatial and temporal distribution of user interaction data collected in three-dimensional virtual worlds. In *Proceedings Sixth International Conference on Information Visualization*, pages 25–31, London, England, 2002.
3. R. Cooley, P-N Tan, and J. Srivastava. Websift: the web site information filter system. In *Proceedings 1999 KDD Workshop on Web Mining*, San Diego, California, 1999. Springer-Verlag.
4. J. Cugini and J. Scholtz. Visvip: 3d visualization of paths through web sites. In *Poceedings of the International Workshop on Web-Based Information Visualization (WebVis'99)*, pages 259–263. IEEE Computer Society, 1999.
5. Stephen G. Eick. Visualizing multi-dimensional data. *IEEE Computer Graphics and Applications*, 34(1):61–67, February 2000.
6. Stephen G. Eick. Visualizing on-line activity. *CACM*, 44(8):45–52, August 2001.
7. B. Huberman, P. Pirolli, J. Pitkow, and R. Lukose. Strong regularities in world wide web surfing. *Science*, 280:95–97, 1997.
8. Ralph Kimball and Richard Merz. *The Data Webhouse Toolkit*. John Wiley & Sons, Inc., New York, New York, 2000.
9. J. Lamping and R. Rao. Laying out and visualizing large trees using hyperbolic space. In *Proceedings of the ACM Symposium on User Interface Software and Technology*, pages 13–14, 1994.

[6] http://www.accrue.com/
[7] http://www.visualinsights.com/
[8] http://www.digimine.com/

10. Steve Lawrence and C. Lee Giles. Accessibility of information on the web. *Nature*, 400(6740):107–109, 1999.
11. N. Minar and J. Donath. Visualizing crowds at a web site. In *CHI '99 Late-breaking Papers*. ACM Press, 1999.
12. M. Spiliopoulou, C. Pohle, and L. Faulstich. Improving the effectiveness of a web site with web usage mining. In *Proceedings 1999 KDD Workshop on Web Mining*, San Diego, California, 1999. Springer-Verlag. In Press.
13. A. Wexelblat and P. Maes. Footprints: History-rich tools for information foraging. In *CHI '99 Conference Proceedings*. ACM Press, 1999.
14. I. Zuckerman, D. Albrecht, and A. Nicholson. Discovering web access patterns and trends by applying OLAP and data mining technology on web logs. In *Proceedings Seventh International Conference on User Modeling*, pages 275–284, Wien, 1999. Springer.

Extreme Temporal Photo Browsing

Adrian Graham, Hector Garcia-Molina, Andreas Paepcke, and Terry Winograd

Stanford University
Gates Computer Science 4A
{adrian.graham|hector|paepcke|winograd}@cs.stanford.edu

Abstract. We developed two photo browsers for collections with thousands of time-stamped digital images. Modern digital cameras record photo shoot times, and semantically related photos tend to occur in bursts. Our browsers exploit the timing information to structure the collections and to automatically generate meaningful summaries. The browsers differ in how users navigate and view the structured collections. We conducted user studies to compare our two browsers and an un-summarized image browser. Our results show that exploiting the time dimension and appropriately summarizing collections can lead to significant improvements. For example, for one task category, one of our browsers enabled a 33% improvement in speed of finding given images compared to the commercial browser. Similarly, users were able to complete 29% more tasks when using this same browser.

1 Introduction

Given the importance of multimedia digital libraries, several projects have developed techniques for searching video, images, and geographic data (e.g. [1, 2, 3]). A majority of other digital library efforts focused on text, because text is most accessible to a variety of search and processing algorithms. Certainly, Personal Digital Library (PDL) facilities remained overwhelmingly textual. By PDLs we mean bodies of information that are mostly of importance to individuals or small groups.

Two changes over recent years have made the development of PDLs with image content urgent. (i) Desktop machine storage has grown cheap and plentiful and (ii) entry-level, point-and-shoot digital cameras have reached large numbers of consumers. Between 1999 and 2002 an estimated 55 million digital cameras have been sold [4]. A majority of 94% of these have been point-and-shoot units [5] that are commonly carried on vacations, to family festivities, and to other typical opportunities for picture taking.

Owners of digital cameras are not necessarily computer-savvy. But even if these consumers are comfortable with their computing equipment, it is their photographs, not the operation of equipment that should be the focus of their activities in viewing or organizing their images. As digital cameras do not require expensive film that can only be used once, users tend to take large numbers of pictures. This in turn results in an image management quagmire similar to the shoebox phenomenon of paper prints. Professional photographers and news reporters face the same challenge.

There have been several approaches to improving the experience of browsing and managing collections of personal digital photographs. The most basic photo browser

K. Börner and C. Chen (Eds.): Visual Interfaces to Digital Libraries, LNCS 2539, pp. 81–97, 2002.

available is a file manager with integrated support for viewing images and thumbnails. Such functionality is found in Windows XP, Mac OS X, and standalone browsers such as ACDSee [6, 7, 8]. Although simple to use, these tools tend to be limited, and do little to help users organize their pictures more effectively than they can in the physical world.

Using a database to manage photographs is another approach to this problem, and systems such as FotoFile and PhotoFinder [9, 10] are designed around that solution. Although these systems offer powerful search functionality, they have failed to catch on, largely because they require time-intensive manual annotation.

PhotoMesa and iPhoto are both examples of photo browsers which automatically group images for display. PhotoMesa groups by folder, year, and month, and iPhoto groups by "roll" (batch of photos downloaded to a computer at once) [11, 12]. This approach helps give more structure to the images than an enhanced file manager, but requires none of the time-intensive annotation of the database systems.

Our approach to this problem builds on the idea of automatically grouping photographs. Instead of using metadata that is directly available in the file system, we analyze the photo creation process and construct a variety of organizational structures. For instance, we can automatically organize photographs by event, realizing that sequences of photos of one event will be taken closer together in time than sequences of other photos.

An additional design goal was to build a system that could scale to visually browse a lifetime's worth of photographs. Photo browsers are notoriously poor at scaling, often unable to provide useful visual information for more than a few hundred images. Through our summarization techniques, we are able to browse tens of thousands of photographs.

These techniques are part of the two-phased approach the Stanford Digital Library Project is taking to PDL image management. During the first phase, we are exploring algorithms that organize and summarize digitally created photographic image collections *without* any curator assistance, using current, mature technologies only. While heuristic and therefore imperfect, we are trying to push this area as far as possible. We are careful during this phase to design our systems such that machine image analysis technologies can be integrated later on.

During the second phase we will examine how gracefully we can take advantage of incremental image captioning efforts that a curator might be willing to invest. The goal is for the collection's user interface to take advantage of coarse, as well as fine-grained captioning as curators create and refine such captions over time. In this chapter we present the current state of our phase I work, which is implemented in prototypes.

We begin with a description of one of our two browsers, the *Calendar Browser*. Next, we describe the heuristic algorithms that underlie this browser. The subsequent section introduces our second *Hierarchical Browser*, and a commercial browser used for comparisons. Finally, we present our experimental comparisons.

2 Calendar Browser

The Calendar Browser (CB) takes advantage of photo time stamps. When taking a picture with a digital camera, the equipment records the date and time when the photo was taken, as well as technical details, such as aperture settings, distance from the focal plane, and whether a flash was used. This data is encoded into the image files, and is available to applications that access these files. The Calendar Browser (Figures 1 and 2) is an intentionally extremely simple interface that uses these date/time stamps to enable drill-down browsing and summarization.

The application window is partitioned into two panes, the control panel and the display panel. The display panel contains space to hold a constant number of fixed-size thumbnails (in our implementation we always show up to 25 images). At any given moment, the user is viewing images from one single time granularity. For example, Figure 1 shows a screenshot of the Calendar Browser at a 'one-year' granularity. At this granularity, all images in the display panel will have been taken during the same year, in this case 2001. Usually, the collection will contain many more than the 25 images. The challenge, therefore, is to select representative images from among the 2001 subset of the collection. The images in the display panel are sorted by time and are labeled with time designations of the next-lower time granularity. In Figure 2 the labels are therefore the months of the year 2001.

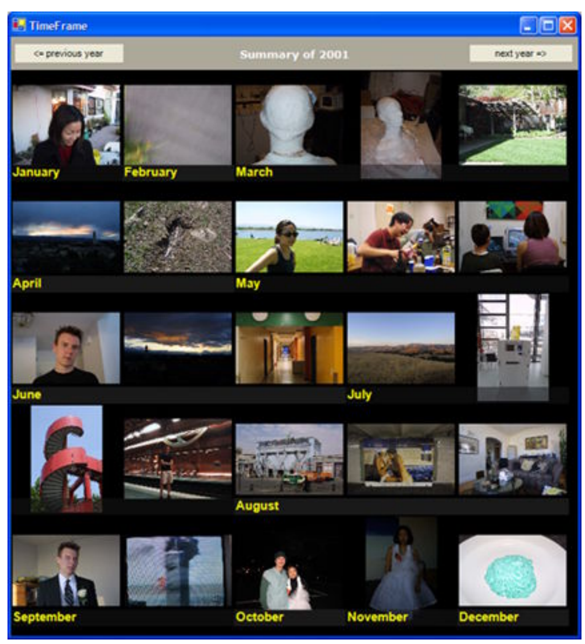


Fig. 1. Calendar Browser: Year View

Notice that not every month is represented with the same number of images. While the browser dedicates only one of the 25 precious display slots to each of January and February, four slots are invested in the month of August. We will explain the allocation strategy in the next section, but roughly speaking, the allocated space is proportional to the number of photos taken during the respective time interval.

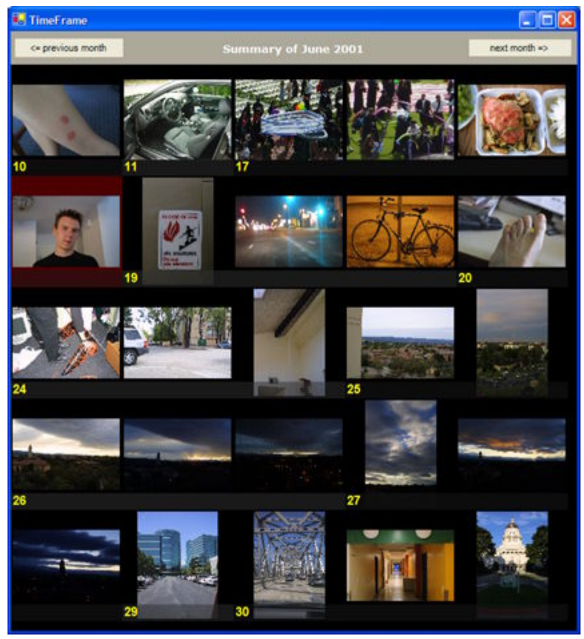


Fig. 2. Calendar Browser: Month View

The two buttons in the control panel allow users to move through time at the current granularity. In Figure 1 this means moving backward and forward by year.

The user 'drills down' into the collection by left clicking on an image. (Right clicking moves us in the reverse direction.) Figure 2 shows the screen after the user left clicked on a June image. Notice that the control panel now indicates that the time granularity has changed to the month level. The control button labels have correspondingly been changed to 'previous month' and 'next month'. Images in the display panel are now labeled by days within the month of June when the respective photos were taken. Again, not every day is represented by the same number of images, but all of the 25 slots are filled.

Inspecting the display panel of Figure 2 further, we find a clue to how representative images are selected for a 'next-coarser' time granularity level. Notice that the largest sequences of image slots in Figure 2 are allocated for the 17th and the 30th of June, which contain five and four images, respectively. The 24th and the 27th are tied with three images. Referring back to Figure 1, notice that one image from each of the two largest June sequences, and one image from the 27th were picked to represent June for the year 2001.

Users may now left click on an image in one of the June days to see a summary of the photos that were taken on that day. If a user left-clicks on an image at the day granularity, that image will be placed in the center of the display panel, and the space around the image will be filled with the photos taken immediately before and after the center image. No summarization is applied at this level. Users may navigate using the previous and next buttons, to move through the un-summarized photographs 25 images at a time.

Earlier versions of this browser had several more controls that afforded the sliding of time windows to control image display down to minute granularity, and other powerful manipulations. Preliminary, informal user tests made it very clear that we were best off with the exceedingly simple design exemplified in the figures. Both novices and experts preferred this simpler interface.

3 Time Based Summarization

People tend to take personal photographs in bursts. For instance, lots of pictures may be taken at a birthday party, but few, if any, pictures may be taken until another significant event takes place. We can derive information about personal collections of photographs based on these irregular, "bursty" patterns, which we analyze through clustering techniques.

The goal of clustering is to expose the structure that is present in a set of personal photographs as a result of the way the user has taken the pictures. Without realizing it, the user gives structure to his image collection by the pattern in which he chooses to take the photographs. Based on these patterns, we can organize the images without any further user intervention. For instance, someone may take a series of photographs at a birthday party. By identifying this burst, we can group these images together, thus creating a structural unit. We call this structural unit a cluster.

In many cases, we can derive even more specific information about the set of photographs in a cluster. Much as we were able to identify the "Birthday burst" by noticing an increased rate of activity, we are able to identify "sub-bursts" by comparing relative rates of activity within the cluster. For example, the rate at which pictures are taken is likely to be higher when someone is opening presents or blowing out candles on a cake. By encoding these sub-events within our cluster structure, we gain an even more accurate portrayal of the collection.

More generally, we can say that the burst structure within collections of personal photographs tends to be recursive, where bursts make up bursts etc. We represent this recursive burst structure using a tree of clusters, where photographs are stored only at the leaf nodes.

Our clustering machinery is the substrate that underlies both phases of our project. First, we can query the cluster structure for information about the patterns present in the collection, such as: the number of images in a cluster, the number of clusters in a time span, the time span of the images within a cluster etc. Based on this information, we can create summarization schemes for a whole class of browsing interfaces. For instance, the Calendar Browser introduced above queries the cluster structure when it populates the screen at different time granularities.

Second, instead of operating on an image at a time, we are able to create user interfaces that operate on entire clusters. For example, this will allow us to build an interface for incrementally adding metadata to a collection (Phase II of project).

3.1 Time Cluster Engine

There are many ways to cluster images based on time metadata. For instance, one might use a K-means algorithm to cluster photos taken at about the same time. Another approach is to use image analysis to help improve the accuracy of clustering photographs by time. AutoAlbum and its successor PhotoToc combine time clustering and image analysis in this way [13, 14], producing time-related clusters of similar images. These systems first cluster photographs by creation time as we do. They then split overly large clusters at points where the difference between the color compositions of two adjacent photos is larger than the local average.

Our work in its current manifestation pushes the use of time further than [13, 14], rather than turning to color analysis. Our goal is to cluster photographs according to the hierarchical burst patterns discussed earlier. This approach finds the same initial events (internally represented by clusters) as [14]. But the procedure then 'cracks open' these events to detect their sub-events (e.g. cutting birthday cake). This technique enables the creation of organizational structure down to an arbitrarily fine granularity.

Starting out towards burst detection, we need to group pictures that are taken at about the same rate. K-means clustering is difficult to use in this case because the specific number of bursts is not known ahead of time.

In order to determine where a burst begins and ends, we need to know the rate at which pictures were taken within the burst. However, in order to find out this rate, we need to know when the burst begins and ends. We overcome this cyclical dependency by initially clustering the images by a constant time difference. That is, we compare consecutive photographs, and if they differ in time by more than a specified amount (see below), we create a new cluster. This gives a reasonable approximation of medium-sized clusters. We then create new clusters by splitting and combining these initial clusters. We next describe this three-step process in more detail.

Step 1: Create initial clusters. We sort the list of photographs in the collection with respect to time, and create the root node of the cluster tree, which is empty. We create the first child cluster and add it to the tree. We then iterate through the sorted list of photographs. Every time two consecutive photographs differ by more than a specified constant time difference, we create a new cluster and add it to the root cluster. Regardless of whether we just created a new cluster, we add the current image to the most recently created cluster. By the time we reach the end of the list, we have a root cluster with as many child nodes as there are initial clusters. This simple, one-tiered cluster tree represents an approximation of the medium-sized bursts in the set of photographs. (This process is equivalent to the time-based clustering used in AutoAlbum [13].)

In step 2 of the clustering process, these initial clusters are split into finer clusters based on the time differences between photographs within each initial cluster. In step 3, the initial clusters are combined into more general clusters based on the time differences between the initial clusters. Thus, the final cluster tree is not heavily dependent on the specific time difference used during initial clustering. We found that choosing initial time differences between 1 and 24 hours resulted in similar final cluster trees. Our current prototype uses an initial time difference of 1 hour.

Step 2: Split initial clusters. In Step 2 we want to further refine these clusters, by splitting them into appropriate sub-clusters (i.e. adding child nodes to the initial clusters).

During clustering, Step 1 could not take into account the rate at which photographs were taken within a cluster. Now that we have approximate clusters, we are able to consider these intra-cluster rates. This enables more accurate clustering, as the rate at which photographs are taken is likely to differ significantly between types of events. For instance, while taking a hike through a forest, someone may take a picture every couple of minutes. In contrast, when photographing a newborn baby for the first time, the time between pictures is likely to be in seconds.

We take these differing rates into account by comparing each pair of consecutive photographs to the 'basic' (see below for details) photographic rate of the cluster. When we come across a pair with a time difference that appears to be outside the normal range for the cluster (an outlier), we create a new cluster. This new cluster contains the images between the previous outlier (or the beginning of the cluster if no previous outlier exists) and the new outlier. This new cluster is then added to the original cluster as a child node. Since photographs are only stored in leaf nodes, the photographs contained by the new cluster are removed from the original cluster.

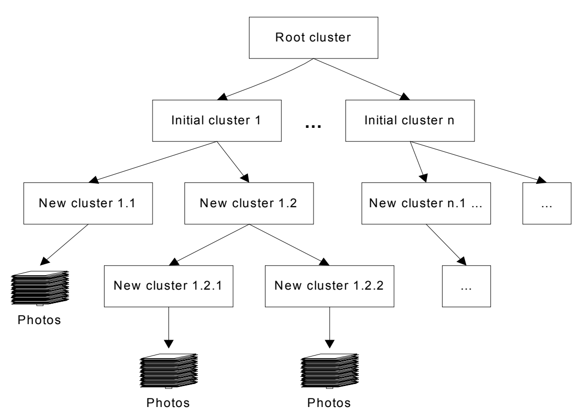


Fig. 3. Partially created cluster tree.

Figure 3 shows a snapshot of this stage in the process. Initial cluster 1 has been subdivided into new clusters 1.1 and 1.2. Cluster 1.2 still contained image bursts that were separated by significant spans of time, so it has been subdivided again. In contrast, cluster 1.1 now contains only images taken during a single burst of picture-taking.

Since bursts in collections of personal photographs tend to be recursive, we repeat this process over each newly created cluster until no outliers remain.

To find outliers in a cluster, we follow standard statistical procedure. We compute all the time differences between consecutive photographs. We then find Q_1, the value that divides the lower 1/4 of the values from the upper 3/4, and Q_3 that divides the lower 3/4 from the upper 1/4. A difference is an outlier if it is greater than $Q_3 + 2.5\times(Q_3\text{-}Q_1)$. (The value 2.5 was selected empirically.) For each outlier, we split the cluster at the time that the large difference occurred.

Step 3: Create parent clusters. This step is much like Step 2, except that instead of splitting initial clusters into more specific clusters, we combine initial clusters into more general clusters. One way is to use a fixed year-month-day hierarchy for the higher levels. That is, for each cluster C discovered in Step 1, we determine the date of its first photograph, and then we make C a child of the appropriate day in the year-month-day hierarchy. Any C children determined in Step 2 will continue to be children. Note that a given day may have multiple clusters in it. This approach is used in our prototype, and is useful for creating summaries that are intended to be navigated using time-based interfaces, as is the case with the Calendar Browser.

An alternative is to create the higher levels of the hierarchy using statistical techniques analogous to the ones of Step 2. For example, instead of analyzing the time between photographs as we did in Step 2, we could analyze the time between clusters. Clusters that are relatively closer together to each other than other groups of clusters could then be combined into parent nodes at a higher, possibly newly created, tier in the cluster tree. This process would be repeated, building the tree upwards from the initial clusters.

3.2 Creating Summaries

Collections of personal photographs typically contain many more photographs than can be displayed on screen at once. This problem applies not only to desktop photo browsers, but even more so to low resolution devices such as televisions, hand-held computers, and displays on digital cameras. Most systems overcome this limitation by allowing users to scroll through sets of images. Unfortunately, scrolling also has limitations. Most notably, scrolling gives no sense of overview, making it easy for users to feel lost in a sea of images.

An alternative to scrolling is summarization, as in the display panel of Figures 1 and 2. Instead of displaying all images, a set of representative images is shown. Once the user finds an image from the event he is looking for, he is able to "zoom in" and see the event in greater detail. When he is finished examining the event, he may "zoom out" to return to the summary view. Detail is hidden when unnecessary, but available whenever needed. Overviews are easily seen by zooming out.

The input to our summarization procedure is a set of sequential clusters **C** at level k in the hierarchy, plus a target T, the desired number of representative photographs. With the Cluster Browser, summarization always starts with a single day, month or year cluster, but in general we can summarize a set of consecutive clusters. The following two steps perform the recursive summarization process:

Step 1: Screen space assignment. If there are T or fewer photographs in C, creating a summary is trivial — we assign screen space to each of the images. Otherwise, we use the following rules to assign screen space:

1. Assign one space to each cluster in **C**. This ensures that there will be at least one photograph from each cluster in the summary. If we are unable to give one space to every cluster, we give priority to larger clusters.
2. Say we assigned M spaces in Step 1. We then assign the remaining $T - M$ spaces to clusters in proportion to their sizes.

For example, say we are summarizing three clusters. Cluster C_1 (including its children) has 20 photos, C_2 has 10 and C_3 has 5. Our target is 10 summary photos. After we assign one slot for each cluster, we are left with 7 spaces, so we give 4 more to C_1, 2 to C_2 and one to C_3. Thus, C_1 gets a total of 5 spaces, C_2 gets 3, and C_3 gets 2.

These rules are based upon the notion that summaries should show as much variety as possible within the given time range. This is achieved by giving summary space to as many clusters as possible, regardless of cluster size. For instance, a given time range may contain two bursts: a vacation when 100 photographs were taken, and an old friend's visit when two photographs were taken. If we simply assigned screen space based on cluster size, a summary of ten images would contain only images from the vacation. Images from different clusters tend to be more distinct than images within a single cluster. Thus, by prioritizing summary images from different clusters, we ensure more variety.

Step 2: Selection of summarization photographs. As a result of Step 1, each cluster C_i in C has been assigned a target number of photos T_i. If C_i is not a leaf cluster, then we recursively repeat Step 1 to allocate the T_i spaces to C_i's children. (Recall that if C_i has children, only its children have actual photographs.) Eventually we obtain a number of spaces assigned to each leaf cluster in C.

There are several options for choosing representative images in leaf clusters:

- *photographs separated by smallest difference in time* - Images with little time between one another are likely to be of the same subject. Any event that warrants multiple photographs is significant enough to be considered representative. Thus, one of the photographs of the sequence would be a good candidate for the summary.
- *photographs separated by largest difference in time* - Images with lots of time between one another may visually differ greatly from one another. Thus, the photographs right before or after the long time interval may be of interest.
- *contrast and resolution information within photographs* - In some cases, choosing a representative image is not as important as choosing a highly visible image. Images with high contrast and resolution tend to be easier to identify within summaries.
- *cluster-wide image analysis* - Based on the image properties of the cluster, it may be good to include an image that best represents the visual characteristics of the cluster.

In our prototype, we use the first two heuristics only. First, we look for the smallest time difference between consecutive photographs, and we select one of the two involved. If we still need additional summary photographs for the leaf cluster, we look at the largest time differences and select the photographs involved. For example, say we need 4 summary photos. If *a* and *b* are the closest photos, *b* is our first choice. Then say that *c*, *d* is the pair with the largest difference, and that *e*, *f* follows. Our remaining choices would be *c*, *d*, and either *e* or *f*. In practice, we have found that the actual details of how photographs are selected are not critical, as long as a reasonable strategy is used.

4 Other Browsers

As we informally tested our Calendar Browser on various users, we found that they liked the time-based approach and the summarizations, but they were in disagreement on the user interface. Some liked the browser's simplicity, others suggested that we borrow the explicitly hierarchical browsing controls of the Windows File Explorer and Macintosh Finder utilities. Figure 4 shows the resulting alternative browser, which we also implemented.

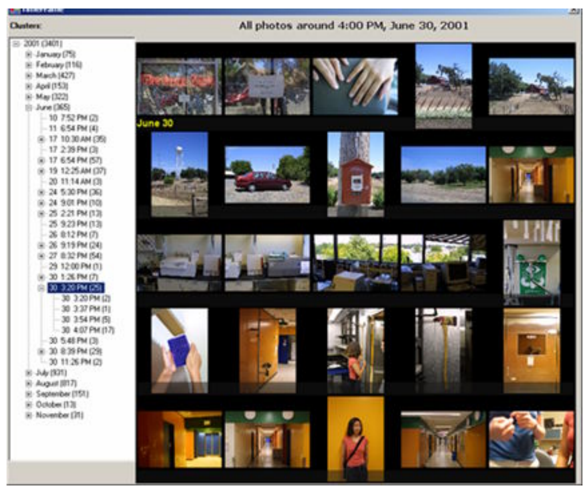


Fig. 4. The Hierarchical Browser

The left panel contains a tree widget whose nodes at its different levels correspond to times at a given granularity. Outermost nodes correspond to years. Once a year is opened up by clicking on the square next to the year's label, months are shown indented. Users may open and close nodes at various levels, just as they do when browsing a hierarchical file structure. The numbers in parentheses next to each node label are the numbers of images nested below that node. For example, Figure 4 shows that on June 30, 2001, 25 photos were taken between 3:20pm and 5:48pm.

The number of entries on each level depend on the results of the clustering process. For example, while all the images taken on June 20 are combined in a single entry, June 30 ended up containing five sub-clusters, each being a burst of picture-taking.

The right pane is the same as the display panel of the Calendar Browser, except that clicking on an image has no effect. The functionality of 'drilling down' is instead covered in the tree control widget. Note that, unlike the Calendar Browser, the Hierarchical Browser allows one to drill down to sub-clusters smaller than a day.

Although the Calendar and Hierarchical interfaces can, of course, be combined, we chose to keep them separate for the purpose of our user experiments. We wanted to test the constituent interface aspects in isolation before considering combination designs. In addition, we were curious how well either design held up against an unsummarized scrollable photo browser. In particular, we were interested in the following questions:

- How quickly can users find a given image, especially when multiple similar images are also part of the collection?
- How effectively will users find images when given a description, such as "mountains with a sunset", "a red telephone", "a woman in a blue T-shirt?"
- Will the time-based summaries facilitate searches for time-related images, e.g., a Halloween or a Christmas picture?
- Do users spontaneously take advantage of the summarizations and time-based organization?
- How good is the summarization when compared to random selections of images?
- Are there browser-related differences among computer novices and experts?

We therefore conducted a controlled study covering our Calendar Browser, our Hierarchical Browser, and a specially configured commercial ACDSee browser. ACDSee is a file manager optimized for browsing directories of images. When a specific directory is selected, as many thumbnails as possible are displayed. When a directory contains more thumbnails than can fit on a single screen, users can scroll vertically using a standard scroll bar. We copied all of the images from our set of test images to a single directory and configured ACDSee to display these images chronologically. Thus, for the purposes of our tests, subjects did not have to navigate through file system folder hierarchies, as typical users of this browser would. Instead, this specially configured browser gave subjects an un-summarized chronological view of our test image sets, similar to the time-based interface used in iPhoto [12]. We refer to this browser as the Scrollable Browser. As well, subjects could find the specific date and time of each image by briefly hovering the mouse pointer over the image.

Both of our browsers are particularly effective for users who originally took the photographs they are exploring. In the case of photographic Personal Digital Libraries, it is very common for the curator also to be the creator of the collection. Nevertheless, as our user experiments show, even strangers to a collection benefit greatly from these browsers.

5 Experiments

We tested 12 subjects, six computer 'novices', and six 'experts'. We defined a novice as someone who uses computers at most to read email, create text documents, and browse the Web. If subjects also used other applications or programmed, they were classified as experts. Ages ranged from 18 to 55+. Among the twelve subjects, six were male, six female. Professions included students, school teachers, and university professors of varying disciplines, other than Computer Science.

We videotaped and recorded all test sessions. The sequence of the subjects' exposure to the three browsers throughout the experiment was balanced to neutralize learning effects across browsers. We timed all interactions with the browsers.

We used two image sets, A and B. Set A contained 1000 images, set B contained 3500. All photographs were in color and taken in a variety of locations, seasons, and times of day over the course of one year.

All values reported vary in their margins of error between ±5% and ±23% at 85% confidence level. Unless otherwise stated, all results reported below are statistically significant at 85% confidence level.

We asked subjects to complete a variety of tasks in six different categories, within a maximum time limit. Tasks in category 0 involved dataset A, all other tasks operated on dataset B. In what follows we describe the task categories, and as we do so, we describe the results obtained.

Category 0. Find a given image in dataset A. Time limit: 5min.

Category 1. Find a given image in dataset B, given also the month during which the image was taken. Time limit: 2.5min.

Subjects performed a total of 9 tasks: three of category 0, each with a different browser, and 6 of category 1. For the latter 6, each browser was used with two separate images. Again, the order in which subjects used the browsers was varied to eliminate biases.

These tasks were intended to test subjects' performance finding specific images when using summarized and un-summarized browsers. Our expectation was that summarization would help users find images more quickly. For instance, when searching for a given image, a user might find a summary image similar to the one being searched for, and then quickly drill down to the target image. Initially we intended to conduct all these tests without giving any clue as to when the photograph was taken. However, in preliminary testing, the un-summarized Scrollable Browser performed so poorly (users often required over 15 minutes to find an image) that narrowing the search was necessary to allow users to complete these tasks in a reasonable amount of time.

Figure 5 shows average completion times for the combined Tasks 0 and 1, for the cases where the subjects did complete the tasks. While the Scrollable Browser (SB) and the Hierarchical Browser (HB) draw just about equal, around 1 minute and 20 seconds, the Calendar Browser (CB) enabled significantly faster completion times at an average of about 50 seconds; an average 33% improvement. In our test runs, this advantage of CB over the other browsers held for computer experts and novices alike. However, we have too few data points in each of these two groups separately to make a general statement on this novice/expert point.

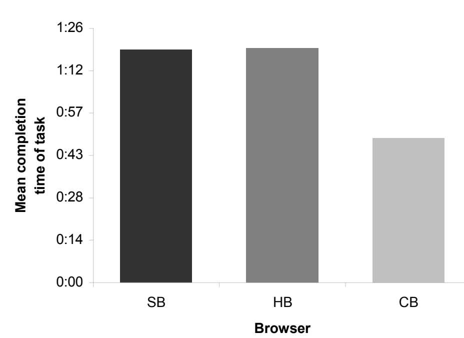


Fig. 5. Average completion times for tasks of categories 0 and 1.

In some cases the subjects did not find the requested images before the time limit expired. The Scrollable Browser did by far the worst, with subjects being unable to complete 31% of the tasks. The Hierarchical Browser was the most reliable for finding the images (6% not found). Users of the Calendar Browser were unable to complete tasks around 11% of the time. The difference in completion success between the Hierarchical Browser and the Calendar Browser are not significant.

The results of these tasks show that the time-cognizant browsers, which perform summarization, can indeed help when subjects are searching for images. Some of the improvements are due to the time-based navigation scheme, but other gains are due to the summarization capabilities that provide subjects with meaningful overviews. These results also indicate that although users were exposed to new browsing interfaces, they were able to understand them well enough with minimal training to perform at levels at least as good as with a traditional browser.

Category 2. Given 10 textual descriptions of images, find as many images that fit any of these descriptions as possible. Time limit: 3min.

Category 2 was designed to measure performance when classes of images were sought, not just a particular image. Our expectation was that summaries would expose users to a variety of images more quickly than the un-summarized view, thus enabling them to find more images that fit the given description within our time limit. For instance, a user might browse at the month level, drilling down only when a related image was shown in the summary.

To generate the 10 descriptions for a category 2 task, we selected at random 10 photographs from the collection. For each image, we then manually wrote a description. For example, the resulting descriptions could be "a sandy beach," or "a man sitting on a bench." Each subject performed two tasks of Category 2, with different description sets, once with CB, and once with SB.

Our results for Category 2 indicate that, when confronted with textual descriptions of images, subjects performed at identical levels in the Scrollable Browser and Calendar Browser. When completing this task, few subjects used the summarization provided by the Calendar Browser. Instead, nearly everyone performed an un-summarized linear search regardless of which browser was used.

Category 3. Find a given image, in the case when a similar image appears in one of the month summaries. In addition to completion time, we noted whether subjects made use of the summarizations, or followed another strategy (usually linear search). Time limit: 5min.

Each query image was selected as follows. A random photo was selected from one of the 12 month summaries. Then a similar (but not identical) photo was selected from the same leaf cluster. The subject was given the similar photo as the query, not knowing how it had been selected. Subjects performed this task with both CB and HB, with different query images.

Category 3 measured the impact of subjects ignoring vs. taking advantage of summarization when looking for a given image, as opposed to working from textual descriptions. In Task 3, 42% of subjects made use of summarization (across both browsers). That is, these subjects first browsed through the year and month summaries; when they found an image similar to the sample they had been asked to find, they drilled down. This 42% who made use of summarization enjoyed a 48% boost in performance.

All of the summary users finished tasks of Category 3 within the allotted time, while 36% of subjects who did not take advantage of summarization were unable to complete the task

Category 4. Find a given image, when the image has a clear clue as to which time of year it was taken. For example: a Christmas tree, fire works, etc. When appropriate, we ensured during the debriefing sessions at the end of the experiment that no cultural differences had prevented a subject from making the time association. Analogous to Category 3, we noted whether subjects recognized and used the time clue. Time limit: 5min.

Category 4 allowed us to examine whether subjects made use of time clues in images, and how this use affects performance. Again, subjects used CB and HB.

Almost all subjects, 92%, made use of the time clues for these tasks. Everyone completed the task. The average completion time was 36 seconds. The 8% who did not use the time clue averaged 1min 18sec. However, 8% of our subject population is too small a data set to make this performance comparison statistically significant.

Category 5. Find images to fit 10 textual descriptions of leaf clusters. Time limit: 5min for each run.

To generate the descriptions, we manually clustered the collection by event, and gave each cluster a short description (e.g., “visit to San Francisco”). For each task instance we selected 10 descriptions at random (no replacement). We limited subjects to the 'year' and 'month' summary views in CB.

Each subject performed two category 5 tasks, using the Calendar Browser each time. For one of the tasks, however, CB was altered so it would select images at random from the proper time span. For the other task, images for the summary were selected as described earlier.

Tasks in category 5 forced users to operate in summary mode, allowing us to examine whether our summarization made any difference when compared to random excerpting from the collection. We found that even at a confidence level of 99% there was a significant difference between random and clustered summarization. We measured a 56% performance improvement.

5.1 Discussion

We are very encouraged by our findings. These were intentionally tough experiments in that we included subjects who were not deeply familiar with computers and, most importantly, were unfamiliar with the collection. In our main target application of Personal Digital Libraries users will know the collection, so summaries and time organization will have even greater positive impact on these users.

We are also delighted by the positive results, because the systems we report on here operate under the most demanding assumption that users invest absolutely no organizing effort. We can expect that even small user efforts, such as high-level captioning will improve upon these results even further.

However, users did not always take advantage of the summarization capabilities. Overall, only 42% of users took advantage of summarization in our Hierarchical and Calendar Browsers. This tendency to overlook the summarization feature was particularly evident when subjects needed to overcome the barrier of abstraction that we erected by the verbal nature of category 2 tasks (finding as many images as possible from 10 textual descriptions). In that case, subjects were also strained by trying to keep multiple image descriptions in mind simultaneously. This strain may have contributed to their choice of the simplest, linear, search strategy.

The fact that 92% of the subjects had a specific time association with certain photographs (Category 5) underscores the importance of making time visible in the interface. Most browsers do not do this; PhotoMesa and the two browsers we developed are notable exceptions [11].

The ingenuity of many subjects impressed us. Some, without computer science training, used binary search algorithms. Others paid attention to the photographic content's connection to the physical world. For example, one subject explained how she looked for sunset images by using the timestamps visible in the tree control of the Hierarchical Browser: "I used the time for the sunset images because I knew I was looking for something late in the day." It is seductive to use some of these observations as the basis for adding search and browse features. However, it is very difficult to keep the interface simple enough that computer novice and expert users can operate the systems with virtually no instruction.

During the debriefing sessions, subjects were clear in their evaluation of the Scrollable Browser: “It seemed like I did a lot of scrolling, when there should have been a more efficient way [to navigate the images].” “The linear one [SB] was more frustrating than the others. You literally felt like you were looking for a needle in a haystack.”

Opinions about the Hierarchical vs. the Calendar Browser were divided. Praising HB: “I like having the orientation of some sort of text on screen;” or “I liked the one [browser] with the tree. It was the easiest to navigate. You could skip around all at once without having to go through the different levels.” On the other hand, other subjects preferred the Calendar Browser: “I'd rather click on a picture than a month [label];” and “The zoom-out zoom-in was more intuitive [than HB]. You didn't have to think about it as much.” Although for testing purposes we kept the HB and CB interfaces separate, we plan to integrate these interfaces in the future, based on the positive response we received from users in both cases.

6 Conclusion

We designed, implemented, and tested two photo browsers that can accommodate thousands of images. These systems are intended primarily for personal digital libraries. Both browsers are built atop a common architecture. This architecture consists of metadata extraction, cluster engine, and summarization modules that are replaceable.

Our browsers use cluster analysis of the times when photographs were taken as the foundation for summarizing large subsets of the images within a collection. The browsers differ in how users navigate the collections.

We conducted user studies that compared various design aspects of the two browsers against a commercial image browser. We found that our Calendar Browser, which affords time navigation through direct-manipulation, enabled a 33% improvement in speed of finding given images in collections. We found that our summarizations improve the search for photographs from textual descriptions by 56% over systems that fill the screen with randomly selected excerpts from the collection.

While summarization clearly improved performance, only 42% of subjects thought of making use of the advantages that summarization provided. We plan to address this issue in subsequent designs.

Organizing principles other than time can, of course, be incorporated into our browser designs. Most notably, image analysis can generate additional complementary clustering, for example, by the presence or absence of faces in images [15], by indoor versus outdoor photography, or by image similarity [16].

Our experiments show that we will be able to make Personal Digital Libraries of digital images manageable and convenient to browse. Digital cameras are rapidly creating large professional and personal image collections. Without substantial support from Digital Library technologies, this exciting new opportunity will be wasted, and turn into the digital version of the unorganized shoebox of photographs in a closet. With the proper tools, on the other hand, the records of individual histories will be fun and instructive to maintain and explore.

Acknowledgments

We are grateful to the JCDL conference reviewers, who provided detailed and thoughtful feedback to our earlier, condensed version. We equally thank the subsequent reviewers of this extended chapter.

The Association of Computing Machinery (ACM) is leading the way towards author-friendly copyright agreements. We thank them for their agreement's built-in facilities that allow authors to build on their own initial work without complications. In our case, [17] is an earlier, condensed version of this chapter.

We are deeply appreciative of our subjects, who mostly live packed lives and have no need to sit through photo browser experiments to fill their days. The many, many minutes and seconds they spent with us turned into all the wonderful data points that make this chapter possible.

References

1. Wactlar, H.D., Christel, M.G., Gong, Y., and Hauptmann, A.G. Lessons Learned from the Creation and Deployment of a Terabyte Digital Video Library, IEEE Computer 32(2): 66-73 (1999).
2. Kobus, B., and Forsyth, D.: Exploiting Image Semantics for Picture Libraries. In *Proceedings of the First ACM/IEEE-CS Joint Conference on Digital Libraries*. ACM Press, 2001.
3. Frew, J., Aurand, M., Buttenfield, B., Carver, L., Chang, P., Ellis, R., Fischer, C., Gardner, M., Goodchild, M., Hajic, G., Larsgaard, M., Park, K., Probert, T., Smith, T., and Zheng, Q.: The Alexandria Rapid Prototype: Building A Digital Library for Spatial Information. In *Advances in Digital Libraries '95*, 1995.
4. InfoTrends Research Group: Global digital camera market size as sales in units from 1999 through 2001 and forecast for 2002, 2004, and 2006. Salomon Smith Barney. March 18, 2002.
5. International Data Corporation (IDC): Global, US, Japan, Western Europe and Asia Pacific shipments comparison of simple VGA, point and shoot, professional mobile and studio digital camera shipments in units for 2000 and 2001, and forecast from 2002 through 2006, with percent compound annual growth rates. Salomon Smith Barney. May 31, 2002.
6. Microsoft Windows XP. 2001. Information at http://www.microsoft.com/windowsxp/.
7. Apple Macintosh OS X. Information at http://www.apple.com/macosx/.
8. ACD Systems ACDSee Browser. Available at http://www.acdsystems.com.

9. Kang, H. and Shneiderman, B.: Visualization Methods for Personal Photo Collections Browsing and Searching in the PhotoFinder. In *Proceedings of IEEE International Conference on Multimedia and Expo (ICME2000)*, pp. 1539-1542. IEEE, New York, 2000.
10. Kuchinsky, K., Pering, C., Creech, M.L., Freeze D., Serra, B., and Gwizdka, J.: FotoFile: a consumer multimedia organization and retrieval system. In *Proceedings of the Conference on Human Factors in Computing Systems CHI'99*, pp. 496-503, 1999.
11. Bederson, B.J.: PhotoMesa: A Zoomable Image Browser Using Quantum Treemaps and Bubblemaps. Proceedings of the 14th Annual ACM Symposium on User Interface Software and Technology. Orlando, Florida. Pages 71-80. November, 2001. ACM Press. Series-Proceeding-SESSION-Artic.
12. Apple iPhoto. 2002. Information at http://www.apple.com/iphoto/.
13. Platt, J.C.: AutoAlbum: Clustering Digital Photographs Using Probabilistic Model Merging. In *Proceedings of the IEEE Workshop on Content-Based Access of Image and Video Libraries 2000*, pp. 96-100, 2000.
14. Platt, J.C., Czerwinski, M., and Field, B.: "PhotoTOC: Automatic Clustering for Browsing Personal Photographs", Microsoft Research Technical Report MSR-TR-2002-17, http://research.microsoft.com/scripts/pubs/view.asp?TR_ID=MSR-TR-2002-17
15. Gross, R., Cohn, J., and Shi, J.: Quo Vadis Face Recognition. Third Workshop on Empirical Evaluation Methods in Computer Vision, IEEE Conference on Computer Vision and Pattern Recognition 2001 (CVPR'01), Hawaii, December 11-13, 2001.
16. Rodden, K., Basalaj, W., Sinclair, D., and Wood, K.: Does Organisation by Similarity Assist Image Browsing? In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems 2001, pp. 190 - 197, 2001.
17. Graham, A., Garcia-Molina, H., Paepcke, A., and Winograd, T.: Time as Essence for Photo Browsing Through Personal Digital Libraries. In Proceedings of 2nd Joint Conference on Digital Libraries (JCDL) 2002.

Accessing News Video Libraries through Dynamic Information Extraction, Summarization, and Visualization

Michael G. Christel

Carnegie Mellon University, Computer Science Department and HCI Institute,
5000 Forbes Ave., Pittsburgh, PA 15213, USA
christel@cs.cmu.edu
http://www.cs.cmu.edu/~christel

Abstract. The Informedia Project has developed and evaluated surrogates, summary interfaces, and visualizations for accessing a digital video library containing thousands of documents and terabytes of data. This chapter begins with a review of Informedia surrogates for a single video document, including titles, storyboards, and skims. Incorporating textual elements, considering user context and emphasizing phrases over words have all led to better video surrogates. These lessons have driven the development of direct manipulation interfaces and visualization strategies for exploring news video libraries. Summarization strategies employed in the Informedia Project for sets of broadcast news video documents are discussed, concluding with the research challenges in building visual interfaces from vast quantities of image data and imperfect automatic video processing.

1 Introduction

The Informedia Project at Carnegie Mellon University has created a multi-terabyte digital video library consisting of thousands of hours of video, segmented into over 100,000 stories, or documents. Since Informedia's inception in 1994, numerous interfaces have been developed and tested for accessing this library, including work on multimedia abstractions or surrogates that represent a video document in an abbreviated manner [1, 2]. The interfaces, including video surrogates, build from automatically derived descriptive data, i.e., metadata, such as transcripts and representative thumbnail images derived from speech recognition, image processing, and language processing. The term "document surrogate" is used in the information retrieval community to label information that serves as the representation for the full document, such as a title, abstract, table of contents, set of keywords, or combinations of these descriptors. In this chapter "video surrogate" is used to label the set of text, image, audio and video that can serve as a condensed representation for the full video document.

K. Börner and C. Chen (Eds.): Visual Interfaces to Digital Libraries, LNCS 2539, pp. 98–115, 2002.

This chapter begins with a review of different types of video surrogates developed by the Informedia Project, discussing their evaluation and evolution. While surrogates summarize a single video document, as the video library grew there was a corresponding need to represent sets of video documents. Visualization strategies were employed to summarize sets of documents and allow interactive navigation. These strategies will be presented following the surrogate review, leading into a discussion of video collages and next steps planned for the Informedia library work. While the Informedia corpus includes broadcast news, documentaries, classroom lectures, and other video genres, this paper will focus on interfaces for broadcast news. Video preview features that work well for one genre may not be suitable for a different type of video [3], and while the discussion here for news may apply equally well for visually rich genres like travel and sports videos, other genres like classroom lecture and conference presentations may need to emphasize audio, dialogue text, and other unique attributes.

2 Informedia Surrogates

Video is an expensive medium to transfer and view. MPEG-1 video, the compressed video format used in the Informedia library, consumes 1.2 Megabits per second. Looking through an hour of candidate video for relevant material could take an hour of viewing time and require downloading over 500 Megabytes of information. Surrogates can help users focus on precisely which video documents are worth further investigation and where to focus attention within those documents, reducing viewing and video data transfer time.

2.1 Thumbnail and Title Surrogates

Consider a user interested in refugee and human rights stories from a corpus of 2001 news. The query produces over 1000 results in the Informedia library of CNN news video during this period. The most relevant 1000 are returned in a series of pages that the user can scroll through and inspect. Figure 1 shows a portion of the first results page holding 30 documents, where each document is represented by a brief title and a single thumbnail image overview. As the user moves the mouse cursor over a document representation, its title is displayed in a pop-up menu.

The layout of Figure 1 communicates the relative relevance of each document to the query as determined by the text search engine, the contribution of each query word for each document (i.e., which terms matched which documents and by how much), a contextual thumbnail image representation, a brief title automatically produced for the document, and the document's play length and broadcast date. In Figure 1 the documents are sorted within the page by relevance, with the pages sorted by relevance as well. Sorting both within the page and for the full set of pages could also be specified by date or size, with other aspects of the display changing as well to emphasize the focus on attributes such as the relative distribution of results across dates [2, 4].

The vertical bar to the left of each thumbnail indicates relevance to the query, with color-coding used to distinguish contributions of each of the query terms. The document surrogate under the mouse cursor, the seventh result, has its title text displayed in a pop-up window, and the query word display is also adjusted to reflect the document under the cursor, as described in [2]. From the query "refugee hunger human rights political asylum" this seventh document matches only "refugee", "rights", and "asylum" with the other query terms grayed out (as shown in Figure 1) to indicate that this document did not mention "hunger", "human" or "political." The search engine uses suffixes and stemming to consider variants of these words as well, e.g., "hungry" and "politics" did not occur in this seventh result. The vertical relevance bar, shown at increased resolution in Figure 2, shows that this document has a relevance score of 30/100 for the given query, with matches on "refugee" contributing a little more than half of the score, matches on "asylum" contributing a bit more than a quarter, and matches on "rights" contributing the remaining score. Other interfaces with a temporal element, such as the storyboard interface and video playback window discussed further below, add views reflecting the distribution of these match terms within the video.

The utility and efficiency of the layout shown in Figure 1 have been reported in detail elsewhere [2, 4], validated through a number of usability methods, including

Fig. 1. Thumbnail results page for Informedia query, with title shown for the 7th result

transaction log analysis, contextual inquiry, heuristic evaluation, and cognitive walk-throughs. In particular, a formal empirical study was conducted to determine the relative merits of such thumbnail menus of results versus similar text menus of titles, document durations and broadcast dates [5]. 30 high school and college students participated in an experiment using a fact-finding task against a documentary video corpus, where dependent measures included correctness, time to complete the task, and subjective satisfaction. That study found that when the thumbnail image is chosen based on the query context, users complete the task more quickly and with greater satisfaction with the interface than when using text menus or a context-independent thumbnail menu, in which each document is always represented by the same thumbnail image of the first shot in the document. Hence, thumbnail surrogates chosen based on context produce a more efficient visual interface.

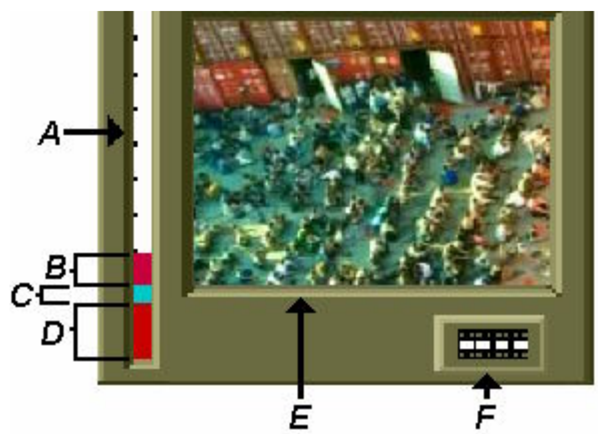


Fig. 2. Large-scale view of thumbnail representation of each video document returned by a query, with relevance bar showing score on a scale of 0 to 100 (*A*), the relevance score for this given result being 30 (30 out of 100 filled in via *B, C, D* color-coded to match query terms asylum, rights, and refugee respectively), a thumbnail image representing this document (*E*), and an interface button (*F*) to click to show the storyboard view

2.2 Storyboard Surrogates

The automatic breakdown of video into component shots has received a great deal of attention by the image processing community [6, 7, 8, 9, 10]. The thumbnail images for each shot can be arranged into a single chronological display, a storyboard surrogate, which captures the visual flow of a video document along with the locations of matches to a query. From Figure 1's interface, clicking on the filmstrip icon (shown as (*F*) in Figure 2) for a document displays a storyboard surrogate like that of Figure 3.

The storyboard interface is equivalent to drilling into a document to expose more of its visual details before deciding whether it should be viewed. Storyboards are also navigation aids, allowing the user to click on an image to seek to and play the video document from that point forward. For example, the mouse is over the red match notch of the third shot shown in Figure 3, a notch corresponding to the word "refugee" which is shown in a pop-up text area. If the mouse is clicked here, the corresponding video is opened and played from that point 1:30 into the document where "refugee" is mentioned and when the room shown in the third shot thumbnail is visible.

Storyboard displays of a simultaneous, ordered set of thumbnail images date back to the advent of digital video. A number of video libraries have converged on the idea of including a thumbnail image in the storyboard for each shot in the video, including CAETI [9], Pictorial Transcripts [6], and the Baltimore Learning Community [11], backed by early research by Zhang, Aoki, and others [10, 12]. Others suggest that more than one image should be added per shot depending on the composition of the shot determined through motion analysis [13]. Other researchers have implemented subsampling; in which a storyboard image is extracted at evenly distributed intervals across a video [14].

Fig. 3. Storyboard view for 8th result of Fig. 1, showing only matching shots

Contextual thumbnail surrogates, a single image per video document as shown in Figure 1, are derived from the set of images representing each shot in that document that make up the storyboard surrogate. Consider the eighth result document in Figure 1, showing a thumbnail image of a man in front of a white building. The document's duration is 3 minutes and 1 second, from which automatic processing identified 37 shots. Figure 3 shows a storyboard view into this document, reducing the 37 shots to only the eight shots that contain matching words from the query. The search engine determined that the matches within the "man in front of building" shot of Figure 3 contributed the most toward the document's ranking for the given query. Hence, that shot's thumbnail is used to represent the whole document in Figure 1.

An area of active multimedia processing research attempts to reduce the number of thumbnails represented in a storyboard to decrease screen space requirements [7, 9, 15]. One method makes use of video processing confidence metrics to set thresholds for inclusion in interface elements like storyboards [16]. The method illustrated in Figure 3 uses the query context to reduce the shots from 37 to 8 by only displaying the shots containing matches. The “x1”, “x2”, “x4”, and “x8” icons shown in the toolbar of Figure 3 are used to present more shots in the same display space by reducing the resolution of each thumbnail image. Taniguchi et al. present densely packed thumbnail arrangements to maximize the number of storyboard elements in a given amount of display space [14]. Researchers at FX Palo Alto developed packing algorithms for storyboard layout that also resized thumbnails such that more important shots are given increased size [15].

2.3 Temporal Surrogates: Storyboards with Text, and Video Skims

Fairly accurate time-aligned transcripts exist for the Informedia collection of CNN news via capturing their closed-captioning, and then determining when each word was spoken through an automatic alignment process using the Sphinx-III speech recognizer [2], filtering the text into a mixed upper and lower case presentation. This timed text enables the creation of the text-augmented storyboards. Based on prior studies that have shown that the presentation of captions with pictures can significantly improve both recall and comprehension, compared to either pictures or captions alone [17, 18], a text-augmented storyboard should have greater summarization utility.

Indeed, Ding et al. found that surrogates including both text and imagery are more effective than either modality alone [11]. This work was confirmed in a recent study [19], which specifically examined the questions of text layouts and lengths in storyboards. 25 university students and staff members participated in an experiment using a fact-finding task against a news video corpus, where dependent measures included correctness, time to complete the task, and subjective satisfaction. Significant differences in performance time and satisfaction were found by the study. If interleaving is done in conjunction with text reduction, to better preserve and represent the time association between lines of text, imagery and their affiliated video sequence, then a storyboard of images plus text with great utility for information assessment and navigation can be constructed. That is, the transcript text should be time-aligned with thumbnail rows in the storyboard, and then reduced to a set of basic phrases important to the particular query context, as illustrated in Figure 4.

Another type of surrogate that presents information temporally rather than in a single display page is the video skim [2]. Ideally, a video skim presents the gist of a larger video by condensing the audio and image highlights into a new playable presentation having duration of only a fraction of the originating video. An early skim experiment using 48 college students, as well as a follow-up experiment with 25 college students, found that skims for public television documentaries and news need to treat the audio narrative carefully, as much information is conveyed for these broadcast video genres within the audio. In particular, skims created from phrases rather than words, with breaks at audio silence points rather than arbitrary breaks, resulted in performance and subjective satisfaction improvements [20]. The improved utility of using phrases as building blocks rather than words was confirmed in evaluations concerning the title and storyboard surrogates as well [2, 19].

These surrogates are built from metadata automatically extracted by Informedia speech, image, and language processing modules, including transcript text, shot boundaries, key frames for shots, and synchronization information associating the data to points within the video [2]. Incorporating textual elements, considering user context, and emphasizing phrases over words have all led to better video surrogates.

Fig. 4. Storyboard augmented with time-aligned transcript text, reduced based on matching phrases to the originating query, which for this example is "air crash"

While the surrogates were put to use effectively, they were not sufficient to deal with the richness of a growing library. As the Informedia collection grew from tens to thousands of hours, the results set from queries grew from tens to hundreds or thousands of documents. Whereas the query of Figure 1 might have produced 30 results that could all be shown on a single screen in a small corpus, against even one year of CNN news this query produces over 1000 results (with the first few shown in Figure 1). Such a set is too large for efficient linear browsing through pages of ranked documents. Visualization techniques are necessary to provide overviews of the full result set and to enable user-directed inquiries into spaces of interest within this result set.

3 Informedia Visualizations

The three main visualization techniques employed in the Informedia library interface are:

- Visualization by Example (VIBE), developed to emphasize relationships of result documents to query words [21].
- Timelines, emphasizing document attributes to broadcast date.
- Maps, emphasizing geographic distribution of the events covered in video documents [1].

Each technique is supplemented with dynamic query sliders [22], allowing ranges to be selected for attributes such as document size, date, query relevance, and geographic reference count. Consider Figure 5, which shows the VIBE overview of the full 1000 results for the query of Figure 1, with the six query words being anchors for the plot. Moving the "refugee" anchor replots all 70 documents that match to this word. The distribution of plot points on this VIBE grid conveys the match term/document relationships for the full set, whereas the relevance bar and query term coloring shown in Figures 1 and 2 conveyed the match terms for only a single document or page of documents. For example, there are a great number of documents

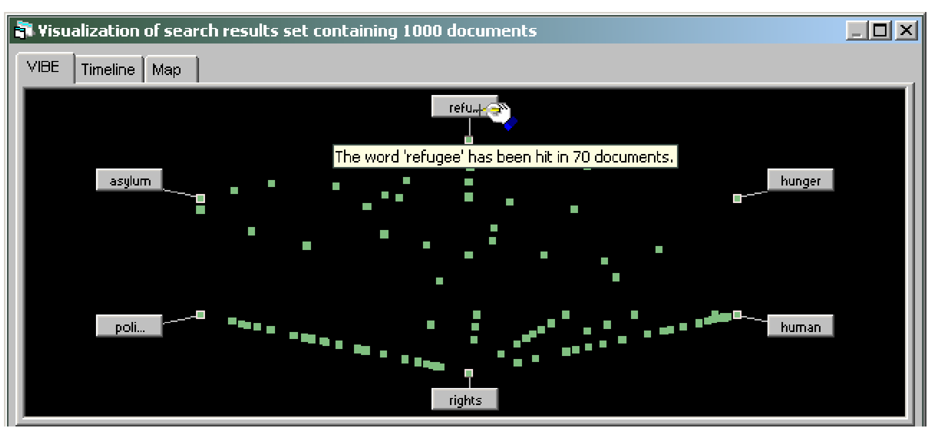


Fig. 5. VIBE scatter plot visualization plotting the 1000 video documents to the 6 query terms of Fig. 1.; the mouse cursor is over the "refugee" anchor where tooltips text informs the user that 70 documents count this term among their matches

matching both "political" and "rights" and hence plotted between those two anchor points, but no documents that match both "human" and "hunger" without also matching at least one other term (since the space on the line between the right-most anchor terms "human" and "hunger" is empty).

The VIBE visualization shown here conveys semantics primarily through positioning, but could be enriched to overlay other information dimensions through size, shape, and color, as detailed elsewhere for the Informedia library [4]. VIBE allows users unfamiliar or uncomfortable with Boolean logic to be able to manipulate results based on their query word associations. For video documents such as a news corpus, there are other attributes of interest besides keywords, such as time and geography. Consider the timeline and dynamic query sliders of Figure 6, which starts from the same working set of 1000 documents used in Figure 5, but drops out all documents of relevance score lower than 7 through the relevance query slider. The remaining 110 documents are plotted by relevance against time, with color-coding by size. As the user moves the mouse over the plot, the distribution of the documents under the

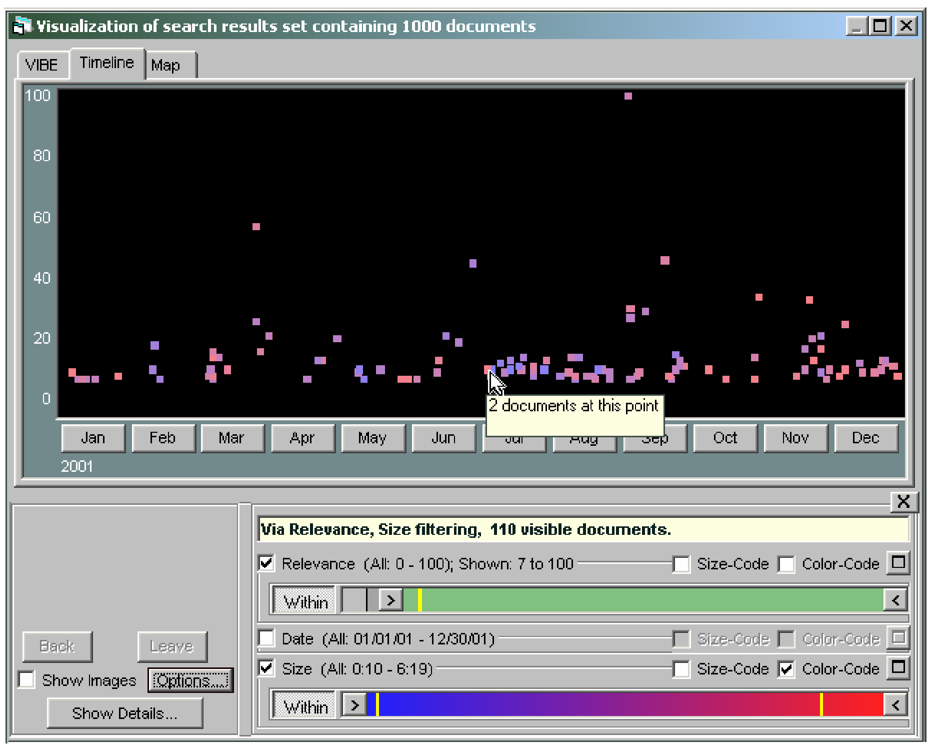


Fig. 6. Timeline view of 1000 documents, reduced to 110 in active view via setting the relevance slider to scores >= 7; the distribution of relevance and size for the two documents at the mouse cursor point is also shown on the two active sliders

mouse is reflected in the sliders. In this simple case, there are two documents under the mouse cursor, both scoring with relevance around ten and having diverse sizes: one around twenty seconds and the other over five minutes, as shown by the yellow marker lines on the relevance and size sliders. These distribution lines on the sliders change with the mouse focus, to give more information pertaining to the documents plotted underneath the mouse cursor point. The sliders also give overall information, e.g., that relevance scores range from 0 to 100 and that the size of documents in this results set range from 10 seconds to 6 minutes and 19 seconds.

To complete the illustration of visualizers against the query set of Figure 1, consider the map plot of Figure 7. The relevance slider is used to restrict the plot to only documents with scores >= 20. The date slider is used to inspect documents from the first quarter, between January 1 and March 31, 2001. As the sliders move, the map replots immediately, giving visual feedback to the user regarding the countries remaining in focus. For the query "refugees hunger human rights political asylum", there are stories pertaining to Canada, Indonesia, Netherlands, Uganda and the United States in this CNN corpus for the first quarter of 2001. For news data, plotting results

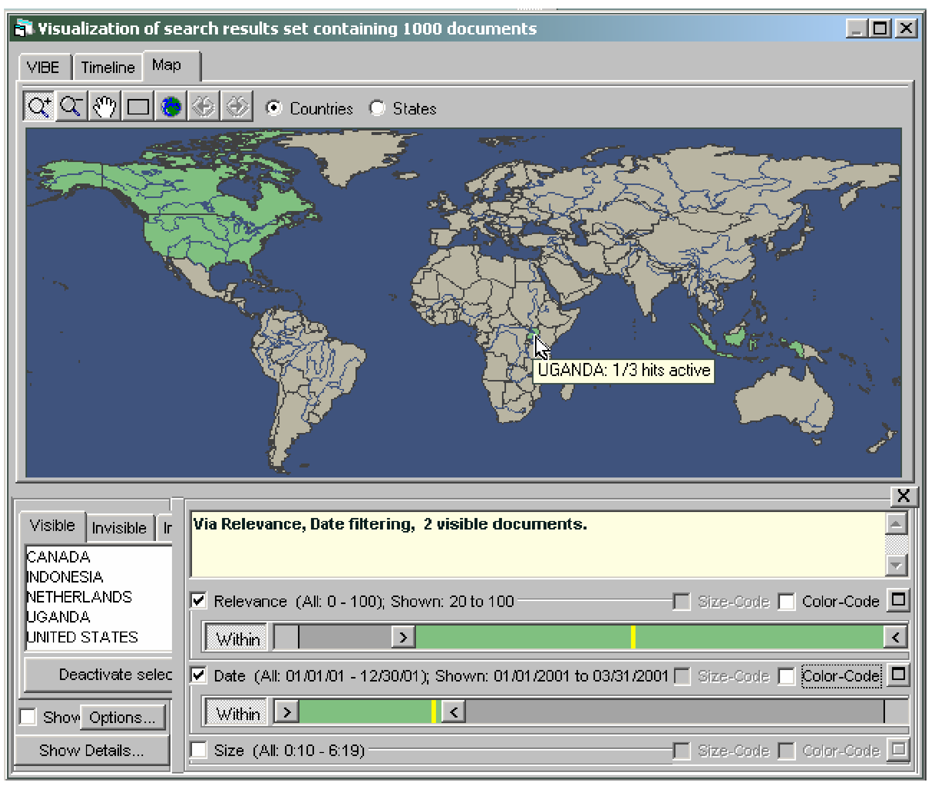


Fig. 7. Map visualization of documents remaining from query of Figure 1 when looking at relevance scores >= 20, date between January 1 – March 31, 2001

against the query terms, time, or geography allows the important dimensions of what, when and where to be controlled and seen by the user. The documents' location references are identified automatically and stored with the video document metadata for use in creating such maps [1]. Timelines and maps are used in the Perseus Digital Library as well for visualizing the geographic and temporal references in its Greco-Roman and London collections [23].

Figure 7 illustrates the use of direct manipulation techniques to reduce the 1000 documents in Figure 1 to the set of 2 documents under current review. By moving the slider end point icon with the mouse, only those documents having relevance ranking of 20 or higher are left displayed on the map. As the end point changes, so does the number of documents plotted against the map, e.g., if Brazil only appeared in documents ranked with relevance score 19 or lower, then Brazil would initially be colored on the map but drop out of the visible, colored set with the current state of the slider shown in Figure 7. Similarly, the user could adjust the right end point for the slider, or set it to a period of say three months in length in the date slider and then slide that three month "active" period from January through March 2001 and see immediately how the map animates in accordance with the active month range. The dynamic query slider is shown in more detail in Figure 8.

By combining multiple techniques, users can refine large document sets into smaller ones and better understand the result space. For example, the 1000 documents of the query in Figure 1 produce the timeline plot shown in Figure 6. By dragging a rectangle bounding only some of the plotted points representing stories, the user can reduce the result set to just those documents for a certain time period and/or relevance range. For more complex word queries, the VIBE plot of documents to query terms can be used to understand the mapping of results to each term and to navigate perhaps to documents matching 2 words but not a third from the query [21].

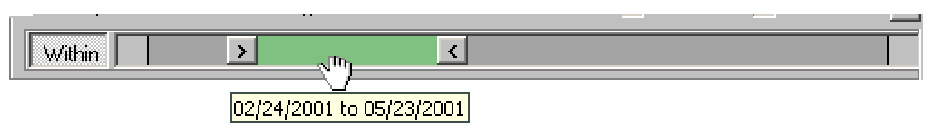


Fig. 8. Dynamic query slider, with adjustable end points, inverse operation via the "Within" button, optional color coding, and tooltips and labels to communicate its current state

Initial interfaces for the Informedia digital video library interface consisted of surrogates for exploring a single video document without the need to download and play the video data itself. As the library grew, visualization techniques such as maps, timelines, VIBE scatter plots, and dynamic query sliders were incorporated to allow the interactive exploration of sets of documents. These visualization techniques allow the user efficient, effective direct manipulation to interact with and change the information display. However, they did not take advantage of the visual richness of the material in the video library. For the Informedia CNN library spanning 5 years of broadcasts, over 1.7 million shots are identified with an average length of 3.38 seconds, with each shot represented by a thumbnail image as shown in Figure 3. Video documents, i.e., single news stories, average 110 seconds in length, resulting in an average image count for document storyboards of 32.6. The next section outlines investigations underway to make use of these shots in summary and navigation interfaces, along with text descriptors that add utility to the VIBE, timeline, and map visualizations presented in the preceding figures.

4 Informedia Collages

Shahraray notes that "well-designed human-machine interfaces that combine the intelligence of humans with the speed and power of computers will play a major role in creating a practical compromise between fully manual and completely automatic multimedia information retrieval systems" [24]. The power of the interface derives from its providing a view into a library subset, where the user can easily modify the view to emphasize various features of interest. A *video collage* is a surrogate with such interactive features, representing multiple video documents through extracted metadata text, images, audio and video [25].

Video collages let the user browse the whole result space without having to resort to the time-consuming and frustrating traversal of a large list of documents. Video collages are generated dynamically based on context, providing a synopsis of the data

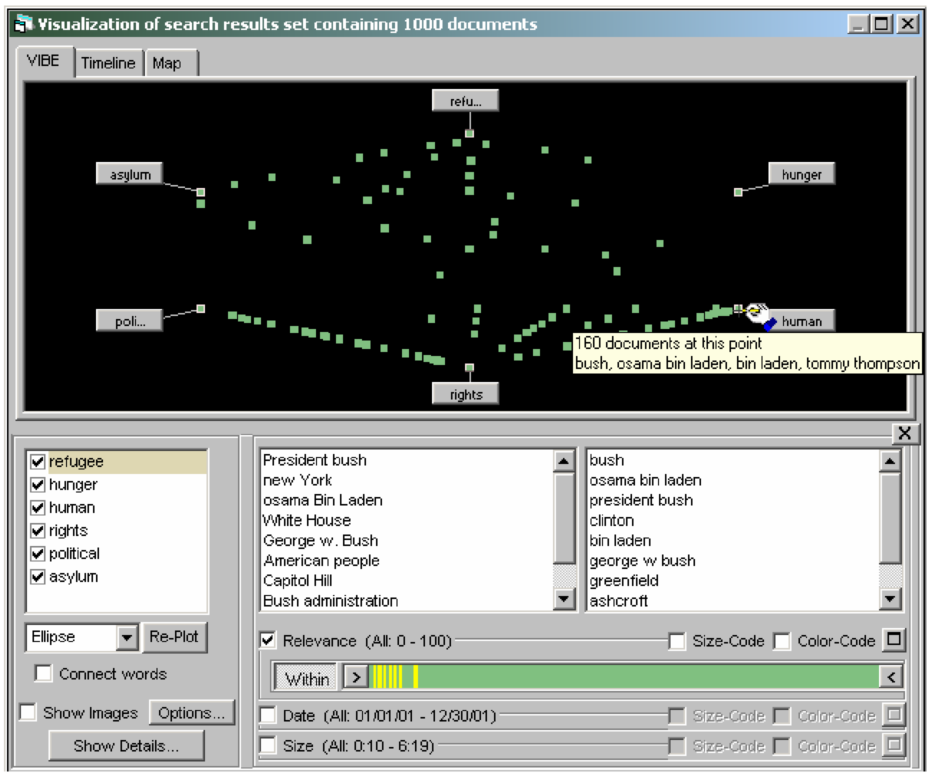


Fig. 9. VIBE collage showing common phrases and common people across the 1000 plotted documents, along with common people at the mouse focus point (160 documents matching only "human")

within a collection of stories. The descriptive data, or metadata, for the stories is produced automatically through Informedia processing. For example, the text transcript can be produced through speech recognition on the audio narrative, and faces are identified through computer vision on the video. Other metadata include overlay text, geographic entities, camera shots, and images representing each shot [1, 2].

Collages were designed to support Shneiderman's Visual Information Seeking Mantra [26]: "Overview first, zoom and filter, then details-on-demand." Collages as overviews allow the rich information space covered by a set of news video documents to be better appreciated and understood. The user can "zoom in" to focus points of interest, such as documents containing particular query words through the use of Visualization by Example (Figure 9), specific time occurrences via timelines (Figure 10), or certain geographic areas via map interaction (Figure 12). Prior work in interacting with VIBE, maps, and timelines has been extended with the introduction of

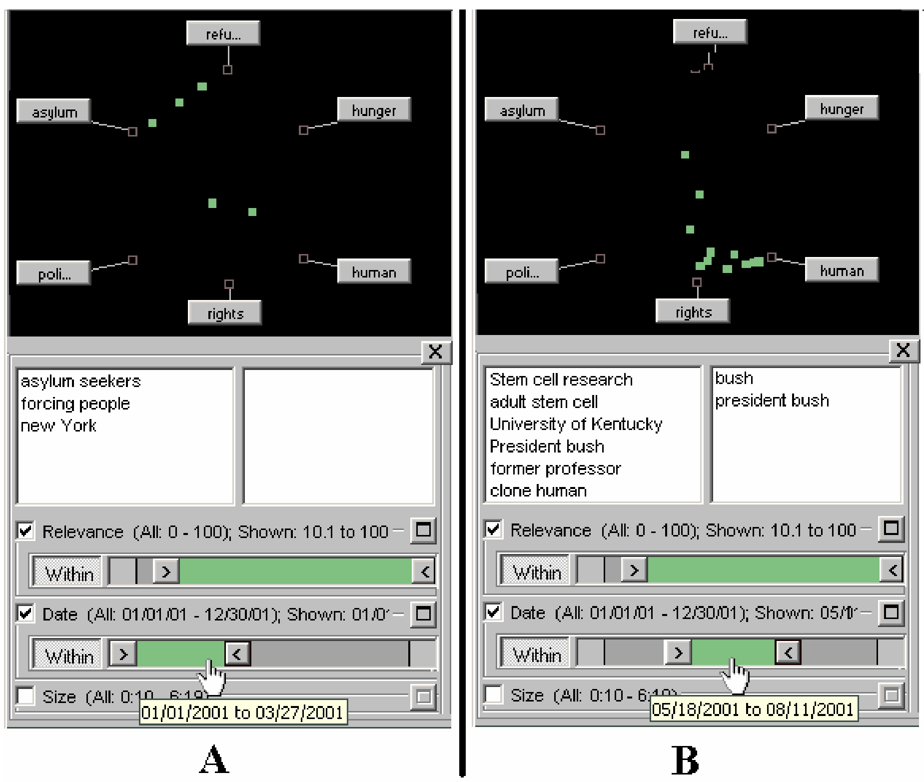


Fig. 10. Two VIBE collages, the left (*A*) showing matches to refugees and asylum early in 2001, with the right (*B*) showing dominance of human rights and stem cell research stories from May to August, 2001

thumbnail images and text phrases into the collage presentation. This section illustrates how these additional attributes improve the utility of the collage when used in an interactive fashion, with Figures 9 – 12 showing collages for the same 1000 documents used in earlier figures.

Consider again the 1000 documents returned from the "refugee hunger human rights political asylum" query shown in earlier figures and densely plotted in Figures 5 and 6. If thumbnails for all the shots in these 1000 documents were to be shown, over 20,000 images would have to be considered. If all the transcript text for these documents were to be shown, over a million characters would be needed. Fortunately, early work with single video document surrogates showed the advantages of using query context to reduce the number of shots considered for representative thumbnails, and other work with titles, storyboards and skims showed the utility of phrases rather than words for surrogates [2, 19]. Rather than considering all the shots, only shots containing matches are considered for inclusion in the collage. Rather than considering all the text, the most common phrases are considered. In addition, through named entity extraction [25, 27], the text is further processed into people, places, and organizations, and the most common of these are considered in the collage. Further refinement is of course possible and desirable to tailor the collage to particular video genres, e.g., for news video the anchorperson sitting the studio rarely offers additional visual information, and so thumbnails of field footage should get preference over anchorperson thumbnails [25].

Common phrases across all the documents in view, and common entities such as the common people across the documents in view, can be displayed along with the plot, as is done in Figure 9, which serves as an enhancement of Figure 5. In addition, common people are shown with the tooltips regarding the documents plotted at the mouse cursor location, letting the user focus inspection to certain areas of the visualization. In the case of Figure 9, the documents that mention only "human" have "Bush", "Osama bin Laden", "bin Laden", "Tommy Thompson" as the most frequent people. Such a list shows an area of improvement needed for collages, to collapse names of the same person into a single representation. Even with its current faults, the tooltips and text provide enough information to support user-directed inquiry and drill-down revealing of detail. For example, by interacting with the VIBE plot to show only the documents plotted at "human" and then selecting to "Show Details" for "Tommy Thompson", you would get 5 documents plotted in a grid as shown in Figure 1. The grid details would reveal that he is the U.S. Health and Human Services Secretary, along with providing a photograph of Thompson.

In addition to providing more detail, the text provides additional description supporting direct, dynamic manipulation of the visualization. Dragging the date relevance slider, for example, can show the change in topics (via common phrases), and key people over time, as shown in Figure 10. With an increased display area per document and reduced number of documents in an active set, more information can be shown for each story cluster. Rather than plot with points or rectangles, thumbnail imagery can be used, as shown in Figure 11, which corresponds to Figure 10 (*A*). Evaluation work like that conducted for Informedia surrogates will need to be performed in order to determine the utility of video collages for navigating, exploring, and collecting information from news libraries.

Other research activities associated with video collages include meaningful clustering mechanisms for the images, accounting for errors and varying degrees of confidence in underlying automatic speech and image processing, and expanding interfaces to incorporate audio and temporal "motion" video as was done with the video skim surrogate. In the near term, text-clustering engines like Vivísimo show promise for organizing text phrases [28]. Image organization and classification will likely rely primarily on its associated, synchronized dialogue transcript, but gradually as more automatic identifiers are developed for video content some domain-specific rules may

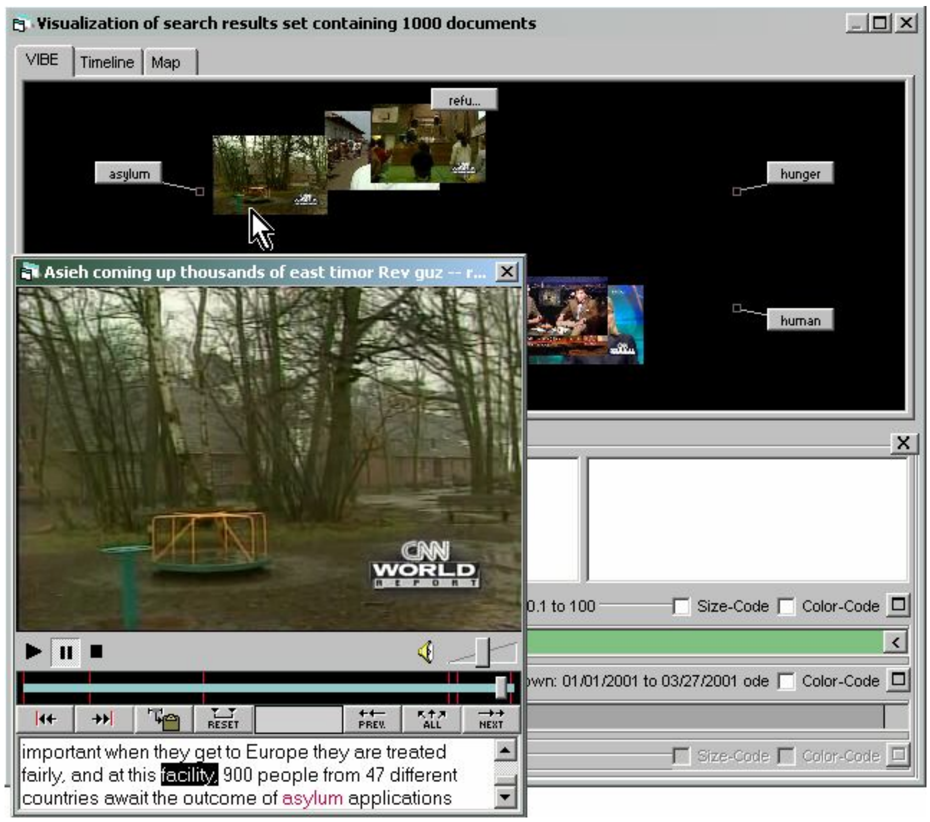


Fig. 11. VIBE collage with thumbnail image overlays, where click on image queues up video document to the point where the thumbnail image was extracted

help with better representing image clusters. For example, work continues on identifying faces within the video library, so the user could choose to emphasize shots with faces, or perhaps shots with no faces.

Font, color, layout, and representation choices all contribute to the effectiveness of collages as used in Figure 11, where the collage could be printed out and used effectively as briefing material regarding vast amounts of video. Figure 12 shows images overlaid on a map, in which the images are tiled to the center of the countries that they represent. The tiled interface does not obscure one image with another, but suffers the disadvantages of covering up small countries with the image and allocating more images to countries with larger areas. Hence, more investigation into the integration of imagery into video collages is necessary.

The collage as a single, viewable representation of information can be greatly enhanced by incorporating expertise from the field of information design. The work of Edward Tufte discusses techniques for improving the persuasive power and coherence of such presentations. In addition, given that the source material is video, collapsed video and audio snippets may enhance the collage's value as briefing material,

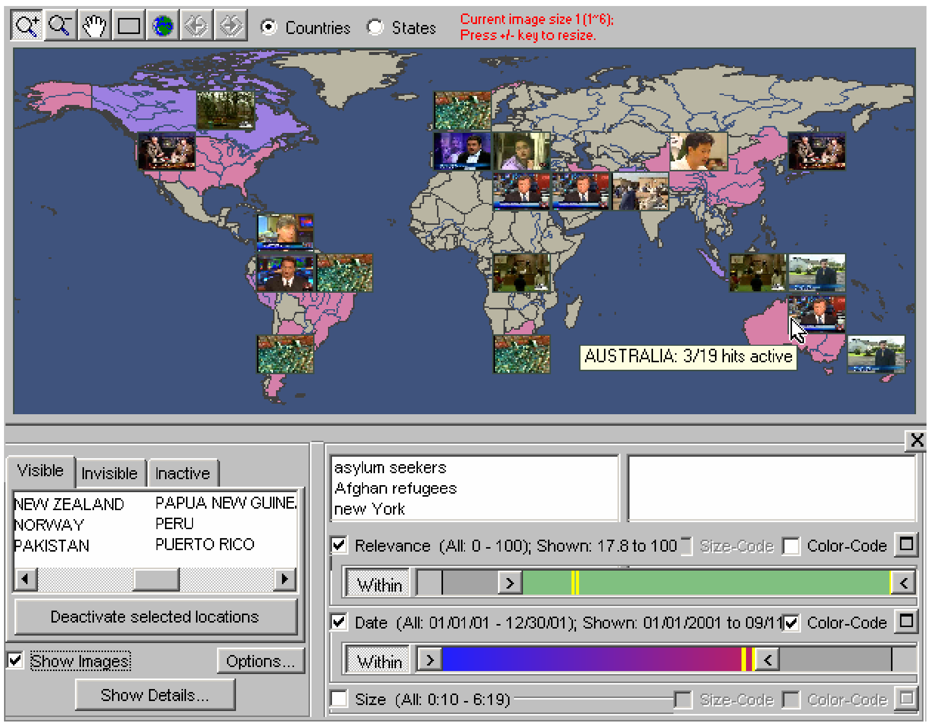


Fig. 12. Map collage with overlaid thumbnails tiled onto the map display

where the collage can be played as an "auto-documentary" covering a set of video material. Earlier work on video skims [20] contributes to such an effort.

Surrogates have proven advantages for allowing information from a video document to be found and accessed quickly and accurately. Visualization techniques addressing text corpora and databases have been shown to be applicable to video libraries as well. Future digital video library interfaces that summarize sets of video documents and leverage from the library's multiple media and rich visual nature need to be designed and evaluated so that the wealth of material within such libraries can be better understood and efficiently accessed.

Acknowledgments

This material is based on work supported by the National Science Foundation (NSF) under Cooperative Agreement No. IRI-9817496. This work is also supported in part by the Advanced Research and Development Activity (ARDA) under contract number MDA908-00-C-0037. CNN supplied video to the Informedia library for sole use in research. CNN's video contributions as well as video from NASA, the U.S. Geological Survey, and U.S. Bureau of Reclamation are gratefully acknowledged. More details about Informedia research can be found at http://www.informedia.cs.cmu.edu/.

References

1. Christel, M., Olligschlaeger, A.M, and Huang, C. Interactive Maps for a Digital Video Library. *IEEE MultiMedia* 7(2000), 60-67
2. Wactlar, H., Christel, M., Gong, Y., and Hauptmann, A. Lessons Learned from the Creation and Deployment of a Terabyte Digital Video Library. *IEEE Computer* 32(1999), 66-73
3. Li, F., Gupta, A., et al. Browsing Digital Video. In *Proc. ACM CHI '00* (The Hague, Neth., April 2000), 169-176
4. Christel, M., and Martin, D. Information Visualization within a Digital Video Library. *Journal of Intelligent Information Systems* 11(1998), 235-257
5. Christel, M., Winkler, D., and Taylor, C.R. Improving Access to a Digital Video Library. In *Human-Computer Interaction: INTERACT97*, Chapman & Hall, London, 1997, 524-531
6. Cox, R.V., Haskell, B.G., Lecun, Y., Shahraray, B., and Rabiner, L. Applications of Multimedia Processing to Communications. *Proc. of IEEE* 86(5) (May 1998), 754-824
7. Lienhart, R., Pfeiffer, S., and Effelsberg, W. Video Abstracting. *Comm. ACM* 40(12), 1997, 54-62
8. Ponceleon, D., Srinivasan, S., Amir, A., Petkovic, D., and Diklic, D. Key to Effective Video Retrieval: Effective Cataloging and Browsing. In *Proc. ACM Multimedia '98* (Bristol, UK, Sept. 1998), ACM Press, 99-107
9. Yeo, B.-L. and Yeung, M.M. Retrieving and Visualizing Video. *Comm. ACM* 40(12), 1997, 43-52
10. Zhang, H.J., Low, C.Y., and Smoliar, S.W. Video parsing and browsing using compressed data. *Multimedia Tools and Applications* 1(1) (1995), 89-111
11. Ding, W., Marchionini, G., and Soergel, D. Multimodal Surrogates for Video Browsing. In *Proc. ACM Conf. on Digital Lib.* (Berkeley, CA, Aug. 1999), 85-93
12. Aoki, H., Shimotsuji, S., and Hori, O. A Shot Classification Method of Selecting Effective Key-Frames for Video Browsing. In *Proc. ACM Multimedia '96* (Boston, MA, Nov. 1996), ACM Press, 1-10
13. Wolf, W. Key Frame Selection by Motion Analysis. In *Proc. IEEE Int'l Conf. Acoustics, Speech and Signal Processing (ICASSP)*, (Atlanta, GA, May 1996), 1228-1231
14. Taniguchi, Y., Akutsu, A., et al. An Intuitive and Efficient Access Interface to Real-Time Incoming Video Based on Automatic Indexing. In *Proc. ACM Multimedia '95*, (San Francisco, CA, Nov. 1995), ACM Press, 25-33
15. Boreczky, J., Girgensohn, A., Golovchinsky, G., and Uchihashi, S. An Interactive Comic Book Presentation for Exploring Video. In *Proc. ACM CHI '00* (The Hague, Netherlands, April 2000), 185-192
16. Foote, J., Boreczky, J., et al. An Intelligent Media Browser using Automatic Multimodal Analysis. In *Proc. ACM Multimedia '98* (Bristol, UK, Sept. 1998), 375-380
17. Large, A., Beheshti, J., Breuleux, A., and Renaud, A. Multimedia and Comprehension: The Relationship among Text, Animation, and Captions. *J. American Society for Information Science* 46(5) (June 1995), 340-347
18. Nugent, G.C. Deaf Students' Learning from Captioned Instruction: The Relationship between the Visual and Caption Display. *Journal of Special Education* 17(2) (1983), 227-234
19. Christel, M. and Warmack, A. The Effect of Text in Storyboards for Video Navigation. In *Proc. IEEE Int'l Conf. Acoustics, Speech and Signal Processing (ICASSP)*, (Salt Lake City, UT, May 2001)
20. Christel, M., Smith, M., Taylor, C.R., and Winkler, D. Evolving Video Skims into Useful Multimedia Abstractions. In *Proc. ACM CHI '98* (Los Angeles, CA, April 1998), ACM Press, 171-178

21. Olsen, K.A., Korfhage, R.R., et al. Visualization of a Document Collection: The VIBE System. *Information Processing & Management* 29(1) (1993), 69-81
22. Ahlberg, C. and Shneiderman, B. Visual Information Seeking: Tight Coupling of Dynamic Query Filters with Starfield Displays. In *Proc. ACM CHI '94* (Boston, MA, April 1994), 313-317
23. Crane, G., Chavez, R., et al. Drudgery and Deep Thought. *Comm. ACM* 44(5), 2001, 34-40
24. Chang, S.-F., moderator. Multimedia Access and Retrieval: The State of the Art and Future Directions. In *Proc. ACM Multimedia '99* (Orlando, FL, October 1999), 443-445
25. Christel, M., Hauptmann, A., Wactlar, H., and Ng, T. Collages as Dynamic Summaries for News Video. In *Proc. ACM Multimedia '02* (Juan-les-Pins, France,, December 2002)
26. Shneiderman, B. The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations. HCI Lab, Inst. Systems Research, Inst. Advanced Computer Studies, *Dept. of Computer Science Tech. Report CS-TR-3665*, Univ. of Maryland, (July 1996)
27. Miller, D., Schwartz, R., Weischedel, R., and Stone, R. Named Entity Extraction for Broadcast News, in *Proc. DARPA Broadcast News Workshop* (Washington DC, March 1999), http://www.nist.gov/speech/publications/ darpa99/html/ie20/ie20.htm
28. Palmer, C., Pesenti, J., Valdes-Perez, R. et al. Hierarchical Document Clustering of Digital Library Retrieval Results. In *Proc. Joint Conf. on Digital Libraries* (Roanoke, VA, June 2001). See also http://vivisimo.com/

Handwritten Notes as a Visual Interface to Index, Edit and Publish Audio/Video Highlights

Anselm Spoerri

School of Communication, Information & Library Studies
Rutgers University, 4 Huntington Street, New Brunswick, NJ 08901, USA.
aspoerri@scils.rutgers.edu
http://scils.rutgers.edu/~aspoerri

Abstract. Digital libraries aim to make media-rich information accessible to "anyone, anywhere, anytime." However, digital audio and video are difficult to search and share. This paper describes Souvenir, a system which enables people to use their handwritten or text notes to retrieve and share specific media moments. Souvenir enables users to take time-stamped notes on a variety of devices, such as the paper-based CrossPad, the Palm Pilot and standard keyboard devices. Souvenir can segment unstructured handwriting into an effective media index without the need for handwriting recognition. People can use their notes to create hyperlinks to random-access media stored in digital libraries. Souvenir has also web publishing and email capabilities to enable anyone to access or email media moments directly from a web page. This paper presents an overview of Souvenir's functionality and describes its handwriting segmentation as well as note and media synchronization algorithms.

1 Introduction

Digital libraries aim to make media-rich information accessible to "*anyone, anywhere, anytime*." However, users need to find it easy to access, use and share this digital information. They have to be able to *personalize* the contents of a digital media library by creating *annotations* that can be used as a personal access index. Users want *flexibility* in how they can interact with digital libraries: 1) Users don't want to have to learn specific query languages. Instead, they want to be able to use familiar and "loosely structured" interaction modes to access highly structured digital libraries. 2) Users want to use diverse access and annotation devices, such as palmtop, laptop or desktop computers. 3) Users want to create and store their digital media annotations in a distributed fashion on different devices and use this meta-data in their workflow without being held captive by a digital media library.

This paper describes Souvenir which enables people to use what comes easy to them, creating handwritten notes, as an access interface and annotation mechanism to do something that is difficult for most, retrieving and sharing specific media moments in digital libraries. The goal of Souvenir is to offer a flexible and comprehensive way for anyone to share and integrate digital media moments into their workflow.

K. Börner and C. Chen (Eds.): Visual Interfaces to Digital Libraries, LNCS 2539, pp. 116–130, 2002.

People are increasingly using handheld computers to record information. Tablets and devices combining paper and digitizing technology, such as the CrossPad or SmartPad, can be used to create handwritten notes that are captured as digital ink. Souvenir supports a variety of note-taking devices to enable people to use their notes to playback or share specific moments in digital media libraries.

1.1 User Need

Digital media content is growing exponentially and represents the next wave of the Internet. Corporations are adopting streaming media to save time, reduce costs and enrich their communications. Knowledge workers need to be able to integrate specific media moments into their workflow. Consumers are acquiring digital media recording equipment and starting to create personal digital media collections. However, audio/video is difficult to search and share in a personal way [1], [6], [8], [10]. General users do not possess the media editing skills and tools that are currently needed to pinpoint and organize specific media moments. What if people could use existing skills, such as taking notes and editing text, to pinpoint specific media moments and create personalized media presentations?

Souvenir offers users flexibility in how they can interact with digital media: 1) Users can use familiar skills and "loosely structured" interaction modes to access digital media libraries. 2) Users can personalize digital audio/video by creating annotations that can be used as a visual access index. They can use a variety of access and annotation devices, such as palmtop, laptop or desktop computers. 3) Users can create hyperlinks to random-access and integrate specific media moments into their workflow without being held captive by how and where the digital media is stored.

This paper is organized as follows: First, previous work is briefly reviewed. Second, Souvenir's functionality is described. Different scenarios are discussed of how users can take media-enabled notes on a variety of devices. We address the strengths and weaknesses of the different note-taking devices supported by Souvenir. A flowchart shows how Souvenir can convert unstructured handwritten notes into media-enabled web pages. A "segment-oriented" framework is introduced to support both handwritten and text notes as well as the playback of different media recordings in the same Souvenir document. Third, we discuss the special issues that arise when handwritten notes, captured as digital ink, are to be used to index digital media. An algorithm for segmenting digital ink is described that exploits spatial and temporal characteristics of how people write without having to recognize their handwriting. We also discuss the need for multiple and complementary ways to index digital media libraries. Fourth, a flexible mechanism for synchronizing notes and media is described. Finally, we provide an informal evaluation of Souvenir and discuss its effectiveness.

2 Previous Work

Previous research has investigated how to provide easy access to digital media collections [1], [4], [5], [8], [9], [10]. Xerox PARC has developed Tivoli [4] for a note-taker to summarize captured meetings. Classroom 2000 [1] automatically captures classes and integrates them with annotations created during class. Marquee [8], Filochat [10], Dynomite [11] and Audio Notebook [6] have clearly demonstrated how personal note-taking can make audio/video retrieval easier.

The use of digital ink to access digital media has been studied by several research groups [1], [3], [4], [6], [8], [10]. [11]. The Classroom 2000 system [2], [7] employs a simple temporal and spatial heuristic to link handwritten notes with audio recorded at the same time. Chiu and Wilcox [3] present a way to generalize simple heuristics into a more general algorithm using hierarchical clustering.

In terms of automatic methods for indexing digital media, there is the major Informedia initiative that combines advanced speech, language and image understanding technology to transcribe, segment and index large video media repositories [9].

Souvenir is innovative in that it offers a flexible and visual solution for anyone to personalize and retrieve audio/video and create hyperlinks to random-access media to share specific moments with others. Souvenir offers users a seamless way to structure their handwritten notes, combine them with text and to publish both digital ink and text as media-enabled web pages.

3 Souvenir

Souvenir is a digital media annotation tool with web publishing and email capabilities. It enables people to use their handwritten or text notes as a personal and visual audio/video index. Souvenir time stamps the user's note-taking activity and uses these time stamps to synchronize the notes with the timeline of a media recording (see Figure 1). Palmtop, laptop or desktop computers can be used to take handwritten or text notes while digital media is being recorded or played back. Users can take notes during (the playback of) a lecture, interview, meeting or movie. Users can be there in person or remotely via phone, teleconferencing or the Internet. At the same time, the audio/video recording or playback can occur on their palmtop, laptop, and desktop computer or remotely on an Internet server. Souvenir enables users to link their notes to the related media file wherever it is stored. Once linked, users can double click any digital ink or text to play the media at the time the note was taken.

Figure 2 provides a flowchart of how Souvenir can convert "unstructured" handwritten notes into structured data that (a) can be used to query and access a digital media, and (b) can be published as media-enabled web pages. Souvenir's functionality can be summarized as follows: 1) Users can take media-enabled notes in multiple ways: (a) the media is recorded or stored on the same device used for taking notes; thus making it easy to synchronize media and notes internally; (b) media and notes

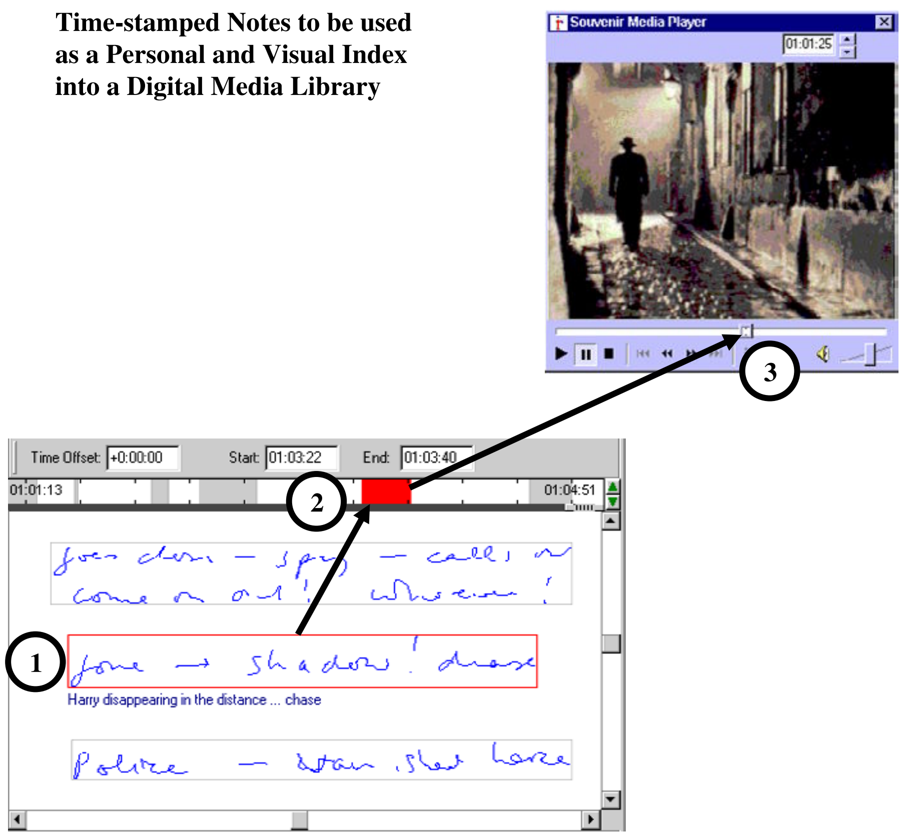
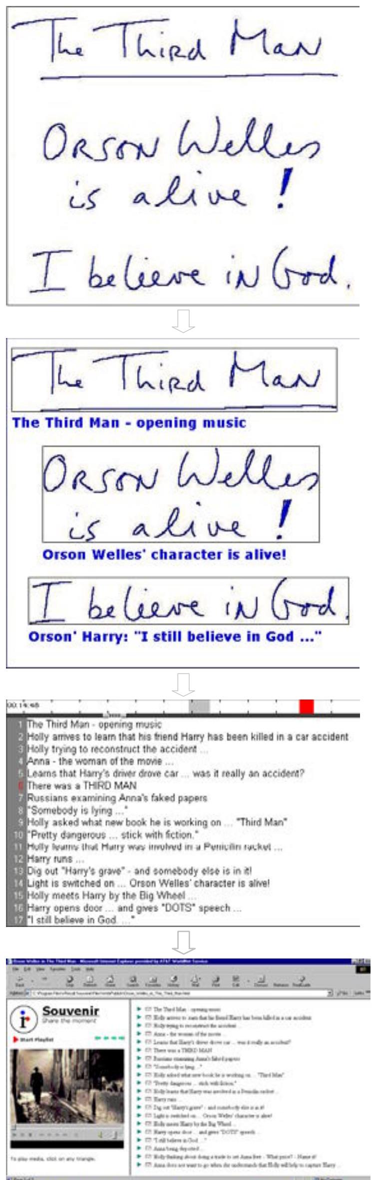


Fig 1: 1) Users take time-stamped notes, which are grouped into segments. 2) Once the timeline of the note activity has been linked to a media recording stored in digital library, a note segment references and indexes a specific media segment on the media timeline. 3) Double-clicking on a note plays the associated media segment. Handwritten notes can visualize in a personal way which parts of a media recording are of interest.

---

are stored on different devices, but they can be synchronized locally; (c) the notes can be synchronized with media hosted on a remote Internet server. 2) The referenced media can be uploaded to or stored in a digital media library. 3) The Souvenir Desktop application offers users a flexible way to link and synchronize their handwritten or text notes via the network with the appropriate media file in a digital library. The Souvenir Desktop enables users to store, synchronize, annotate, edit and publish their notes created by a palmtop, laptop or desktop computer: (a) handwritten notes can be segmented and annotated with text; (b) only the text components of the segments can be displayed and shared with others. 4) Users can publish their Souvenir documents as a set of web pages. 5) Users can email hyperlinks to specific media moments directly from a web page or using the Souvenir Desktop application.

Converting "Unstructured" Handwritten Notes into Media-Enabled Web Pages



Unstructured Handwritten Notes
Handwritten notes are initially unstructured and need to be grouped to enable both digital ink and text in the same Souvenir document.

Media-enabled Notes
Souvenir time stamps the user's note-taking activity so that the notes can be synchronized with the timeline of an audio/video recording.

Segmentation of Handwritten Notes
Souvenir organizes digital ink into segments by exploiting spatial and temporal characteristics of how people write without the need to recognize the digital ink.

Text Annotation of Digital Ink
Users can annotate an ink segment, where the associated text is displayed directly below it.

Media Synchronization
Souvenir can automatically synchronize notes and media if (a) the audio/video is recorded on the same device used for taking notes, or (b) the media playback occurs in the Souvenir Media Player. In all other cases, users need to identify a specific media moment to be linked with a specific note. After that, Souvenir does the rest. Double clicking on the digital ink or text will play the associated media.

Text View of Notes
Users can use the "text only" view to expand their initial, quick notes to create a full report to be shared with others.

Web Publishing & Email Sharing
Souvenir documents can be published as a set of web pages so that others can access or share via email specific media moments directly from a web page. The media owners do not lose control over their content, because users are only creating and sharing pointers to specific media moments instead of actually copying the content.

Fig. 2. shows how Souvenir can be used to convert "unstructured" handwritten notes into structured data that (a) can be used to query and access a digital media library, and (b) can be published as media-enabled web pages.

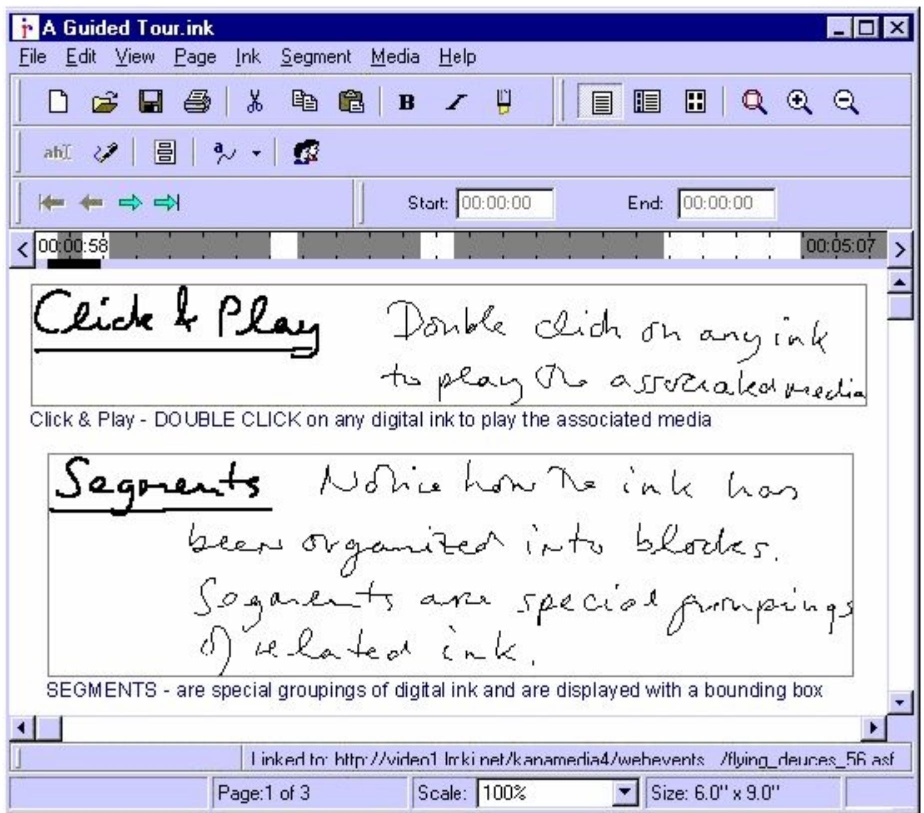


Fig. 3. The *Souvenir Document* window consists of a *media timeline* and a *document display*, which shows digital ink segments surrounded by a bounding box and the associated text directly below. Double-clicking on the digital ink or text plays the associated media. The media timeline shows how the segments, displayed in dark gray, are related to the media.

3.1 Souvenir Desktop

The Souvenir Desktop application enables users to store, synchronize, annotate, edit, publish and share their media-enabled notes that were created using a palmtop, laptop or desktop computer. The Souvenir Desktop lets users synchronize their notes via the network with the appropriate media wherever it is stored. Once synchronized, double-clicking on the digital ink or text plays the associated media. The Souvenir document window consists of a media timeline and a document display (see Figure 3). The media timeline shows how the digital ink and/or text segments (displayed in dark gray on the timeline) are related to the media. Users can interact with the media timeline to fine-tune the relationship between their notes and the associated media. The document display supports three display modes (ink only, text only, and ink & text), thumbnails and different magnifications views.

3.2 Supported Annotation Devices

Souvenir is flexible in terms of the devices that can be used to create time-stamped annotations. Souvenir currently has companion applications for the *Palm Pilot* and the *CrossPad*, a portable digital notepad that digitizes notes written on paper, to enable the capture of handwritten notes as time-coded digital ink. Notes taken with a Palm Pilot or CrossPad need to be uploaded to a PC.

The Souvenir Palm Application enables users to create time-stamped notes, where each note has three components: 1) “Digital Ink” – handwritten notes created by writing with a stylus on the Palm screen. 2) “Text” – created using Graffiti or a keyboard. 3) “Keywords” – up to six keywords can be assigned to a note.

Users can think of a Souvenir Palm note as a “post-it” note on which they can scribble short handwritten notes, add text notes on the backside and categorize it using up to six keywords at the same time. The notes created with the Souvenir Palm application are uploaded to a PC each time the user performs a HotSync and are automatically transferred to the Souvenir Desktop application.

The note-taking devices supported by Souvenir have different strengths and weaknesses in terms of creating media-enabled notes: (a) The CrossPad enables note-taking on paper, which is the preferred way for most people to take notes, and its writing area is large enough to take extensive handwritten notes. But, the CrossPad can not record or playback digital audio or video. CrossPad notes have to be uploaded to the Souvenir Desktop to be linked to the related media recording. Stifelman [6] has developed a digital notepad prototype that can record digital audio, but it could not be successfully commercialized. Further, the CrossPad and digital notepads like it have only been adopted by a small user population so far. (b) The Palm Pilot has been widely adopted because of its effective interface and small size, which in turn makes it difficult to take extensive digital ink notes. This is why the Souvenir Palm application also supports text notes. The Palm Pilot is ideal for taking short handwritten or text notes in a way that is non-intrusive during a meeting or class. (c) Keyboard-based devices are universally used. However, most people can not type as fast as they can write by hand. Users need to be skilled typists to take notes that do not "lag behind" the media moments they wish to pinpoint. (d) Tablet computers can be used to take media-enabled notes, but their current costs are prohibitive.

As the above discussion shows, there does not currently exist an "ideal device" for taking extensive media-enabled notes easily, quickly and affordably. Thus, Souvenir Desktop application has been designed to support notes created by a variety of devices, which are currently affordable and could be adopted or are being used by a large user population.

3.3 Segment-Oriented Framework

A Souvenir document is composed of *segments*, whose data structure stores: (a) digital ink and/or text; (b) the associated media file and its playback start and end times. Souvenir employs a "segment-oriented" framework to support: (a) both handwritten and text notes as well as (b) the playback of multiple media recordings in the same document, (c) to enable users to interact with digital ink at a higher-level of organization instead of having to manage a large number of individual strokes, and (d) to create hyperlinks to random-access media wherever it is stored. These hyperlinks make it easy for anyone to share specific media moments without the need to transfer large media files. The media owners do not lose control over their content, because users only share pointers to specific media moments instead of copying them.

Souvenir is built on the premise that users' initial notes are "imperfect" or "noisy" in terms of how precisely they pinpoint the intended media moment and how completely they describe or annotate it. Souvenir aims to make it easy for users to "edit and polish" their initial Souvenir notes to create a report that can be published and shared with others via the Internet. Souvenir has been designed so that users can use their text editing skills to organize and refine their initial media-enabled notes. In particular, Souvenir organizes handwritten notes into segments so that users can interact with digital ink at a level of visual organization that is familiar and makes it easy to edit and refine their handwritten notes. Users can easily annotate digital ink segments with text or add new text and link it to a specific moment in a media recording. Users can edit and rearrange their notes as well as "copy & paste" notes from other Souvenir documents that reference different media files. Users can then publish their report as set of web pages so that others can access or share via email specific media moments directly from these pages (see Figure 2).

3.4 Media Editing

Initial notes can have a tendency to lag behind the moment in the media recording users wish to pinpoint. Souvenir makes it easy for users to correct for this "lag problem." On the one hand, users can specify a general "playback offset" that is added or subtracted from the “start” and “end” times of all segments in a Souvenir document. On the other hand, users can fine-tune the relationship between a specific segment and its associated media. Users can edit and change the “start” and “end” media times of a segment via edit controls in the document window or by interacting with the media timeline. The media timeline shows how the ink and text segments are related to the timeline of the associated media recording(s). The segments are displayed in dark gray on the media timeline (see Figures 1 and 3). Users can change the start and/or end time of a segment by changing its spatial area in the timeline.

4 Structuring Handwriting

Handwriting is ideal for taking quick notes. However, handwritten notes are "loosely structured" whereas text notes are highly structured. Hence, handwritten notes need to be structured to enable both digital ink and text in the same Souvenir document and to be an effective index for digital media. Furthermore, people need to be able to interact with digital ink at a level of visual organization that is easy and familiar. People most commonly interact with text at the word, sentence or paragraph level, where the latter two units of organization are used for attaching annotations in text documents.

At the lowest level of organization, handwritten notes are a collection of strokes, where each stoke represents the pixels touched between successive "pen-down" and "pen-up" events. Figure 5(b) shows a collection of strokes that are surrounded by rectangular bounding boxes. Human handwriting is structured and governed by constraints. People (in the West) have a tendency: (a) to write consecutive or related words spatially close together; and (b) to create straight lines of text. Further, people have a tendency to write a few (lines of) words describing their current thought, to pause for a while and then to start a new thought or to add more to the current or to a previous thought. These constraints can be used to create a simple, yet robust digital ink segmentation algorithm, which assumes that users create lines of text and write top to bottom (see Figure 4). However, the algorithm makes no assumptions about the direction, orientation or line height of the hand-written text and can adapt to the user's current writing style. The algorithm segments digital inks into units of organization that are equivalent to text paragraphs (see Figure 5(c), 5(d), or 5(e)). Users can now interact with their handwritten notes in a familiar way, where the notes are organized in such a way to make it easier to annotate, edit or refine them.

However, digital ink in close spatial proximity is not always created close in time. This fact matters because we want to use the temporal properties of handwritten notes to index audio and video. Without access to the creation times of the digital ink, it is very difficult to detect that notes spatially close have not been created close in time. Souvenir uses note-taking devices, such as the CrossPad or Palm Pilot, that can automatically time-stamp handwritten notes. One of the goals of Souvenir is to make visible when digital ink was added later in time in the vicinity of already existing ink. Figures 5(c) and 5(d) show digital ink in close spatial proximity that has been detected by Souvenir as distinct segments because a certain amount of time elapsed without note-taking activity or some of the digital ink was added later in time.

The purpose or function of a note in relation to a media file can be: (a) to just comment on a specific moment in time; (b) to describe a limited time period in a media recording; (c) to fully capture the spoken content in a one-to-one relationship with the written notes, as is the case in closed captioning, but which can only be created by a specialist; or (d) to be an addition to an existing note that pinpoints a media moment other than the current media time. Now, it takes a certain amount of time to create a note and the note-taking timeline is linear and monotonic increasing. Additional information is needed to determine the specific purpose of a note in relation to the media timeline. For example, if the state of the media recording or playback device is in "pause" or "stop," then the written words are most likely to describe or comment on the current media time. Souvenir takes into account this piecewise linear relationship between media playback activity and simultaneous note-taking activity when synchronizing Souvenir notes and media (see Figure 6).

As the above discussion shows, structuring digital ink as well as synchronizing it to the appropriate media moments can be difficult. Souvenir offers users a seamless and flexible way to structure and synchronize their handwritten notes with digital media wherever it is stored.

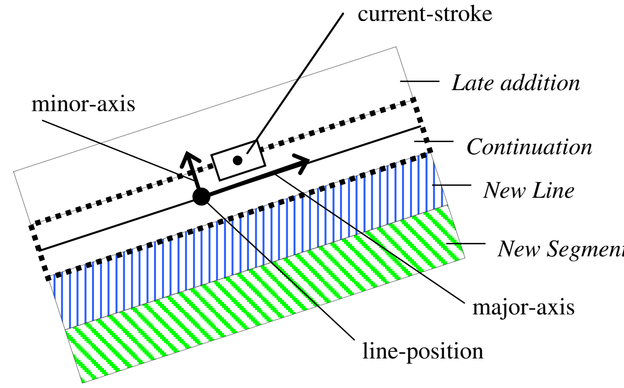


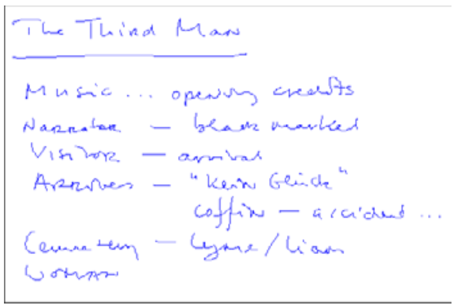
Fig. 4. The *Souvenir segmentation algorithm* computes the line-position, major-axis and minor-axis for the current-line to determine if the current-stroke lies in the "Late Addition," "Continuation," "New Line," or "New Segment" spatial band.

4.1 Digital Ink Segmentation

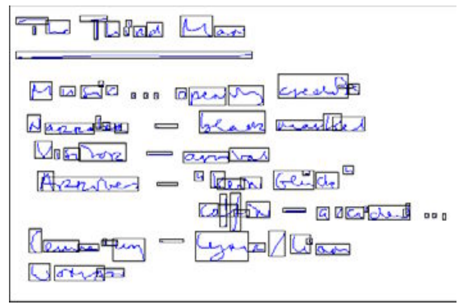
Souvenir automatically groups the digital ink into segments to identify interesting entry points into a media recording. The digital ink segmentation algorithm exploits spatial and temporal characteristics of how people write without having to recognize their handwriting. The handwriting activity can be categorized as follows: the stroke just being written (a) is part of the "current segment" of strokes or words, or (b) forms the beginning of a "new segment," or (c) is a later addition to an "old segment." Souvenir assumes that users write "top to bottom" so that a later addition represents a "backwards motion" up the page. Souvenir makes no assumptions about the writing direction and computes a "current-line" estimate so that "dotting the i's, crossing the t's and underlining," which often represent a backward motion of the pen, do not get categorized as a later addition to an existing segment.

The Souvenir segmentation algorithm continuously calculates an estimate of the line currently being written to determine if the stroke next in time signals the start of new segment. In particular, it computes the line-position, major-axis and minor-axis for the current-line. The line-position is equal to the weighted averages of the x and y coordinates of all the strokes assigned so far to the current-line. The algorithm also computes major and minor axes based on the spatial distribution of the strokes so far assigned to the current-line so that the variance is maximal and minimal in the direction of the major-axis and minor-axis, respectively. The orientation of the current-line is equal to the orientation of the major-axis. The current line-height is proportional to the length of the minor-axis. The estimate of the current-line adjusts as more strokes are added. The handwriting segmentation algorithm loops through all the digital ink strokes in order of creation. The current-stroke is grouped with the current-line if it lies within a spatial band defined by the estimates of the major-axis and minor-axis for

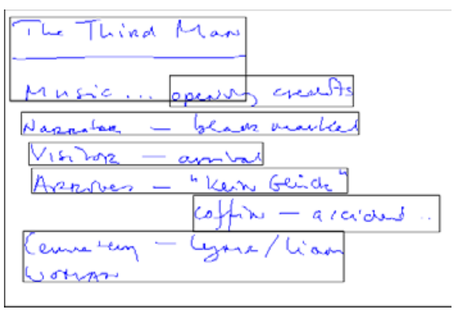
Digital Ink Segmentation



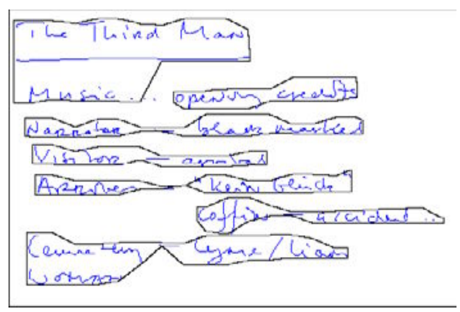
(a) Digital ink.



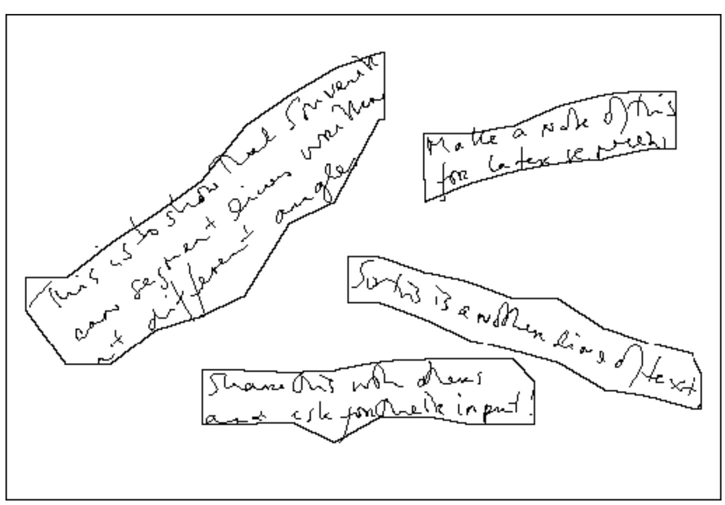
(b) Individual strokes surrounded by bounding boxes.



(c) Detected segments with rectangular bounding boxes.



(d) Detected segments with minimal area boxes.



(e) Detected segments for words written at different angles in the same document.

Fig. 5. shows how Souvenir can segment and structure digital ink that initially consists of a collection of strokes. In (c) and (d) "opening credits" is detected as a separate segment because of a significant time gap in note-taking activity.

the current-line (see Figure 4). Otherwise, the current-stroke forms the seed for the estimate of a new current-line. If the vertical distance between the new current-line and previous current-line is greater than a multiple of the current estimate of the line-height, then a "new segment" is created. If this vertical distance is negative, then the new segment is categorized as a "later addition." A "new segment" is also automatically created if a certain amount of time has elapsed without note-taking activity.

Figure 5 shows how Souvenir can segment handwritten notes that initially consist of a collection of strokes: (a) displays the digital ink without segmentation; (b) shows the individual strokes, which represent the pixels touched between successive "pen-down" and "pen-up" events, surrounded by bounding boxes; (c) shows the segments detected by Souvenir and displays them with rectangular bounding boxes - "opening credits" is detected as a separate segment because of a significant time gap in note-taking activity; (d) shows the detected segments and displays them with minimal area bounding boxes to make the individual segments easier to see; (e) demonstrates that Souvenir can segment text written at different angles in the same document.

Once Souvenir has segmented the handwritten notes, users can easily edit the detected ink segments by merging, splitting, regrouping or annotating them manually. For example, users can split a segment by selecting it and moving the cursor inside of it - the segment is then split in two based on the cursor position in an interactive way.

4.2 Multiple & Complementary Indexes for Digital Media Libraries

Audio/video content can be described at multiple levels of organizations to facilitate the creation of a retrieval index. There are major research efforts to automatically identify the content of individual image frames or movie sequences [9]. The media index created by user annotations represents an additional and complementary way to index digital media. On the one hand, human media annotations are time consuming to create and have required so far specialized software tools to produce. On the other hand, the user annotations represent "value-added" information that can not be easily inferred by automatic techniques. Souvenir aims to "liberate" the creation of user media annotations: (a) a variety of inexpensive devices and tools can be used to create them; (b) the media-enabled annotations can be leveraged without being held captive by where and how the media is stored in a digital library; (c) users can use their time more productively, because they get a personalized media index "for free" when they take Souvenir notes while experiencing digital media; (d) the handwritten notes can visualize in a personal way which parts of a media recording are of interest.

5 Note & Media Synchronization

Souvenir time stamps the user's note-taking activity. It uses these time stamps to link and synchronize the notes with the timeline of a media recording. This way the notes can be used to access and playback specific moments in a media recording. The synchronization of the notes and media timelines is conceptually straightforward: iden-

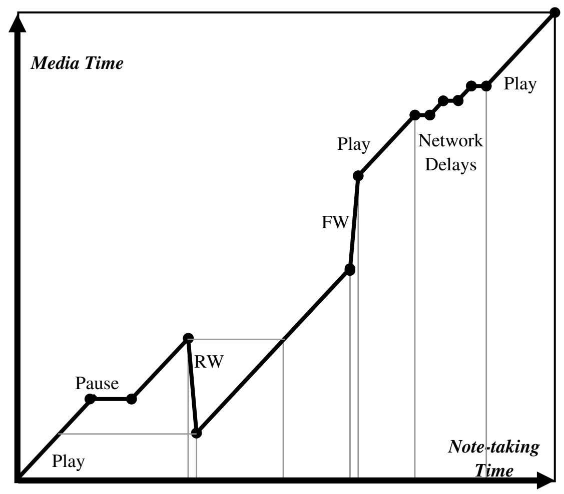


Fig 6: shows the piecewise linear relationship between media playback and simultaneous note-taking activity that needs to be considered when synchronizing notes and media.

tify a specific media moment to be linked with a specific note and the resulting relationship between their respective time-codes can be used to compute the media playback offsets for all the other notes. This presupposes that the recording or playback of the media is a strictly monotonic, linear event. However, users can pause, play, rewind or fast forward the media while taking notes. If the media is delivered via a network, playback delays can also occur.

The Souvenir Media Player keeps track of when a user plays, pauses, rewinds or fast-forwards a media recording. This player, which can play local as well as Microsoft or RealNetworks streaming media, also monitors the network performance if the media is streamed. Souvenir can use this database of media playback activity to link the user's note-taking activity with the appropriate media moments.

Souvenir automatically performs the synchronization of notes and media if (a) the audio/video is recorded or played back on the same device used for taking notes, or (b) the media playback occurs in the Souvenir Media Player. In all other cases, a wizard will guide users through a simple process where they need to identify a specific media moment to be linked with a specific note. After that, Souvenir completes the synchronization of notes and media, taking into account the piecewise linear relationship between media playback activity, network performance and simultaneous note-taking activity (see Figure 6).

Souvenir is built on the premise that users want the flexibility to use different devices to capture and playback digital media. However, users will want to be able to share their media-enabled notes with others. Hence, they will want to upload or store the referenced media in a digital library that accessible to others via the Internet. Souvenir makes it easy for users to update or change the location information of the media recordings referenced in a Souvenir document.

6 Evaluation and Discussion

Souvenir has been downloaded by quite a few people. Informal tests of Souvenir's digital ink segmentation algorithm are encouraging (see Figures 1, 2, 3 and 5). The algorithm can segment notes created by different note-takers without the need for any training or special instructions. However, a formal evaluation is needed and planned to test the effectiveness of the segmentation algorithm more rigorously. Considering the high variability of human handwriting, the algorithm tracks simple properties that are robust enough to detect segments that are useful and match how the note-takers would visually parse their notes. As mentioned, handwritten words in spatial proximity can be assigned to different segments when they were not all created at the same time. This can be "visually counter-intuitive" until the user takes into account that the timeline of how the notes were created is not always easy to see.

Souvenir offers users a flexible mechanism for linking notes and related media. In the "worst case," users have to identify a specific media moment to be linked with a specific note and then Souvenir does the rest. However, general users will adopt Souvenir only if the synchronization is automatic and transparent. So far users have had to use specialized tools to pinpoint and share specific media moments. Using Souvenir, users can create a media "edit list," which stores the start and end times of media clips, without having to learn new skills. Souvenir has been designed to enable users to use their note-taking and text editing skills to manipulate digital media without having to be concerned about where and how the referenced media has been stored.

Souvenir makes it easy for anybody to bookmark specific moments in streaming media hosted on the Internet. However, some media sites make it difficult to identify the URLs of their media content to prevent "deep linking." Having access to the URL makes it possible to play the streaming media in the Souvenir Media Player, which in turn makes it easy for Souvenir to automatically link the notes with the related media. The reluctance of media sites to make the media URLs easily available is a pity, because Souvenir encourages its users to only create and share pointers to specific media moments instead of actually copying the content and media owners losing control over their content. Actually, the Souvenir media annotations could be used for community building and data mining purposes. The Souvenir notes make it possible to access and share specific media moments without the need to view the entire recording, resulting in a more efficient use of server resources. Content providers can use the personal media annotations and the resulting targeted access of their content to visualize frequently accessed moments.

7 Conclusion

Souvenir is a versatile note-taking tool to pinpoint, edit and publish audio/video highlights stored in digital libraries. Souvenir is innovative in that it offers a comprehensive solution for users to use their handwritten or text notes to create a visual and effective audio/video index. Souvenir gives users flexibility in how they can access and integrate digital media moments into their workflow. Users can take media-enabled notes on a variety of devices, such as the paper-based CrossPad, the Palm Pilot and standard keyboard devices. Souvenir can organize digital ink into segments, which are detected using a segmentation algorithm that exploits spatial and temporal characteristics of how people write without having to recognize their handwriting. Souvenir employs a "segment-oriented" framework to support both handwritten and text notes as well as the playback of multiple media recordings in the same document. Souvenir is flexible in terms of how to synchronize notes and media, taking into account the piecewise linear relationship between media playback and note-taking timelines. Souvenir also has web publishing and email capabilities to enable anyone to access or share specific media moments directly from a web page.

Acknowledgement

Souvenir was developed while the author was at i-Recall Inc. Special thanks to Nathaniel Polish, Daniel Magill and Oliver Daniel for their invaluable development help as well as to Nick Belkin (Rutgers University) for valuable feedback and input.

References

1. Abowd, G. D., Atkeson, C. G., Brotherton, J. A., Enqvist, T., Gulley, P., and LeMon, J. "Investigating the Capture, Integration and Access Problem of Ubiquitous Computing in an Educational Setting," Proc. of CHI '98 (April 1998), ACM Press, 440-447.
2. Brotherton, J. A., Bhalodia, J. R. and Abowd, G. D. "Automated Capture, Integration, and Visualization of Multiple Media Streams," Proc. of IEEE Multimedia '98, July, 1998.
3. Chiu, P. and Wilcox, L. "A Dynamic Grouping Technique for Ink and Audio," Symposium on User Interface Software and Technology '98, (San Francisco, Nov 1998), 195-202.
4. Cruz, G. and R. Hill. "Capturing and Playing Multimedia Events with STREAMS", Proc. of ACM Multimedia ‘94, October 1994, pp. 193-200.
5. Moran, T. P., Palen, L., Harrison, S., Chiu, P., Kimber, D., Minneman, S., van Melle, W., and Zelweger, P. "`I'll Get That Off the Audio,': A Case Study of Salvaging Multimedia Meeting Records," Proc. of CHI '97 (Atlanta, April 1997), ACM Press, 202-209.
6. Stifelman, L. J. "Augmenting Real-World Objects: A Paper-Based Audio Notebook," Proc. of CHI '96 (Vancouver, Canada, April 1996), ACM Press, 199-200.
7. Truong, K. N. and Abowd, G. D. "StuPad: Integrating Student Notes with Class Lectures," Proceedings of CHI '99, (Pittsburgh, May 1999), ACM Press, Volume 2, 208-209.
8. Weber, K., and Poon, A. "Marquee: A Tool for Real-Time Video Logging," Proc. of CHI '94 (Boston, April 1994), ACM Press, 58-64.
9. Wactlar, H., Kanade, T., Smith, M., and Stevens, S. "Intelligent Access to Digital Video: The Informedia Project." IEEE Computer, Vol. 29, No.5, May 1996.
10. Whittaker, S., Hyland, P., and Wiley, M. "Filochat: Handwritten Notes Provide Access to Recorded Conversations," Proc. of CHI '94 (Boston, April 1994), ACM Press, 271-277.
11. Wilcox, L., Schilit, B. and Sawhney, N. "Dynomite: A Dynamically Organized Ink and Audio Notebook," Proc. of CHI '97 (Atlanta, April 1997), ACM Press, 186-193.

Term Co-occurrence Analysis as an Interface for Digital Libraries

Jan W. Buzydlowski, Howard D. White, and Xia Lin

College of Information Science and Technology, Drexel University,
Philadelphia, Pennsylvania 19104
{janb, whitehd, linx}@drexel.edu

Abstract. We examine the relationship between term co-occurrence analysis and a user interface for digital libraries. We describe a current working implementation of a dynamic visual information retrieval system based on co-cited author maps that assists in browsing and retrieving records from a large-scale database, ten years of the Arts & Humanities Citation Index, in real time. Any figure in the arts or humanities, including scholars and critics, can be mapped, and the maps are live interfaces for retrieving co-citing documents.

1 Introduction

The first wave of research on digital libraries focused on making data physically available. While digital libraries promise to deliver huge amounts of content immediately, this very abundance exacerbates the problem of separating relevant from irrelevant material. The next wave of digital library research must address the ever-present problem of intellectual access—that is, matching requests appropriately in information systems so that users can find what they are looking for. This often requires human intervention to represent the raw material—documents—through synoptic language such as descriptors, keywords, and abstracts. Given the volume of material contained in digital libraries, such representation is a large-scale, formidable task.

There have been many efforts, described in, e.g., [1], to automate the process of representing documents with suitable indexing terms or abstracts. There have also been many attempts to display those documents *en masse* in a format that allows users to better understand the entire collection, e.g., [2] and [3]. Much of the current work in presenting an entire collection, e.g., [4] and [5], has focused on the visualization of the materials. This is a positive trend, in that visualization helps one see patterns that cannot be readily absorbed in any other way [6].

Unfortunately, many visualizations are being built on trivial collections. Others are marred by cryptic symbolism and/or inadequate labeling in the displays. Most in the bibliometric tradition are static rather than dynamic interfaces. In this paper we present a new way of visualizing a large, real-world collection of bibliometric data based on dynamic queries from users. We describe an implementation of such a system and illustrate its use with some examples.

K. Börner and C. Chen (Eds.): Visual Interfaces to Digital Libraries, LNCS 2539, pp. 133–144, 2002.

2 Method

In each document there are sections that contain various terms. A fruitful method for rendering a collection of documents is through analysis of their co-occurring terms—that is, the words, phrases, author names, and so on that appear together in designated spans of text in the same document. For example, a number of salient words can be extracted from the title of a work or from its abstract (eliminating stop words and perhaps using word stemming). The authors cited in a bibliography section can also be extracted, as can the journals in which those authors publish. Such co-occurrences can then be counted and statistically processed to serve as indicators of the substantive contents of the collection.

Each document must be parsed to reveal the salient terms. Fortunately, this parsing can be automated, as described in, e.g., [1]. Since the number of co-occurring terms in segments of each document can vary, they constitute a repeating field. It is easy, however, to cast this information in the form of a set of records with each record containing a document number and a single term for each of all of the terms in a section.

The advantage of term co-occurrence analysis lies in the fact that all of the information is derived from the documents themselves and requires no human intervention. The more documents one has, the more information becomes available. This makes it ideal for digital libraries.

2.1 Term Co-occurrence Analysis

Information retrieval is often divided into two categories: searching and browsing. Searching implies that you have a good-to-perfect idea of what you want. Browsing implies that you will be able to recognize what you want when you see it.

Browsing helps a user examine a new field before doing more in-depth research. If one were starting to do research in a new specialty, it would be helpful to see what subject areas are involved and how they are related, who the specialty's major authors are, what its major journals are, and so on. For instance, if an undergraduate wanted to explore philosophy, she would probably have a name like Plato in mind. It would be helpful if she could use Plato's name as an input to learn the names of other philosophers, such as Aristotle, Hegel, or Kant, who are repeatedly co-mentioned with him in scholarly writings. Similarly, if a reader likes the novelist John Steinbeck, it would be helpful for him to know the other novelists, critics, and scholars who are most often mentioned in Steinbeck studies.

How, then, does one use term co-occurrence analysis to help in searching or browsing? Our technique makes use of *term seeding*. The general principle is that given a single word or phrase as a "seed," or starting point, we can retrieve the other terms that most frequently occur with the seed in designated fields across a large collection. Documents that contain the seed term are systematically examined to return the related terms, rank-ordered by frequency of occurrence. For instance, if

Plato is the seed author, we can retrieve the other authors that co-occur most frequently with him in the references of journal articles.

It is useful to have the list of top-ranked related terms, given a single input term. However, once a list is retrieved, it is also of interest to know how each item in the list is related to every other item. All of the items in the list are related to the seed, but they have sub-relations as well when they are systematically paired. For instance, although the four philosophers mentioned above are studied in various combinations, most educated persons would expect Plato and Aristotle to be conjoined more often than, say, Aristotle and Hegel; conversely, Hegel and Kant would be conjoined more often than Plato and Kant. Our goal is to make explicit all such sub-relations, which means extending a ranked list of term-counts into a square matrix of term-counts for every pair in a set of terms.

To determine the various relationships between all of the terms returned after the seed term is entered, the analysis of co-occurrences will give a metric to determine the strength of an association. The strength of the co-occurring terms comes from the number of times two terms occur together within the collection. In our example Plato and Aristotle will presumably co-occur a large number of times and so will Hegel and Kant. The more times one author co-occurs with another author, the stronger the association those two authors have; the less frequent, the less strong. In this example we are working within the framework of author co-citation analysis (ACA), developed over a 20-year period in, e.g., [7], [8], and [9]. The same methodology and arguments for ACA's validity can be applied to appropriate repeated terms to support a more general analysis of term co-occurrence.

2.2 Display Techniques

The data structure that represents the co-occurrence of terms is called a co-occurrence matrix. The row and column headings represent the terms of interest, and the intersections of the rows and columns hold counts of the number of times the pair of terms co-occur. Since the matrix is symmetric (e.g., Hegel and Kant have the same count as Kant and Hegel), it can be represented by either its upper or lower half.

Once the pairwise co-occurrences have been derived, it is possible to examine the raw co-occurrence frequencies directly. However, when they are numerous (e.g., 25 terms produce 25(24)/2 = 300 unique co-occurrences of pairs), it is difficult to grasp what all of the numbers jointly mean. Fortunately, there are several visualization techniques that allow one to present all of the co-occurrences simultaneously, tapping into the human ability to process pictures better than numeric matrices. Three visualization, or mapping, techniques that have been used to render multiple co-occurrences are multidimensional scaling (MDS), Kohonen Self-Organizing Maps, and Pathfinder Networks.

MDS is a mathematical technique that reduces the high dimensionality of the co-occurrence matrix to a more visually representable two or three dimensions. It is a well-established methodology that produces maps in which authors are positioned as points on a page. The basic metaphor in such mapping is that greater similarity of authors, as measured by their co-occurrence counts, is rendered spatially as greater proximity of their points. MDS is often used in conjunction with clustering algorithms that permit cluster boundaries to be drawn around groups of related points.

A second mapping algorithm used is that of self-organizing maps (SOMs), also called Kohonen Maps after their Finnish inventor. This display technique uses self-training neural networks to determine the placement of terms in two or three dimensions, e.g., [10]. A SOM is similar to a MDS map in that authors are represented as points on a page, but it also automatically groups similar authors into explicit word or concept areas. Again, closeness of points or point areas implies relatively strong relationships in the raw data.

A third mapping algorithm is Pathfinder Networks (PFNETs). The algorithm was developed by Schvaneveldt and other cognitive scientists [11] to retain the more salient links and eliminate less salient links in verbal association networks. The latter were often generated by having human subjects make similarity choices between paired stimuli and then simplifying the pairwise data as a PFNET. Since any co-occurrence matrix can be represented as a network, all of the PFNET techniques also apply to term co-occurrence analysis and have been used in our project to generate maps. Spring embedder algorithms, e.g., [12], [13], are needed to render or embed the PFNET on a screen or page, and the resulting visualization positions terms as points on a page with the additional feature of explicitly connecting with lines the terms that are most directly associated with each other (and not connecting terms that are linked via less salient intermediate terms).

The format of SOMs and the presentation algorithm of PFNETs arguably convey at least as much information as the MDS algorithm [14], and they are highly tractable for real-time implementation. Thus, our current research in visualization of co-occurrence terms favors SOMs and PFNETs.

3 Implementation

We have created a system based on the methodology described above. It is a generalized Web-based interface that can work with any term-based collection. The data are contained in a specialized database, the visualization algorithms are coded in C for speed, and the interface is coded in Java for portability and Web-browser execution.

There are currently two data sources on which the system is implemented. One, which we presently call ConceptLink, maps co-occurring descriptors from the Medline database of the National Library of Medicine. Another, presently called AuthorLink, is based on a citation index from which we can extract co-cited authors. This paper will describe the latter in detail.

Our system has three tiers. The front tier is a Web interface in Java and HTML. The middle tier is an application server including various implementations of data-mapping procedures (in Java Servlets, C, and CGI). Due to the modularity of the architecture, the back-end tier can be any database or search engine. The back end for AuthorLink is a BRS/Search Engine (BRS formerly was a commercial rival to Dialog). The back end for ConceptLink is Oracle 8i, which stores and processes a large co-occurrence database, and the PUBMED search engine.

The data source for AuthorLink is the Arts & Humanities Citation Index (AHCI) for the decade 1988 to 1997. The data were given to our college by the Institute for Scientific Information, based near Drexel University in Philadelphia, as a research grant in 1998. There are approximately 1.26 million bibliographic records. The resultant database consists of approximately 7 million terms. Although the size of the database is nontrivial, the results below are achieved in real time—retrievals and mappings are accomplished in a few seconds.

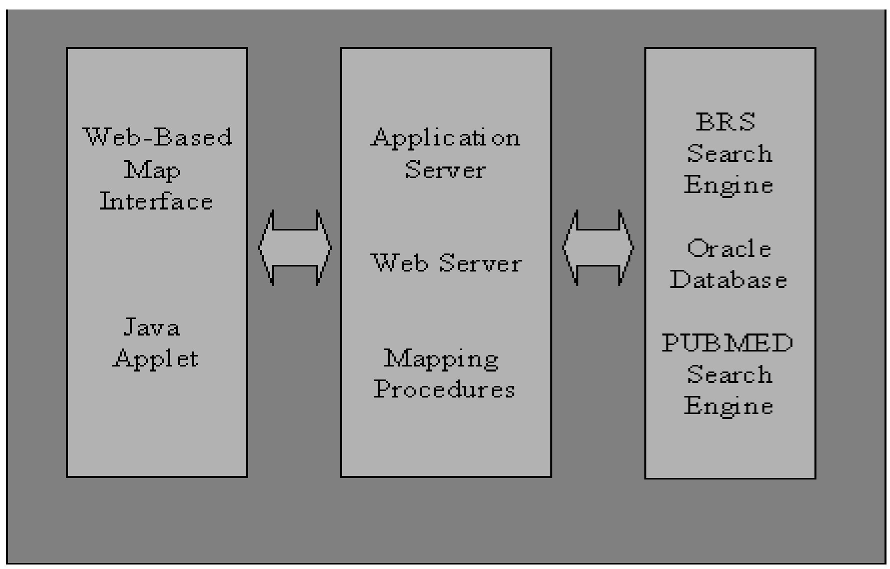


Fig. 1. Architecture of the AuthorLink System

3.1 The Interface

We have simplified the input to ACA from what it has been in the past. Rather than requiring a whole list of authors as input, we require only a single nameseed, such as Plato or Steinbeck or Margaret Atwood, to generate the analysis. This greatly lightens the cognitive load on the user, who gets considerable information back for minimal input. We can map not only world giants such as Rembrandt, Mozart, or Shakespeare, who are known to most educated people, but literally thousands of other people in the humanities as well, as long as they are cited in the journals covered by AHCI.

Initially the user is presented with a screen explaining the system and with a blank text box in which to enter an author name, the nameseed, to retrieve a list of the most closely associated authors, as in Figure 2.

Fig. 2. Initial Login Screen

After the user enters the nameseed, the database is queried to determine the authors who are most often co-cited with the nameseed author. The list of author names is returned, in descending order of frequency, along with the number of times each author co-occurred. (The default return is the first 25 authors, the seed plus 24 others, but this can be easily adjusted.) For instance, if the user selected Plato as the nameseed, then the authors co-cited with Plato would be returned as in the left box in Figure 3.

To determine the strength of associations for the 300 co-occurrences of the top 25 authors, the user selects the “Map It Now” button and the database is again queried. The resulting co-occurrence matrix is passed to the mapping algorithm. The initial map displayed is that of a PFNET. An example of what the map looks like is shown in Figure 4.

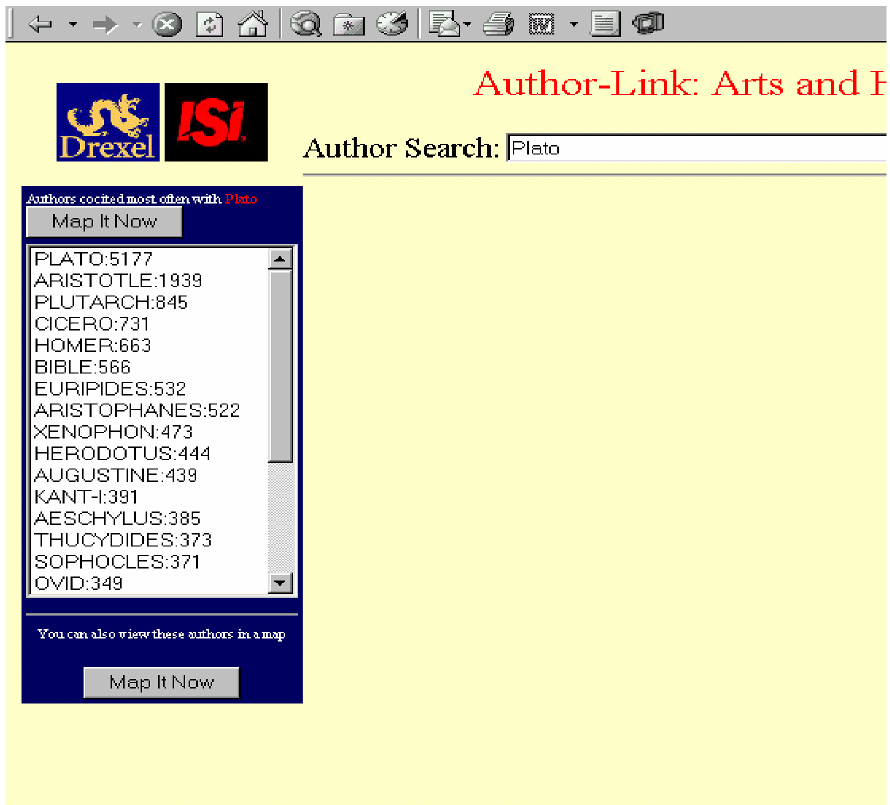


Fig. 3. List of Authors Related to Plato

The names directly linked by the PFNET form highly cogent groupings. For example, the relationships between Plato and Aristotle and between Hegel and Kant that were presumed above are borne out by the actual links that are formed from the 1988-97 AHCI. More generally, the PFNET suggests a virtual "Great Books Course" built around Plato. We take this as evidence that AuthorLink has considerable promise as an aid to browsing in a digital library space.

Experts on a given author will be able to see that author's intellectual affiliates, some of which may be new to them, as derived from contemporary citation patterns. This is indexing by use, similar to that of "recommender systems." Novices or even experts on an author may not be able to interpret all of the connections (which are based on the perceptions of citing scholars), but they will at least have rich leads as to authors that are worth reading together.

The user can toggle on or off the number of co-citations for each linked pair. They are "toggled on" and appear as numbers over the links in Figure 4. If the cursor is placed on an author's name, that author's co-citation count with Plato pops up.

The user can also choose from a list box to display the same co-occurrence matrix as a SOM. The SOM for Plato is shown in Figure 5.

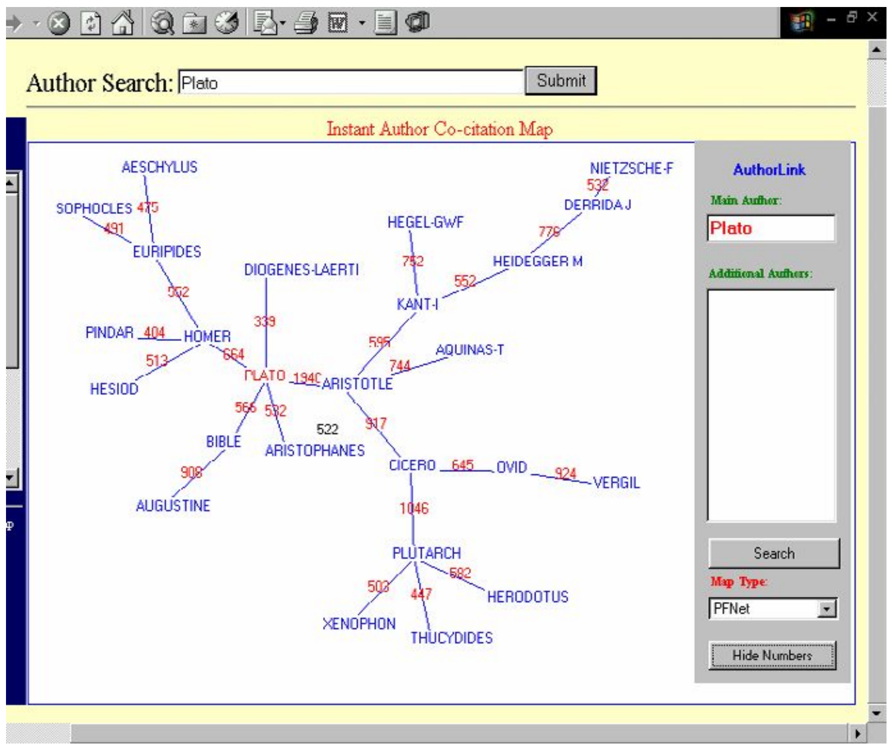


Fig. 4. PFNET of Authors Related to Plato

Plato appears near the center of the map. (Seed authors are not always central.) Near him are other closely connected authors, such as Aristotle, his only rival in Greek philosophy. Plato's eminence in the Western intellectual tradition is such that he is implicated in the 1988-97 AHCI not only with medieval and modern philosophers (Augustine, Aquinas, Hegel, Kant, Nietzsche, Heidegger, Derrida) but with Greek dramatists (Aristophanes, Aeschylus, Sophocles, Euripides), Greek and Roman poets (Homer, Hesiod, Pindar, Ovid, Vergil), Greek and Roman historians (Herodotus, Thucydides, Xenophon, Plutarch, Cicero), and the Bible. Plato's biographer, Diogenes Laertius, is also present. We can show that all of these names are automatically placed in appropriate groupings by the Kohonen algorithm.

But what of retrieval? The AuthorLink interface is connected to the full 1.26 million bibliographic records in the 1988-97 AHCI. If the user wants to pursue Plato's connections with any other author(s) in the map, the literature making those connections can be retrieved. Plato's name is automatically entered as the seed author in the Additional Authors list box to the right of the map (as shown in Figure 4). The user can select one or more additional names, say Ovid and Vergil, to search with Plato in a Boolean AND relationship. The user drags-and-drops the names into the Additional Authors list box, clicks on Search, and the articles in AHCI that contain those names are returned, as in Figure 6, which displays a BRS/Search Web interface.

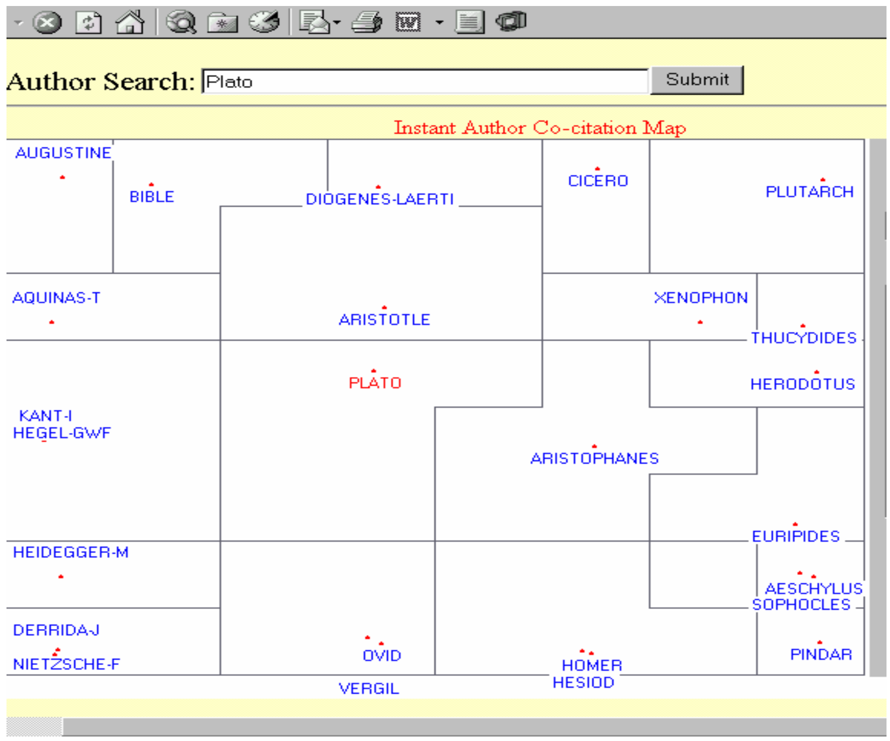


Fig. 5. Self-organizing Map of Authors Related to Plato

Clicking on the hyperlinked title of any article will bring up its full bibliographic record plus a listing of the works it cites. In the latter, the cited works by the ANDed authors (e.g., Plato, Ovid, and Vergil jointly) are highlighted. Eventually it should be possible to bring up full-text copies of many of the cited articles rather than their bibliographic records. This will come about as the Institute for Scientific Information and its many academic library customers move from print to fully digitized and hyperlinked resources.

It must again be emphasized that our maps are produced from the data alone, without any human intervention. The data source is substantive and important. The queries are dynamic, based on any user-supplied seed, and the results (list of names and maps) are returned in seconds. This allows for the real-time interaction with the nameseeds, associated author lists, and maps to support online browsing and searching.

4 Conclusion

White and McCain [3] suggest the following list (abridged) for evaluating a visual information retrieval interface:

1. Is the display an improvement over a simple list?
2. Does it provide new capabilities?

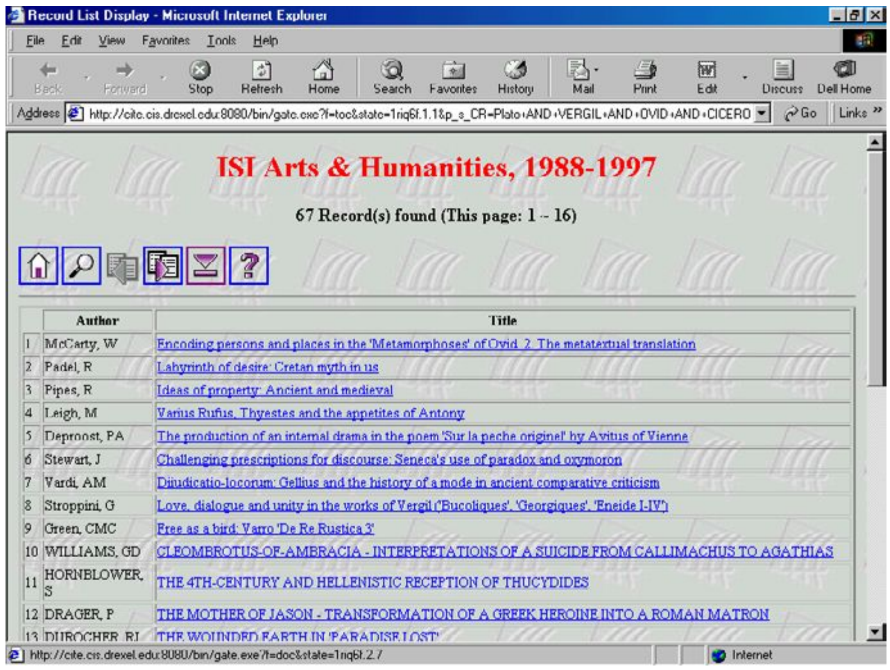


Fig. 6. Retrieval Interface

3. Is it rapidly intelligible?
4. Is it helpful in real time (or with an acceptable wait)?
5. Is it tied to an important collection?
6. Is it scalable upward to collections greater in size?

We believe the system we have built allows us to answer these questions in the affirmative.

For Question 1, a map does provide more information than a simple list. The information provided in Figure 4 or Figure 5 on interconnections of authors is significantly greater than the information provided in Figure 3.

For Questions 2 and 4, our system does provide new capabilities by being dynamic and working in real time so that a user can perform term analysis and retrieval iteratively. Because of its responsiveness, it is quite fun to use for exploring the ties of authors in whom one is interested.

For Question 3, a study has just been completed by the first author of this paper that compares the mental, or cognitive, maps of 20 experts in the arts and humanities to the visual maps produced from the AHCI data described above. The results show that both map types, the SOM and the PFNET, correspond very well to the mental maps of the experts, which were elicited through card sorts. Correlations of both the SOM and the PFNET with the card-sort data were statistically significant ($p < 0.001$ in each case). The SOM's correlation with the card sorts was higher than the PFNET's. However, the 20 experts were equally divided when asked to choose an overall preference for one type of map over the other, with many indicating that the two types are complementary and that both are needed to do a thorough exploration.

For Question 6, the current implementation is based on a data source of significant size, e.g., 7 million terms. We are seeking data sources of even greater size, e.g., the entire AHCI collection, to test the bounds of the system.

Finally, for Question 5, the AHCI database for 1988-97 is already of significant size and importance. When it is combined with AuthorLink, one can readily explore the bibliographic relationships of many thousands of writers, of all magnitudes of eminence, in the humanities. This is a potentially valuable resource for students and researchers everywhere, whether they want to browse or retrieve documents or both. Moreover, the technology AuthorLink represents is transferable to other significant digital library collections.

References

1. Salton, Gerard. Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer. Addison-Wesley Publishing Company (1989)
2. Chen, Chaomei. Visualizing Semantic Spaces and Author Co-Citation Networks in Digital Libraries. Information Processing & Management. Vol. 35. (1999) 401 – 420
3. White, Howard D., McCain, Katherine W. Visualization of Literatures. Annual Review of Information Science and Technology. Vol. 32. (1997) 99 – 168
4. Chen, Chaomei. Information Visualization and Virtual Environments. Springer-Verlag (1999)
5. Ding, Ying, et al. Bibliometric Information Retrieval System (BIRS): A Web Search Interface Utilizing Bibliometric Research Results. Journal of the American Society for Information Science. Vol. 51. (2000) 1190 – 1204
6. Cleveland, William S. Visualizing Data. Hobart Press (1993)
7. White, Howard D. Author Co-Citation Analysis: Overview and Defense. In Scholarly Communication and Bibliometrics, Christine L. Borgman (ed.) Sage Publications (1990) 84 – 106
8. McCain, Katherine W. Mapping Authors in Intellectual Space: A Technical Overview. Journal of the American Society for Information Science. Vol. 41. (1990) 433 – 443
9. White, Howard D., and McCain, Katherine W. Visualizing a Discipline: An Author Co-Citation Analysis of Information Science, 1972-1995. Journal of the American Society for Information Science. Vol. 49. (1998) 327 – 355
10. Lin, Xia. Map Displays for Information Retrieval. Journal of the American Society for Information Science. Vol. 48. (1997) 40 – 54
11. Schvaneveldt, Roger W. (ed.) Pathfinder Associative Networks. Ablex Publishing Corporation (1990)

12. Kamada, Tomihisa and Kawai, Satoru. An Algorithm for Drawing General Undirected Graphs, Information Processing Letters. Vol. 31. (1989) 7 – 15

13. Fruchterman, Thomas and Reingold, Edward. Graph Drawing by Force-Directed Placement, Software—Practice and Experience. Vol. 21. (1991) 1129 – 1164

14. White, Howard D., Buzydlowski, Jan, Lin, Xia. Co-Cited Author Maps as Interfaces to Digital Libraries: Designing Pathfinder Networks in the Humanities. Proceedings, IEEE International Conference on Information Visualization. (2000) 25 – 30

Information Visualization, Human-Computer Interaction, and Cognitive Psychology: Domain Visualizations

Kevin W. Boyack, Brian N. Wylie, and George S. Davidson

Sandia National Laboratories*, Computation, Computers & Math Center,
P.O. Box 5800, MS-0318, Albuquerque, NM 87185 USA
{kboyack, bnwylie, gsdavid}@sandia.gov

Abstract. Digital libraries stand to benefit from technology insertions from the fields of information visualization, human-computer interaction, and cognitive psychology, among others. However, the current state of interaction between these fields is not well understood. We use our knowledge visualization tool, VxInsight, to provide several domain visualizations of the overlap between these fields. Relevant articles were extracted from the Science Citation Indexes (SCI and Social SCI) using keyword searches. An article map, a semantic (co-term) map, and a co-author network have been generated from the data. Analysis reveals that while there are overlaps between fields, they are not substantial. However, the most recent work suggests areas where future collaboration could have a great impact on digital libraries of the future.

1 Introduction

The amount of information becoming available in digital form is increasing exponentially. Many institutions, while interested in providing digital information to their users, are only slowly making the shift from paper to digital libraries. There is a pronounced need for advances in techniques and tools to aid the individual user in finding and gleaning knowledge from relevant information.

It is felt by many researchers that such advances would be enhanced by insertion from the fields of information visualization, human-computer interaction, and cognitive psychology. Yet, to date, it is unclear how much interaction there has been between these fields, and how much impact each has had on advances in digital libraries.

The purpose of this paper is to explore the history and current state of overlap between these four fields (including digital libraries) by analysis of bibliographic information. We employ our visualization tool, VxInsight [1], which was developed to

* Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy under Contract DE-AC04-94AL85000.

K. Börner and C. Chen (Eds.): Visual Interfaces to Digital Libraries, LNCS 2539, pp. 145–158, 2002.

build and explore maps of technology using data from the Science Citation Index (SCI). Over the past few years, we have found that VxInsight has broad application to mapping and navigation of many different types of data [2-4]. In this paper, we provide an overview of related work and tools, some background on the VxInsight tool and process, and several different visualizations of the domain comprised of the four fields mentioned. We close with a summary and suggestions for future work.

2 Related Work

2.1 Literature Maps

Various efforts to map the structure of science from literature have been undertaken for many years. The majority of these studies are performed at the discipline or specialty level. Maps are often based on similarity between journal articles using citation analysis [5], co-occurrence or co-classification using keywords, topics, or classification schemes [6,7], or journal citation patterns [8,9]. Latent semantic analysis (LSA) has been used to map papers based on co-occurrence of words (or authors) in titles, abstracts, or full text sources [10,11]. In addition, domain maps based on author co-citation analysis [10,12] are becoming more common. Many of these studies probe the dynamic nature of science, and the implications of the changes.

Once a similarity matrix is defined, algorithms are used to cluster the data objects (e.g., articles or patents). Common clustering or ordination methods for producing maps include multidimensional scaling, hierarchical clustering, k-means algorithms, Pathfinder network scaling, and self-organizing maps. The standard mapping output for early literature studies was a circle plot where each cluster was represented by an appropriately sized circle. Links between circles provide relationship information. Traditionally, map outputs have been paper-based and only resolve structure at a few discrete levels. However, in recent years, several systems have been reported that use a computer display and allow some navigation, browsing, and filtering of the map space.

2.2 Visualization Tools

SENTINEL [13] is a Harris Corporation package that combines a retrieval engine using n-grams and context vectors for effective query with the VisualEyes visualization system. The visualization tool allows the user to interact with document clusters in a three-dimensional space. Chen [10,14] uses a VRML 2.0 viewer in conjunction with Generalized Similarity Analysis to display authors (as spheres) and the Pathfinder linkage network based on author co-citation analysis. Citation rates are shown as the 3rd dimension in these VRML maps. Börner [11] uses the CAVE environment at Indiana University to interface with documents in a virtual library. Documents are clustered using latent semantic analysis. Features such as shape, color, and labeling are used to identify features of each document. Document details are available on demand through a hypertext link.

Self-organizing maps have been used in many venues, including the organization of document spaces [15]. This technique is used to position documents, and then display them in a two-dimensional contour-map-like display in which color represents density.

Two packages that are more similar to Sandia's VxInsight are SCI-Map, developed by ISI [16], and the SPIRE suite of tools that originated at Pacific Northwest National Laboratory [17,18]. SCI-Map uses a hierarchically-nested set of maps to display the document space at varying levels of detail. This nesting of maps allows drilling down to subsequent levels. Each map is similar to the traditional circle plot, where the size of the circle can indicate the density of documents contained in the circle or some other measure of importance. Relationships at each discrete level are indicated by links between circles.

Like VxInsight, SPIRE maps objects to a two-dimensional plane so that related objects are near each other, and provides tools to interact with the data. SPIRE has two visualization approaches. In the Galaxies view, documents are displayed as a scatter plot. The interface allows drilling down to smaller sections of the scatter plot, and provides some query and summarization tools. In the Themescape view, a terrain display, similar to that in VxInsight, is used. Themescape visualizes specific themes as mountains and valleys, where the height of a mountain represents the strength of the theme in the document set.

Themescape and VxInsight are examples of so-called 2.5D visualizations. Here the purpose of using the third axis, height, is to make the clusters of documents more visible than they would be in a 2D visualization such as a self-organizing map. Use of the terrain metaphor can also aid in helping the user to learn and remember the information space, much as we use visual 3D clues (mountains, buildings, etc.) to learn and remember the physical world in which we live.

None of the systems mentioned above is interactive in the sense that it would interface to a large digital library in real-time. One system that shows promise for the future is a new query-based visual interface at Drexel University. Buzydlowski and coworkers [19] have developed an interface to over 1.2 million records from the Arts & Humanities Citation Index (AHCI). The user types in the name of an author of interest, and a map of the 25 authors most linked to the query author is returned. The user can drill down through an author to find individual works.

3 VxInsight Description

VxInsight is a PC-based visualization tool for exploring data collections. It works by providing access to data in an intuitive visual format that is easy to interpret and that aids natural navigation. VxInsight exploits the human capability to visually detect patterns, trends, and relationships by presenting the data as a landscape, a familiar representation that we are adept at interpreting, and which allows very large data sets to be represented in a memorable way.

Figure 1 shows the general process through which data must pass to produce a VxInsight map. A typical database, represented as a spreadsheet in the figure, would consist of a few thousand objects (the rows), with one or more attributes arranged in tables (the columns). These must be processed to compute similarities for each pair of data elements, which are then used to construct an abstract graph. In this graph of nodes and arcs, the nodes represent individual data elements and the weighted arcs are the similarities between the elements.

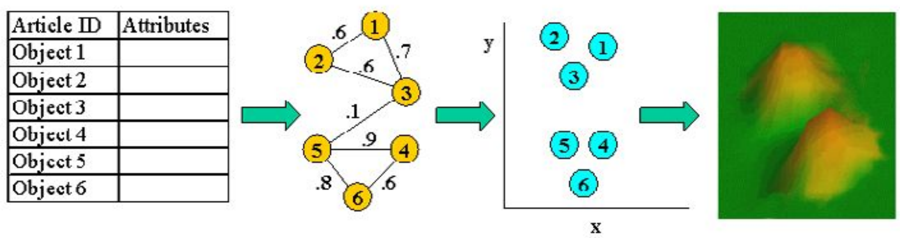


Fig. 1. Data processed into a VxInsight map.

VxInsight uses a force-directed placement algorithm, VxOrd [3], to cluster data elements using pairwise similarity values as input. This algorithm uses a random walk technique, where the criteria for moving nodes is the minimization of energy given by:

$$E_{x,y} = \left[\sum_{i=0}^{n} \left(w_i \times l_i^2 \right) \right] + D_{x,y} \; . \qquad (1)$$

where $E_{x,y}$ is the energy of a node with n edges at a specific x,y location, w_i are the pairwise similarities, l_i is the Euclidean distance between this node and the node connected by edge i, and $D_{x,y}$ is a density measure with respect to the area around point x,y. Use of a density field for the repulsive term is an order O(N) process, thus ordination using VxOrd can be done rapidly. Another advantage of VxOrd is that the number, size, and position of clusters are dictated by the data, not by an arbitrary assignment of the number of clusters. The output from this clustering process is an x,y location on the abstract visualization surface for each data element.

In VxInsight, these coordinates are used to generate the mountain terrains. The height of each mountain is proportional to the number of objects beneath it. Labels for peaks are generated dynamically from any attribute in the database by showing the two most common words, phrases, numbers, etc. in a cluster for that attribute. This reveals the content of the objects that comprise each peak, and provides context for further navigation and query.

VxInsight supports multi-resolution zooming into the landscape to explore interesting regions in greater detail, which reveals structure on multiple scales. Following each mouse click, the landscape and labels are recalculated, to give a new, higher resolution view of the desired terrain. Temporal data can be viewed using a time slider to reveal growth and reduction in areas of interest, new emerging areas, and bridged regions that have merged together.

Data access and retrieval is achieved via an ODBC connection to the user's data source. Clicking on a single data element provides detail on demand for that item (such as author, source, title, etc.) A query window allows the user to interrogate the data source, resulting in colored markers on the terrain showing those items matching the query. The distribution of query markers in the context of the terrain with its labels can be very meaningful to the analyst. Various analysis tasks can be accomplished by combining navigation, multiple queries, and time sliding functions.

4 Domain Visualizations

Several different visualizations were prepared to show the domain comprised by the fields of information visualization (IV), human-computer interaction (HCI), cognitive psychology (CP), and digital libraries (DL). One of the main advantages of domain visualization is the ability to combine and explore related work from different fields.

The first step in this process was to procure an appropriate set of bibliographic data. Data were retrieved from the Science Citation Indexes (SCI and Social SCI) through the SciSearch[1] web interface used at Sandia. A short list of search terms related to the four fields was compiled (see Table 1) and was queried against titles, abstracts, and keywords for years 1991-mid 2001 for the SCI, and 1995-mid 2001 for the SSCI. The number of articles retrieved by each query is shown in Table 1, along with unique numbers of articles once duplicates were removed.

A total of 4478 unique articles were thus retrieved, with approximately 700, 800, 2000, 370 articles in the IV, HCI, CP, and DL fields, respectively. The majority of the duplicate articles were the result of overlap between the SCI and SSCI, rather than overlap between search terms. The term 'mental models' was included in the original search, but was found to have little overlap with fields other than CP. Thus, it will not be discussed further. Also, the terms 'exploration,' 'navigation,' and 'browsing' can often be associated with digital libraries; however, given the narrowing of these queries with the terms 'information' and 'data', the results can be expected to fit with the IV field more than with the DL field, as will be confirmed later (see discussion on query overlaps in section 4.1)

Three different domain maps based on these data were produced: an article map, a semantic (keyword) map, and an author map. Each will be described further below.

[1] * The SciSearch engine supports automatic stemming and verb tense variation.

Table 1. Search terms and numbers of articles retrieved.

Search term and field		'91-'01 SCI	'95-'01 SSCI	Unique SCI	Unique SSCI
cognitive model	(CP)	363	430	363	270
cognitive science	(CP)	386	450	375	280
cognitive psychology	(CP)	236	415	216	289
cognitive system	(CP)	162	125	148	49
information visualization OR visualization of information OR data visualization OR visualization of data OR interactive visualization	(IV)	434	51	434	14
Exploration*	(IV)	149	34	143	21
navigation*	(IV)	53	14	50	6
browsing*	(IV)	26	4	26	3
digital library	(DL)	246	219	238	127
human computer interaction	(HCI)	565	234	539	72
human computer interface	(HCI)	204	49	170	13
mental model		322	532	292	340
TOTALS		3146	2557	2994	1484

* search perturbations for these terms were the same as those shown for the 'visualization' terms.

4.1 Article Map

A map of articles in the combined IV/HCI/CP/DL domain was generated based on the number of ISI keywords in common between each pair of articles. The similarity metric used (w_{ij}) is a cosine similarity given by the expression:

$$w_{ij} = T_{ij} / \text{sqrt}(n_i * n_j) \ . \qquad (\mathbf{2})$$

where T_{ij} is the number of keywords in common for articles i and j, and n_k is the number of keywords for any article k. A threshold value for w_{ij} of 0.2 was applied to keep low w_{ij} values from dominating the clustering. The single highest w_{ij} value for each article whose maximum w_{ij} was below the 0.2 threshold was also kept, in order to not exclude those articles from the map.

Clustering was performed using VxOrd, resulting in a map of 3142 articles. 1336 articles were not given positions on the map since they had no keywords in common with any other article. 60% of the DL articles are in this category, suggesting that additional text-based analyses (e.g. using latent semantic indexing) are needed to fully understand the DL overlaps. Figures 2-4 show the domain map for three separate 2-year time periods.

Examination of Figures 2-4 reveals that although there are some overlaps in the four fields, they are not extensive. Peaks in the middle and right portions of the terrain are dominated by CP (magenta dots), with but few IV and HCI papers in some areas. IV work (green dots) is found near the top center of the terrain, and does show significant overlap with DL (white dots) in one peak. Most of the HCI work (blue dots) is in the peaks at the far left and lower left. These peaks show perhaps more overlap between the four fields than any others. Detailed views of these two HCI peaks are shown in Figure 5, which covers the entire time period from 1991-mid 2001. The far left peak (Figure 5a) contains mostly HCI material with a scattering of IV and CP at the edges. The peak at the lower left (Figure 5b) is actually a ridge of two clusters with bridging material between them, where the left-most peak has more HCI material and the right-most peak has more CP material. Close examination of the contents of each group indicates that the left-most cluster is concerned with interfaces and design, while the right-most cluster is more concerned with systems and cognition. The work in the center deals with relevance of HCI design.

Figures 2-4 also show trends in publishing in the four fields. DL work first appears in the 1995-1996 time frame, and grows through the 1999-2000 time frame. In addition, some DL work has moved from the core DL work in the center of the terrain to the HCI peaks by year 2000. This indicates that DL may be receiving some benefit or insertion from the HCI/CP work that forms the HCI clusters. This observation is tempered somewhat by the lack of many direct query overlaps. In fact, only 8 papers retrieved with the DL query were also retrieved using any of the other queries. Thus, while DL work may be benefiting indirectly from the IV/HCI/CP work, very little work connects them directly.

Additional trends from Figures 2-4 include a slight shift in HCI work (lower left peaks) from interface design to a system design with more cognitive modeling input. This is indicated by the shift in peak sizes and query marker (colored dot) distributions in Figures 2-4.

Perhaps the most exciting overlap from a DL standpoint occurs in the DL peak near the top center of Figure 4 (years 1999-2000), which shows a significant overlap with IV. A detailed view of this region is shown in Figure 6. Here, *retrieval design* and *database retrieval* are sandwiched between DL and IV clusters. Several HCI and CP papers are also found in this region of convergence between IV and DL. This topic of *retrieval* is thus at the overlap of all four fields, and is an area in which DL can benefit from collaborations across the other three disciplines.

This conclusion that future work based on convergence between the IV, HCI, CP, and DL fields should focus on information retrieval may appear elementary. However, this analysis provides a formal basis to that conclusion, and may also suggest specific work that can be built upon for such studies. A list of the articles that comprise the view in Figure 6 is too large to include here, but may be obtained from the author.

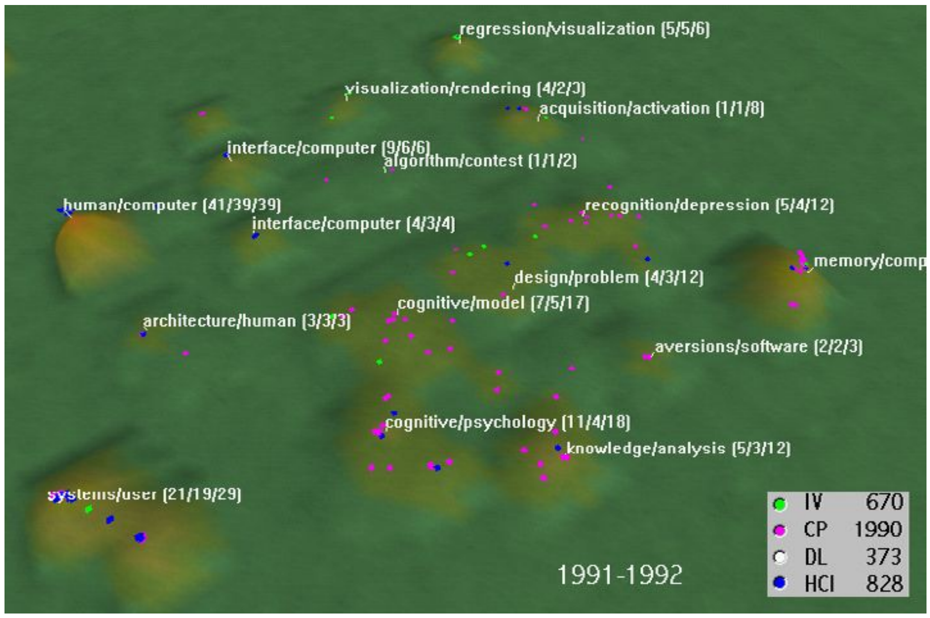


Fig. 2. IV/HCI/CP/DL domain for the 1991-1992 time period.

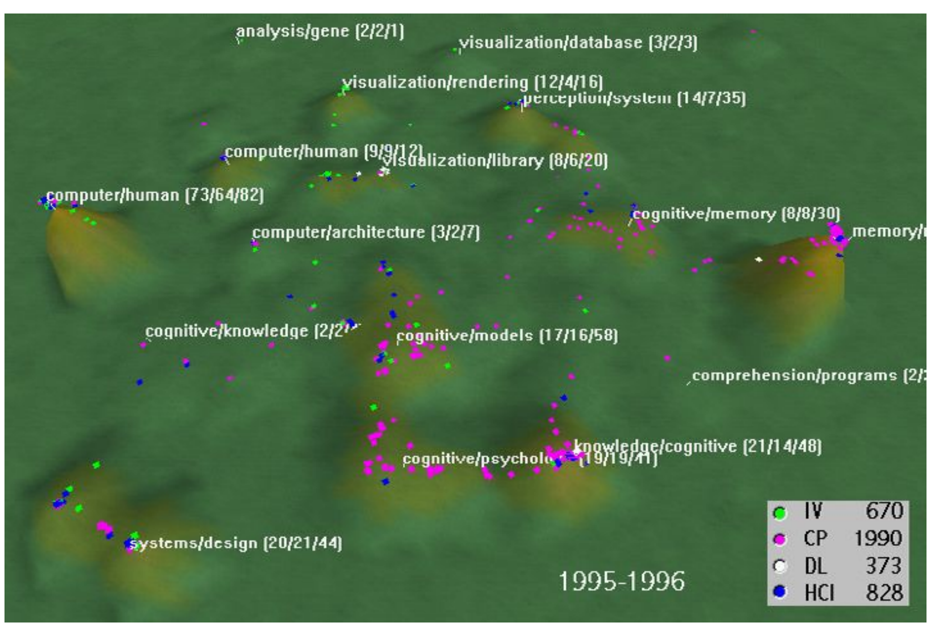


Fig. 3. IV/HCI/CP/DL domain for the 1995-1996 time period.

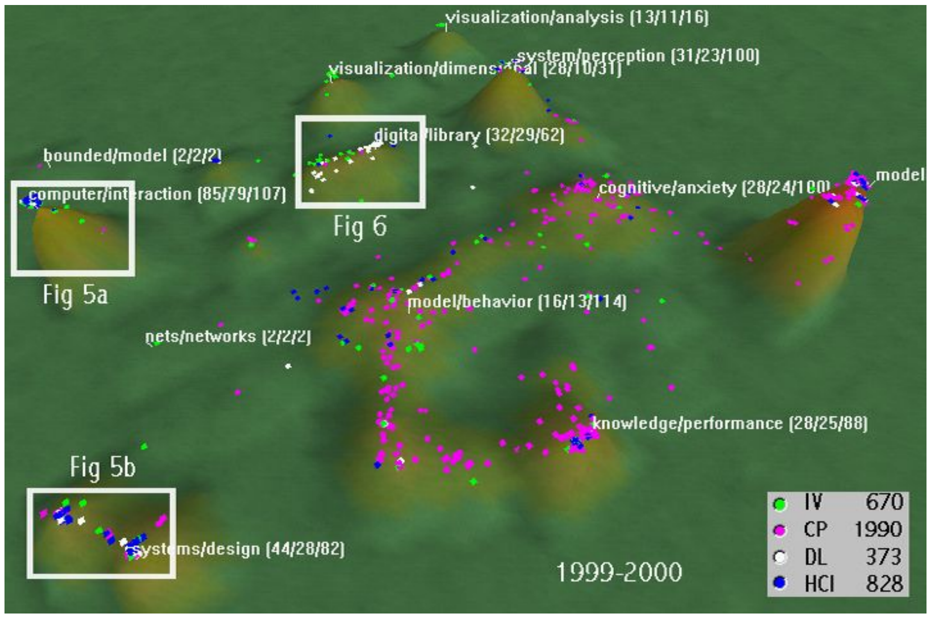


Fig. 4. IV/HCI/CP/DL domain for the 1999-2000 time period.

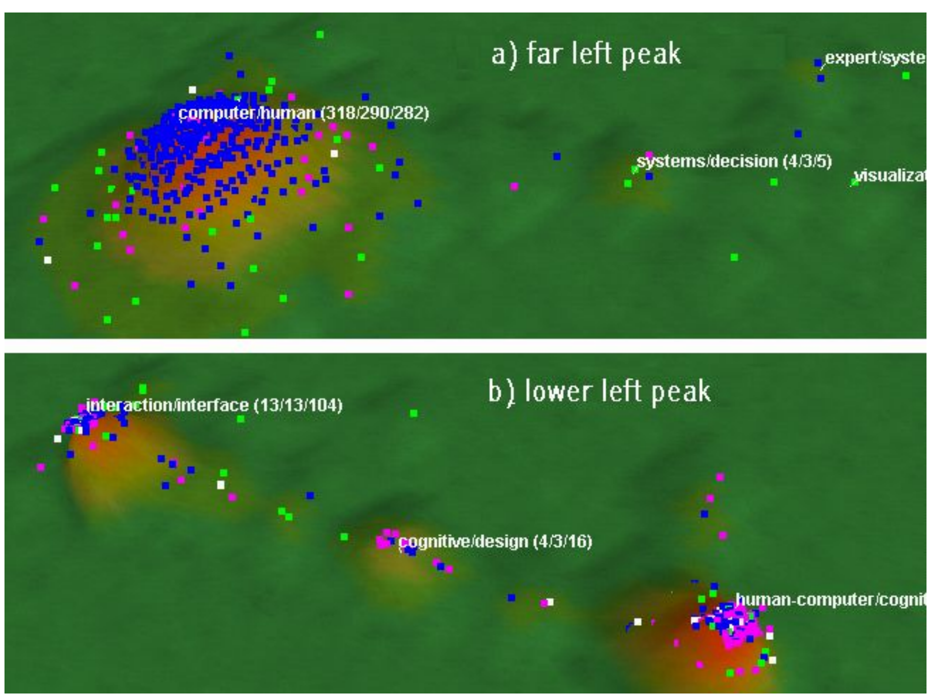


Fig. 5. Detail on human-computer interaction (HCI) peaks from Figures 2-4 for the entire time period of 1991-mid 2001.

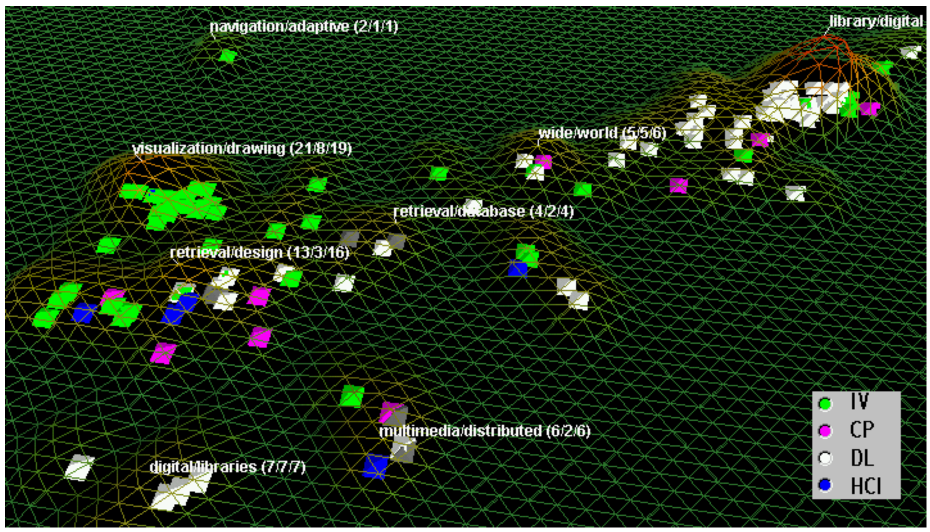


Fig. 6. Detail on the digital libraries / information visualization (DL/IV) peak from Figures 2-4.

4.1 Semantic Map

In addition to the article map described above, a semantic map (or map of terms) representing the domain was also created. This map was created using ISI keywords as terms. No words parsed from titles or abstracts were used as terms. A cosine similarity between terms *i* and *j* was used (see equation 2) where T_{ij} is the number of articles for which terms *i* and *j* were both designated as keywords, and n_k is the number of occurrences for term *k*. This analysis was restricted to terms occurring at least twice in the corpus of documents, comprising 2373 terms in all. No similarity threshold was employed and VxOrd was used to calculate the term positions.

Figure 7 shows those terms in the semantic map that occur 35 times or more in the document corpus. The center map shows the spatial relationship between the three main clusters of terms, while the other three segments show individual terms and their spatial relationships.

The cluster at the upper left contains terms related to three of our four disciplines: IV, HCI, and DL. The term *information retrieval* is found in this cluster, which indicates its prominence, and which adds credibility to the conclusion reached above that it is a current topic of overlap between the four fields, and should be the focus of future work.

The cluster at the upper right is concerned with the medical side of cognitive psychology, while the large cluster at the bottom of the terrain focuses on various cognitive processes and information structures. Yet, there is a gap between the computer-related terms in the upper left cluster and the cognitive processes of the lower cluster, indicating that the bridge between the cognitive sciences and processes and the IV/HCI/DL fields is not as strong semantically as it needs to be to have a significant impact on the digital libraries of the future.

It is interesting to examine the change over time in terms lying between the upper left and lower peaks from Figure 7. Terms such as *automatic analysis* and *graph algorithms* are bridging the space between the IV/HCI/DL terms and the cognitive sciences. Tools with these attributes may prove fruitful in inserting more of the benefits from cognitive sciences in digital libraries of the future.

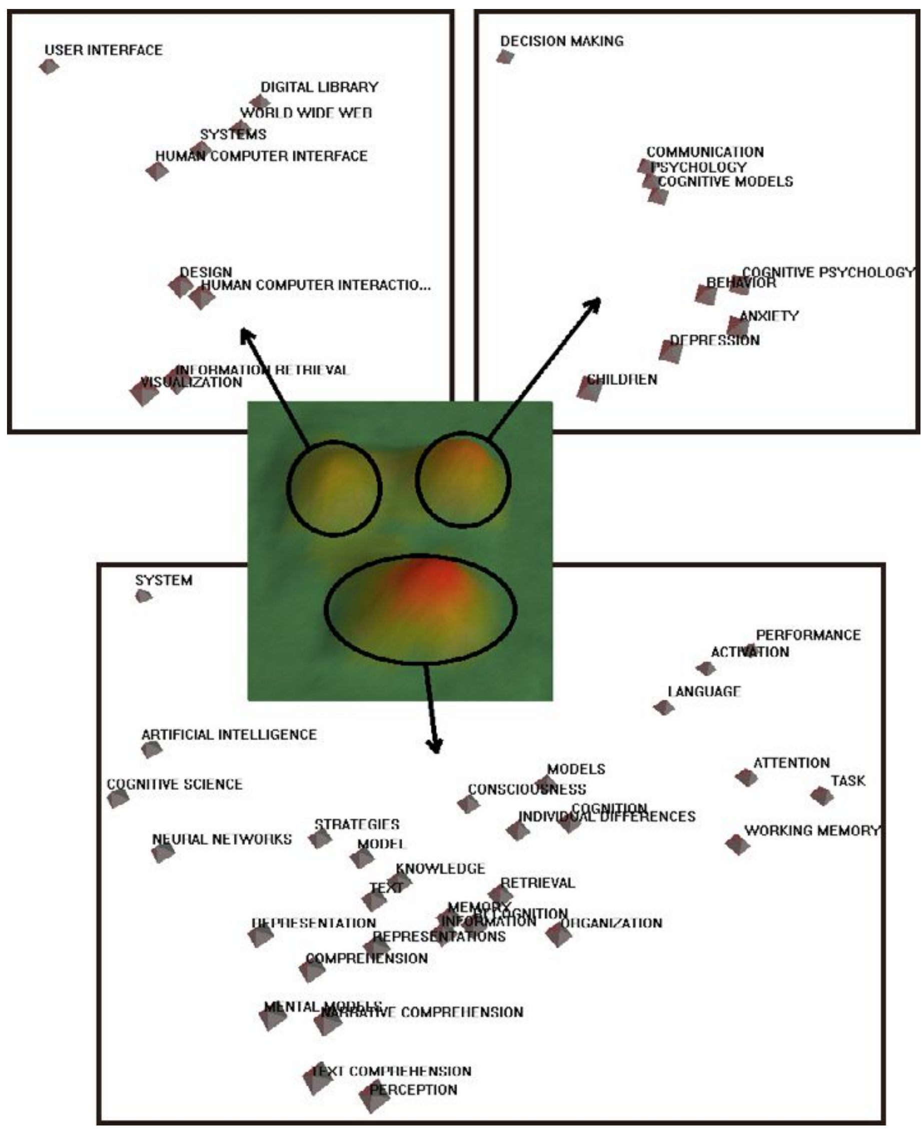


Fig. 7. Semantic map for the IV/HCI/CP/DL domain showing spatial relationships between the most common terms (ISI keywords) for the domain.

4.2 Co-author Network

A co-author network for the IV/HCI/CP/DL domain has also been generated. Only authors with 2 or more papers in the document corpus were included. 885 authors matched this criterion. Association was once again calculated using a cosine similarity where the union term T_{ij} denotes the number of papers co-authored by a pair of authors.

The co-author network is represented in Figure 8. Authors whose papers were retrieved by queries to the four fields are shown as dots of different colors. There are very few instances where dots of more than one color occur in the same local cluster. In addition, there are many, many clusters in this map, but only one arrow visible (at the scale indicated by the figure) that join more than one cluster. This indicates that there really is no co-author network established in this domain. Rather, the majority of researchers have interactions with a small group of others, doing research in one of the four fields. For a convergence in the IV, HCI, CP, and DL fields to truly occur, much more collaboration across fields is needed.

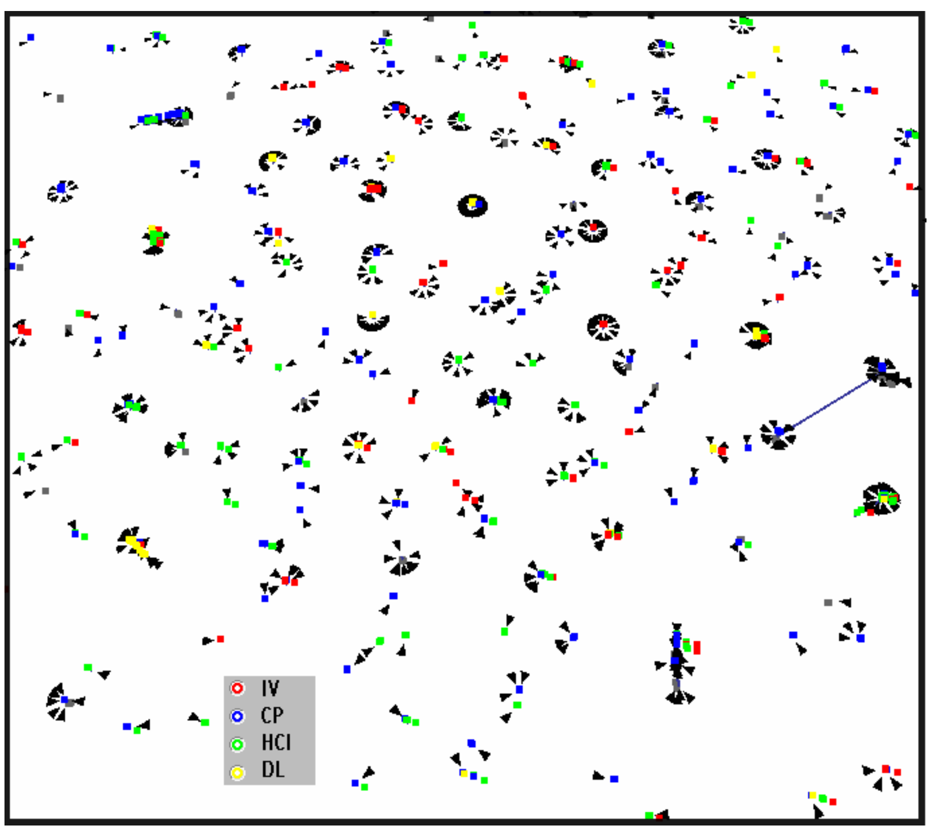


Fig. 8. Co-author network. Arrows show connections between authors (co-authorship). All but a few arrows are within clusters rather than between clusters.

5 Summary and Future Work

We have produced three visualizations of the domain comprised of the fields of information visualization, human-computer interaction, cognitive psychology, and digital libraries. Analysis based on dynamic views of the maps indicates that the current overlap between these fields is not substantial. However, the analyses also indicate that there are areas where recent research has occurred, and where future research should be focused to greatly benefit the digital libraries of the future.

Collaborative work between researchers in the four fields on the topics of *information retrieval*, *advanced graphing,* and *automated analysis algorithm*s are currently hot topics at the intersection of these fields. Such collaborations should not only benefit digital libraries, but other areas of focus between pairs of the disciplines.

Given that 60% of the DL articles had no keywords in common with any other article, and thus were left out of the article map of section 4.1, we advocate a more substantial textual analysis of the data, perhaps using LSA to extract words from titles and abstracts. If there are additional overlaps between DL and the other fields, they should be uncovered using those methods.

We also advocate a further analysis of authors across the four fields using an author co-citation analysis rather than the co-author study performed here. A co-citation analysis could point out a more extensive research network than we are led to believe exists based on the co-author network alone, and could suggest researchers and institutions that should collaborate more fully.

References

1. Davidson, G.S., Hendrickson, B., Johnson, D.K., Meyers, C.E. & Wylie, B.N. (1998). Knowledge mining with VxInsight: discovery through interaction. Journal of Intelligent Information Systems 11, 259-285.
2. Boyack, K.W., Wylie, B.N., Davidson, G.S. & Johnson, D.K. (2000). Analysis of patent databases using VxInsight. ACM New Paradigms in Information Visualization and Manipulation ‘00, McLean, VA, Nov. 10, 2000.
3. Davidson, G.S., Wylie, B.N. & Boyack, K.W. (2001). Cluster stability and the use of noise in interpretation of clustering. Proc. IEEE Information Visualization 2001, 23-30.
4. Boyack, K. W., Wylie, B. N., & Davidson, G. S. (2002). Domain visualization using VxInsight for science and technology management. Journal of the American Society of Information Science and Technology 53(9), 764-774.
5. Small, H. (1997). Update on science mapping: creating large document spaces. Scientometrics 38, 275-293.
6. Noyons, E.C.M. & Van Raan, A.F.J. (1998). Advanced mapping of science and technology. Scientometrics 41, 61-67.
7. Spasser, M.A. (1997). Mapping the terrain of pharmacy: co-classification analysis of the International Pharmaceutical Abstracts database. Scientometrics 39, 77-97.
8. Leydesdorff, L. (1994). The generation of aggregated journal-journal citation maps on the basis of the CD-ROM version of the Science Citation Index. Scientometrics 31, 59-84.

9. Bassecoulard, E. & Zitt, M. (1999). Indicators in a research institute: A multi-level classification of scientific journals. Scientometrics 44, 323-245.
10. Chen, C. (1999). Visualising semantic spaces and author co-citation networks in digital libraries. Information Processing and Management 35, 401-420.
11. Börner, K. (2000). Extracting and visualizing semantic structures in retrieval results for browsing. ACM Digital Libraries '00, San Antonio, TX, June 2000.
12. White, H. D. & McCain, K. W. (1998). Visualizing a discipline: An author co-citation analysis of information science, 1972-1995. Journal of the American Society for Information Science 49(4), 327-355.
13. Fox, K.L., Frieder, O., Knepper, M.M. & Snowberg, E.J. (1999). SENTINEL: A multiple engine information retrieval and visualization system. Journal of the American Society for Information Science 50(7), 616-625.
14. Chen, C., Paul, R.J. & O'Keefe, B. (2001). Fitting the jigsaw of citation: Information visualization in domain analysis. Journal of the American Society for Information Science and Technology 52(4), 315-330.
15. Honkela, T., Kaski, S., Kohonen, T. & Lagus, K. (1998). Self-organizing maps of very large document collections: Justification for the WEBSOM method. In I. Balderjahn, R. Mathar & M. Schader (Eds.) Classification, Data Analysis, and Data Highways. Berlin: Springer.
16. Small, H. (1999). Visualizing science by citation mapping. Journal of the American Society for Information Science 50(9), 799-813.
17. Hetzler, B., Whitney, P., Martucci, L., & Thomas, J. (1998). Multi-faceted insight through interoperable visual information analysis paradigms. Proceedings of IEEE Information Visualization '98, 137-144.
18. Wise, J.A. (1999). The ecological approach to text visualization. Journal of the American Society for Information Science 50(13), 1224-1233.
19. Buzydlowski, J.W., White, H.D. & Lin X. (2003) Term co-occurrence analysis as an interface for digital libraries. Lecture Notes in Computer Science. Springer. THIS BOOK - CHANGE reference as necessary.

On Geometry and Transformation in Map-Like Information Visualization

André Skupin

Department of Geography
University of New Orleans
New Orleans, LA 70148
askupin@uno.edu

Abstract. A number of visualization techniques have been put forward that implement a map metaphor to display abstract, non-georeferenced information. This paper refers to these as *map-like information visualizations* that are distinguished from other information visualization approaches in a number of ways. It interprets some of the principles underlying these techniques within a framework informed by geographic information science (GIScience). Recent geographic efforts in this research area have linked ideas about the nature of geographic information to cognitive schemata proposed by cognitive linguists. This paper draws on the arguments that have emerged from those efforts regarding the nature and usefulness of geographic metaphors. It proposes to discuss particular projection techniques, like multidimensional scaling or self-organizing maps, with reference to the geometric primitives they employ. These primitives will drive the choice of geometric and symbolic transformations that are necessary to achieve a particular visualization. Designers of map-like visualizations are thus challenged to seriously consider the implications of particular computational techniques and the consequences of symbolization choices.

1 Introduction

Two-dimensional representations have become a pervasive theme in the development of visual exploration and retrieval tools for digital libraries. Many of the proposed visualizations are decidedly "map-like", exhibiting graphic elements and design characteristics of traditional maps. What makes a visualization of abstract information, such as a document corpus, *map-like*? Like traditional maps, such visualizations are constructed from geometric primitives that then become associated with certain map symbols and displayed on a flat display surface. Traditional geographic maps as well as map-like information visualizations are the result of some form of projection of a higher-dimensional configuration into a two-dimensional display space. However, map-like visualizations are not maps in the traditional sense, because they depict abstract information spaces, instead of geographic space.

What sets these visualizations apart from other information visualization techniques? General information visualization techniques include three-dimensional displays. Map-like visualizations are typically restricted to two-dimensional displays, with the possible exception of landscape visualizations. The latter type is often

K. Börner and C. Chen (Eds.): Visual Interfaces to Digital Libraries, LNCS 2539, pp. 161–170, 2002.

referred to as 2.5-dimensional. Since these landscape visualizations are constructed by interpolation or extrusion of numeric attributes from a two-dimensional geometric configuration, they are still dependent upon the particular characteristics of the two-dimensional techniques discussed in this paper.

Map-like visualizations are distinguished from other two-dimensional visualizations by how coordinate axes are defined. It is different from all those methods in which two dominant variables of a data set are directly mapped to the two axes. While map-like visualizations are in agreement with those methods regarding the primacy of location for human observers [1, 2], they do not exhibit such clear association between each axis and one of the input variables. They do also not give preferential treatment to any particular axis. Classic tree layouts, for example of dendrograms created through hierarchical clustering, are not considered to be map-like, due to the direct mapping of cluster distances onto one axis. Axes in map-like visualizations are defined very differently, compelling Shneiderman et al. [3] to refer to them as "non-meaningful." Indeed, by far the most frequently asked first question of users confronted with such visualizations is: "What do the axes mean?" The two axes are not meaningless, but rather they reflect aspects of *all* the input dimensions (i.e., variables) in a complex manner, the particulars of which are determined by the employed projection technique. It is exactly this underlying mix of input variables that gives these visualizations the potential to portray high-dimensional information spaces in a map-like form.

2 Relationship to Other Work

By its very nature, research in information visualization crosses the boundaries of individual disciplines. Researchers in this rather young area do however emerge from distinct, established academic disciplines and this is necessarily reflected in the particular approaches that are being pursued. A look at dominant publication outlets confirms that the core of the information visualization research community consists of computer scientists. They have put forward most of the individual techniques as well as overarching taxonomies. Of particular relevance to this paper are efforts to develop taxonomies of visualizations [4-8].

Geography and cartography have a long history of information visualization activities, if we were to include *geographically referenced information*, which is typically visualized in the form of maps. Geographic and cartographic interest in the provision of visual representations for *abstract information* is a more recent phenomenon. A particularly influential guiding principle of those efforts has been Waldo Tobler's *First Law of Geography,* published in a 1970 paper, according to which everything is related to everything else, but closer things are more closely related [9]. It was a logical predecessor to Tobler's later observation, in 1979, of the parallels between multidimensional scaling and the surveying technique of trilateration [10].

Any discipline that wants to make scientific contributions outside its established territory needs to first clarify the nature of its domain of inquiry and define a set of core concepts. For geography, geographic space represents the core domain. Among its core concepts are location, region, distribution, spatial interaction, scale, and change [11]. How are these concepts relevant to map-like interfaces to abstract information? If *spatial metaphors* represent a useful basis for the design of user interfaces, then geographic concepts dealing with *space* should be given serious consideration. As for *metaphors*, the work of cognitive linguists [12, 13] has been particularly influential. Couclelis [14] convincingly links those metaphor notions with geographic concepts. She argues that there are three fundamental groups of questions that arise in this endeavor:

- questions regarding the *meaning of geographic concepts* in visual representations of abstract information,
- how geographic concepts can help to perform relevant *cognitive tasks*, and
- issues surrounding the particular concepts, tools, and methods for incorporating geographic concepts into a *visual presentation* of abstract information.

Fabrikant and Buttenfield [15] draw on lessons from geography, cognitive science, and human computer interaction (HCI) to distinguish three spatial frames of reference. These are grounded in geographic space, cognitive space, and Benediktine space [16], respectively. Each is associated with metaphors of distinct character, with consequences for the design of actual interfaces. Fabrikant's ongoing research extends much beyond previous empirical work by testing the relevance and usefulness of particular components of the geographic metaphor, such as distance or scale [17].

Skupin [18, 19] proposes to consider the relevance of cartography to information visualization beyond an appreciation of the map metaphor. This refers particularly to the principles underlying map projection techniques, to problems of graphic complexity, to the choice and positioning of labels, and to map design principles.

3 Geometric Configurations

Research and development efforts in information visualization have matured to a point at which the provision of meaningful taxonomies of the various techniques is a necessary step towards the creation of a coherent theoretical framework on which further progress will depend. A number of taxonomies have now been proposed. Some of these treat the information visualization field in its entirety [4, 5, 8]. Other taxonomies are devoted to specific groups of techniques. Examples are papers on graph visualization [6] and pixel-oriented techniques [7].

This paper proposes to distinguish projection techniques used for map-like information visualizations according to the geometric primitives they employ. Distinct techniques exist to project elements of a high-dimensional information space in order to create two-dimensional configurations made up of these basic geometric primitives, which are either zero-, one-, or two-dimensional. This proposed division of techniques derives from Couclelis' argument regarding the cognitive rationale behind use of the spatial metaphor [14]. She argues that experiential space is made up of certain elementary building blocks that correspond to the geometric primitives of mathematical space. *Places*, *ways*, and *regions* are fundamentally distinct experiential categories. If we are to make the map metaphor believable and useful, then we have to give serious consideration to how the corresponding geometric primitives of *points*, *lines*, and *areas* are created, transformed, and ultimately visualized.

3.1 Points

Zero-dimensional primitives are employed by such techniques as multidimensional scaling (MDS), principal components analysis (PCA), and spring models. Information space elements enter these methods as discrete units, typically in some form of vector space model [20], with only implicit representation of inter-document relationships. In the case of MDS, distance between documents is made explicit and computed as dissimilarity for each pair of documents. This method makes a fairly overt attempt to preserve distance relationships. In the case of nonmetric MDS, which is appropriate for non-Euclidean, nonmetric dissimilarity measures, this is based on the rank order of dissimilarities. This also helps to allow the bridging of large dimensional gaps between vector space and low-dimensional display space, since some contraction of unused vector space portions and expansion of dense portions can occur. However, MDS implementations typically do not convey these distortions to the user at all. Zero-dimensional configurations can be visualized in a straightforward manner, by linking point symbols and labels to the computed point locations.

Point configurations are useful for the creation of landscape visualizations. Feature attributes can be linked to point locations as elevations and interpolated to form a continuous surface [21]. While this can result in a visually attractive representation, the attribute to be interpolated as well as the interpolation function and parameters have to be chosen carefully. Existing proposals in this direction rarely consider how meaningful the mixture of continuous surfaces with discrete feature labels is [18], or how to visually represent uncertainty associated with different portions of an interpolated surface.

3.2 Lines

Typical for one-dimensional configurations are graph layout methods. To this category belong tree graph methods, which are used to visualize hierarchies such as those obtained by hierarchical clustering procedures. Graph layout methods have also been developed for non-tree structures, such as the topological structure of hypermedia information spaces. Herman et al. [6] provide a comprehensive survey of graph visualization methods. Examples for map-like graph visualizations are H-tree layouts, balloon views, and hyperbolic views.

The added dimensionality of links between node locations provides graph visualizations with an opportunity to directly visualize distortions, since links themselves can be symbolized according to the degree of distortion. Methods like the hyperbolic tree make use of the added affordance of a linked representation in a different way, by introducing distortions to provide spatial context.

3.3 Areas

Elements of an information space could be represented in two-dimensional form as areas. In geographic representations, areas are often conceptualized as topologically disconnected entities (e.g., lakes or metropolitan areas) and visualized accordingly. In map-like information visualization it is far more common to create topologically connected areas. This typically amounts to the tessellation of a given display surface.

The tree map method [22] is the prime example for such a tessellation-based area representation. Tree maps are frequently used whenever hierarchically structured data, such as from the Open Directory Project, are encountered. Tree maps provide a complete tessellation with areas of different sizes being assigned to leaves and nodes of the hierarchy. Variation of area sizes on tree maps is akin to what cartographers call cartograms, in which the size of geographic objects, such as individual countries, is changed to reflect some numeric attribute.

Less explicitly structured input data, such as high-dimensional vector spaces, can also be visualized with tree maps, after computation of a hierarchical clustering solution.

3.4 Fields

Geometric configurations based on points, lines and areas reflect a conceptualization of information spaces as consisting of discrete objects. Alternatively, one could interpret elements of a digital library as sample observations of an information *continuum*. Phenomena exhibiting continuous, gradual variation are commonly referred to as *fields*.

The most common information visualization technique implementing a field concept is the self-organizing map (SOM) method [23]. It creates a regular tessellation using uniform area units, akin to raster elements used in digital imagery and GIS. SOMs indeed behave similarly to standard raster data models, compared to the vector-like behavior of the object conceptualizations discussed in the previous sections. For example, how a SOM can be used very much depends on its resolution. A SOM with very fine resolution, i.e. a large number of neurons or nodes, will enable the creation of a detailed visual representation, including the eventual ability to distinguish individual documents [19]. On the other hand, using a coarse SOM amounts to a document classification.

The objective function of the classic Kohonen algorithm is similar to k-means clustering and attempts to preserve topological neighborhood relationships. Kohonen maps perform rather well at this, but at the cost of a pronounced contraction of those map areas that correspond to thinly populated portions of the high-dimensional information space.

3.5 Alternative Geometric Configurations

Researchers developing map-like interfaces have to consider that fundamental spatial relationships, such as proximity and neighborhood, are highly dependent on the specific method used to create the initial configuration. It is often possible to use alternative projection techniques, but the most fundamental differences are found when the employed geometric primitives are of a different dimensionality.

Consider as one example the task of visualizing a hypermedia network, such as a set of linked Web pages. One possibility would be to compute a distance matrix based on network distance, i.e. the number of hyperlinks that one would have to traverse to jump from one node to another. This distance matrix is fed to a MDS procedure. Connecting the resulting node locations with straight-line segments, according to the hyperlink structure, then finishes the depiction of the hypermedia network. Now imagine an alternative solution, in which a graph layout method is used to determine coordinate locations for nodes with explicit consideration of the link structure. Even though both results could eventually employ identical symbolization, i.e. identical point and line symbols, the two visualizations would be fundamentally different.

For another example, imagine that one would take the vector space model of a document corpus as input and create alternative visualizations using MDS and SOM. One approach would compute a similarity matrix and use MDS to derive points directly. The second approach would first train a Kohonen map and then find the set of neurons that best fit the input data set. The MDS configuration will provide explicit coordinates for each document, but will be less flexible when additional documents are to be mapped, because the geometric space between points is not explicitly defined in terms of the input feature space. Within the area tessellation of the Kohonen map the high-dimensional vector space of the training data set is represented in explicit chunks. Every portion of the trained SOM is thus explicitly associated with a portion of the information continuum. That makes it very easy to map out documents that were not part of the SOM's training data set. The chunking of vector space comes at a price though. Depending on the coarseness of the grid of neurons, individual neurons can become associated with multiple documents, preventing the assignment of discrete document coordinates. Choosing a finer resolution SOM can counteract this. However, training a high-resolution SOM exacts a computational toll, much like the processing of high-resolution satellite imagery does.

The division of techniques according to the dimensionality of geometric primitives follows from cognitively useful distinctions in experiential space. Higher-level concepts, such as *region* or *scale*, derive from the more basic concepts. However, there may be fundamental differences of the degree to which individual techniques meaningfully support higher-level concepts. Consider the differences between Kohonen maps and treemaps. Using either method we could communicate the existence of *regions* via area fill color. To the human observer, the two map-like visualizations are then functionally identical. However, differences in the principles underlying the two techniques mean that regions are actually constructed in very different ways. Borders between regions will be defined more locally and strictly for tree maps and more holistically and fluidly for Kohonen maps. Related to this, the concept of *scale* will be embedded very differently in these two visualizations. How is this of concern to interface designers? Users expect map-like visualizations of non-geographic information to *function* like geographic maps, at basic and higher levels. Any mismatch between this expectation and the reality of an interface should be of concern and at the very least be communicated to the user.

4 Map Transformations

Some cartographers have long seen mapmaking as a sequence of transformations [10], similar to the visualization reference model proposed by Card et al. [24]. Understanding this transformational character of maps can help designers of map-like interfaces to more fully realize the potential of the geographic metaphor.

4.1 Transformations Between Geometric Configurations

The categorization of techniques presented in the previous section does not consider any particular method of symbolizing or otherwise transforming the geometric configuration that might occur *after* the projection. Transformations between different geometric arrangements are common in Geographic Information Systems (GIS) and easily modifiable to fit the needs of non-geographic visualization. For example, a point configuration obtained by MDS can be turned into a continuous surface representation through surface interpolation. If the goal is the delineation of point territories, then tessellation into Voronoi polygons is an easy choice. Use of geometric transformations can help to mitigate some of the problematic issues encountered with particular projection techniques. For example, one can derive individual document coordinates from a Kohonen map by randomly distributing document points inside their respective neurons. Depending on the coarseness of the neuron grid, this will result in the kind of solution typically obtained by MDS, but without the scalability problems of that method.

Error, uncertainty, and distortion characteristics of the original data set as well as of the original projection will of course propagate throughout all further processing and should be considered when visualizations are eventually created.

4.2 Alternative Visualizations from One Geometric Configuration

Cartographic representations are also transformational in the sense that a single geometric configuration could lead to a number of valid visualizations that might encode equivalent information, but are not equal. The data set underlying the visualizations shown here is a human subject test used to investigate aspects of geographic ontology [25]. Subjects were asked to list examples for geographic “things”, with five variations in how the specific question was posed (“geographic feature,” “geographic object,” “geographic concept,” “something geographic,” and “something that could be portrayed on a map”). A vector space model is created from the responses consisting of 31 objects and five variables. Then a Kohonen map is computed as the basis for a series of transformations. In the first visualization, unique two-dimensional coordinates are displayed for each of the input objects (Figure 1). Then, a hierarchical clustering solution is computed for the original 31 objects and the resulting ultrametric tree is projected onto the point configuration (Figure 2). Line thickness corresponds to distance levels such that thicker lines connect points that are closer in feature space. Notice how similar feature space distances may correspond to very different 2-D distances (e.g., the *river-mountain* pair vs. the *road-city* pair). One could also use the point configuration to investigate the five input variables by producing five interpolated surfaces and displaying them in a form similar to Tufte’s small multiples [26](Figure 3). In another approach, two levels of a hierarchical clustering solution of the five-dimensional SOM neurons are shown using the Viscovery SOMine software (Figure 4). Finally, a pie chart map is displayed, constructed from the relative proportion of subject responses for each term (Figure 5). All of the figures, with the exception of figure 4, were produced by combining a given SOM-derived point configuration and transforming it using statistical and geometric operators, and finished in standard GIS software.

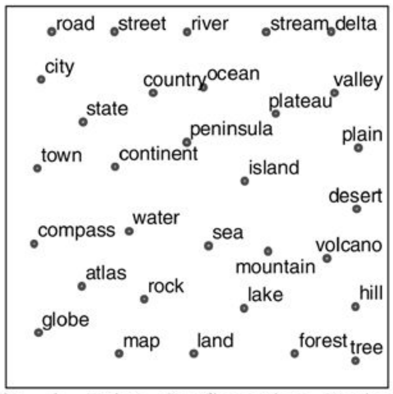


Fig. 1. Point Configuration Derived from Self-Organizing Map

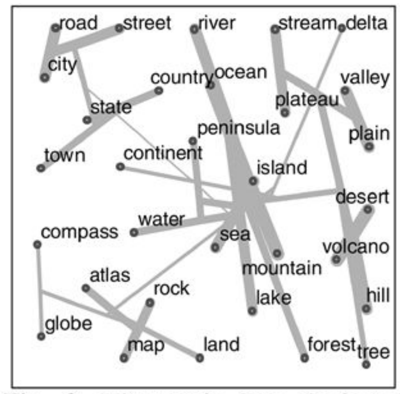


Fig. 2. Ultrametric Tree Projected onto Point Configuration

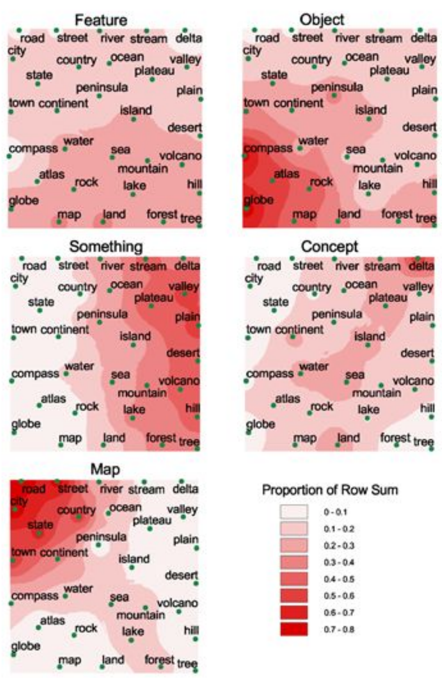


Fig. 3. Visualization of Five Input Variables as Interpolated Surface (from [25])

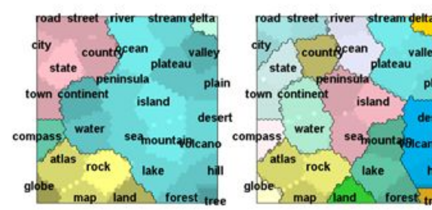


Fig. 4. Two Levels of a Hierarchical Classification of SOM Neurons Visualized on the SOM Geometry. Five- and Fifteen-Cluster Solutions Shown (from [25])

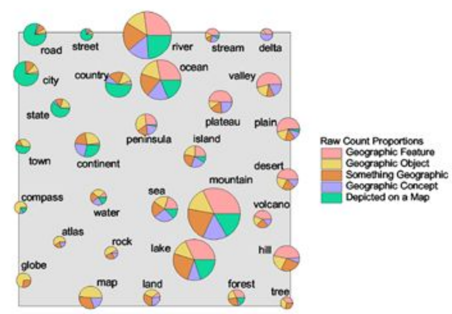


Fig. 5. Visualization of Five Input Variables as Pie Charts Associated with Point Locations

5 Conclusions

Those attempting to produce useful map-like representations of abstract information are faced with a multitude of design decisions. There are a number of techniques available that will transform an information space into a low-dimensional geometric configuration. This paper argued that grouping these techniques according to the employed geometric primitives helps to understand their conceptual underpinnings, for example regarding the difference between object and field conceptualizations of an information domain. This helps to point toward the transformations that can turn two-dimensional configurations into a variety of geometric and topological structures. Add to this a choice among symbolization techniques and it becomes clear that the eventual appearance of a visualization does not have to be driven by the initial projection method. One implication of this is that similar symbolization techniques can be applied to the results of such diverse techniques as self-organizing maps, multidimensional scaling or pathfinder network scaling. Comparative studies of different visualization techniques with respect to issues of computation, perception, and comprehension should thus be much more feasible than the current lack of such studies suggests.

Acknowledgments

Support provided through the Louisiana Board of Regents Support Fund, grant # LEQSF(2002-05)-RD-A-34.

Note: This is an expanded and revised version of a paper titled “Cartographic Considerations for Map-Like Interfaces to Digital Libraries,” presented at the Workshop on Visual Interfaces to Digital Libraries, Roanoke, Virginia, June 28,2001.

References

[1] J. Bertin, *Semiology of Graphics: Diagrams, Networks, Maps*. Madison, WI: University of Wisconsin Press, 1967/1983.
[2] A. M. MacEachren, *How Maps Work*. New York: The Guilford Press, 1995.
[3] B. Shneiderman, D. Feldman, A. Rose, and X. F. Grau, "Visualizing Digital Library Search Results with Categorical and Hierarchical Axes," presented at Digital Libraries 2000, San Antonio TX, 2000.
[4] S. K. Card and J. D. Mackinlay, "The Structure of the Information Visualization Design Space," presented at InfoVis '97, Phoenix AZ, 1997.
[5] E. H. Chi, "A Taxonomy of Visualization Techniques Using the Data State Reference Model," presented at InfoVis 2000, Salt Lake City UT, 2000.
[6] I. Herman, G. Melancon, and M. S. Marshall, "Graph Visualization and Navigation in Information Visualization: A Survey," *IEEE Transactions on Visualization and Computer Graphics*, vol. 6, pp. 24-43, 2000.
[7] D. A. Keim, "Designing Pixel-Oriented Visualization Techniques: Theory and Applications," *IEEE Transactions on Visualization and Computer Graphics*, vol. 6, pp. 59-78, 2000.
[8] B. Shneiderman, "The Eyes Have It: A Task by Data Type Taxonomy for Information Visualization," presented at Visual Languages '96, Boulder CO, 1996.
[9] W. Tobler, "A Computer Model Simulating Urban Growth in the Detroit Region," *Economic Geography*, vol. 46, pp. 234-240, 1970.
[10] W. Tobler, "A Transformational View of Cartography," *The American Cartographer*, vol. 6, pp. 101-106, 1979.

[11] NRC, *Rediscovering Geography: New Relevance for Science and Society*. Washington DC: National Academy Press, 1997.
[12] G. Lakoff and M. Johnson, *Metaphors We Live By*. Chicago IL: The University of Chicago Press, 1980.
[13] G. Lakoff, *Women, Fire, and Dangerous Things: What Categories Reveal About the Human Mind*. Chicago IL: The University of Chicago Press, 1987.
[14] H. Couclelis, "Worlds of Information: The Geographic Metaphor in the Visualization of Complex Information," *Cartography and Geographic Information Systems*, vol. 25, pp. 209-220, 1998.
[15] S. I. Fabrikant and B. P. Buttenfield, "Formalizing Semantic Spaces for Information Access," *Annals of the Association of American Geographers*, vol. 91, pp. 263-280, 2001.
[16] M. Benedikt, "Cyberspace: Some Proposals," in *Cyberspace: First Steps*. Cambridge MA: MIT Press, 1991.
[17] S. I. Fabrikant, "Evaluating the Usability of the Scale Metaphor for Querying Semantic Spaces," in *Spatial Information Theory: Foundations of Geographic Information Science (Lecture Notes in Computer Science 2205)*, D. R. Montello, Ed. Berlin: Springer-Verlag, 2001, pp. 156-171.
[18] A. Skupin, "From Metaphor to Method: Cartographic Perspectives on Information Visualization," presented at InfoVis 2000, Salt Lake City UT, 2000.
[19] A. Skupin, "A Cartographic Approach to Visualizing Conference Abstracts," *IEEE Computer Graphics and Applications*, vol. 22, pp. 50-58, 2002.
[20] G. Salton, *Automated Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer*. Reading MA: Addison-Wesley Publishing Company, 1989.
[21] J. A. Wise, J. J. Thomas, K. Pennock, D. Lantrip, M. Pottier, A. Schur, and V. Crow, "Visualizing the Non-Visual: Spatial Analysis and Interaction with Information from Text Documents," presented at InfoVis '95, Atlanta GA, 1995.
[22] B. Johnson and B. Shneiderman, "Treemaps: A Space-Filling Approach to the Visualization of Hierarchical Information Structures," presented at Visualization '91, San Diego, 1991.
[23] T. Kohonen, *Self-Organizing Maps*. Berlin: Springer-Verlag, 1995.
[24] S. K. Card, J. D. Mackinlay, and B. Shneiderman, "Readings in Information Visualization: Using Vision to Think,". San Francisco: Morgan Kaufmann Publishers, Inc, 1999.
[25] D. M. Mark, A. Skupin, and B. Smith, "Features, Objects, and Other Things: Ontological Distinctions in the Geographic Domain," in *Spatial Information Theory: Foundations of Geographic Information Science (Lecture Notes in Computer Science 2205)*, D. R. Montello, Ed. Berlin: Springer-Verlag, 2001, pp. 488-502.
[26] E. R. Tufte, *Envisioning Information*. Cheshire CT: Graphics Press, 1990.

GeoVIBE: A Visual Interface for Geographic Digital Libraries

Guoray Cai

School of Information Sciences and Technology
Pennsylvania State University
002K Thomas Building, University Park, PA 16802
Tel.: (814) 865-4448
cai@ist.psu.edu

Abstract. Users of spatial digital libraries normally select documents of interest based on both their geographical and topical relevancy in relation to specified queries. This paper proposes an integrated and flexible geographic information retrieval and browsing tool, GeoVIBE, which combines a geographical model and a vector space model to form document representations that are tightly linked. At the user interface, GeoVIBE consists of two types of browsing windows, GeoView and VibeView, which work in coordination for visual navigation in the document space. GeoView imposes a geographical order to the document space based on the idea of document "footprints" that may be cartographically visualized to reveal the spatial structure of the document space. VibeView is a visual presentation of document space defined by multiple reference points, similar to that of VIBE interface by Olson and colleagues (1993). By smooth integration of two browsing strategies, GeoVIBE enables users to search information using geographic clues as well as thematic clues.

1 Introduction

Human activities are, for the most part, geographically constrained, and will benefit from information services that would allow users to specify geographical scope together with topical interests [1]. In the library tradition, textual geographical information tend to be indexed using library catalog methods and retrieved by keyword searching, while cartographic maps and other visually-encoded geographical information are treated as special collections of books. However, a user is likely to be interested in all types of geographical information within certain *spatial* and *thematic* scopes (irrespective of media types). To this criterion, current information retrieval systems are ineffective for retrieving multimedia geographical information in the sense that they do not support user's interactions with the document collection through user-perceived spatial and thematic relevance criteria. Researchers have approached the problem by emphasizing either spatial dimensions (as in geographical information systems) or topical dimensions (as in information retrieval systems) for discriminating relevant documents from irrelevant ones, but the two paradigms of information retrieval have rarely been related and integrated (for exceptions, see [2-4]).

K. Börner and C. Chen (Eds.): Visual Interfaces to Digital Libraries, LNCS 2539, pp. 171–187, 2002.

This paper is an attempt to synthesize the information retrieval and visualization techniques from the disciplines of geographical information science (GIScience) and information retrieval (IR) studies. In particular, we focus on the design of a visual interface, GeoVIBE, which associates a geographical view of the document space (GeoView - using multilayered interactive maps) with a thematic view of the document space (VibeView - visualizing thematic term vectors in relation to multiple reference points) in a tightly coupled fashion. The coupling of the two visualization techniques creates new ways of engaging users in interacting with geographical information in digital libraries.

Geographical information may exist in visual forms (such as cartographic maps, remote sensing images, photographs) or in textual forms (such as field survey descriptions, technical papers, and reports). To facilitate automated retrieval of relevant documents from a collection of documents, a digital library commonly maintains one or more representations of these documents that serve as surrogates. Creating informational surrogates and using them properly has been a long-standing goal that motivates many research efforts in digital libraries, and some of the useful guidelines have been developed by Greene and colleagues [5]. These guidelines have influenced this research on the design of the GeoVIBE interface for geographical information, particularly where the use of multiple surrogates and visual interfaces for navigating within and linking across surrogates are concerned. Our experiences from the development of GeoVIBE also suggest extending the existing guidelines to include the coupling of multiple surrogate types to facilitate visual and interactive filtering and navigation.

Surrogates for geographical information documents should be designed to serve three functions in an information retrieval system. First, they are abstractions and homomorphic reductions of the original documents that require less time to process and analyze but at the same time provide enough semantic cues for users to judge relevance. For example, most textual IR systems use a set of term vectors as surrogates of the original documents, and all analysis and visualizations are performed in the term vector space.

The second function of document surrogates is to mediate the differences (or bridge the gaps) between the *literal space* in which the documents were originally encoded and the *semantic space* with which users perceive relevance. By literal space, I mean a representation space that comprises attributes that are directly extractable from the documents, such as content-bearing terms, visual attributes (color, texture, composition of images), and legends (from maps). Semantic space is comprised of those attribute dimensions that are most salient to the users' judgment of relevancy.

For geographical information, semantic space can be divided into two kinds of subspaces: a *geographical space* and a *thematic space*. These two document subspaces are inherently different. *Geographical space* is a two-dimensional space corresponding to the surface of the Earth and is most commonly represented by a geographical coordinate system, such as latitude and longitude. A document may be concerned with a small area on the earth surface, which can be geometrically represented as points, lines, or areas in a geographical coordinate system, and can be used as the footprint of the document in the geographical space. *Thematic space* is a multidimensional space, where documents are placed according to their thematic concerns. The number of dimensions in thematic space may vary depending on how specific the concepts in the document are categorized into themes. With the definition of literal space and semantic space, document surrogates can be constructed from selected dimensions in literal space, or selected dimensions from semantic space, or some mix of dimensions taken from both spaces. Literal space surrogates are easiest to construct because they can be extracted directly from the original document, and can potentially be automated. However, constructing document surrogates in semantic space means that we have to take a user-centered perspective to identify informational objects that convey meaning, and is generally more difficult to do because it often involves significant content transformations and abstractions from literal space representations, and may require human knowledge and intervention. The challenge for designing surrogates for geographical information documents stems from the dual nature of their literal space (visual and textual) and the dual nature of their semantic space (spatial and thematic). In GeoVIBE, document surrogates are defined separately in spatial subspace (using geographical "footprints") and thematic subspace (using term vectors). The process of deriving these surrogates is defined as the mapping from the literal spaces to the two semantic subspaces.

The third function of document surrogates is to serve as interface objects to assist users' navigation and relevance judgments in a visual and controllable environment. The works by Greene, Marchionini, Plaisant, and Shneiderman [5] on preview and overview surrogates fall into this category. Overview surrogates as interface objects aid the retrieval of relevant documents by presenting to the user a comprehensible "picture" of the document space for maximum comprehension and minimal disorientation. Meanwhile, preview surrogates help users decide whether the original document should be accessed. Interface level surrogates should also give users control of what attributes and objects to focus on in the display. Natural metaphors for interactive browsing and retrieval include coordinated multiple views [6, 7] and zoomable overviews [8]. All these techniques have been applied to the interface design of GeoVIBE.

When people approach digital libraries with their information needs, they are likely to have only fragmented and vague clues of "where" and "what" they are looking for. Accordingly, their searching attempts will also be fragmented, using geographical search for a while and then switching to thematic search, and back and forth. This behavior is consistent with our knowledge from cognitive science and linguistics studies that people appear to have two separate cognitive facilities: one that deals with space, and one that deals with language and other symbols [9][10]. These two separate facilities have separate ways of representing knowledge: spatial and conceptual. Both facilities participate in the understanding of the structure and contents of a geographical information space, but they do so in rather different ways – one generates spatial structure of the information space, and the other generates a conceptual structure of the information space. The two ways of representing information are likely to be interlinked and mutually enhancing.

The existence of two fundamentally different cognitive facilities (spatial and conceptual) and their complementary roles in making sense of geographical information call for dual representations of information space in digital geolibrary interfaces [11]. This serves as another guiding principle for the design of GeoVIBE. By adopting an integrated multi-view approach [12], GeoVIBE allows users to navigate the document space at will, using all the knowledge they possibly have, finding new clues, and eventually narrowing the search space down to relevant documents [6].

The rest of the paper is organized as follows. The next section describes the functionalities of GeoVIBE at its current implementation. This is followed by a discussion of the information representation schemes used to enable GeoVIBE (Section 3). Section 4 relates our work on GeoVIBE with other similar efforts, and spell out the similarities and differences. Finally, further research questions and plans of future development are discussed, and conclusions are drawn.

2 GeoVIBE Interface

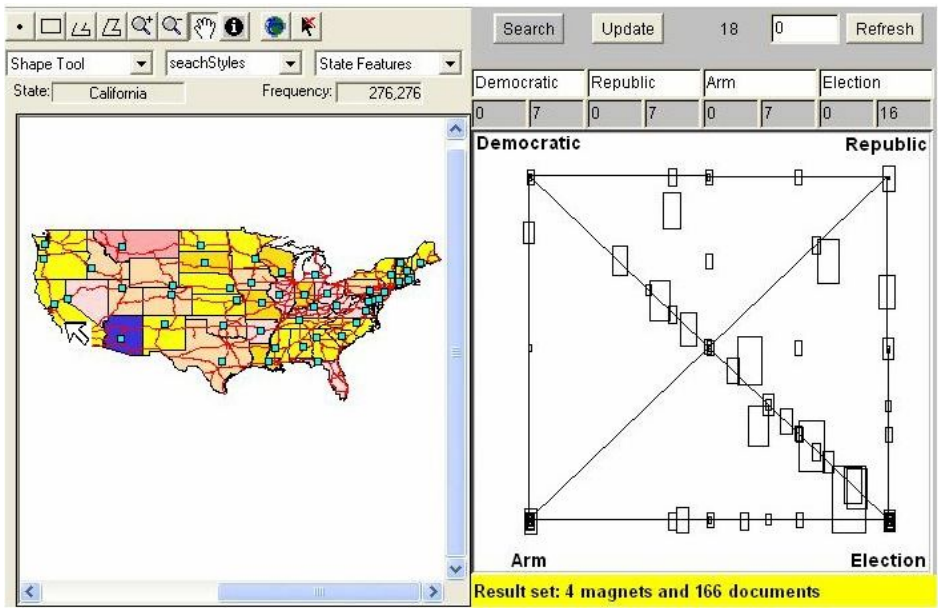
The main feature of GeoVIBE is that it supports visual interaction with the document space utilizing the user's common-sense geographic knowledge as well as thematic concepts. Figure 1 shows a snapshot of the GeoVIBE system. The display consists of two opened views of the document space. The sub-window at the left is the GeoView, which shows a multi-layered map with clickable icons of different shape and sizes linked to document items. The window is the VibeView, where all the documents are presented in a coordinate system defined by Points of Interest (POI) on the display. In the following discussions, we will show how the two views of the document space work together.

2.1 GeoView

GeoView imposes a geographical order on the underlying document space. In this information space, documents are represented by the cartographic characterization of their geographical scopes, and are visualized as a multilayered map that places document "footprints" within a comprehensible context established by a number of "base" map layers. GeoView has a number of functional regions: a map view region, a tool tip region, a spatial query mode region, and a toolbar region (see Figure 1). The toolbar region provides many of the standard desktop GIS tools for manipulating a map view, such as 'zoom in', zoom out', 'identify', and 'pan'. In addition, it includes tools for drawing query shapes (point, rectangle, polyline, and polygon) so that users can formulate geographical and spatial queries.

Information queries to a geographical digital library can be arbitrarily complex, but the most common spatial queries are believed to be of four types: *point*, *region query*, and *buffer zone* [2, 13, 14]. A point query retrieves any documents or geographic datasets that contain, surround or refer to a particular spot on the surface of the earth. A region query defines a polygon in that space and asks for information regarding anything that is contained in, adjacent to, or overlapping the polygonal area so defined. A buffer zone query identifies a neighborhood of a spatial feature as the spatial criteria for selecting documents. GeoView currently supports point-in-polygon queries and region queries for selecting and filtering documents, but more queries types (as discussed by Larson [2]) are planned for future extensions.

Besides spatial query functions, GeoView also provides users with *preview* and *overview* of the document space. An example of *preview maps* can be observed in Figure 2, where the map view clearly shows the geographic "footprint" of a document



GeoView VibeView

Fig 1. A Snapshot of the GeoVIBE Interface.

(titled "Former Congressman Urges….") when a user clicks on it in the document list window (lower right corner). The map is color coded to show different degrees of relevance of the document to any location on the map. To resolve the difficulties of interpreting color codes, GeoView implements a tool-tip, called "Frequency", which shows the associated frequency measure of a document on a location when the user moves the cursor to that location. There are two frequency readings (separated by a comma) in the "Frequency" field: the first reading is the frequency of the currently selected document, and the second reading is the total frequency measure of all the documents. The ratio of these two frequencies gives the reader the sense of the relative strength of the binding of the current document in relation to other documents.

In contrast to preview maps that visualize geographical footprints of a single document, an overview map is the one that shows an overall picture of how a collection of documents is related to different parts of the geographic space. An example of *overview maps* is presented in Figure 1. Again, the degree of association between a document collection and a particular locality is visualized by a color coded frequency map, where darker colors mean higher frequency, in general. Users can also get a reading of the frequency value on any location simply by moving the cursor over it. In the GeoView of Figure 1, the frequency field reads 276 while the cursor is over a location within California. Central to understanding and using previews and overviews in GeoView is the concept of "*document-location frequency*" (DLF). DLF is a measure of what the system believes is the degree of relevance of a document (or a set of document) to a geographical location or region. A DLF of zero indicates no association between the document(s) and the location, and higher DLF indicate higher

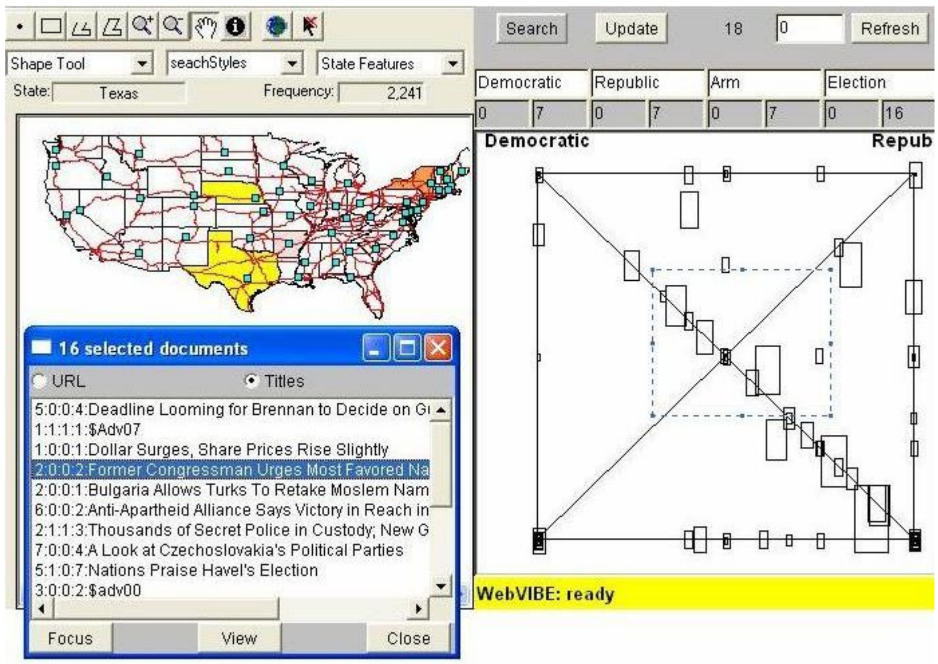


Fig 2. A snapshot of GeoVIBE, showing: (1) overview and selection functions of VibeView, and (2) the preview function of GeoView.

degree of document-location association. DLF can be calculated from the geographical indexing structure of GeoVIBE, represented as *document-shape frequency matrix*, which will be formally introduced later in Section 3.

GeoView enables users to judge relevance in a rich geographical context. It adopts the abstract sense of the world (places and locations) as a user interface metaphor for browsing a large number of documents. When documents are put into the context of the geographical world, the potential spatial interactions between places (diffusion, movement of information through space) and/or the spatial patterns of document distribution provide rich clues for judging relevance of a document in its associated geographical context. A map-based graphical interface tends to be intuitive and comprehensible to anyone who is familiar with maps [2]. Morris [15] suggested that when users are given a choice between menu (text-based) and map-based graphical interfaces to a geographic database, they prefer the maps.

2.2 VibeView

VibeView is similar to the interface of the VIBE system [16]. First, the visual space in the view is a coordinate system that is established by defining a set of *points of interest* (POI) on the display. Each POI consists of a vector of key values describing a subject of interest to the user and a unique icon placed on a position within the VibeView window. Currently, VibeView allows up to four (4) POIs to be specified by a user through entering salient keywords, but theoretically the number of POIs can be more or less and each POI may be chosen to be any of the following: (1) user queries expressed in vector space model; (2) personal interest profiles; and (3) some known documents. After POIs are defined, the placement of a document icon is determined by the relative similarity of the document to the POIs. The position of a document icon gives an indication of the contents of that document. The size, color, and shape of a document icon may vary according to a user-defined function (e.g., the length of the document). Document visualization through VibeView is especially useful for identifying groups of interesting documents in a collection that does not fit a hierarchical structure.

VibeView supports exploration of the content structure of a document space by providing an effective spatial "overview map". This overview can easily be understood because it is based on the familiar "desktop" metaphor as its information organizing principle. The "desktop" metaphor encourages the thinking of a computer screen as an office desk that organizes information spatially as a number of piles of documents, each pile holding similar documents. If a document is related to more than one pile, it is put between those, closest to the most relevant pile. The desktop metaphor allows people to manipulate digital documents just as they manipulate piles of files on their desks. Desktop metaphors have been widely used in developing content awareness information retrieval interfaces, such as the VIBE system [16] and the InfoCrystal [17].

VibeView implements a subset of the functions of the VIBE interface [16]. A common user session of VibeView starts with a user entering 2 to 4 keywords (known as POIs) and then clicking on the "Search" button. This will cause the system to search for documents that contain at least one of those keywords, and generate a new display. The VibeView window on Figure 1 shows a screen layout after a user enters four keywords: "democratic," "republic," "arm," and "election." Clusters of document icons indicate groups of similar contents in relation to the four points-of-interest (POIs). The closer an icon is to a particular POI, the more similar the document is to the meaning of the POI term.

The spatial encoding of document icons within VibeView is designed to exploit human spatial reasoning capabilities in judging document similarities and relevancy. Here the size of a document icon indicates the size of that document, but it can also be used to describe other attributes of the document, such as how recent the document is. Individual POIs can be dragged and re-placed to another location by the user, and the layout of document icons are updated automatically.

Querying documents in VibeView involves only simple user actions such as clicking and dragging. A double-click on a single document icon brings a preview on the document ID number. A double-click on a group of document icons will bring up a document list enumerating all the documents clustered in that display region. A user can drag a rectangle box to select a set of documents into a document-list dialogue window. An example of the document-list dialogue window is shown in Figure 2 (the window on top of GeoView window). This window allows the user to focus attention on individual documents. Clicking the "View" button will bring up a separate window that shows the original content of the highlighted document. If a user decides to focus only on the documents in the list window and remove other documents, (s)he can simply click on the "Focus" button.

2.3 Coupling GeoVIEW and VibeVIEW

Although GeoView and VibeView each provide preview, overview, and query functions for interacting with the document collection, it is the tight coupling of these two complementary views that constitutes the uniqueness of the GeoVIBE environment. This section will show some of the unique features that are available only in GeoVIBE.

2.3.1 Simultaneous Spatial and Thematic Overview

The graphical overviews of the document space presented by GeoView and VibeView can convey immediate understanding of the size and extent of the content coverage, as well as how documents relate to each other. The tight coupling of GeoView and VibeView allows users to interact with the same subset of documents through interpreting and manipulating two overviews, a capability that provides a much higher value to the users than the simple sum of the values of individual overviews. With GeoVIBE, it is possible for the user to make inferences on both the geographical and thematic structures of the document collection with great ease. For example, Figure 1 is a coupled overview map of documents selected by the four keywords: “democratic,” “republic,” “arm,” and “election.”

By reading the two linked overviews in Figure 1, the following conclusion can be quickly drawn: “There are a total of 166 documents that mention one of the themes defined by the four key words, and there seems to be a large portion of the selection that are concerned with the themes of both ‘democratic’ and ‘election’. Geographically, this set of documents covers almost all the states in the US, but it seems that it covers the state of Arizona in more frequency than the coverage on the state of Nevada”. Because both views can be directly manipulated by users through zooming in geographic view and changing search keys in VibeView, GeoVIBE allows users to focus on their complex information problem at hand.

Our informal user studies showed that users actively engage with both overview windows and shift their focus back and forth between the two. Users also reported that having the two overview windows side-by-side facilitates the immediate integration of knowledge learned from individual views.

2.3.2 User Controlled Spatial-Thematic Navigation

The spatialization of document space representations in GeoVIBE attempts to maximize the transfer of users’ skills and knowledge in learning and navigating a natural spatial environment to facilitate their ‘navigation’ in digital information space. Cognitive studies on human acquisition of spatial and geographical knowledge [18-20] showed that people construct their knowledge about an environment by a combination of map reading (survey), navigating, and recording landmarks. Similar learning activities are supported in GeoVIBE through overview, query, and preview functions. When interacting with GeoVIBE, users’ information searching process typically involves interweaving of overview, preview, and query activities - a sequence guided only by users’ focus on learning about the document space. GeoVIBE allows users to switch their mode of interactions freely among the following alternative modes:

(1) Thematic overview; (2) thematic query; (3) thematic preview;
(4) spatial overview; (5) spatial query; and (6) spatial preview.

As an example, Figure 2 demonstrates a small section of a user's session, where a user starts with an overview in VibeView (mode 1), followed by a selection query in VibeView (mode 2), which in turn invokes a document selection list dialogue. From this time, a user can go to the preview mode in GeoView by selecting individual documents in the document list dialogue (mode 6), or see a geographical overview of selected documents in GeoView (mode 4), or cancel the document list dialogue and continue interacting with the overview map in VibeView (mode 1). In summary, GeoVIBE facilitates users' efforts to learn about the information structure of a geographical document collection.

2.3.3 Progressive Spatial and Thematic Focusing

In many cases, a document collection is so large and so widely dispersed in its content that using either geographical constraints (in GeoView) or thematic constraints (in VibeView) in isolation will still leave a large number of documents undecided on their relevancy. As an example, suppose a user wants to find information about "democratic," "republic," "arm," and "election" and in the same time cover some part of the States of New York and Pennsylvania. A thematic query by searching the four terms results in a selection of 166 documents from total of 635 documents in our test collection. If we do a spatial query by the state boundaries of New York and Pennsylvania, 103 documents will be selected. However, both of the selections have some documents that are not relevant to the user, and the independent operation of GeoView and VibeView (without coupling) cannot easily get to the answer set that should be 38 documents. The successful linking of GeoView and VibeView in GeoVIBE enables the user to quickly focus on the relevant documents in two easy steps: one thematic query, followed by a spatial query, all through direct manipulation.

Progressive focusing in GeoVIBE means that a complex query can be subdivided into a number of simple queries that can be any combination of spatial or thematic selections. GeoVIBE maintains a common selection set for GeoView and VibeView at any time, and coordinates the refreshing and updating of the screen on both views when queries in one view make changes on the common selection set that should be reflected on the other view.

To explain further the utility of progressive spatial-thematic focusing of GeoVIBE, Figure 3 provide another scenario where a sequence of simple, direct manipulation queries on GeoView and VibeView results in fast focusing into relevant documents. A transcript of a user session that moves the system status from Figure 1 to Figure 3a and to Figure 3b is given below:

(1) User: Enter keywords "democratic," "republic," "arm," and "election" in VibeView, and then click "Search" button.

(2) GeoVIBE: Show overview of 166 selected documents in GeoView and VibeView. (see *Figure 1*)

(3) User: Drag a query box at the lower-right corner of VibeView, and release the mouse

(4) GeoVIBE: Bring up a separate window (Document List Dialogue, or DLD) showing a list of selected documents.

(5) User: Click on the "Focus" button on the Document List Dialogue

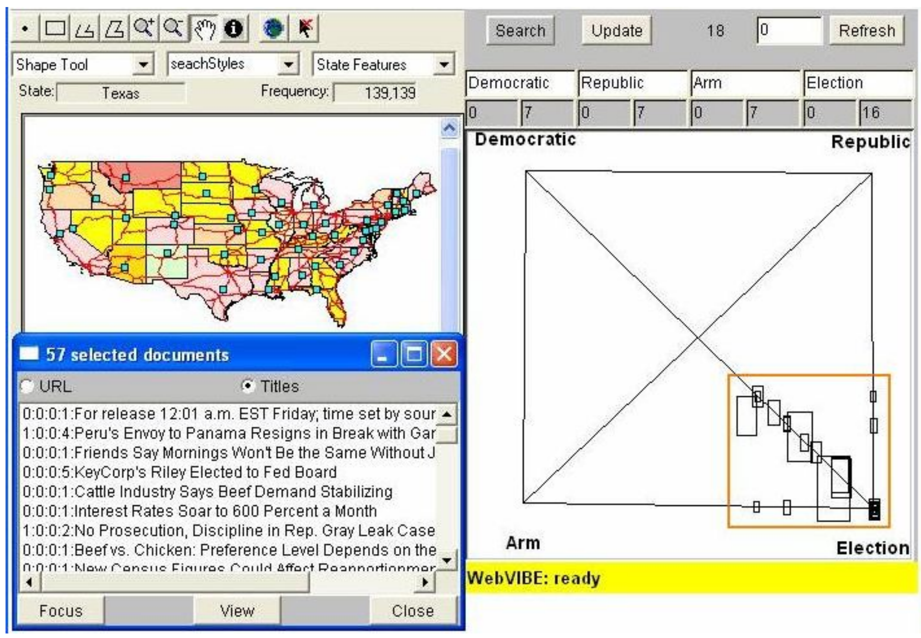


Fig. 3a: The resulting screen after a thematic selection query on Figure 1.(57 documents left in the selection, compared to 166 documents in Figure 1)

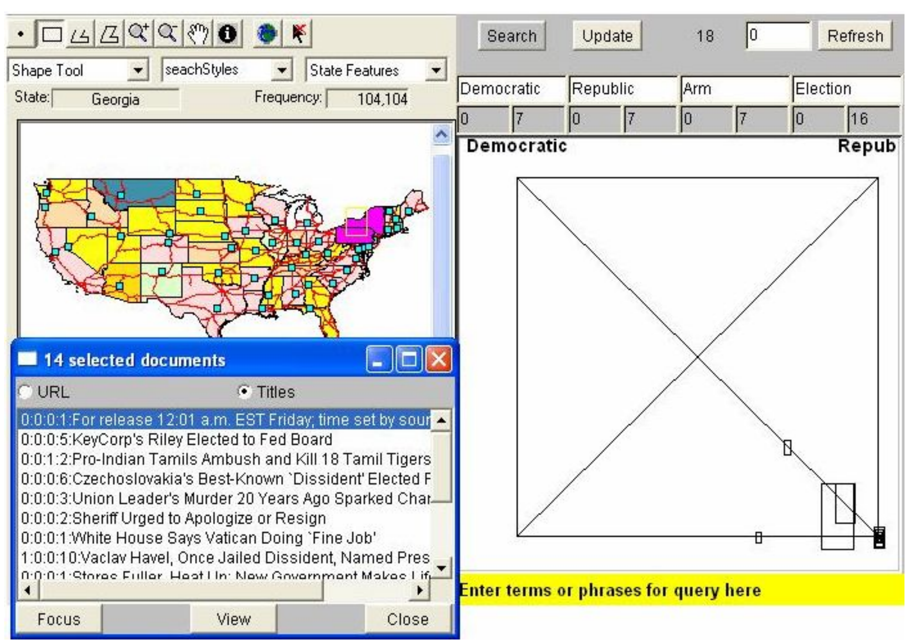


Fig. 3b: The resulting screen after a spatial selection query on Figure 2. (14 documents left in this selection, compared to 57 documents in Figure 3a)

(6) GeoVIBE: Refresh both the GeoView and VibeView windows, with documents outside the previous selection removed (see *Figure 3a*). 57 documents are kept in the views.

(7) User: In GeoView, drag a box to select the state boundaries of New York and Pennsylvania, and then go to VibeView and click "Update" button.

(8) GeoVIBE: Redraw both GeoView and VibeView to keep only those documents whose geographical coverage overlap with the State of New York or the State of Pennsylvania. (see *Figure 3b*) At this time, 14 documents are in the selection, which is the desired set.

The above user session exemplifies the type of complex queries that can be handled well by GeoVIBE. Progressive focusing also supports user's query refinement process as the user knows better about the information structure of a collection.

3 GeoVIBE Architecture

Although this paper focuses on the interface design of GeoVIBE, the picture would not be complete without mentioning the underlying information representation and retrieval models that enable GeoVIBE. The underlying theory of GeoVIBE is the GeoVSM model [11] that reflects an integration of geographical information retrieval model [2] and the vector space model [21]. Within GeoVIBE, a document is represented by two sets of surrogates: thematic surrogates in the form of a term vector for its thematic content, and geographic surrogates in the form of spatial "footprints" (geometric shapes defined in a geographic coordinate system) to represent its geographic scope. These surrogates are mapped into interface objects that form two information spaces: a GeoView that reveals geographical orders of the document space, and a VibeView that reveals thematic orders of the document space.

The operation of GeoVIBE assumes the pre-existence of both geographical surrogates and thematic surrogates for the same document set. Here we will briefly discuss two related issues: (1) how geographical and thematic surrogates are represented, and (2) how they are derived.

3.1 Representation of Geographical Surrogates

For each document D_i, we assume that there exist *m* geometric shapes $\{S_{ij}, j=1,\ldots,n\}$. For each S_{ij} in D_i, we record a frequency measure F_{ij} that reflects the degree of association between D_i and the geographical location represented by S_{ij}. Like the document-term frequency matrix used for characterizing vector space model [21, 22], we define the concept of *document-shape frequency matrix*, DSFM= $\{ f_{ij}, i=1\ldots M, \text{and } j=1\ldots N \}$. The structure of DSFM is illustrated in Figure 4. The rows of the matrix correspond to a set of geometric shapes (which can be a mix of points, lines or polygons), and the columns of the matrix represent individual documents. The value of the (i, j) element, f_{ij}, in the matrix is the frequency measure representing the magnitude of the association between document *i* and shape *j*.

Geometry	**D_1**	**D_2**	**…….**	**D_i**	**……**	**D_{M-1}**	**D_M**
Shape 1	0	3		1		5	0
Shape 2							
….							
Shape j				f_{ij}			
…….							
Shape N-1							
Shape N							

Fig. 4. A model of geographical information space described by *document-shape frequency matrix* (DSFM)

After a DSFM is constructed for a set of geographical documents, all the interface objects and their attributes in GeoView can be derived from the DSFM. In particular, a geographical overview map for a document selection can be generated by selecting those shapes from the DSFM that have non-zero (cumulative) frequencies for the document selection. Likewise, the geographical footprints of a document can be generated by a cartographic presentation of those shapes in the DSFM that has a non-zero frequency for this document. Region selection queries can be performed by spatially intersecting the query shape with the index shapes in DSFM and calculating a new selection of documents.

3.2 Representation of Thematic Surrogates

Thematic surrogates in GeoVIBE are represented as a term-document frequency matrix (TDFM) similar to the representation in a vector space model [21]. This model assumes that each document can be approximately described by a vector of (content-baring) keywords that are generally considered *pairwise orthogonal*. Under this model, an information retrieval system stores a representation of a document collection using a *document-by-term* matrix, where the element at position (*i*, *j*) corresponds to the frequency of occurrence of j^{th} term in the i^{th} document. By representing all the objects (terms, documents, queries, concepts, etc) as vectors in a vector space, the model can compute a similarity coefficient that measures the similarity between a document and a query or another document. Those documents whose contents are most similar to the query are considered the most relevant. For more detail on vector space model and associated representations, see [21], and more recently [22] and [23].

3.3 Deriving Geographical and Thematic Surrogates

The general process of deriving the dual representation of documents as geographical and thematic surrogates involves a number of steps:

(1) analyze and extract the literal features of the source documents (LF);
(2) separate literal features into two groups: one bearing geographical content (LF_{geo}), and the other bearing thematic (non-geographical) content (LF_{th});
(3) derive geographical surrogates DSFM from LF_{geo}, and thematic surrogates TDFM from LF_{th}.

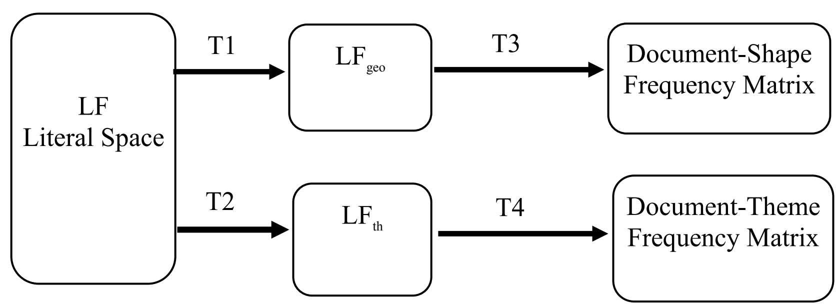


Fig. 5. Deriving geographical and thematic surrogates

The constituents of literal features can be different depending on whether textual encoded information or visually encoded geographical information is involved. For textual documents, literal features are extractions of content-bearing terms from the original documents. For cartographical maps and georeferenced images, literal features may include such diverse forms as geographical coordinates representing location and spatial extent, extraction of terms from map legends, expert interpretations of perceived contents, and other metadata. It is still quite problematic for automated thematic indexing of visual materials [24] and automated indexing of the geographical content in textual documents by geometric shapes [25]. The general topics of deriving semantic representation from literal feature representation are too broad to be covered here. In a very abstract level, we can thinks of the whole problem as defining four transformations T1, T2, T3, T4, shown Figure 5.

4. Relations to Other Works

Research on visual interfaces for digital geolibraries has explored the use of spatialization for visualizing both geographical structures and the thematic structures of document contents, but rarely put them in tight coordination as GeoVIBE does. Exceptions include the Informedia Project [26] and the Dynamic Homefinder application [27]. Informedia uses a combination of timelines, keywords, and geographical surrogates for visualizing digital video libraries. Dynamic Homefinder combined map-like interface with slider bars to enable dynamic queries similar to the progressive focusing in GeoVIBE.

The Alexandria Digital Library (ADL) [28] project made extensive use of hypermap concepts [13] to integrate visualizing geographical contents with browsing maps and image objects. More recently, the Alexandria Digital Earth Prototype (ADEP) explored the use of digital earth metaphor for the visualization of thematic space [29], but it is not clear at this time if the two kinds of document representations are coupled at the interface level.

InfoCrystal [17] provides a spatially compact interface for complex Boolean or vector space queries. An InfoCrystal interface has two sets of icons: criteria icons represent user-defined criteria are placed at the surrounding edges with even space; interior icons representing unique types of queries are placed according to proximity and ranking principles. Queries can be specified by selecting individual or a group of interior icons in a graphical manner.

Microsoft TerraServer [30] is a multimedia geographical data warehouse that serves aerial, satellite, and topographic imagery. It uses multi-layered index maps that categorize imagery into "themes" by data source, projection system, and image "style." It indexes source images and photos by scenes and tiles. A user may query images in three styles: coverage map, place query, and coordinate query.

GeoWorlds [Neches, 2001 #873] was a digital library project developed by the University of Southern California Information Science Institute. It shares the same vision principles with GeoVIBE in that both recognized the needs to integrate geography-based and document-based information retrieval methods for searching large geospatial information collections. GeoWords also allows users to exploit both geographical and thematic clues in narrowing down to relevant documents. GeoVIBE is different from GeoWorlds in two major aspects. First, GeoVIBE does not force documents to follow strict hierarchical categories like GeoWorlds does. Second, GeoVIBE employs two similarity retrieval engines, one for geographical relevance and the other for thematic relevance. In comparison, GeoWorlds relies on textual-based search for both geographical relevance and thematic relevance, and its map view is merely an interface for users to issue queries.

5 Conclusions and Future Work

GeoVIBE represents a new class of visual interfaces for digital geospatial libraries. The most important contribution of GeoVIBE is the tight coupling of two visual interface representations, a GeoView and a VibeView, which work in coordination to provide overview, preview, and query functions through direct manipulation. The design of GeoVIBE represents a systematic application of the principles on the choice of document surrogates, which led to the use of two complementary surrogate sets. Informal user studies are underway, and have partially demonstrated the feasibility of the design idea behind GeoVIBE.

Many questions about the usability of GeoVIBE will not be clear before more formal user experiments are conducted. People may have difficulties in interpreting maps and map-like interfaces, and having two windows side-by-side may be too overwhelming for some. Future work is also required to address the issue of reliably and efficiently generating geographical and thematic surrogates from the literal extraction of the source documents.

Acknowledgement

The author is thankful for many constructive comments from anonymous reviewers. The design ideas and implementation of GeoVIBE have benefited from interaction with Emily Morse and other members of the Information Retrieval Research Group at the University of Pittsburgh. Special thanks to Junyan Luo, who helped Java programming on the GeoVIBE implementation, and to Frank Ritter who helped improve the quality of this paper.

References

1 National-Research-Council, "Distributed Geolibraries - spatial information resources," Mapping Science Committee, National Research Council, Washington,DC, Summary of a Workshop (1999).

2 Larson, R. R., "Geographic Information Retrieval and Spatial Browsing," in *GIS and Libraries: Patrons, Maps and Spatial Information*, L. Smith and M. Gluck, Eds. Urbana-Champaign: University of Illinois(1996), pp. 81-124.

3 Buttenfield, B. P. and Kumler, M. P., "Tools for browsing environmental data: The Alexandria Digital Library interface," Proceedings of the Third International Conference on Integrating Geographic Information Systems and Environmental Modeling., Santa Fe (1996).

4 Fabrikant, S. I. and Buttenfield, B. P., "Formalizing Semantic Spaces for Information Access," *Annals of the Association of American Geographers*, vol. 91 (2001), pp. 263-280.

5 Greene, S., Marchionini, G., Plaisant, C., and Shneiderman, B., "Previews and overviews in digital libraries: Designing surrogates to support visual information seeking," *Journal of the American Society for Information Science*, vol. 51 (2000), pp. 380-393.

6 Masui, T., Minakuchi, M., Borden IV, G. R., and Kashiwagi, K., "Multiple-view Approach for Smooth Information Retrieval," Proceedings of the 8th Annual Symposium on User Interface Software and Technology, Pittsburgh (1995).

7 North, C. and Shneiderman, B., "A Taxonomy of Multiple-Window Coordination," University of Maryland, Computer Science Dept CS-TR-3854, (1997).

8 Fabrikant, S. I., "Evaluating the Usability of the Scale Metaphor for Querying Semantic Spaces," in *Spatial Information Theory: Foundations of Geographic Information Science - Proceedings of COSIT 2001 International Conference on Spatial Information Theory, Morro Bay, CA, USA, September 19-23*, *LNCS 2205*, D. R. Montello, Ed.: Springer-Verlag(2001), pp. 156-172.

9 Allen, B., "Information space representation in interactive systems: relationship to spatial abilities," Proceedings of the Third ACM Conference on Digital Libraries, Pittsburgh (1998).

10 Jackendoff, R., *Languages of the Mind*. Cambridge, MA: MIT Press (1992).

11 Cai, G., "GeoVSM: An Integrated Retrieval Model For Geographical Information," in *LNCS 2478: Geographic Information Science-Second International Conference GIScience*, M. Egenhofer and D. Mark, Eds. Boulders, Corolado: Springer(2002).

12 Carrico, L. and Guimaraes, N., "Integrated multi-views," *Journal of Visual Languages and Computing*, vol. 9 (1998), pp. 287-297.

13 Laurini, R. and Milleret-Raffort, F., "Principles of geometric hypermaps," Proceedings of the 4th International Symposium on Spatial Data Handling, Zurich, Switzerland (1990).

14 Laurini, R. and Thompson, D., *Fundamentals of Spatial Information Systems*. London: Academic Press (1992).

15 Morris, B., "CARTO-NET: Graphic retrieval and management in an automated map library," *Special Libraries Association, Geography and Map Division Bulletin*, vol. 152 (1988), pp. 19-35.

16 Olson, K. A., Korfhage, R. R., Sochats, K. M., Spring, M. B., and Williams, J. G., "Visualization of a Document Collection: the VIBE System," *Information Processing and Management*, vol. 29 (1993), pp. 69-81.

17 Spoerri, A., "InfoCrystal: a visual tool for information retrieval," Proceedings of the IEEE conference on visualization, Cambridge MA (1993).

18 Landau, B. and Jackendoff, R., ""What" and "where" in spatial language and spatial cognition," *Behavioral and Brain Sciences*, vol. 16 (1993), pp. 217-265.

19 Strohecker, C., "Cognitive Zoom: From Object to Path and Back Again," in *Spatial Cognition II, LNAI 1849*, C. Freksa, W. Brauer, C. Habel, and K. F. Wender, Eds. Berlin Heidelberg: Springer-Verlag(2000), pp. 1-15.

20 Golledge, R. G., "The nature of geographical knowledge," *Annal of Association of American Geographers*, vol. 92 (2002), pp. 1-14.

21 Salton, G., Yang, C., and Wong, A., "A vector space model for automatic indexing," *Communications of the ACM*, vol. 18 (1975), pp. 613-620.

22 Grossman, D. A. and Frieder, O., *Information Retrieval : algorithms and heuristics*: Kluwer Academic Publishers (1998).

23 Korfhage, R. R., *Information Storage and Retrieval*: John Wiley & Sons, Inc. (1997).

24 Jörgensen, C., "Access to Pictorial Material: A Review of Current Research and Future Prospects," *Computers and the Humanities*, vol. 33 (1999), pp. 293-318.

25 Woodruff, A. G. and Plaunt, C., "GIPSY: Geo-referenced Information Processing System," *Journal of the American Society for Information Science*, vol. 45 (1994), pp. 645-655.

26 Christel, M. G., "Accessing News Video Libraries through Dynamic Information Extraction, Summarization, and Visualization," The First ACM-IEEE Joint Conference on Digital Libraries, workshop on visual interfaces, Ronake, VA (2001).

27 Williamson, C. and Shneiderman, B., "The dynamic HomeFinder: evaluating dynamic queries in a real-estate information exploration system," Proceedings of the Fifteenth Annual International ACM SIGIR conference on Research and development in information retrieval, Copenhagen Denmark (1992).

28 Smith, T. R., "A digital library for geographically referenced materials," *IEEE Computer*, vol. 29 (1996), pp. 54-60.

29 Ancona, D. and Smith, T., "Visual Explorations for the Alexandria Digital Earth Prototype," The Second ACM-IEEE Joint Conference on Digital Libraries, workshop on visual interfaces, Oregon, WA (2002).

30 Barclay, T., Gray, J., and Slutz, D., "Microsoft TerraServer: a spatial data warehouse," Proceedings of the 2000 ACM SIGMOD on Management of data, Dallas, TX (2000).

31 Neches, R., Yao, K.-T., Ko, I.-Y., Bugacov, A., Kumar, V., and Eleish, R., "GeoWorlds: Integrating GIS and Digital Libraries for Situation Understanding and Management," The New Review of Hypermedia and Multimedia, vol. 7 (2001).

Interactive Information Visualization in the Digital Flora of Texas

Teong Joo Ong[1], John J. Leggett[1], Hugh D. Wilson[2], Stephan L. Hatch[3], and Monique D. Reed[2]

[1]Center for the Study of Digital Libraries, Texas A&M University, USA
{mong, leggett}@csdl.tamu.edu
[2]Department of Biology, Texas A&M University, USA
{monique, wilson}@mail.bio.tamu.edu
[3]S. M. Tracy Herbarium, Texas A&M University, USA
s-hatch@tamu.edu

Abstract. The Digital Flora of Texas project is a collaborative research effort that relies on botanical research centers in Texas to create and maintain digital library collections on the flora of Texas. The project focuses on developing web-accessible collections for botanists and botanically-interested non-specialists. The Herbarium Specimen Browser (HSB) is the main portal to the collections of the Digital Flora of Texas. The HSB allows examination of the collections through several types of interactive information visualizations: a hierarchical taxonomic browser, specimen distribution and density maps, and stackable bar graphs of temporal specimen data. The interactive information visualization tools of the HSB are designed to help users extract as much information from the data as possible while minimizing the associated learning cost and complexity of these tools.

1 Introduction

The Digital Flora of Texas project is a collaborative research effort that relies on botanical research centers in Texas to create and maintain digital library collections on the flora of Texas. Currently, three-fourths of the 680 plant species listed as in danger of becoming extinct in the next ten years occur in Texas, California, Florida, Hawaii, and Puerto Rico. Texas is the only one of this group that lacks a comprehensive geographical knowledge of its plant resources and the most recently written comprehensive flora for the state is obsolete [1]. A single, consolidated source of high quality botanical data is needed and this need will increase as the population density and economy of Texas grow. The digital flora produced by this project will be a valuable resource for students, educators, scientists, government agencies, and, indeed, anyone with an interest in the botanical heritage of Texas [2].

The project focuses on developing web-accessible collections for botanists and botanically-interested non-specialists. It includes refinement and expansion of specimen label data capture systems and deployment of these to new sites, development of data translation programs [3], refinement and elaboration of web-based data access systems, development and curation of distributed plant image libraries, and exploration of protocols employed to produce digital library collections

K. Börner and C. Chen (Eds.): Visual Interfaces to Digital Libraries, LNCS 2539, pp. 188–198, 2002.

from distributed participants who contribute taxonomies, images, specimen data, keys, and descriptions [4].

2 The Herbarium Specimen Browser

The Herbarium Specimen Browser (HSB) is the main portal to the collections of the Digital Flora of Texas [5]. It currently provides access to approximately 300,000 specimen records contributed by thirteen collaborating herbaria. In addition to detailed specimen listings and full-text querying on all fields [6], the HSB allows examination of the collections through several types of interactive information visualizations: a hierarchical browser giving a high-level taxonomic overview, maps showing the distributions of specimens and species, and stackable bar graph displays showing temporal aspects of the specimens such as collection date or flowering periods. The HSB also has the capability of on-the-fly filtering of all displayed information by county, herbarium, and other criteria, allowing users to answer questions by visualizing subsets of the information [7, 8]. The interactive information visualization tools of the HSB are designed to help users extract as much information from the data as possible while minimizing the associated learning cost and complexity of these tools.

2.1 Browsing the Collections

The HSB in its initial *Main Display* mode (Figure 1) shows:

- The specimen selection criteria currently in force (specified by the filtering controls in the *control frame* on the left)
- A line indicating the number of specimens which meet those criteria, and the numbers of species, genera, and families represented by those specimens
- A list of all families represented by specimens meeting the current criteria, the number of genera, species, and specimens they contain, and several links.

The family names in this display are link anchor sites. Selecting one causes an "expansion" in taxonomic context that shows a listing of the genera (represented by specimens) contained in that family. The user can then select one of the genera to see a terse list of species in the genus or a list of the full specimen data sorted by *Herbarium* and accession number.

The lists remain expanded until the user contracts them with a subsequent click on the genus or family name. The map anchor dynamically displays the specimen distribution map of a particular family, genus or species depending on the map level that was chosen in the control-frame. The mapping system is capable of providing, at a quick glance, the density and distribution of a particular family, genus or species organized in terms of specimens and species in the counties of Texas, counties of Texas and adjacent states (shown in Figure 2), and states/provinces of North America.

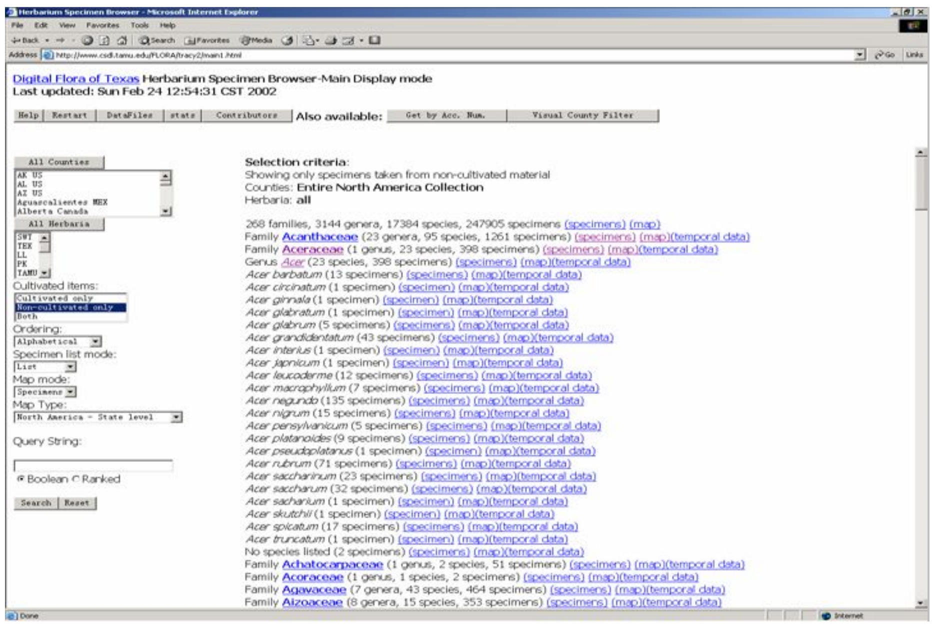


Fig. 1. Browsing in taxonomic context on the family Aceraceae and genus Acer

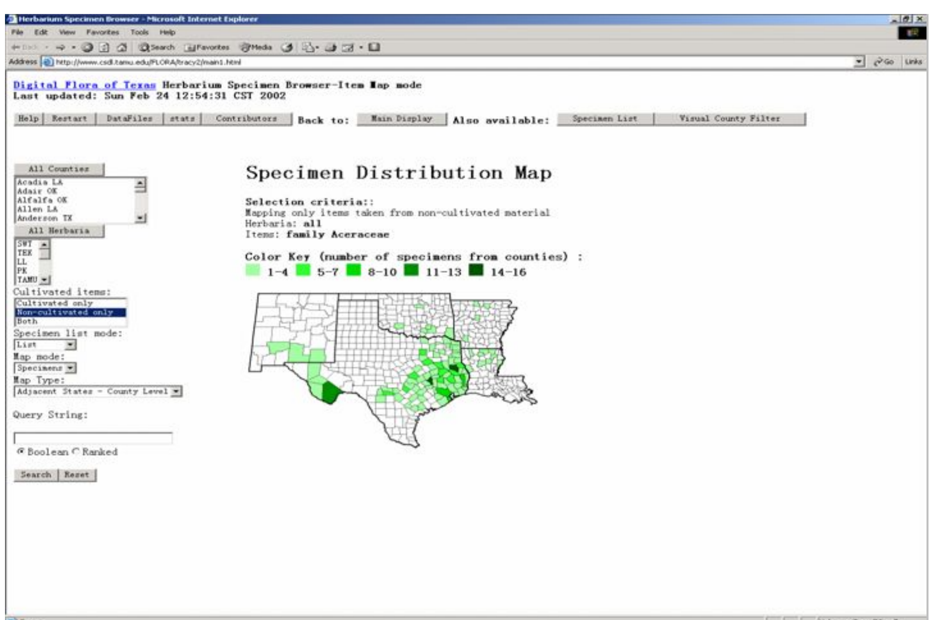


Fig. 2. Specimen distribution map for Texas and adjacent states by counties

Clicking on any colored region of the map will produce a specimen listing that consists of the related specimen data from a particular region [9]. This feature allows the user to quickly find density, location, collector and botanically related information in a few clicks. These clickable distribution maps can be easily generated using the system discussed in [10]. A similar system has been developed by the Jepson Flora Project for the University and Jepson Herbaria Collections at the University of California at Berkeley [11].

2.2 Visual County Filter

In the example shown in Figure 3, the user is interested in answering questions that concern only the counties that are adjacent to his/her county (Brazos county in the figure). For example, "I know plant X is in my county. Does it also occur in surrounding counties?" The user may simply click the surrounding counties to filter (subset) the collections. Clicking on the 'Main Display' button will take the user to the hierarchical browsing display of the plants that have been collected in the adjacent counties. This tool provides a convenient way for users who are not familiar with the names of the 254 counties of Texas to locate botanical information simply by highlighting counties on the map. An option at the bottom allows the user to select a set of regions to answer questions such as "Which plants inhabit the Southwestern region of Texas?"

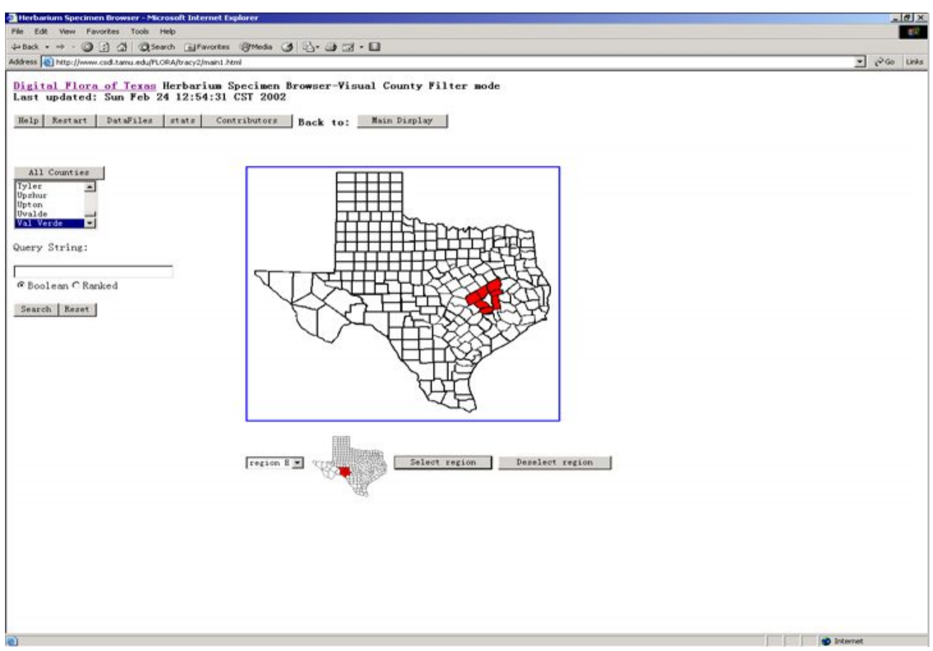


Fig. 3. Using the Visual County Filter to select a region from the Texas counties

2.3 Querying the Collections

Browsing a collection can sometimes be time consuming because users have to navigate to a particular location within the specimen browser to get to the families, genera or species of interest. A query interface to the collection provides the user with the means to locate data more directly than browsing through the alphabetical listing of families in the *main display frame*. Since all text data is full-text indexed, the user can quickly find common names, locations, collectors, accession numbers, dates, etc. Figure 4 illustrates the listing of specimens sorted by Herbarium, Accession number, Family, Genus and Species fields from a simple query string of 'Aceraceae' using the query control on the *control frame*.

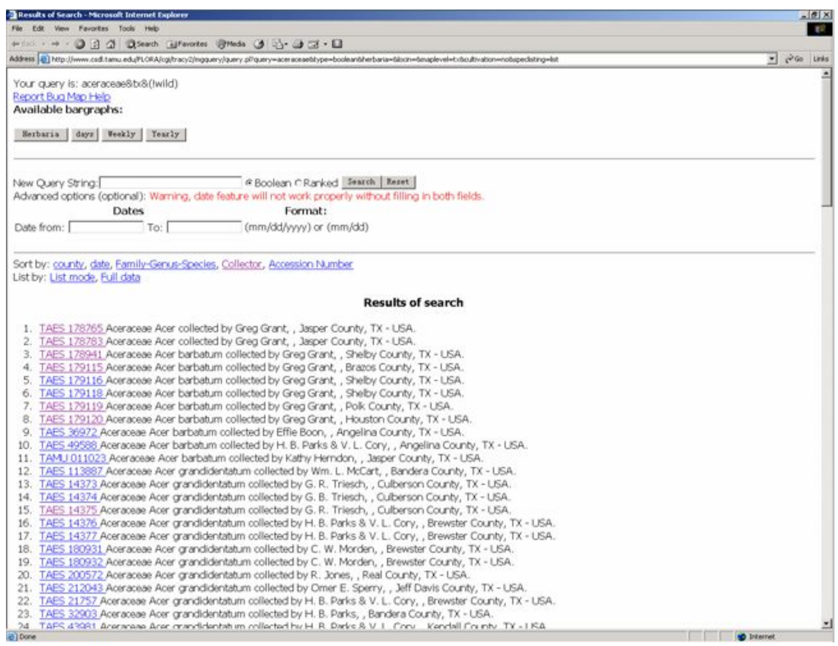


Fig. 4. Specimen listing generated with the query string 'Aceraceae'

2.4 Visualizing Temporal Specimen Data

Temporal information can be difficult to formulate in query strings since it involves creating the relevant query, filtering the result set and searching through the results based on the relevant fields. During our regular weekly meeting session, tools were requested that could summarize the temporal dimension of the specimen collections from a high level and also help in making quick comparisons among herbaria specimen data. Using these tools, one could discover the earliest specimen records of the herbaria, the collectors who collected them, and when and where they were collected. This request resulted in an extension to the HSB in which interactive stackable bar graph visualizations were designed for temporal specimen data.

In conjunction with the specimen distribution maps, temporal data allow users to derive answers to many questions that are otherwise hidden in plain numerical data, such as: “Has a species changed its range over the years?”; “When was a species first collected in a state, county or country?”; “During the occurrence of a particular event, such as World War I, which collectors were still actively collecting specimens?”; “Where were they collecting?”; “How many species were collected?”; ”What were the species collected?”; and, “How active was a herbarium in specimen collecting activities during a certain period in time?”

2.4.1 Visualizing Collections by Year

Figure 5 presents the bar graph generated from clicking the ‘Yearly’ button in Figure 4. The graph summarizes the number of specimens collected by different herbaria, represented with different colors, over a certain time span. The bars on the graph are clickable to allow users a quick glance at the specimen records that were collected by a particular herbarium for a particular period of time. Using the ‘Yearly’ bar graph in conjunction with the specimen listings, visual county filter, and specimen distribution maps of the HSB, a user can quickly discover: the earliest date in the year a particular species has been collected, the latest date a species has been collected, the oldest specimen of a particular species, the years that a certain herbarium was actively involved in the collection of certain species, and whether a certain species is relatively more or less common now than in the past. For example, to answer “What was the earliest collected specimen in the family Aceraceae, who collected it, and which herbarium contains the record?” The user would simply click on the leftmost bar in Figure 5.

Other than providing hints on temporally related data, the ‘Yearly’ bar graph can offer insights on trend information and the impacts of certain historical events on a botanical taxon that can help answer questions such as “During World War I, did the collection of taxon X drop in the Western Region of Texas?” By using the Visual County Filter to select the counties located in the Western Region of Texas, we can obtain the corresponding specimen subset in the hierarchical display of the specimen browser after clicking on the ‘Main Display’ button (as in Figure 1). From the ‘Main Display’ screen, we can locate the family we are interested in to obtain the specimen distribution map as well as query (by clicking on the ‘temporal data link’ beside the ‘map’ link) to find out the number of specimens that are collected from that region. The yearly bar graph generated from the query allows the user to make a quick comparison on the number of specimens collected over the years and the distribution map provides information on the relative density and number of specimens collected in the selected counties.

2.4.2 Visualizing Herbarium Collections

We can further narrow our search for answers that correspond to an individual herbarium by clicking on the ‘Herbaria’ button to summarize the specimen collections of herbaria neatly, thus freeing the user from wasting time counting the specimen data in the result set (Figure 6). Furthermore, by looking at the graph summary and the peaks of the bar graphs, users can find answers to questions such as “Which herbarium collected the most specimens from the Family and Genus ‘Aceraceae *Acer*’?”, “Which are the most frequently collected species from a given family” and “Does a particular herbarium contain any specimens of this family?”

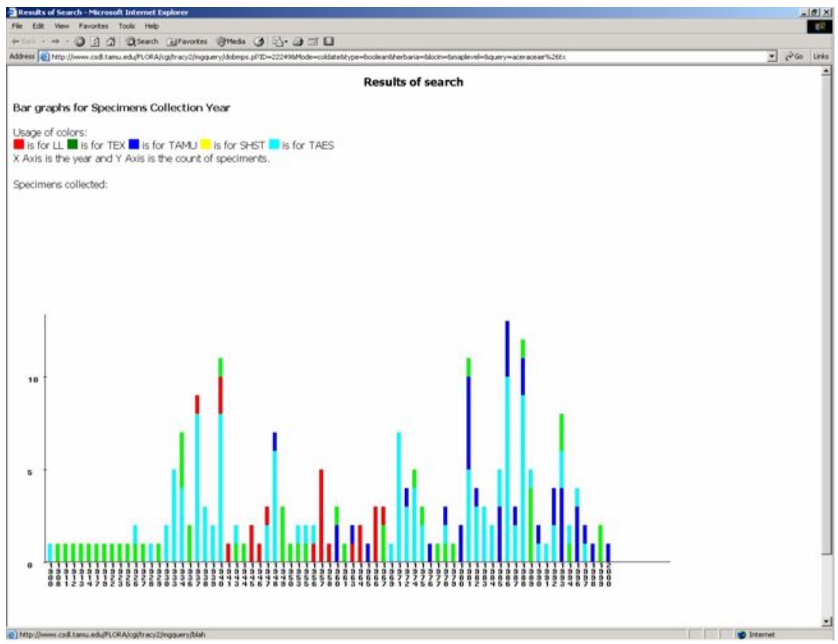


Fig. 5. Years in which specimens for the Aceraceae family were collected. Stacked column bars indicate herbaria specimen counts

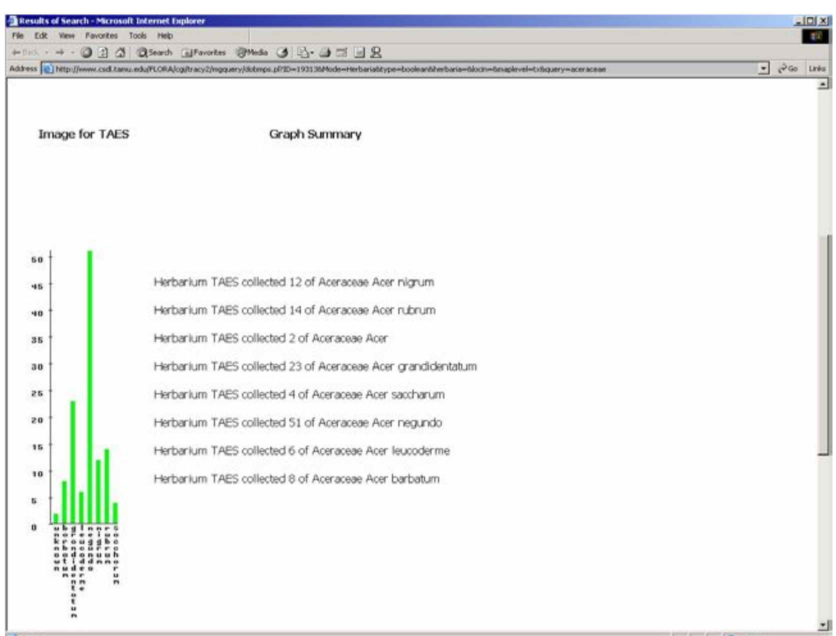


Fig. 6. One of the bar graphs that summarizes the specimen counts of a herbarium from the result set

2.4.3 Visualizing Collections by Day

Figure 7 presents the bar graph generated from clicking the ‘Days’ button in Figure 4. It is arranged according to the day and month fields while taking leap years into account. With different colors used to represent different herbaria across the country, a user can quickly find answers to questions such as: “When are the flowering and fruiting times of a particular species?”; “Do the flowering and fruiting times of that species vary by location?”; “When is the best time in the year for certain species to be collected?”; “When and where will a particular species (usually) be collected by a particular herbarium?”; and, “Does data from an institution support what the user has seen either in the field or in another institution's holdings?” If users know the name of a plant that currently faces danger of extinction, they can see its distribution throughout the years using the bar graphs and the mapping system of the HSB in different granularities. Furthermore, if all specimen records contribute to the graphs, we can answer questions such as “When are herbaria most actively engaged in collection activities?”

Fig. 7. The ‘Days’ bar graph of the Aceraceae family organized according to day and month fields

For example, if users would like to find information on the flowering period and collection locations of a particular plant family, they can query on the family name and visualize the results in a day graph. Assuming that the flowering period of the plant is in spring, we should see a concentration of peaks from March to June representing the number of specimens collected, as well as the herbaria that collected them. If we would like to find out about the specimen records collected by the Herbarium TAES, we can click on the bars with the corresponding color to obtain detailed information about each of these records. Using this graph in conjunction with other information visualization tools of the HSB, we can derive hidden information otherwise obscured by numerical data alone.

2.4.4 Visualizing Collections by Week

The bar graph depicted in Figure 8 provides a quick summary of the specimen records organized in weeks of the year according to their day and month fields. It allows users to quickly determine the approximate time of the year in which a specimen is collected by a certain herbarium. This graph provides users with an overall view on all of the query data, thus preventing inaccuracy from potential noise or outliers in the data. Furthermore, proper organization of the data into weeks eliminates the need to manually inspect each of the peaks in the ‘Daily’ bar graph or the specimen listings to determine the approximate time of specimen collection or flowering period throughout the year. With the bars organized according to the four weeks of a month, users can easily understand and locate the information according to the week numbers that are placed below the graph.

For example, if a user learns from Figure 7 that ‘Aceraceae’ flowers in spring and summer and the TAES and TAMU herbaria have been actively collecting specimens of this family, he/she can use the ‘Weekly’ bar graph to summarize these data characteristics and display them in a minimal number of bars. Using the ‘Weekly’ bar graph, the user can gain a bigger view of the picture with the data being averaged out according to the week number in months. Let’s assume that we are interested in the first few bars in the Spring season ‘Days’ bar graph that represents specimen collections of TAMU. We can obtain a listing of all of these specimens by clicking on the red bar in the second column of the ‘Weekly’ bar graph. One other trend we can observe from the ‘Weekly’ bar graph is that TAMU collection of this family is mainly active only during spring (less in summer). The ‘Days’ would have misguided us because there are significant number of bars representing TAMU specimen collection in summer season. Thus, the ‘Weekly’ bar graph is a helpful addition that allows users to inspect the specimen data from many different perspectives and granularities.

3 Summary

The capability to display otherwise obscured botanical information, such as flowering and collection periods, specimen collections of the various herbaria, and simple numerical comparisons of specimens held by various herbaria, removes the burden of query formulation and data processing from the shoulders of our users and allows them to answer important questions through interactive information visualization.

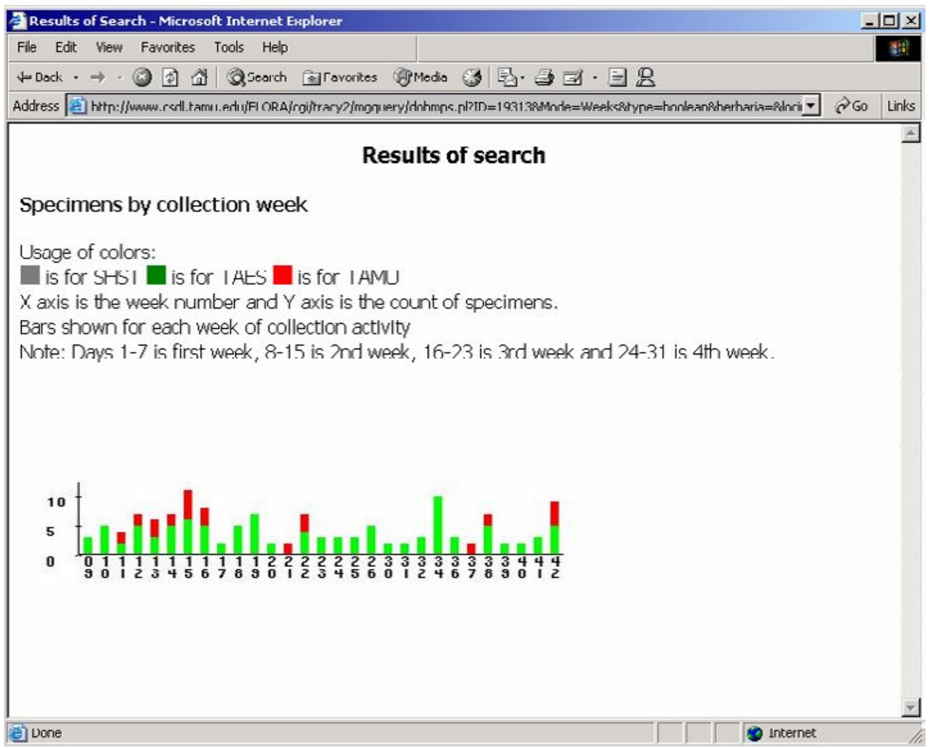


Fig. 8. Bar graph depicting the weeks of the year that specimens for the Aceraceae family are collected

4 Future Work

Currently the software that generates the bar graphs is only capable of representing information on a daily, weekly or yearly basis. In the future, we would like to expand functionality by incorporating the dynamic regional distribution mapping capability of the Herbarium Specimen Browser into the querying mechanism so that users can click on a bar graph to directly obtain a specimen regional distribution map in addition to a detail listing of the collected specimens. Lastly, data mining techniques could be employed to take the step of information extraction and visualization further. For example, mining plant distributions and suggesting plant collecting activities by the nearest or most closely focused herbarium.

Acknowledgments

This research was supported in part by the Texas Advanced Research Program under Grant Number 010366-0041C-1999 and the Telecommunications and Informatics Task Force, Texas A&M University.

References

1. D. S. Correll and M. C. Johnston. Manual of the Vascular Plants of Texas. 1970, Texas Research Foundation, Renner, Texas.
2. Digital Biodiversity: The Flora of Texas Project http://www.csdl.tamu.edu/FLORA/ftc/arp99/arp99.htm
3. Flora of Texas Consortium - Data Specification. http://www.csdl.tamu.edu/FLORA/ftc/ftcffld4.htm
4. Gaurav Maini, John J. Leggett, Teongjoo Ong, Hugh D. Wilson, Monique D. Reed, Stephan L. Hatch, and John E. Dawson, "Enhancing Scientific Practice and Education Through Collaborative Digital Libraries," *Proceedings of the EdMedia Conference*, 2002, pp. 1206-1210.
5. The Digital Flora of Texas homepage http://www.texasflora.org
6. Ian H. Witten, Alistair Moffat and Timothy C.Bell. Managing Gigabytes, Compressing and Indexing Documents and Images. 2nd. Edition, 1999, Morgan Kaufmann Publishing Company.
7. C. Ahlberg and B. Shneiderman, "Visual Information Seeking: Tight Coupling of Dynamic Query Filters with Starfield Displays," *Proceedings of CHI'94, ACM Conference on Human Factors in Computing Systems,* 1994, pp. 313-317.
8. Erich R. Schneider, John J. Leggett, Richard K. Furuta, Hugh D. Wilson and Stephan L. Hatch, "Herbarium specimen browser: a tool for accessing botanical specimen collections," *Proceedings of the ACM Digital Libraries Conference,* 1998, pp. 227-234.
9. C. Runciman and H. Thimbleby, "Equal Opportunity Interactive Systems," *International Journal of Man-Machine Studies,* 25 (4), 1986, pp. 439-451.
10. Teong Joo Ong, John J. Leggett, Hugh D. Wilson, Stephen L. Hatch, Monique D. Reed, Gaurav Maini, Erich R. Schneider, and Peter J. Nürnberg, "A Web-Based Run-Length Encoded Map Generating System," *Proceedings of the EdMedia Conference*, 2002, pp. 1506-1507.
11. Museum Informatics Project, University of California at Berkeley http://www.mip.berkeley.edu

Visual Explorations for the Alexandria Digital Earth Prototype

Dan Ancona, Mike Freeston, Terry Smith[1], and Sara Fabrikant[2]

[1]Alexandria Digital Earth Prototype
{da, freeston}@alexandria.ucsb.edu, tsmith@cs.ucsb.edu
[2] Department of Geography, UCSB
sara@geog.ucsb.edu

Abstract. The Alexandria Digital Earth Prototype project addresses issues of access, browsing, delivery and understanding of georeferenced library items. Two visualization approaches that address these issues in different ways are under development: interfaces between some existing digital earth systems and the digital library, and spatial but nongeoreferenced information spaces. Both the abstract spatial and georeferenced projects are being evaluated for their educational potential in classroom and lectures, in labs and as tools for students to use on their own.

1 Introduction

> ""...To see through the topography and see how the rocks lie in three dimensions beneath the topography is the hardest thing to get across to a student." After a mile of silence, he added cryptically, "Left-handed people do it better.""
> - Tectonicist Eldridge Moores, quoted in *Assembling California* [1]

The Alexandria Digital Earth Prototype Project (ADEPT) is the research arm of the Alexandria Digital Library (ADL) at the University of California, Santa Barbara (UCSB). The ADL main collection currently indexes more than 2 million items, with more than 17,000 of these available online. The ADEPT middleware used to connect the ADL databases to web- and application-based front ends is designed to sit in front of relational databases, and support for searching across Z39.50 protocol and other data stores is under development. Searches across spatial and other terms are aggregated under a novel bucket system. This system is designed to support universal spatial searching across heterogeneous datasets.[2]

On top of this system, two general categories of visualization are being developed: georeferenced "digital earth" based systems, and abstract concept spaces. The relationships between the existing system and this work is shown in the architecture diagram in Figure 1. This dual approach is somewhat similar to the GeoVIBE and

K. Börner and C. Chen (Eds.): Visual Interfaces to Digital Libraries, LNCS 2539, pp. 199–213, 2002.

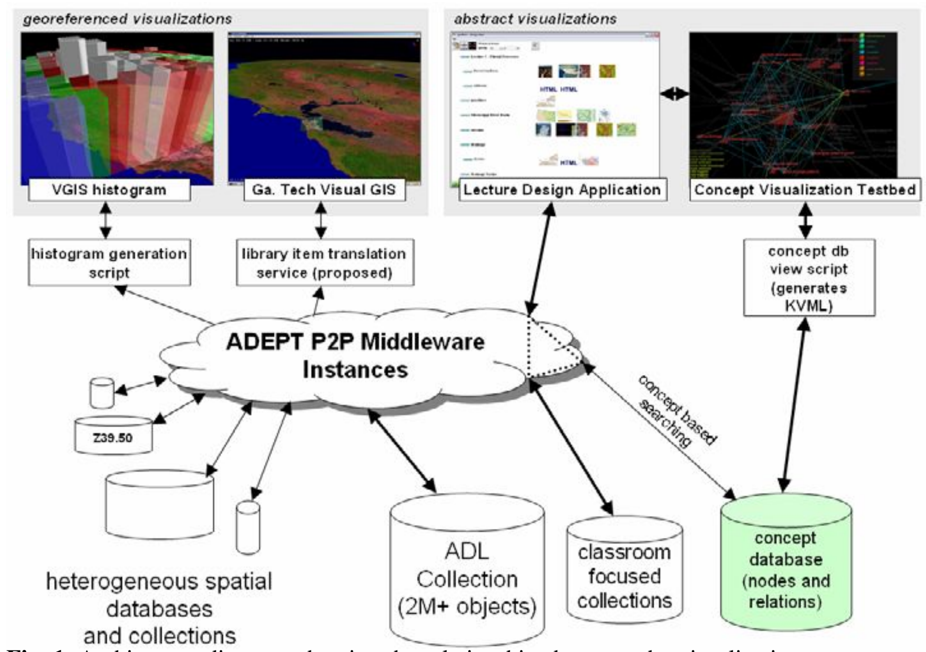


Fig. 1. Architecture diagram, showing the relationships between the visualization components and the rest of the Alexandria system.

VibeView approach presented by Cai in 2001.[3] The vast majority of objects currently in the library are maps and imagery; the connection between the middleware and existing visualization systems should be straightforward for these types of data (from the user's perspective, if not in implementation). Laying imagery down on 3D terrain and rendering it interactively is an example of this kind of service. There are a large number of terrain and imagery visualization systems currently available that this functionality could be implemented with. The two with closest ties to ADEPT are the GT-VGIS (the Georgia Tech Visual Geographic Information System)[4] and VTP (the open source Virtual Terrain Project).[5]

The second category of visualization being developed relates to abstract, nongeo-referenced spaces, with a focus on knowledge domain visualization via a concept-based approach. Based on graph visualization algorithms, this work is in the tradition of SEMnet[6] and Spire,[7] and Herman's[8] excellent overview of graph layout techniques. It draws inspiration from and has potential application to hypertext visualization work - work that traces it's past to Vannevar Bush's original, prescient vision of a hypertext system in 1945[9], and is still being pursued under the auspices of Englebart's Bootstrap Institute,[10] the nanotechnology development environment described by the Foresight Institute's Web Enhancement project,[11] and elsewhere. While the datasets rendered by the current concept space aren't specifically hypertext related, graph visualization techniques have potential application to hypertext systems, as illustrated by Park's NexistWiki system[12] and others.

While the abstract and georeferenced approaches are quite disparate, both address the core library issues of access, browsing, delivery and contextual understanding of library items and data. The central linkage between the two aspects is that since the abstract spaces considered here are relatively simple linear 3D spatializations, it's likely that techniques, interaction mechanisms and widgets (and the code to implement these features) can be shared between the two.

The current user base of the library consists mostly of researchers in the Earth Sciences. A primary goal of ADEPT is to extend the software beyond the research applications the library is currently used for and into use in educational contexts. Scenarios have been generated that involve the library software in lectures, labs and as a general resource for students. In support of this educational focus, the engineering team based at UCSB works with a separate education and evaluation (E&E) with members at both UCSB and UC-Los Angeles. The E&E team does feature capture and scenario development, and evaluates everything from cognitive walkthroughs and screenshot designs through to mockups and functional demos.[13]

1.1 Process

The development process for these applications is a hybrid of iterative user centered, scenario based design and the kind of small and fast development described by Verplank's spiral[14] or the futurist programming manifesto.[15] The current stage of the process involves development of scenarios and general feature descriptions (as captured by the E&E team) into screenshot proposals, specifications and small but functional applications. While VTP and GT-VGIS are both relatively heavyweight applications (requiring machines with graphics acceleration, gigahertz class processors and gigabit disks), the concept space work is designed to be as light as possible, so that it can run on classroom and lab machines, and even instructor's laptops. Simple XML formats are used as authorable file formats and for easy database/application interchange.

One goal common to all the designs presented here is to allow for the benefits of 3D and spatial visualizations (improved aesthetics and user experiences, top down and context-inclusive views of data, etc.) with the ease of use and familiarity of conventional 2D interfaces. One strategy for this is the employment of simple interaction techniques, such as highlit, clickable links that navigate the user from one viewpoint in the space to another and obviate the need for full 3D navigation. This kind of "push here" functionality is necessary for the transfer of this technology out of the lab and into user domains. The adoption of new visual language elements and new user interface affordances is a significant barrier to wider use of these kinds of systems.

This is a critical area for both good design work (one basis of which is described in the following section) and for evaluation of these designs.

1.2 Design Principles

The current user experience of both the web and web-based digital library systems is akin to reading materials by having someone stuff the pages through a small hole at you. 3D spaces have the potential to change this. Since there are typically some usability costs to users in 3D spaces (such as the necessity of having to free navigate through the space), there have to be significant benefits for the space to be successful. These potential advantages include answering user's questions, such as:

- How much stuff is here?
- What kind of stuff is it, and how big is the stuff?
- How is the stuff related to itself, and to other stuff?
- Where can I go from here?
- Where have I been?
- Who else is here?

The key is to *let these benefits drive the design*.

In addition to these motivating questions, Fabrikant and Buttenfield present a number of spatial concepts that form a design theory of information spaces[16], in a

Fig. 2. Sketch of an urban scale visualization, with georeferenced web pages.

framework similar to that originally proposed by Benedikt[17]: identity, location, direction, distance, magnitude, scale and change. While some of these questions can be adequately addressed in 2D, 3D has the potential to allow designers more flexibility to create designs that begin to answer these questions.

A sketch of how this might work in practice is shown in Figure 2. It shows a small web locale, perhaps a dozen or so pages, spatialized around an urban-scale visualization. This is a descendent of Ancona's former industry based work,[18] applied to the Alexandria library problem space. This neighborhood view (without the annotation layer) was produced by the U. of Toronto Centre for Landscape Research's software, but it's similar to the kinds of 3D spaces being made possible by both recent GIS tools or the Virtual Terrain Project. Another sketch of

Figure 4 (below, in section 2.3) is another sketch showing the application of these principles to a georeferenced space, as described in more detail in section 2. Benefits of spatialization for abstract, nongeoreferenced spaces are itemized in section 3.2.

2 Georeferenced Visualizations

The root metaphor behind this aspect of our work is that of the digital earth – an imagined whole earth simulation that would afford access to any kind of data: imagery, which is already well supported, but also eventually process simulations, visualizations, documents, and general datasets with georeferenced components. Any data with any kind of georeferenced component could be accessed through such an interface.

The following scenario illustrates this: starting at an interactive globe representation of the whole earth, the user zooms smoothly in through continental and macro-regional scales, locating the exact area of interest. Zooming further in, elements of the built environment as well as data associated with the region become visible. This data might be geologic, social scientific, archaeological, meteorological, historic, etc. – but it is available across all the resolutions zoomed through (at variable levels of detail and aggregation, with the interactivity serving to modulate the overmapping that would invariably occur), with street or building level granularity at the bottom.

The designed flexibility of the ADEPT middleware makes it an excellent starting point for providing the data retrieval and translation capabilities necessary for the implementation of this scenario. The rest of section 2 describes the incremental steps being taken towards the development of a system that could make this scenario possible.

2.1 Connections to Visual Geographic Systems

While the VTP and VGIS systems take significantly different technical approaches to the digital earth problem, both could benefit from integration with a digital library system. Features that could be implemented include:

- Automatic terrain draping of library items
- Visualization of collection and result set distributions
- Automatic retrieval of collection objects and object metadata through the Labeling of features via gazetteer integration
- visual interface

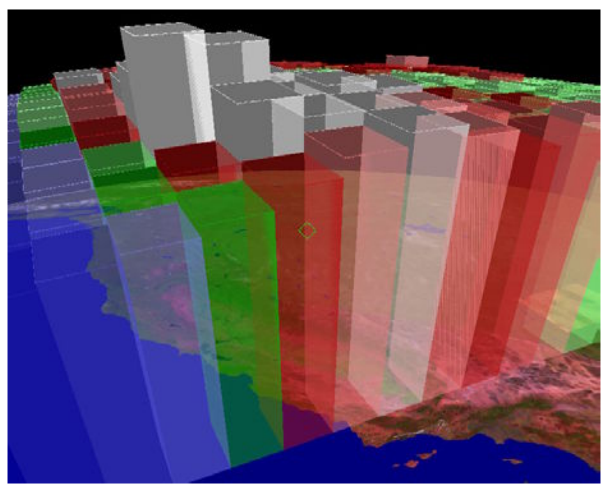


Fig 3. GT-VGIS system with ADL spatial histogram data layer, with height of columns showing number of items per 10 degree square.

2.2 Spatial and Temporal Histograms

One existing visualization of the library is the spatial histogram created as part of the collection level metadata for each collection. A 2D map based view of the spatial histogram, generated straight from the database, is included in the collection level metadata attached to each collection. Figure 3 shows part of an interactive, 3D representation of the same data, as displayed by the GT-VGIS System. The detail of this representation, which clearly indicates the relative number of items in each cell, is an incremental improvement over the 2D histogram, which shows only gross differences in the coverage by rendering each sampled square in a different color. This visualization hints at one possible advantage of moving to a 3D representation: answering the "how much is here?" question.

2.3 Search Results and Personal Collections

The ADEPT middleware will soon feature personal collections, or user created and maintained subsets of library items. The footprint visualization sketch shown in Fig. 4 is a screenshot mockup of proposed personal collection visualization features. This is an extension of the histogram visualization of Fig. 3, with some added features: individual footprints of maps, images and other library objects are visible, as is a tour of visible links connecting them.

This is a proposed implementation of the “iScape” (for Information landScape) client functionality originally imagined for the Alexandria Library. The concept space visualization presented in the next section presents another aspect of iScape development.[19]

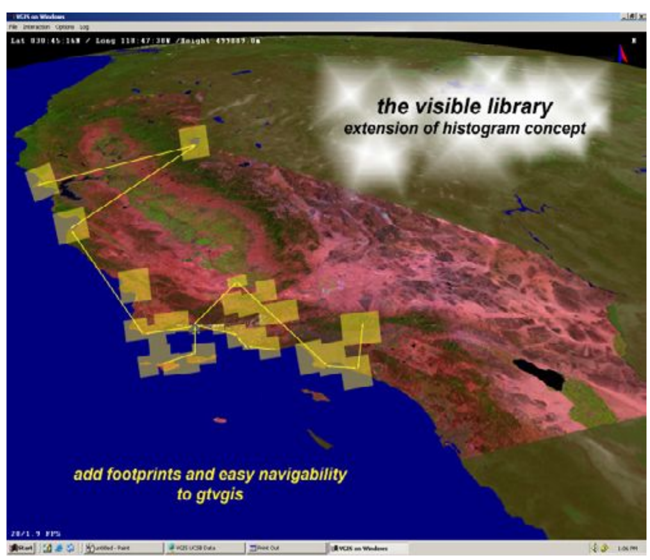


Fig. 4. GT-VGIS based mockup of proposed personal collection library item footprints, and a tour path. (proposed features)

3 Nongeoreferenced Visualizations

In addition to the georeferenced visualization work, we are also investigating the use of spatial and geographic metaphors for information presentation and visualization.

This work is similar to that of Chen[20], Buzydlowski[21] and of Fabrikant.[16] Fabrikant's need for a flexible, nonproprietary client for her current latent semantic indexing work partially drove the development of this testbed.

The concept space graph shown in Fig. 5 is generated from a database of concepts produced by the ADEPT knowledge organization team. Each concept record in this knowledge base includes any number of relationships to other concepts - relationships that were determined and coded by domain experts, and not automatically via latent semantic or other indexing. These relationships can be of several yet to be firmly decided types, and the knowledge base is scripted to generate the list of nodes and edges in an XML format. This attempt to map the concept space by looking directly at the concepts and their relationships to one another is complementary to the approach taken by Chen[22] and Boyack[23] whose strategies are to create thinly connected graphs based on authors and citations. Their approach, based on the Pathfinder Network[24] tree-pruning algorithm also described by Buzydlowski[21] was not deemed appropriate for this application due to the *a priori* existence of relationships in the database. These relationships were manually derived and loaded into each record by hand, and their visualization was a primary goal.

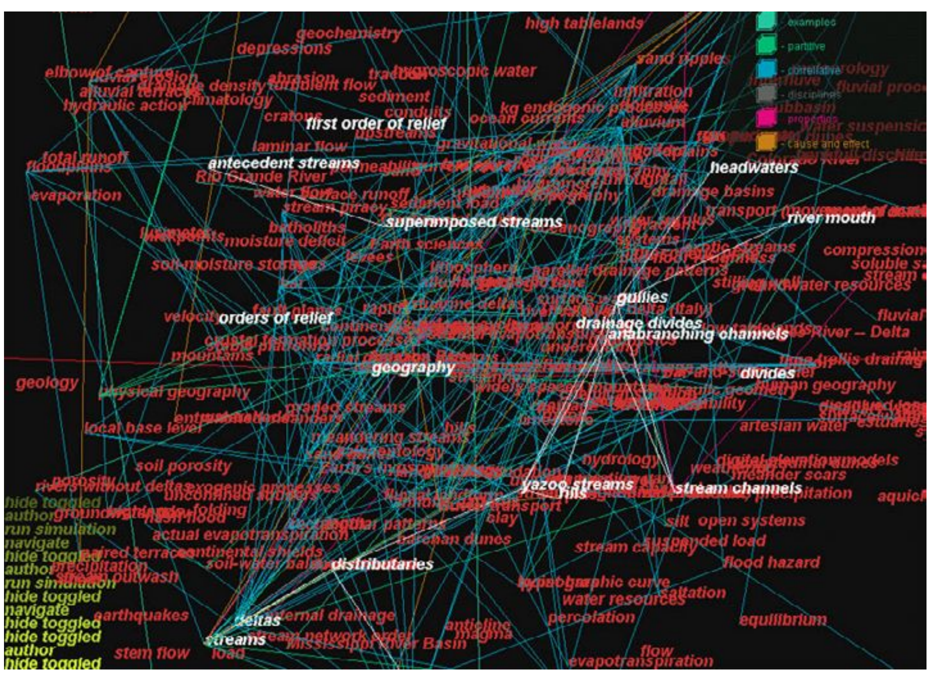


Fig. 5. Full display of the entire navigable concept space by the visualization estbed system. The interface for selection and deselection of relationship types is shown in the upper right hand corner.

Specific design goals for this software were for it to:

- Serve as a testbed for *reusable* interaction/navigation techniques
- Utillize a simple, flexible code
- Apply separation of style and data principles to a 3D space
- Be small, fast, lightweight
- Be unencumbered and released as open source
- Be a visualization of all nodes and relationships in the existing database

3.1 Concept Space Testbed

The testbed is a general purpose, interactive OpenGL based graph visualization tool, and a static screenshot is shown in Figures 5, 6 and 7. The program initializes by reading in an XML file that describes the styles and colors for the space. This file includes either the node and edge data, or a pointer to a file with this data in it, cleanly separating the data from the presentation.

The concept base can be searched via a popup text box. Nodes can be selected for addition or removal into a savable subset of the whole graph, and their positions can be manually adjusted. A simple binary force springs algorithm can be applied - repulsion is calculated between all nodes, and attraction along any activated edges. This algorithm is similar to the Fruchterman and Reingold algorithm describe by the chart found in Chen.[25] The details of this algorithm are shown in Equations (1) and (2),

$$f_a = (k_{ae} - d) / d \tag{1}$$

$$f_r = d / (k_r d / 2)^{1/2} \tag{2}$$

where f_a is the attractive force, k_{ae} is the attractive constant (which can be set per each edge set in the XML interface), d is the linear distance between the nodes, f_r is the repulsive force and k_r is a repulsion constant. Repulsion is calculated for each node relative to each other node and attraction is calculated along each node's connected edges. The attractive constant k_{ae} generates distances proportional to its value, unlike in a more traditional physics based algorithm, where a higher spring constant results in closer nodes. This factor can be used conveniently to alter the scale of the space, along with the size of the boundaries. The algorithm can also be run on subsets of the graph, either by selecting a subgraph and then hiding the rest of the graph, or by turning edge sets on and off with buttons provided in the interface.

While this algorithm is approximately of order $O(N)^3$, the algorithm runs at approximately five frames per second on a Pentium II 300Mhz laptop with a data set of around 300 nodes. Because of the lack of large geometry and the streamlined

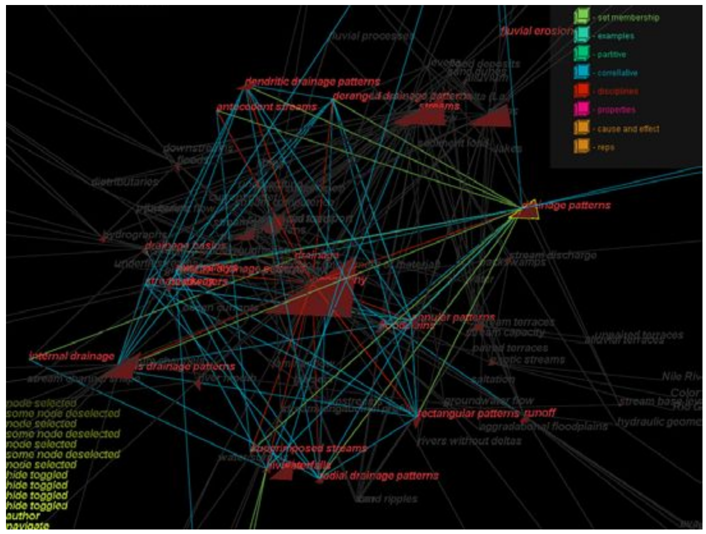


Fig. 6. A slightly smaller space than that shown in Figure 5, shown after the force springs algorithm was run, and with a highlit subgraph. The visibly different widths of the nodes are a function of that node's number of outgoing links.

C/OpenGL framework, it runs adequately without 3D hardware acceleration, although the framerate during navigation is increased for large datasets with the pres
ence of graphics acceleration hardware.

Since this system works on a database of concepts (there will eventually be a few thousand, and most views will be much smaller) and not on library items themselves (which can number in the millions for some collections, although many personal collections are on the order of hundreds), this is an acceptably scalable solution for this application. Intentional deformation techniques (such as hyperbolic projection[26]) were eschewed as being contrary to the small and fast design principles adopted for this implementation, as well as due to the apparent usability costs of such systems. Also, the navigation and manipulation code for such a space would be more specific and less likely to apply to other designs - contrary to the goal of flexibility and reusability.

Ongoing challenges facing this technique include difficulties with the inherent complexity of the graph. Fig. 7 shows a first attempt at a solution to this issue, a subset of the graph trimmed down to a manageable size for presentation to nonexperts. Early feedback from expert users of the software indicate that the benefits of answering the "how much is here?" and "where else can I go?" questions outweigh the complexity of the interface, but this balance is shifted negatively for nonexperts. The size of this graph was determined by the application of the "7±2" heuristic[26], which describes the approximate number of items a person can keep in short term

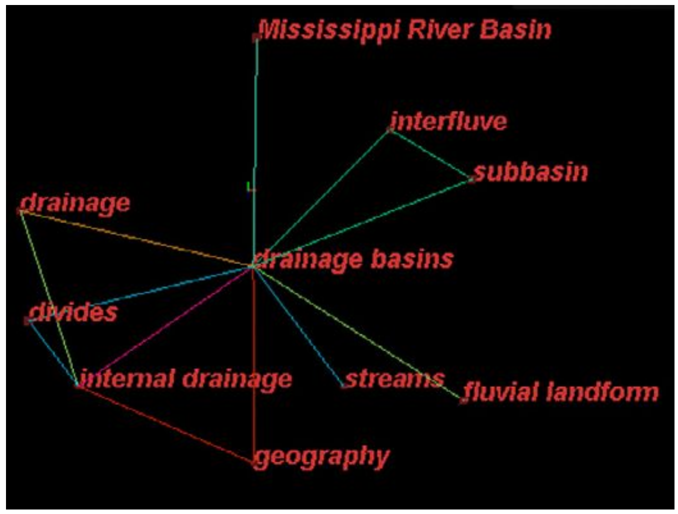


Fig 7. Display of a manageable subset of the concept space shown in Figures 5 and 6. Edges of different types are rendered in different colors.

memory – or "7±2, plus an extra" in this case. Further experimentation with color, highlighting, planar projection and navigation may result in other solutions, and a 2D application for these small graphs is under development.

3.2 Benefits and Features of This Approach

Interactivity and 3D allow for the view of spaces that would otherwise be impractical to view in 2D. Traditional edge crossing minimization algorithms work fine for loosely connected graphs, or ones that have been modified according to a Pathfinder network or other edge culling algorithm, but for a fixed dataset like this the annoyance of edge crossings is ameliorated by the interactivity of the space.

Landmarks (described in Chen[28] and suggested for use in the testbed by Fabrikant) were implemented by making the width of each node proportional to the number of outgoing links, and the height proportional to the incoming links. This feature is visible in the differently sized triangles shown in Figure 6.

Another ongoing challenge in this area is to make nonambiguous use of as many of the spatial concepts identified by Fabrikant and Buttenfield as possible; again, these are identity, location, direction, distance, magnitude, scale and change.[29] Several of these are challenged in the current design: first, the labels are of fixed size and take up as much or more screen real estate as the node shapes themselves. Second, the nodes themselves are different sizes – while this is helpful in giving the user more information about the graph structure, it does preclude the possibility of using node size to gauge distance. Early user feedback has indicated that the perception of distance in the space is problematic, at least when the space is being projected onto a screen and not directly interacted with. These problems could be ameliorated with improved navigation (such as a spaceball), viewing the space through a simulated 3D projection system such as GeoWall,[30] or perhaps through the use of fog to provide an increase in depth cueing.

3.3 Applications of the Testbed

This testbed system is currently being or about to be used in the following four situations, all related immediately to the teaching of an introductory level geography course:

Knowledge space construction. The team that has undertaken the truly massive task of generating the knowledge base (with several thousand records so far) has been generating views and using the resulting spaces to double check their work.

Lecture authoring. The instructor and TAs designing the lectures based on the concept base are using the testbed for lecture development. The TAs have shown interest in using the browsing capability of the software for generating lecture frameworks. A rough but functional mockup (labeled "Lecture Authoring Application" in Figure 1) has been created and is being evaluated by the E&E team.

Classtime. Smith is planning on using the testbed to present simplified concept spaces to students as a part of the lectures. There are some unresolved issues with both the rendering of simple diagrams, and with detailed usability issues surrounding the creation of those simple spaces. It is possible that 2D versions of the knowledge spaces may be utilized for these purposes, as the main benefit of 3D (it's ability to display the full, complex graph of the space) becomes less significant with small graphs.

Evaluation. The testbed has been successfully integrated with a simple lecture design prototype system that is in the process of being evaluated by the E&E group in the project.

4 Future Work

The georeferenced work presented here will someday allow instructors to design spaces that allow students (whether they're left or right handed) to make the kinds of leaps of understanding that Eldridge Moores talked about. The ease of laying maps on terrain that could be implemented with a connection between VGIS or VTP and ADEPT is an incremental first step towards this goal.

The feature list roadmap for the abstract space testbed has grown lengthy even with the few users already experimenting with it. Experimentation with simplified, geometric 2.5D layout algorithms, widgets for addition and manipulation of nodes and edges, interoperability with the X3D standard for crossplatform viewing of spaces and numerous navigational improvements are all possibilities. Some of the feature requests are along the lines of those proposed for what would eventually become the (unfinished) Vizbang data visualization client.[31] In the near term, the testbed may be applied to LSI generated frameworks, although it's difficult playing catch-up with the proprietary software currently being used. It could also potentially be applied to other domains, such as social network visualization or other graph structures.

5 Conclusions

This work is ongoing and only preliminary conclusions can be drawn. Not all of the designs presented here will turn into useful applications. However, the four application areas of the concept space testbed presented in Section 3.3 are an encouraging sign.

With user interface design stuck in the local minimum of Windows, Icon, Menus and Pointers, some dead ends and negative results are to be expected. Dissatisfaction with WIMP interfaces is widespread, if not universal. Since all data has context (either georeferenced or not), it can be spatialized, and it’s possible that investigations along these lines will result in new paradigms for general users. For example, the concept space work presented here could potentially serve as a content aggregation strategy, or a way of visualizing general digital library contents. Both the concept space and digital earth designs address the same basic set of problems: access, browsing, delivery and understanding of library items. Because of these common underlying motivations, it’s likely that there will be interactions between these two threads in the future.

This work tends to provoke strong feelings – some users can’t stand it, but others find it appealing and are drawn to it. Solutions to the objections of skeptical users can be harnessed in the process of design, evaluation, improvement and redesign. There is no intrinsic reason why 3D information spaces should be more difficult to use than 2D ones. As these solutions are designed, this work will play a role in the development of significant, new and broadly appealing user interface paradigms.

Acknowledgements

Thanks to Olha Buchel, Tim Tierney and the rest of the ADEPT Knowledge Organization team for the creation and scripting of the concept space database, and to Greg Janée for his edits on the first edition. The work described herein was partially supported by the NSF-DARPA-NASA Digital Libraries Initiative and the NSF NSDL initiative under NSF IR94-11330, NSF IIS-9817432, DUE-0121578, and UCAR S02-36644.

References

1. McPhee, John. *Assembling California*. Farrar, Straus and Giroux: New York, NY. 1993.
2. Janée, Greg and James Frew. The ADEPT Digital Library Architecture. *Proceedings of the Second ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL '02)*, Portland, OR, July 14-18, 2002.
3. Cai, Guoray. GeoVIBE: A Visual Interface to Geographic Digital Library. *Proceedings of the First Visual Interfaces to Digital Libraries Workshop*, Roanoke, VA. June 28, 2001.
4. Koller, David, with Peter Lindstrom, William Ribarsky, Larry F. Hodges, Nick Faust, and Gregory Turner. 1995. *Virtual GIS: A Real-Time 3D Geographic Information System. Proceedings of Visualization '95*. GVU Technical Report 95-14.
5. The project website can be found at http://www.vterrain.org/
6. For a brief description and screenshot of SEMnet, see http://www.birkhauser.com/hypermedia/cyb52.html
7. http://www.pnl.gov/infoviz/spire/spire.html
8. Herman, Ivan, with Guy Melançon and M. Scott Marshall. 2000. Graph Visualization and Navigation in Information Visualization: A Survey. *IEEE Transactions on Visualization anc Computer Graphics*, Vol. 6 No. 1, January-March 2000.
9. Bush, V. As We May Think. *The Atlantic Monthly* July: 101-8 1945.
10. Engelbart, D. Knowledge-Domain Interoperability and an Open Hyperdocument System. *Proceedings of the Conference on Computer-Supported Cooperative Work,* Los Angeles, CA, October 7-10, 1990, pp. 143-156.
11. Drexler, K. Eric. Engines of Creation, The Coming Era of Nanotechnology. 1986. http://www.foresight.org/EOC/index.html, see Ch. 14 ("The Network of Knowledge") in particular.
12. Park, J. Thinking about Markup Languages in the Context of Complex, Urgent Problems. 2002. http://www.nexist.org/em2002.
13. Leazer, Gregory H., Anne J. Gilliland-Swetland and Christine L. Borgman. Evaluating the Use of a Geographic Digital Library in Undergraduate Classrooms: ADEPT. *Proceedings of the Fifth ACM Conference on Digital Libraries*, 2-7 June 2000, San Antonio, TX (ACM Press, 2000): 248-249.
14. As described in Holmquist, Lars Erik, Breaking the Screen Barrier - http://www.viktoria.se/play/publications/2000/phd/leh/
15. Haeberli, Paul and Bruce Karsh, http://www.sgi.com/grafica/future/

16. Fabrikant, S. I. and Buttenfield, B. P. "Formalizing Semantic Spaces for Information Access" *Annals of the Association of American Geographers*, 91: 263-280. 2001.
17. Benedikt, M. Cyberspace: Some Proposals. In *Cyberspac: First Steps*, edited by M. Benedikt. Cambridge, MA: MIT Press. 1991.
18. Ancona, D. 3DXML: Authoring, Publishing and Viewing Structured Information. *VRML Consortium's Database Working Group*. 1998 whitepaper. http://www.vrml.org/WorkingGroups/dbwork/ancona/home.html
19. Gilliland-Swetland, Anne J. and Gregory H. Leazer. Iscapes: Digital Libraries Environments for the Promotion of Scientific Thinking by Undergraduates in Geography. *Proceedings of the First ACM-IEEE Joint Conference on Digital Libraries*, 24-28 June 2001, Roanoke, VA.
20. Chen, C. & Paul, R. J. Visualizing a knowledge domain's intellectual structure. *IEEE Computer*, 34(3), 65-71. 2001.
21. Buzydlowski, Jan W., Howard D. White and Xia Lin. 2001. *Proceedings of the First Visual Interfaces to Digital Libraries Workshop*, Roanoke, VA June 28, 2001.
22. Chen, C. Visualising Semantic Spaces and Author Co-citation Networks In Digital Libraries. *Information Processing & Management* 35(3), 401-420. 1999.
23. Boyack, Kevin W., with Brian N. Wylie and George S. Davidson. Information Visualization, Human-Computer Interaction, and Cognitive Psychology: Domain Visualizations. *Proceedings of the First Visual Interfaces to Digital Libraries Workshop*, Roanoke, VA June 28, 2001.
24. Schvaneveldt, Roger W., ed. Pathfinder Networks: Theory and Appliations. AblexNorwood NJ. 1990.
25. Chen, C. Information Visualisation and Virtual Environments. London: Springer-Verlag. 1999.
26. Munzner, Tamara. Interactive Visualization of Large Graph Networks. Ph.D. dissertation, Stanford University. 2000.
27. Miller, George. The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information. *The Psychological Review*, 1956, vol. 63, pp. 81-9.
28. *ibid*, pp. 38-39.
29. *ibid*.
30. http://www.geowall.org/
31. *ibid*.

A Lightweight Protocol between Digital Libraries and Visualization Systems

Rao Shen, Jun Wang, and Edward A. Fox

Virginia Tech., Computer Science Department,
Blacksburg, VA 24060, USA
{rshen,juwang2,fox}@vt.edu

Abstract. A lightweight protocol, VIDI, is proposed to enhance the interoperability of digital libraries (DLs) and visualization systems (VIS). VIDI is related to the Open Digital Library project, which encourages a component-based approach to the construction of digital libraries, implemented by extending the Open Archives Initiative Protocol for Metadata Harvesting. VIDI adds in the concept of a registry of transformers to convert between common metadata and visualization formats. Design and implementation discussions of VIDI explain its feasibility, flexibility, and generality.

1 Introduction

Most of today's digital libraries (DLs [1], [2]), have simple query interfaces and offer no geometric display of data. Thus, more and more research now is being done in information visualization because of its relevance and advantages in a variety of domains, including in the field of DLs as described by Dan Ancona in Part III in this book.

Consider the case of coupling visualization with digital library technology, as we have done in the ENVISION_MARIAN project [5], [12]. Then, a visualization system (VIS [13]) communicates with the user to form the request and sends it to the back-end DL. It transforms the unseen internal semantic representation of data into visible geometric displays. However, in such a coupled system, the DL and VIS are usually bound directly to each other; they transfer query and result data in some format that perhaps only they can recognize.

DLs are often huge, with sizes still growing. It is fundamentally important to find a way to pass the data required into various VIS tools to achieve users' different needs. It is impractical and unnecessary to generate a VIS tool supporting all advanced visualization techniques and all kinds of data formats, especially when both technologies are advancing quickly. So, naturally, multiple visualization tools should be used.

Ideally each DL and VIS could be weakly coupled and freely connected, so that users can access both DL and VIS synchronously and freely as shown in Fig. 1. Thus, for example, VIS1 could be used to access DL1 and DL2, and DL2 could be accessed by communicating with either VIS1 or VIS2.

K. Börner and C. Chen (Eds.): Visual Interfaces to Digital Libraries, pp. 217–226, 2002.

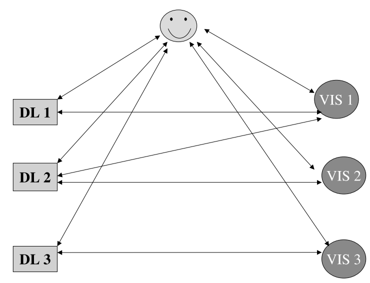


Fig. 1. User can access multiple DLs and VIS through the VIDI protocol

VIDI [11], a lightweight protocol extended from the Open Archives Initiative (OAI) Protocol for Metadata Harvesting (OAI-PMH [7]) is proposed. There are two classes of participants in the OAI-PMH framework: data providers and services providers. The available online lists of registered data providers and service providers found at the OAI home page (http://www.openarchives.org/) provide an indication of the growing number of participants in OAI.

Based on the simplicity, wide acceptability, and success of the OAI-PMH, VIDI presents a new solution to achieve interoperability between DL and VIS. VIDI has an interoperability and scalability framework with two participants: DL providers and visualization providers. DL providers administer DLs that support the VIDI protocol as a means of exposing metadata about the content in their systems. Visualization providers issue VIDI protocol requests to the systems of DL providers and use the returned data as a basis for visualization and presenting results to the users. Any VIS that adopts the VIDI protocol could be a *Visualization provider*. Thus, any VIS, such as the SomFrontEnd [6] may register as a visualization provider with its name and the website of its VIDI interface offered to a central open registry. SomFrontEnd is a tool developed by the Artificial Intelligence (AI) Lab at University of Arizona for viewing classification output. In our implementation of the ODL_NP/SOM prototype system, we make SomFrontEnd act as a visualization provider.

2 VIDI Protocol

The VIDI protocol extends from OAI-PMH, which defines a mechanism for harvesting XML-formatted metadata from repositories. A harvester is a client application that issues OAI-PMH requests, while a repository is a network accessible server that can process the six OAI-PMH requests. To support OAI-PMH, the HTTP Post method is recommended for the VIDI implementation because there is no limitation on the length of the arguments for POST. Each VIDI protocol request has one key=value pair that specifies the VIDI protocol request as an OAI-PMH protocol request. Responses to VIDI requests are formatted as HTTP responses with appropriate HTTP header files. Each request returns a Content-Type of text/xml.

This extension of OAI-PMH is an application of the concept of Open Digital Libraries, first expounded in [8], and elaborated in [9]. The ODL approach argues for using lightweight protocols that are well defined for a particular functionality in order to connect the components that make up digital library systems. For VIDI, we expand this concept in order to more easily build integrated DL-VIS environments. In particular, as is shown in Table 1, we adapt two of the five OAI requests, and add in three others (two of which are take-offs on OAI verbs).

Table 1. Comparision of OAI and VIDI Requests

OAI	**OAI&VIDI**	**VIDI**
GetRecord	Identify	ListVisdataFormats
ListIdentifiers	ListMetadataFormats	ListTransformers
ListRecords		ListResultSet
ListSets		

Unlike OAI-PMH (with six verbs), only five commands are identified and defined in the VIDI protocol. Further, all the commands defined in VIDI, except the *RequestResultSet* command, have options to be issued to either DLs or VIS. The *RequestResultSet* command is the only one for transferring query data. The other four are used for exchanging system information, and can be implemented in Client-Server or Server-Server mode. Their respective roles are:

Identify: It is a mandatory command used to retrieve system information about a DL or VIS necessary for the protocol communication. The response to this request includes the name for the DL or VIS, the base URL for the DL or VIS, the timestamp for the last time the data structure changed, the version of the VIDI protocol, and the email address of the administrator of the system. This command needs no argument, and the request example may be as follows:
http://DL_host:8080/DL/servlet/DL?verb=identify

ListMetadataFormats: It is used to retrieve the metadata formats available from a DL. It needs no argument, and the request example may be as follows:
http://DL_host:8080/DL/servlet/DL?verb=ListMetadataFormats

ListVisdataFormats: It is used to retrieve the data format VIS understands. The format of data that VIS can visualize is called visdata format. This command needs no argument, and the request example may be as follows:
http://VIS_host:8080/VIS/servlet/VIS?verb=ListVisdataFormats

ListTransformers: It is used to retrieve the transformers that VIS supports to transform the metadata format to visdata format. It has three optional arguments that constrain the transformers for specific use by the supported metadata format, visdata format, or by the identifier of the DL. This command request example may be as follows:
http://DL_host:8080/DL/servlet/DL?verb=ListTransformers

RequestResultSet: It is used to request the query result set from a DL. It has three required arguments of the request URL of the VIS, the visdata format the VIS needs, and the unique query identifier for result set from the DL. It also has two optional arguments of the name of the VIS, and transaction ID of the request. This command request example may be as follows:
http://DL_host:8080/DL/servlet/DL?verb=requestresultset&
name=SOM&visformat=SomMap&baseURL=
http://VIS_host:8080/VIS/servlet/VIS&resultsetid=
verb=ListRecords&metadataPrefix=oai_dc&set=odlsearch1/+creator=
robert+title=computer/1/25

For each of the above request commands, an XML schema is developed for the response format.

3 VIDI Implementation

Implementation of the VIDI protocol is flexible, and can be either in Client-Server or Server-Server mode. The tradeoff is between simplicity and functionality. ENVISION_ODL and ENVISION_MARIAN were implemented using Client-Server mode by Wang [11], where the VIS (ENVISION [5], [12]) acts as a client, sending requests to the DL (either ODL [8], [9] or MARIAN [3]). Development of another prototype system ODL_NP/SOM is presented below to illustrate the feasibility, flexibility, and generality of the VIDI protocol. A block diagram for our first attempt to work with NP/SOM is given in Fig. 2. The flow graph of the Client-Server protocol implementation is given in Fig. 3.

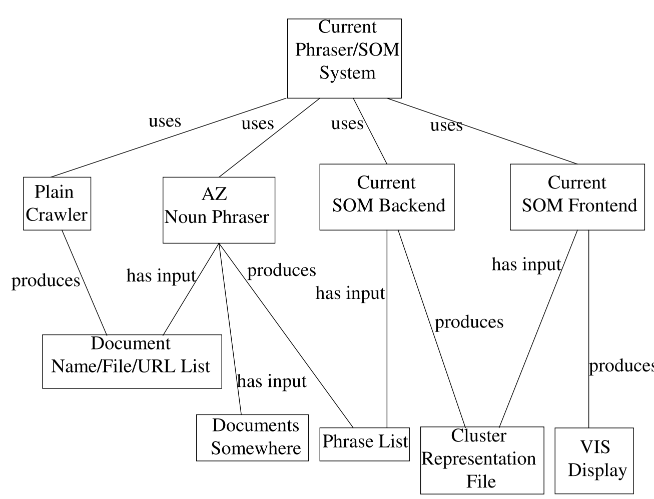


Fig. 2. Block diagram of NP & SOM system

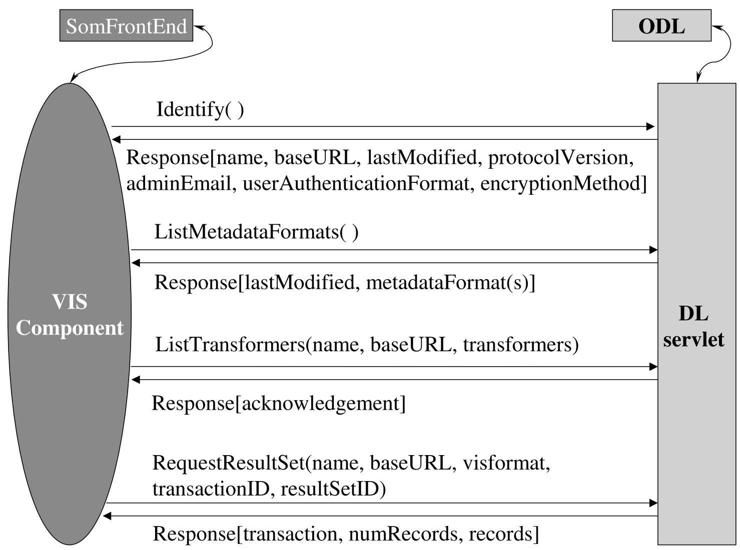


Fig. 3. Flow graph of Client-Server protocol implementation

3.1 Protocol Implementation on VIS Side

The client functions of VIS are encapsulated in a component named *VISComponent*, which implements VIDI requests sent to the DL. *VISComponent* can be either a Java servlet or a Java application. It sends VIDI commands on behalf of VIS to the DL and forwards the retrieved result to VIS for visualization. Two VIS systems, ENVISION and SomFrontEnd, can connect to DL through *VISComponent* developed as a Java application using the VIDI protocol.

3.2 Protocol Implementation on DL Side

A Java servlet is set up to talk with DLs as a proxy. The servlet can respond to request commands from VIS. The response for *Identify* and *ListMetadataFormats* are static and predetermined by the DL with which it communicates. For the *ListTransformers* command, the servlet retrieves the transformer and returns nothing but an acknowledgement. For the *RequestResultSet* request, the servlet retrieves the *reslult set ID*, which is a query identifier used to request the result set from the DL. The servlet applies the transformation designated by VIS through the *ListTransformers* command on the result set it gets from the DL, and then the wrapped result with corresponding *visdata format* can be visualized by VIS.

Currently, two DL systems can be connected to VIS using the VIDI protocol, ODL, and MARIAN. MARIAN is an indexing, search, and retrieval system optimized and extended for digital libraries. Open Digital Libraries (ODLs) are systems originally built by Hussein Suleman as networks of extended Open Archives. As illustrated in Fig. 4, the data was aggregated into a central archive

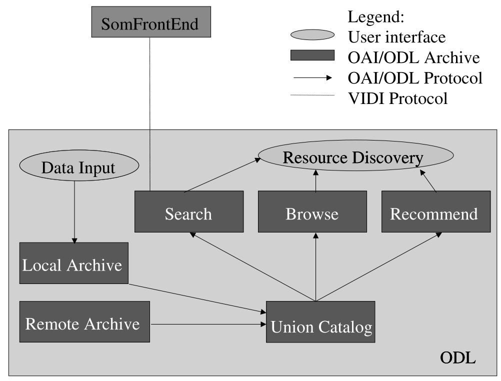


Fig. 4. ODL_NP/SOM prototype system

for use by local services; three high-level services were provided using this data: Search, Browse, and Recommend; search indexed the data and exposed an OAI-like interface for specifying keyword queries; SomFrontEnd then communicates with the Search component using the VIDI protocol. Thus, VIDI is an extension of the ODL concept to extend beyond DL to also VIS.

3.3 Protocol Implementation on Transformer

A transformer plays an important role in the implementation of the VIDI protocol. It can be as simple as an XSLT stylesheet that transforms some specific XML metadata format supported by DL to some XML visdata format supported by VIS. It can also be as complex as an application that accepts the data in some specific metadata format and outputs the data in a specific *visdata format*.

A transformer is sent to the DL through the ListTransformers command; it can be either an XSLT stylesheet or a URL of an online accessible application. In the ENVISION_ODL and ENVISION_MARIAN prototype systems, an XSLT stylesheet is developed to convert the data in Dublin Core format to visdata format supported by ENVISION. In ODL_NP/SOM prototype system, a chain of servlets is developed to perform transformation.

Theoretically, if we have d DLs supporting d different metadata formats and v VIS systems supporting v various *visdata formats*, we would need to code $d*v$ transformers for interoperability as shown in Fig. 5. However, not all VIS systems might be appropriate to display all DLs, so we might need less transformers. If there are $m \ll d$ *metadata formats* supported by d DLs and $n \ll v$ *visdata formats* supported by VIS systems, only $m*n$ transformers are needed. Compared with the situation in Fig. 5, only three transformers are needed in Fig. 6. Both DL1 and DL2 support the Dublin Core metadata format, which is a simple yet effective element set for describing a wide range of networked resources. We plan to register the transformers in an open transformation registry. The registration serves as a publicly accessible list of VIDI conformant repositories, making it easy for both DLs and VIS to discover the transformers.

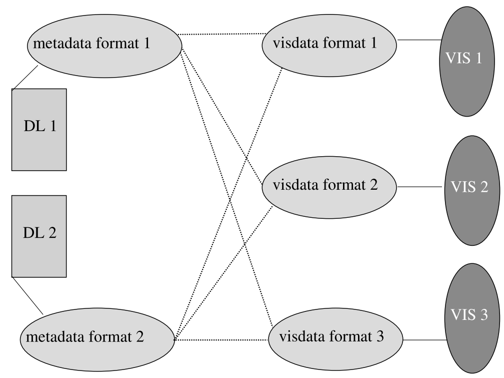


Fig. 5. Need 2*3 transformers for interoperability

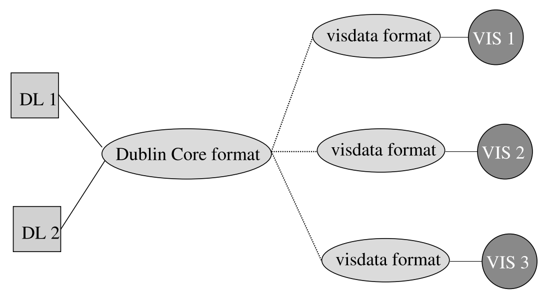


Fig. 6. Need 1*3 transformers for interoperability

3.4 VIDI Example: NP and SOM

To illustrate our approach, we consider the visualization of DL content based on an analysis of noun phrases (NPs) and their occurrences, using self-organizing maps (SOMs). Fig. 2 illustrates the various parts of such a system, using the form of a concept map.

SomFrontEnd can visualize data generated from the Noun Phrasing (NP [10]) and Kohonen Self-Organizing Map (SOM [4, 6]) applications which were developed by the AI Lab at the University of Arizona. The AZ Noun Phraser is a noun phrase generation tool for extracting high-quality phrases from textual data. SomBackend is a tool for classifying textual documents. In our ODL_NP/SOM prototype system, a chain of servlets is developed to perform transformation. It encapsulates the functions of the AZ Noun Phraser and SomBackend to transform data into a visdata format supported by the SomFrontEnd. Fig. 7 illustrates the flow of data in accord with the VIDI protocol. The XML schema for ODL's metadata format is based on the Dublin Core Schema, and the XML schema for SomFrontEnd's visdata format is based on two files generated from SomBackend.

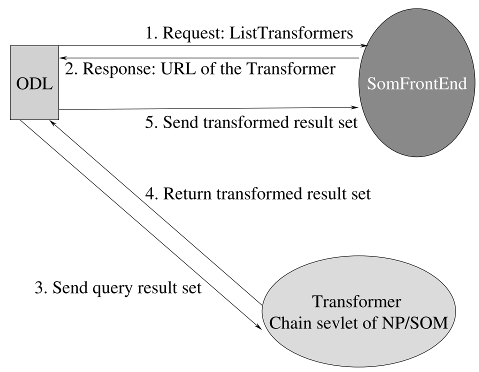


Fig. 7. ODL_NP/SOM system

3.5 Technology Used

XML technologies: XML is a data format that represents data in a serialized form that can be transported over the network from one endpoint to another. So the use of XML for data exchange is very important in the VIDI implementation.

JAXP: The Java API for XML Processing is a native Java interface to the industry standard XML parsing APIs, SAX (Simple API for XML) and DOM (Document Object Model), along with a pluggable interface to an XSLT (XML Stylesheet Language Transformations) engine. Using JAXP in the VIDI implementation gives us a full-featured API for accessing, modifying, and creating XML documents in Java.

XSLT: XSLT is a mechanism to convert an XML document from one schema to another. We use XSLT to transform metadata format supported by DL to visdata format understood by VIS.

Java Servlets: A servlet is a request/response-base Java object that runs within the managed container environment. We use a lightweight servlet to handle VIDI requests and responses on DL side. We also use a chain of servlets to deal with complex transformation in the ODL_NP/SOM prototype system, as shown in Fig. 7.

4 Conclusion and Future Work

There are many efforts and applications of the Open Archives Initiative (OAI). For example, the Open Citation Project is working towards becoming a registered service provider with the OAI. Also, the Computing and Information Technology Interactive Digital Educational Library (CITIDEL) extends OAI to achieve higher performance across the broad Internet [1]. In this paper we present VIDI, an extended OAI protocol. Our conclusions from this research work are:

– VIDI can enhance the interoperability between visualization systems and digital libraries.

– The feasibility of VIDI can be understood by considering the implementation of the ODL_NP/SOM system.

Future research may include:

– In-depth analysis and additional implementations of the VIDI protocol. These need to be completed in order to thoroughly evaluate the protocol.
– Providing registration service for DL_VIS transformers. These need to be implemented in a secure and convenient fashion so that many of the developers of DL and VIS systems find them convenient and worth supporting and advertising. We encourage the involvement of other researchers in this field in this project. We hope that the basic idea, enhanced by the comments of others, will lead to greater use of VIS systems in conjunction with DLs.

Acknowledgements

This research work was funded in part by the NSF through grants: CDA-9312611; DUE-0121741, 0136690, 0121679; IIS-0080748, 0086227, 0002935, and 9986089. Thanks go to all colleagues in the Digital Library Research Lab at Virginia Tech for their very useful and important suggestions and to the Artificial Intelligence Lab at University of Arizona for software and assistance with NP and SOM.

References

1. Fox, E.A.: Advancing Education through Digital Libraries: NSDL, CITIDEL, and NDLTD. In Proc. of the International Conference: Digital Library IT Opportunities and Challenges in the New Millennium, Beijing (2002), pp. 107-117
2. Fox, E.A., and Urs, S.R.: Digital Libraries. Annual Review of Information Science and Technology, ed. Blaise Cronin, Vol. 36, Ch. 12, pp. 503-589 (2002)
3. Gonçalves, M.A., France, R.K., and Fox, E.A.: MARIAN: Flexible Interoperability for Federated Digital Libraries. In Proceedings 5th European Conference on Research and Advanced Technology for Digital Libraries, ECDL'2001, September 4-8, 2001, Darmstadt, Germany, pp. 173-186
4. Lin, C., Chen, H., and Nunamaker, J.F.: Verifying the Proximity Hypothesis for Self-Organizing Maps. Journal of Management Information Systems, 16(3):57-70, 2000
5. Nowell, L., France, R., Hix, D., Heath, L., and Fox, E. Visualizing Search Results: Some Alternatives to Query-document Similarity, in Proc. SIGIR'96, Zurich, Switzerland, Aug. 18-22, 1996, pp.67-75.
6. Orwig, R.E., Chen, H., and Nunamaker, J.F.: A Graphical, Self-Organizing Approach to Classifying Electronic Meeting Output. Journal of the American Society for Information Science. 48(2):157-170 (1997)
7. Sompel, H.V., and Lagoze, C.: The Open Archives Initiative Protocol for Metadata Harvesting. (2001) Available online: http://www. openarchives.org/OAI/openarchivesprotocol.htm
8. Suleman, H., and Fox, E.A.: A Framework for Building Open Digital Libraries. D-Lib Magazine, December 2001, 7(12)

9. Suleman, H., and Fox, E.A.: Designing Protocols in Support of Digital Library Componentization. In Proceedings of the 6th European Conference on Research and Advanced Technology for Digital Libraries (ECDL2002), September 16-18, 2002, Rome, Italy, pp.568-582
10. Tolle, K.M., and Chen, H.: Comparing noun phrasing techniques for use with medical digital library tools. Journal of the American Society for Information Science, 51(4):352-370, 2000
11. Wang, J. (2002). VIDI: A Lightweight Protocol between Visualization Systems and Digital Libraries. Master's Thesis, Dept. of Computer Science, Virginia Tech
12. Wang, J., Agrawal, A., Bazaz, A., Angle, S., Fox, E.A., and North, C. Enhancing the ENVISION Interface for Digital Libraries. Short paper in Proc. JCDL'2002 Second Joint ACM / IEEE-CS Joint Conference on Digital Libraries, July 14-18, 2002, Portland, pp.275-276.
13. Wise, James A., Thomas, James J., Pen-nock, Kelly, Lantrip, David, Pottier, Marc, and Schur, Anne. Visualizing the non-visual: Spatial analysis and interaction with information from text documents. In Proc. of the Information Visualization Symposium, pp. 51-58. IEEE Computer Society Press, 1995.

Top Ten Problems in Visual Interfaces to Digital Libraries

Chaomei Chen[1] and Katy Börner[2]

[1]College of Information Science and Technology, Drexel UniversityPhiladelphia, PA 19104-2875, USA
Chaomei.Chen@cis.drexel.edu
[2]School of Library and Information Science, Indiana University, Bloomington, IN, 47405, USA
katy@indiana.edu

Abstract. Many research communities periodically and collectively deliberate about their most significant and challenging problems as a way to agree upon pressing questions and promising research directions. Identifying top-ten problems is a useful way to reflect on what a community has achieved and define a new research agenda for the future. This chapter introduces a set of such problems for research in visual interfaces of digital libraries in order to simulate studies in this area.

1 Motivation

David Hilbert presented one of the most famous lists of challenging mathematical problems to the International Congress of Mathematicians in Paris in 1900. His twenty-three mathematical problems influenced and continue to influence mathematical research all over the world.

Research on visual interfaces of digital libraries grew out of several interrelated areas such as information visualization, digital libraries, and information retrieval. As a result, there is a limited consensus on the set of major research questions and their priority, making it hard for researchers in the field to define their research, to find collaborators and combine efforts, and to contribute to a general theory of information visualization. So far, there exists no central software code repositories and very few data sets are shared, which might be due to the fact that information visualization as a field itself is very young.

In this chapter, we first outline some top-ten problems identified in fields that are closely related to visual interfaces in particular and information visualization in general. Then we make our first attempt to crystallize the top-ten problems based on what we learned from the two workshops in 2001 and 2002[1]. Many stimulating ideas are presented in earlier chapters of this book; we conclude by listing which of these initiatives we think are most significant and what remains to be done in the future.

[1] http://vw.indiana.edu/visual01 and http://vw.indiana.edu/visual02/

K. Börner and C. Chen (Eds.): Visual Interfaces to Digital Libraries, LNCS 2539, pp. 226–231, 2002.

2 Related Top-Ten Problem Lists

Various "top-ten problems" lists have emerged in research communities related to visual interfaces of digital libraries and information visualization. For example, Foley identified the ten most challenging problems [1] in computer graphics. He emphasized that each of them shares common concerns and also represents a particular perspective of the discipline. To get a sense of what a top-ten list looks like, Foley's top-ten problems are shown below:

1. Fill the gap between image-based and geometric modeling techniques
2. Fill the gap between motion-capture animation and simulation/procedural animation
3. Creative information visualization
4. Automated creation of information and scientific visualizations
5. Abstracting away from reality
6. Display more pixels
7. Display fewer pixels
8. Unified graphics architectures
9. User interfaces for 3D creativity
10. Truly immersive virtual reality

Problem 3 is particularly relevant to research on visual interfaces of digital libraries. Foley defined information visualization as "creating representations of non-geometric information by adding geometry to the information." Foley had data warehousing in mind when he listed this problem, but this is equally valid for digital libraries. Information visualization will grow in importance as digital libraries become more and more common. The content of a digital library does not always come with an inherent geometry. The major challenges are how to extract new and more complex types of relationships and visualize them so that they can make the contents of digital libraries more accessible and manageable to users.

Information retrieval is another topic relevant to this book. Visual interfaces aim to support retrieval as well as browsing. Croft in 1995 listed the top ten research issues for information retrieval [2]:

1. Efficient, flexible indexing and retrieval
2. Integrated solutions
3. Distributed information retrieval
4. Vocabulary expansion
5. Interfaces and browsing
6. Routing and filtering
7. Effective retrieval
8. Multimedia retrieval
9. Information extraction
10. Relevance feedback

Problem 5 is particularly relevant to our mission. Croft states in [2] that, "The interface is a major part of how a system is evaluated, and as the retrieval and routing algorithms become more complex to improve recall and precision, more emphasis is placed on the design of interfaces that make the system easy to use and understandable." Croft further points out that, "Interfaces must support a range of functions including query formation, presentation of retrieved information, feedback, and browsing. The challenge is to present this sophisticated functionality in a conceptually simple way." Until recently, this has been a relatively under-researched issue. However, Croft predicted this will change as more work in information visualization appears.

Hibbard, in May 1999, listed top-ten problems for visualization [3]. He grouped the problems into visual quality, integration, information, interactions, and abstractions. The ten problems are as follows:

1. Realistic visual displays
2. Integrated virtual reality and physical reality
3. Integration of visualization with networking, voice, artificial vision, computation and data storage
4. Optimal visual interactions
5. Visualization of high-dimensional numerical information
6. Visualization of non-numerical information
7. Direct manipulation with visualizations
8. Visual idioms for collaborative interactions
9. Abstractions for visualization and user interaction processes
10. Reconciliation of expressiveness and easy of use

A number of problems listed here are related to the design of visual interfaces.

3 Top-Ten Problems in Visual Interfaces of Digital Libraries

The two workshops held in 2001 and 2002 on Visual Interfaces to Digital Libraries generated lively discussions. The participants offered many good suggestions and valuable feedback (see section 2.1 on Socio-Technical Challenges). Here we have assembled an initial list of top-ten problems and outline what should be addressed by future research in this area.

1. Theoretical Foundations

Research in visual interfaces of digital libraries as a whole lacks solid theoretical foundations. Although principles for perception and cognition, computer graphics, and human-computer interaction do exist, they do not readily lend themselves to formation of design principles. Many principles are tightly coupled with particular environments and are hard to generalize. More often, the same fundamental problem disguises itself in different forms, which also complicates the process of putting available theories into practice. Foundation works such as [4] are urgently needed. Theoretical contributions can significantly influence our practice.

The first problem is concerned with how we should build on the success of current? visual interfaces and explore promising application areas. Some exciting candidates include bibliometrics, scientometrics, knowledge tracking, and knowledge discovery. These areas have special requirements for their unique tasks and many have already developed their own approaches for handling complex information visually. Insights, experiences, and lessons learned from these fields are valuable sources of inspiration.

2. Empirical Foundations

This essence of this problem is, “to know where we are.” It is crucial to make clear what has been empirically proven to be useful and beneficial. What are the common elements that have been found in every successful example of visual interfaces of digital libraries? Which features have worked well in some cases, but not others? Which features so far have shown no conclusive benefits? Examples in areas such as visual information retrieval, visual information exploration, and empirical studies of information visualization [5] are likely to provide valuable clues. New methodologies and taxonomies of exemplar systems should be carefully considered.

3. Scalability

Digital libraries often face problems of scalability because of their varying sizes and contents. Will the algorithms and solutions that have been tried and worked on small-scale digital libraries break down in large-scale digital libraries? Researchers in information visualization are striking for faster responses, incremental updates, and a scale-proof layout performance. Computing and data processing power is continually growing more powerful, as the volumes of the data we need to handle increase. Visual scalability is the capability of visualization tools to display large datasets effectively, in terms of either the number or the dimension of individual data elements [6]. Interfaces that can handle multiple scale representations are among the most promising solutions, e.g., zoomable user interfaces [7].

4. Labeling

The problem of labeling visual interfaces may be divided into two areas: displaying readable labels and selecting meaningful labels. When numerous objects are displayed in a visual interface, one has to selectively choose the objects that get labeled. Some objects should be labeled prominently, some should be labeled moderately. Still others should not be labeled at all. Making meaningful labels is equally challenging if not more challenging than making readable labels. However, natural language processing and automated indexing may provide useful techniques for this purpose. Users’ interventions as part of the labeling process should also not be completely ruled out. A related question is, “meaningful to whom?” To create quality labels, one may need to take into account the nature of tasks and characteristics of the users of a particular digital library.

5. Individual Differences

The fifth problem concerns individual differences and how visual interfaces should accommodate such differences. In reality, one size can hardly fit all. One user's favorite visual interface may be another user's nightmare. Individuals have relatively stable cognitive preferences and abilities that can be measured by psychometric tests. For example, spatial ability indicates an individual's ability to recognize and handle spatial relationships of objects [8, 9]. Research in human-computer interaction has shown that individual differences can be the most significant factor in one's performance.

6. Supporting Collaborative Work

Given the individual differences we need to accommodate and the diversity of social norms in cyberspace, supporting collaborative work is a challenging task in its own right. Translation of collaborative work into a visual process entails overcoming a variety of obstacles. For instance, how should we represent participating parties through a visual interface? How should we integrate social structures with the organization of the underlying digital library? How should we evaluate whether a particular visual interface is useful in a collaborative setting?

7. Benchmarking and Standardization

The provision of commonly accessible and comparable test collections has been proven useful in several fields, especially test collections in information retrieval and associated text retrieval conferences (e.g., TREC at http://trec.nist.gov). A test collection must be similar to a real digital library in terms of size and content. At the same time, it must be simple enough to provide a sound base for performance analysis. Standardization is a closely related issue, which involves more efficient and effective integrations of services, protocols, and interfaces at various levels. The lack of benchmarks and standards in part contributes to the current suboptimal situation.

8. Evaluation

Evaluation is an integral part of the design and development of visual interfaces as well as digital libraries as a whole. Evaluative studies are needed to find out what has worked/is working? for both users and designers?. The development of research methods may benefit from the huge literature in human-computer interaction. This problem is also closely related to problem 7. Common test collections are necessary to make evaluative studies more comparable in terms of the strengths of evidence and conclusions.

9. Personalization

Pro-active, customized, and personalized information delivery is an increasing trend in digital libraries. Visual interfaces are in a good position to organize and reorganize the way an underlying digital library is presented to a client, for instance, tailored according to the client's background and access history. This is also related to collaborative support, or collaborative recommendation. The online bookstore Amazon.com provides a collection of recommended books for each user based on the titles purchased by the user and what others bought along with those same titles. While personalization and individual differences are a pair of closely interrelated problems, they are markedly different. Individual differences emphasize the types of abilities that are relatively stable, whereas personalization is driven by what information is needed. Although individual differences could be a factor, personalization focuses more on the type of information, the content, rather than the style.

10. Standardization

Modularization and standardization of digital library and information visualization services will save valuable resources and prevent designers from constantly "reinventing the wheel"[10]. Interfaces which link data storage and representation formats need to be standardized (see previous chapter in this book). As well, interoperation ability and cross-platform performance can be considerably improved.

We conclude this book with these challenging problems and hope that they will stimulate the creation and examination of more "top-ten" problems for this young, interdisciplinary field.

Acknowledgements

Special thanks to Stephen Eick and Sougata Mukherjea for their valuable feedback on an earlier version of the chapter.

References

1. Foley, J., *Getting there: The ten top problems left.* IEEE Computer Graphics and Applications, 1999.
2. Croft, W.B., *What do people want from information retrieval?* D-Lib Magazine, November 1995 (http://www.dlib.org/dlib/november95/11croft.html).
3. Hibbard, B., *Top ten visualization problems.* SIGGRAPH Newsletter, 1999. **33**(2).
4. Furnas, G.W. *Effective view navigation*. in *CHI '97*. 1997. Atlanta, Georgia: ACM Press.
5. Chen, C. and M. Czerwinski, *Empirical evaluation of information visualizations.* International Journal of Human-Computer Studies, 2000. **35**(5): p. 631-635.
6. Eick, S.G. and A.F. Karr, *Visual scalability.* Journal of Computational Graphics and Statistics, 2002. **11**(1): p. 22-43.
7. Hightower, R.R., et al. *Graphical multiscale Web histories: A study of PadPrints*. in *9th ACM Conference on Hypertext and Hypermedia (Hypertext '98)*. 1998. New York, NY: ACM Press.
8. Chen, C. and M. Czerwinski, *Spatial ability and visual navigation: An empirical study.* New Review of Hypermedia and Multimedia, 1997. **3**: p. 67-89.
9. Chen, C., M. Czerwinski, and R. Macredie, *Individual differences in virtual environments: Introduction and overview.* Journal of the American Society for Information Science, 2000. **51**(6): p. 499-507.
10. Börner, K. and Y. Zhou. *A Software Repository for Education and Research in Information Visualization*. in *Fifth International Conference on Information Visualisation*. 2001. London, England: IEEE Press.

Author Index