IEEE Visualization Conference,
IEEE Information Visualization Conference and
IEEE Symposium on Visual Analytics Science and Technology

October 28 - November 1, 2007
Sacramento, California, USA

Conference Compendium

SPONSORED BY IEEE COMPUTER SOCIETY VISUALIZATION AND GRAPHICS TECHNICAL COMMITTEE

IEEE Information Visualization Conference

Posters
Chairs: Kwan-Liu Ma and Martin Wattenberg

Doctoral Colloquium
Chair: William Pike

IEEE Symposium on Visual Analytics Science and Technology

Doctoral Colloquium
Chair: William Pike

IEEE Visualization Conference,
IEEE Information Visualization Conference and
IEEE Symposium on Visual Analytics Science and Technology

October 28 - November 1, 2007
Sacramento, California, USA

InfoVis Posters

Interactive Poster: Name That Cluster – Text vs. Graphics

James Abello† Hans-Jörg Schulz‡ Benoit Gaudin+ Christian Tominski‡

DIMACS, Rutgers University University of Rostock University College Dublin University of Rostock

ABSTRACT

Given a user query, search engines generally return a very sizeable collection of possible answers. Clustering has been proposed as a tool to partition the possible answer set into more manageable subsets of related results. There is no current agreement on the preferred mode of presentation of these clusters. Currently, most search engines display the set of results in an almost purely textual form. However, relatively recently we have witnessed some timid attempts to use some graphical representations. To elucidate when and why text appears to outperform graphics for certain fundamental clustering related tasks, this work presents a preliminary user study with three interfaces to display flat clusters of user queries.

Keywords: applications of information visualization, graph visualization, user studies.

1. THE SETUP

When deciding on how to present a partition of clustered search engine results to the user, there are basically three general choices: a pure textual interface or a graphical interface – or, since each of these two choices has its Pros and Cons, maybe even a combination of both with some graphical and some textual elements. The ongoing user study [2] is an attempt to provide some directions to support this design decision. Other works like [4] aim to incorporate user feedback to help determine the criteria influencing clustering quality, but our emphasis is not on any clustering technique but rather on the presentation mode and associated interaction methods. In that respect, our work is more related to [2] since it considers graphical techniques to present textual clustering. However, our study also includes pure textual representation and clustering related tasks.

Since we wanted to mimic as close as possible the scenario of a user's query web search without encumbering the study with web page content descriptions or extra URL clicks we settled for presenting the user with sets of related web search queries instead of web page descriptions. Namely, after choosing an input query from a scrolling query list the user's task is to *Select*, *Explore*, and finally *Grade and Name* a set of query clusters offered as related to the input query.

In the course of the study, we collect the user's response as well as the time used to explore, rate and name the clusters. This allows us to correlate the different interface types with the measured performance, cluster sizes and user responses. An interface could then be considered better, when it outperforms another interface in terms of completion time or completeness of the answers (less unnamed or ungraded clusters).

2. INTERFACE DESCRIPTIONS

Even though, different approaches and techniques have been used to implement the different interfaces, close attention was paid to make the interfaces as functionally similar as possible regarding the used color schemes or the evaluation.

† abello@dimacs.rutgers.edu

+ benoit.gaudin@ucd.ie

‡ {hjschulz, ct}@informatik.uni-rostock.de

The **textual interface** (Figure 1) presents clusters of queries as a scrollable list. It is implemented as a DHTML/JavaScript web page. The clustered queries associated with an input query are displayed in rows on the screen, together with the form fields needed to evaluate each cluster.

The **graphical interface** (Figure 2) represents queries as round, marble-like items scattered across a 2-dimensional canvas (call them graphical query-items). Elements of the same cluster are positioned close to each other and a polygonal frame is drawn around each cluster. The graphical query-items are annotated with textual labels using a level-of-detail approach that shows more labels depending on the level of zooming being applied to a cluster. This interface has been implemented in JAVA/OpenGL and it reuses some of the mechanisms offered by the CGV-platform [1]. The described graphical setup is in our view a logical extension of the 1-dimensional, list-like display used in the textual interface.

We also offer a third **hybrid interface** (Figure 3) that incorporates in our view the "best" of the textual and the graphical interfaces. Concretely, the clusters are now presented both in textual and graphical form. These views are linked with each other in a coordinated way using a Model-View-Controller Pattern offered in the CGV platform. Thus, navigational changes from inter- or intra-cluster exploration are reflected by both views in a coordinated way. This way, the interface indicates both: the current position of the item the user is currently viewing in the textual representation, and corresponding visual information on the cluster appearing in the graphical component. For the latter, the density of a cluster is visible at a glance.

3. HOW TO PARTICIPATE

There are two ways to help our effort to compare the described interfaces:

- There will be an opportunity to test the three interfaces by joining our user study at the poster presentation during the InfoVis conference.
- We are now working on extensions of our experiment set up to a web-based platform. This will enable us to reach out to a larger and broader set of participants. The URL of this web user study will be made available at the InfoVis conference.

Since the web experiment is expected to draw a large number of participants, we plan to apply classical statistical analysis to this "vox populi". Such a statistical approach applied to results obtained from a large number of participants could make our results robust to "unusual behaviors" of just a few participants.

REFERENCES

[1] J. Abello et al, CGV – Coordinated Graph Visualization, Interactive Poster, InfoVis 2007

[2] M. Ghoniem et al, A Comparison of the Readability of Graphs Using Node-Link and Matrix-Based Representations, InfoVis 2004

[3] J.E. Swan II, Experimental Design and Analysis for Human-Subject Visualization Experiments, IEEE InfoVis 2006 Tutorial.

[4] C. Ware, et al, Cognitive measurements of graph aesthetics, Information Visualization, Vol.1 No.2, 103-110, 2002.

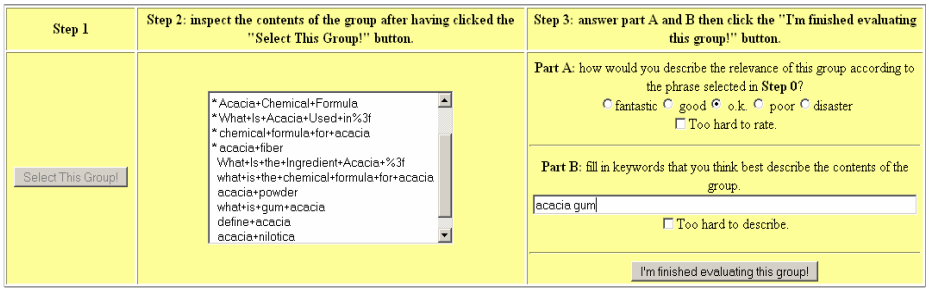


Figure 1: The textual interface.

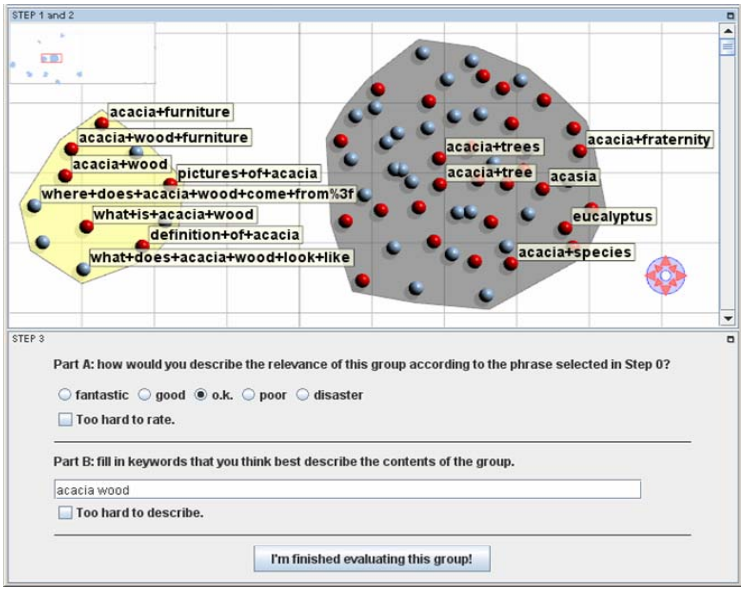


Figure 2: The graphical interface with one cluster already evaluated (gray) and one currently being in the process of evaluation (yellow).

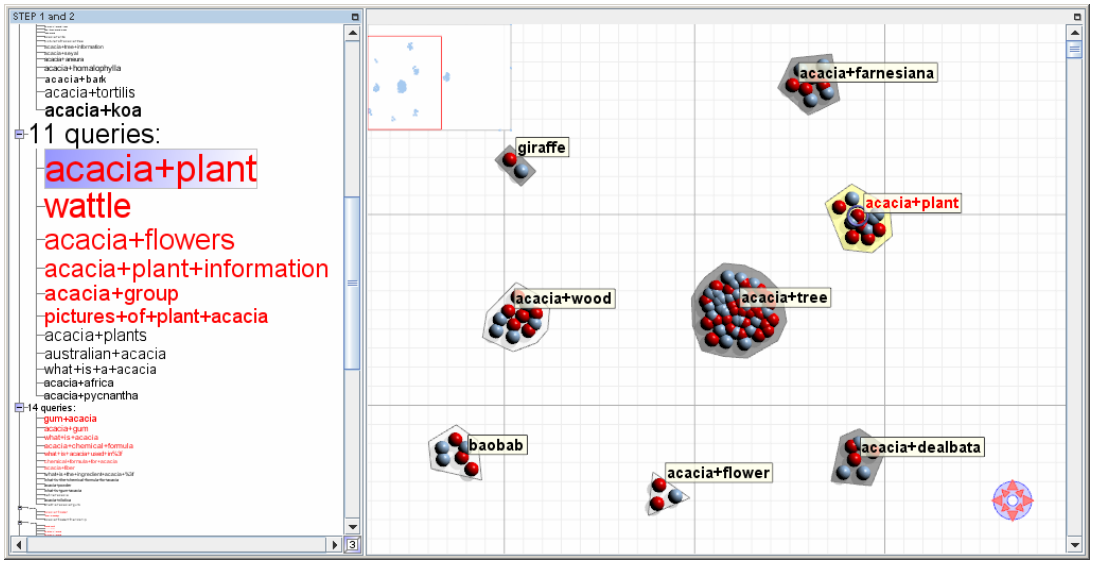


Figure 3: Hybrid interface showing textual and graphical components. The flat cluster list is visible on the left; the graphical representation is at the right. Both are linked, so that the selection of an item in either one of them is reflected in both views – the highlighted item in the tree view corresponds to the highlighted item in the graphical cluster. User evaluations are performed through an interface identical to the one presented in Figure 2.

Interactive Poster: Exploration of the 3D Treemap Design Space

Hans-Jörg Schulz* Martin Luboschik† Heidrun Schumann‡

University of Rostock, Germany

ABSTRACT

Inspired by Venn diagram layouts, the Treemap [6] is one of the most prevalent implicit tree visualization techniques. Ever since its publication, it has been modified and extended in many ways. This work presents a way to generate 3-dimensional Treemap visualizations by a 4-step procedure. It can be used for rapid prototyping and comparing different 3D Treemap layout approaches, to devise user studies on 3D Treemap layouts or for educational purposes.

Keywords: tree visualization, 3D Treemap, implicit graph layout

1 INTRODUCTION

The original Treemap [6] is a **2D**, **implicit** layout technique that uses **containment** of **rectangles** to indicate parent-child-relationships. These rectangles are aligned **parallel to the axes**, alternating between horizontal and vertical layout (**Slice and Dice**). Over the years, researchers have modified and extended the original Treemap with regard to all of these characteristics:

- the dimensionality has been extended to 3D, i.e. as in Steptrees [4] or Treecubes [10]
- the implicit edge representation has been partially modified to explicitly drawn edges in Elastic Hierarchies [13]
- the containment relationship has been substituted by overlap in the Beamtree technique [11]
- the used graphics primitives have been changed from rectangles to circles [12] and convex polyhedra [2]
- the alignment to axes has been turned into radial arrangements (as in Pietrees [8])
- the layout mode has been enhanced from the original Slice-and-Dice method [6] to other techniques like Squarified Treemaps [5] and Quantum Treemaps [3]

All of the above characteristics can be changed in combination (e.g., using overlap of circles in a radial arrangement), yielding a large number of possible Treemap configurations. Or framework is the first to provide the means to systematically explore this vast set of Treemap techniques. In this work, we focus on 3-dimensional Treemaps that completely rely on implicit edge representations. This pretty much fixes the first two items in the above list, but leaves the others to be freely combined within our framework.

2 A NEW 3D TREEMAP IN 4 STEPS

When devising a 3D Treemap configuration from within our framework, its parametrization is done in four steps:

*e-mail: hjschulz@informatik.uni-rostock.de
†e-mail: luboschik@informatik.uni-rostock.de
‡e-mail: schumann@informatik.uni-rostock.de

1. **Specify the containment relationship:** In this step, the user specifies how the parent-child-relationship should be encoded in the implicit representation. There are three possible choices: containment, adjacency and overlap. All of them are exemplified in Figure 1 using cuboids as graphics primitives.

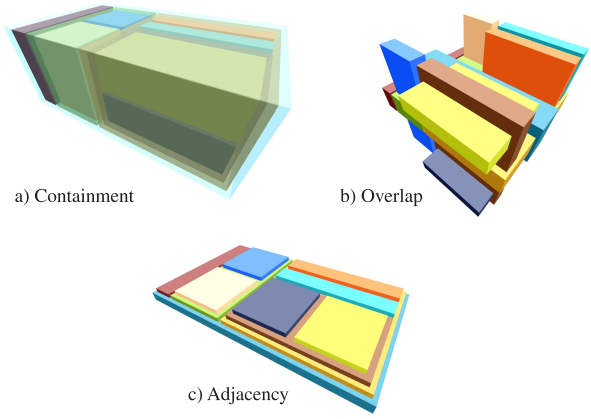


Figure 1: Implicit edge representations exemplified.

2. **Choose the graphics primitive:** After deciding about the edge representation in the first step, this step is about specifying the 3-dimensional representation of nodes. Primitives that have been used for 3D Treemaps up to now are cuboids [4, 10], cylinders [11, 12] and frustums of pyramids [1]. Yet, as shown in Figure 2, other graphical primitives like spheres can be imagined. Hence, it makes no sense to consider only a fixed set of graphics primitives, instead primitives should be provided to the framework in a plug-in manner.

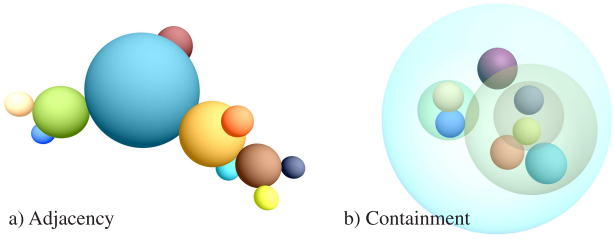
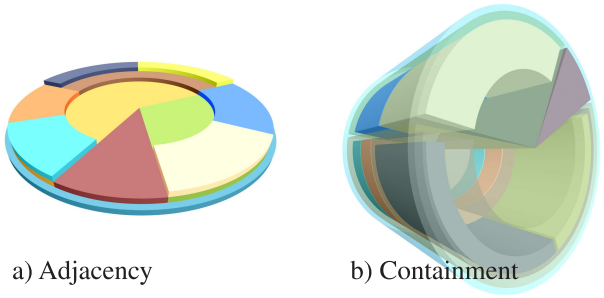


Figure 2: Two possible 3D Treemap configurations that use spheres as graphical primitves.

3. **Select a layout method:** Layout methods basically describe how the available space is distributed among the leaves of the hierarchy. While basic strategies like "Slice and Dice" or "Sphere Packing" are available by default, more sophisticated techniques can be realized using the plugin concept as the need arises. This flexibility allows users to create and experiment with new techniques that use alternative or uncommon layout methods. Examples for such are depicted in Figure 3.



a) Adjacency b) Containment

Figure 3: Two possible 3D extensions of the Pietree technique [8], in which radial alignment is combined with an axes-parallel stacking of the used primitives resulting in a hybrid alignment approach.

4. **Decide upon its alignment:** This step specifies the alignment of the layout. It is dependent on the layout method and provides an alignment parameter to it. This dependency is inevitable, as not all layout techniques can easily be used for both, axes-parallel and radial arrangement. Furthermore, besides configurations that are entirely axes-parallel (like those in Figure 1) and configurations that are entirely radial (like the ones depicted in Figure 2), there exist hybrid alignment approaches (like the ones depicted in Figure 3).

3 Our Framework

Our framework provides an interface that allows users to specify 3D Treemap configurations according to the 4-step specification process described above. Then, one or more specified configurations can be used to visualize a given hierarchy. The framework can be used to view different 3D Treemap configurations side by side to allow for their simultaneous exploration. Additionally, a standard treeview is linked to these views to ensure that all nodes of the hierarchy can be selected easily, even if they are occluded or otherwise hard to pick from the 3D representation. Thus, highlighting a node in the treeview is propagated to all of the other views and highlights the very same node there. This mechanism is very useful for comparing new 3D Treemap prototypes with existing techniques. Moreover, comparing and contrasting with familiar techniques makes it easier for users to get accustomed to novel alternative designs.

Secondly, the described concept of a modular, stepwise parameterization of 3D Treemap configurations provides a new systematical basis for conducting user studies. It is possible to analyze the influence of the individual characteristics by keeping everything else fixed and altering only the parametrization of interest. This allows to investigate questions like "Which implicit edge representation technique is best suited for identification and comparison of data items in case of axes-parallel cuboids that are arranged in a Slice-and-Dice fashion?"

A prototype of the described framework will be available to the InfoVis-attendees for a hands-on demonstration at the poster desk. So far, it includes all edge representations, primitives, alignments and layouts that are necessary to specify most of the existing 3D Treemap configurations like Treecube [10] or StepTrees [4]. Since they are provided in the said modular fashion, also other possible combinations of these edge representations, primitives, alignment- and layout-strategies can be generated with the prototype.

4 Conclusion

With our framework we have developed a platform for rapid prototyping of 3D Treemap visualizations and their interactive evaluation. Different layout aspects have been singled out into a modular concept that allows to easily put together new Treemap configurations from known building blocks. This enables the user to go beyond the number of known 3D Treemap techniques by systematically exploring the range of possible layout combinations. That way, the user can really find the very technique that perfectly fits task and data, even if it has not been described before. To further increase the number of choices for the user, future work will include the adaptation of our framework to 3-dimensional extensions of other well-known implicit techniques like Sunburst [9] and Icicle Plot [7].

References

[1] K. Andrews, J. Wolte, and M. Pichler. Information pyramids^{TM}: A new approach to visualising large hierarchies. 1997.

[2] M. Balzer and O. Deussen. Voronoi treemaps. In *InfoVis 2005*, pages 49–56, 2005.

[3] B. B. Bederson, B. Shneiderman, and M. Wattenberg. Ordered and quantum treemaps: Making effective use of 2d space to display hierarchies. *ACM Transactions on Graphics*, 21(4):833–854, 2002.

[4] T. Bladh, D. A. Carr, and J. Scholl. Extending tree-maps to three dimensions: A comparative study. In *APCHI 2004*, pages 50–59, 2004.

[5] M. Bruls, K. Huizing, and J. van Wijk. Squarified treemaps. In *Data Visualization 2000*, pages 33–42, 2000.

[6] B. Johnson and B. Shneiderman. Tree-maps: a space-filling approach to the visualization of hierarchical information structures. In *Visualization 1991*, pages 284–291, 1991.

[7] J. B. Kruskal and J. M. Landwehr. Icicle plot: Better displays for hierarchical clustering. *The American Statistician*, 37(2):162–168, 1983.

[8] R. O'Donnell, A. Dix, and L. J. Ball. Exploring the pietree for representing numerical hierarchical data. In *HCI 2006*, 2006.

[9] J. Stasko and E. Zhang. Focus+context display and navigation techniques for enhancing radial, space-filling hierarchy visualizations. In *InfoVis 2000*, pages 57–65, 2000.

[10] Y. Tanaka, Y. Okada, and K. Niijima. Treecube: visualization tool for browsing 3d multimedia data. In *IV 2003*, pages 427–432, 2003.

[11] F. van Ham and J. J. van Wijk. Beamtrees: Compact visualization of large hierarchies. In *InfoVis 2002*, pages 93–100, 2002.

[12] W. Wang, H. Wang, G. Dai, and H. Wang. Visualization of large hierarchical data by circle packing. In *SIGCHI 2006*, pages 517–520, 2006.

[13] S. Zhao, M. J. McGuffin, and M. H. Chignell. Elastic hierarchies: Combining treemaps and node-link diagrams. In *InfoVis 2005*, pages 57–64, 2005.

Visual Support for Exploration within Web Search Results Lists

Orland Hoeber*
Memorial University of Newfoundland

Xue Dong Yang†
University of Regina

ABSTRACT

The static list-based representation of Web search results remains essentially unchanged since the early days of Web search. This poster presents a simple and easy to understand visualization method to support users in the exploration of Web search results. A term frequency histogram provides a visual representation of the frequencies of the terms appearing in the title and snippet of the top search results. Users can interactively select positive and negative relevancy for terms in the histogram, resulting in the colour-coding of the corresponding terms within the search result list. This selection of terms also produces a re-sorting of the search results within the list, based on the use of the selected relevant and non-relevant terms. In addition to an interactive demo, results from a preliminary evaluation are presented in the poster.

Index Terms: H.5.2 [Information Interfaces and Presentation]: User Interfaces—User-centered design; H.3.3 [Information Storage and Retrieval]: Information Search and Retrieval—Search process

1 INTRODUCTION

The common method for representing Web search results is a list-based format in which searchers must consider each document one-by-one, and to some degree, in the order provided by the search engine. The interfaces used by the top Web search engines provide little or no ability to manipulate and explore the search results. Evidence of the challenges users encounter when dealing with Web search results have been brought to light by a number of studies focusing on the behaviour of Web searchers. In these studies, it was shown that users seldom venture past the third page of Web search results [5, 6].

When users' information needs are very specific and when they are able to craft an accurate query, it may be possible to find the answers they seek among these first three pages of search results. However, in many situations, there may be a high degree of vagueness within users' information needs. It is in these situations that an exploration tool is very valuable in supporting users in their tasks of finding relevant documents among many non-relevant documents in the search results lists.

As noted by Nguyen and Zhang [3], "Web search result visualization is not merely a simple way of information presentation, displaying results for a query. It also provides an interactive environment for users to explore, discover, and analyze information." Our research follows this same approach, providing tools and features that allow the user to take an active role in the Web information retrieval process.

The system described in this poster, WordBars2, builds upon our previous research on using term frequency histograms to support both interactive query refinement and interactive search results exploration [2]. Although WordBars2 continues to support query refinement processes, the focus here is on the new features which support visualization and exploration within the search results lists.

*e-mail: hoeber@cs.mun.ca
†e-mail:yang@cs.uregina.ca

2 WORDBARS2

2.1 Term Frequency Histogram

WordBars2 employs the services of the Google API to retrieve the top 100 document surrogates for a given user query. As the set of search results are retrieved, the *title* and *snippet* from each of the document surrogates are combined in a bag-of-words approach producing a document descriptor text string. Commonly used terms, as well as terms that are less than three characters long are ignored. All other terms are reduced to their root forms using Porter's stemming algorithm [4]. The frequency of each unique stem in the document descriptor is counted, the outcome of which is a term frequency vector corresponding to document surrogate ds_i:

$$\vec{F_i} = (f_{i1}, f_{i2}, \cdots, f_{in})$$

where f_{ij} is the frequency of the stemmed term j within the document surrogate ds_i, and n is the total number of unique stems appearing in all the document surrogates processed.

A single master vector $\vec{M}$ is generated to represent the sum of the frequencies of the stems over the entire set of search results obtained. Sorting this vector results in the terms that are used most frequently appearing at the top, and those that are seldom used appearing at the bottom. Providing a visual representation of the top end of the sorted master vector gives users an impression of the relative frequencies of the commonly used terms within the title and snippet of the top search results. The result is that users can *see* this frequency information without having to read it.

A vertically-oriented, colour-coded histogram is employed for this purpose (see Figure 1). Both the sizes of the bars in the histogram, as well as the intensities of the colours, represent the frequencies of the commonly used terms in the top search results. Features of the opponent process theory of colour [1] were used in the selection of a colour scale, which varies on the yellow-blue colour channel, as well as the luminance channel. As a result, frequently appearing terms are represented using large, dark blue bars; infrequent terms are represented using small, light yellow bars.

Term labels are provided to the right of each frequency bar. All the terms that are present in the query are coloured orange; all others are dark grey. This use of colour allows users to easily identify their query terms within the histogram. Due to space considerations, only the 40 most frequently used terms are displayed in the term frequency histogram. While there may be relevant terms beyond this cut-off mark, we assume that the most beneficial terms are those that are used frequently within the title and snippet of the top search results.

2.2 Term Highlighting Within the Search Results List

An important feature within WordBars2 is the ability to highlight relevant and non-relevant terms within the search results list. To the left of each element in the term frequency histogram are two icons. The positive icon (in the shape of a plus sign) is used to indicate that the corresponding term is relevant to the user's information need. The negative icon (in the shape of a minus sign) is used to indicate that the corresponding term is not relevant.

Initially, these icons are presented as subtle outlines (following Tufte's principle of *smallest effective difference* [8]). When a user clicks on one of the icons, the outline icon is replaced with a full-colour icon. Shades of green and red were selected to represent the

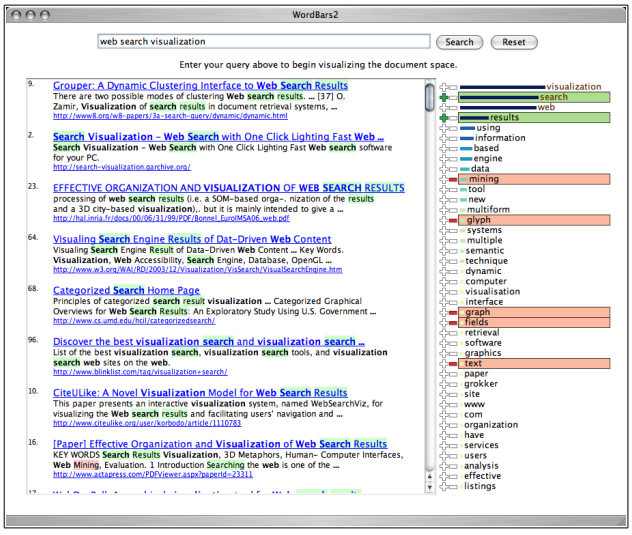


Figure 1: WordBars2 provides users with the ability to select relevant (green) and non-relevant (red) terms from the term frequency histogram, resulting in colour-coding of the terms in the search results.

positive (relevant) and negative (non-relevant) icons, respectively. Not only do these colours fall within a different colour channel than those used by the term frequency histogram, they also can be decoded into positive and negative colours based on a traffic light metaphor.

In addition to indicating the selected terms with the colour icons, the background of the term within the term frequency histogram is highlighted using a low-luminance version of the same colour (i.e., low-luminance green for the relevant terms, and low-luminance red for the non-relevant terms). The luminance on these colours is reduced in order for there to be sufficient luminance contrast between the background colour and the font colour.

Within the search results list, every term which has the same root as the selected terms is also highlighted with the corresponding positive or negative colour. The luminance of these colours is reduced even further than those used in the term frequency histogram. The goal was to avoid a "visual war with the heavily encoded information" [7], yet still effectively communicate to users the locations of the selected terms within the search results list. The inspiration for using this technique in WordBars2 comes from a similar highlighting technique used in the "find" feature in the Firefox Web browser.

The end result is that as users select a relevant term by clicking on the corresponding positive icon, the icon is switched to its "on" state, the term in the histogram is highlighted in green, and all the instances of terms with the same root are highlighted green within the search results list. A similar process holds for the selection of negative terms, which are highlighted in red. An example showing the selection of both relevant and non-relevant terms is provided in Figure 1.

2.3 Search Results Re-Sorting

Although the original WordBars system supported search results re-sorting, this method has been extended in WordBars2 to reflect the selection of both the relevant (positive) and non-relevant (negative) terms from the term frequency histogram. As users select relevant and non-relevant terms, a selection vector $\vec{S}$ is generated:

$$\vec{S} = (s_1, s_2, \cdots, s_n)$$

where n is the total number of unique stems that are present in the search results set processed, and the selection index s_i is given by:

$$s_i = \begin{cases} 1 & \text{if term } i \text{ is selected as a relevant term} \\ 0 & \text{if term } i \text{ is not selected} \\ -1 & \text{if term } i \text{ is selected as a non-relevant term} \end{cases}$$

The re-sorting of the search results is based on the dot product between the sort vector $\vec{S}$ and each term frequency vector $\vec{F_i}$: $sv_i = \vec{S} \bullet \vec{F_i}$. As a result of this calculation, each document surrogate within the search results is assigned a sort value sv_i. The search results list is re-sorted based on this value, in descending order. The outcome is that those document surrogates with a high sort value (i.e., containing relevant terms) are moved to the top of the list, and those with a low sort value (i.e., containing non-relevant terms) are moved to the bottom of the list. Note that the speed of these calculations is fast enough to provide the illusion of an instantaneous re-sorting of the search results based on selections made in the term frequency histogram.

Clicking on any document surrogate will open the corresponding document in a new window, and will change the link colour from blue to purple (as per the de-facto standard for visited links in a Web page). This allows users to easily identify documents that have already been visited, even after the search results are subsequently re-sorted by the user.

3 Conclusion and Future Work

In the poster, the details regarding the visual and interactive features for Web search results exploration supported by WordBars2 will be presented (a video of which is available on the author's web site[1]). In addition to an illustrative example, an live demo of the system will be available for use. The results of a preliminary study on the use of WordBars2 will also be included, indicating that the visual and interactive features are both easy to understand and effective for exploring documents within Web search results lists.

A negative aspect of the visualization techniques used in WordBars2 becomes apparent when a user selects many relevant and non-relevant terms from the term frequency histogram. As more and more terms are selected, the colour-coded highlighting in the search results list moves from an aid in identifying the corresponding terms within the list, to visual noise within the interface. However, when only a few select terms are selected, and when the selections are made in an interactive and exploratory manner, the highlighting of the corresponding terms within the search results list can be a very valuable feature.

References

[1] E. Hering. *Outlines of a Theory of Light Sense (Grundzge der Lehr von Lichtsinn, 1920)*. Harvard University Press, 1964.

[2] O. Hoeber and X. D. Yang. Interactive web information retrieval using WordBars. In *Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence*, 2006.

[3] T. N. Nguyen and J. Zhang. A novel visualization model for web search results. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):981–998, 2006.

[4] M. Porter. An algorithm for suffix stripping. *Program*, 14(3):130–137, 1980.

[5] C. Silverstein, M. Henzinger, H. Marais, and M. Moricz. Analysis of a very large Web search engine query log. *SIGIR Forum*, 33(1):6–12, 1999.

[6] A. Spink, D. Wolfram, B. J. Jansen, and T. Saracevic. Searching the web: the public and their queries. *Journal of the American Society for Information Science and Technology*, 52(3):226–234, 2001.

[7] E. Tufte. *Envisioning Information*. Graphics Press, 1990.

[8] E. Tufte. *Visual Explanations*. Graphics Press, 1997.

[1] `http://www.cs.uregina.ca/~hoeber/WordBars2/`

Hairograph: Synthesizing Statistics with Hair

Berkay Kaya*
Sabanci University

Can Çeçen†
Sabanci University

ABSTRACT

In this paper we refer to our perspective of representing the statistics about a country using the concept of human hair and its style which differs from person to person. We map demographics and socio-economics data sets to different properties of hair styles. Since national statistics involve many variables, a single hair strand plays an essential role on representing multivariate data sets. We picture the comparable differences between two different countries by stressing on their hairstyles.

Keywords: Data Visualization, Information Visualization, Statistics, Multivariate, Comparative

1 INTRODUCTION

National statistics enclose information about education, economy and the population. In our project, Hairograph, we try to represent these statistics about a nation by using hair strands as the feature for representing multivariate data. In general terms, Hairograph is an interactive tool for people who want to know about the statistics of a nation in a representation of a human head where each person or a group of people is shown by hair strands.

2 RELATED WORK

In the area of representing multivariate data sets, many different techniques are implemented. A commonly used method is combining different volumes into one visualized volume. A project uses the same technique by using contextual cues in order to create a visual volume for representing multivariate data [1]. In Hairograph, a hair strand is capable of visualizing different variables by simply modifying its default parameters such as color and length.

For the study of visualizing national statistics, Worldmapper, uses morphing techniques to demonstrate how a country evolves for the given parameter such as migration, population or tourism [2]. The land areas of all countries in the world map are morphed into new areas by regarding the statistics chosen. In our project, different hairstyles are generated for different countries regarding the selected statistics by using the combination of hair strands.

3 MOTIVATION

Ordinary graphs such as bar graphs or pie charts do not have much difference than merely reporting numerical comparisons about the distribution of the population with respect to the emphasized statistics. Moreover, they do not have artistic representation where a person may get influenced by only looking at these visuals. With Hairograph, a country is visualized as a human head where its hairstyle gives brief information about the highlighted statistics.

* e-mail: berkayk@su.sabanciuniv.edu
† e-mail: cancecen@su.sabanciuniv.edu

Our main motivation is the similarity of the role of a single person or a group of people which has a certain homogenous characteristic in the society and the contribution of a hair strand or bunch of hair to the whole hair style. Although only one hair strand may not change the hairstyle, the combination of more hair strands may significantly affect the hairstyle. Therefore, hair style is a powerful differentiating feature among people, which can also be implied to national differences. Moreover, hair is an effective feature for multivariable representation since every hair strand carries multiple properties such as branchiness, length and color.

3.1 Representative Variables

The first variable we used for hair strands is *branchiness*. Having more branches makes the individual look worse and vice versa. *Branchiness* is the most essential variable showing the corresponding statistics such that when comparing two different countries on economic issues, the country which has less branched hair strands is the country having the better economy and same goes for the opposite. In this feature, we designed 5 different scales from best to worst, which are also indicated at *Figure – 1*.

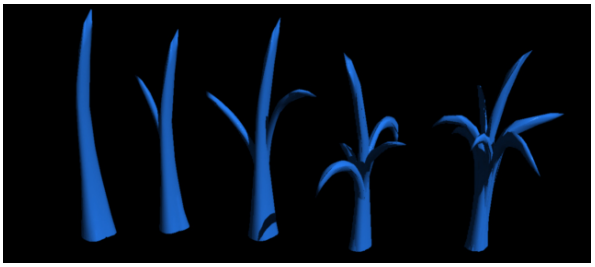


Figure 1. Five levels of differentiation with respect to branchiness

The second variable we use is *the length of the hair strands.* For representing people's ages, we use length as a variable where longer hair represents the older people and shorter hair stands for the young. Therefore, length will be used as a common variable which does not provide any information about the well-being of the statistics; however, it will give some information about the overall statistics like the distribution of old/young people in the country.

The third variable we use is *the color of hair strands*. We assign blue and pink as the color of hair strands where we follow the standard convention for representing sex.

As *the fourth parameter*, we chose the *density of hair among the head*. We employ the human head as a grid to our hair layout where density gives information about the population of the country.

Apart from all variables, we also have the distribution of the hair which is fixed in such a way that the longer hairs are placed on the back of the head, whereas shorter hairs are standing on front. This also gives the user the chance to compare two countries with regarding their younger and older population since

while locating the strands we consider the percentage of the young and old aged population.

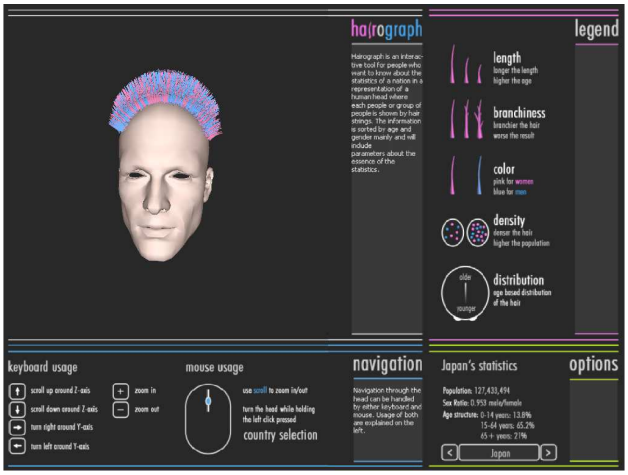


Figure 2. Snapshot from the application featuring Japan's education statistics

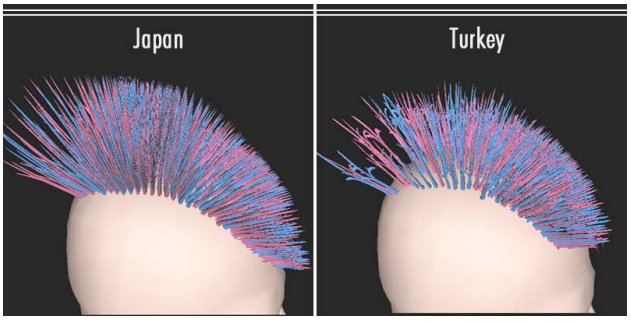


Figure 3. Snapshot showing the comparison between Japan and Turkey regarding their education statistics

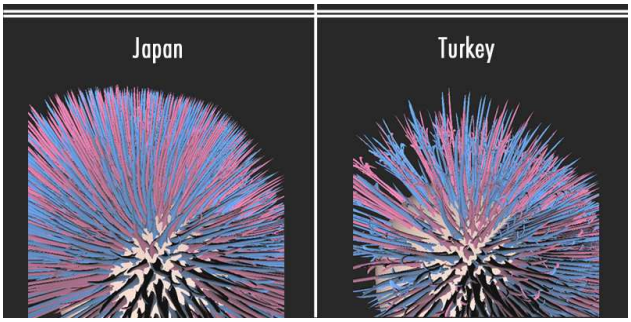


Figure 4. A closer look to Japan -Turkey comparison

3.2 Implementation

The application is implemented in OpenGL environment while the models are designed in Carrara software package as deformed three variable functions and imported as wavefront objects. For handling large sets of models, we use display lists to improve performance.

4 Discussion

Upon researching hair rendering, we gave up the idea of realistic hair models since we agreed that it would not be as informative as a data visualization application would require to be.

In different branchiness levels, the number of branches is not increasing linearly to ensure the different hair strand quality is perceived.

4.1 Difficulties Faced

We were planning to distribute the strands over the head with respect to the geographical properties of the country; however, it would not give satisfactory results because of several reasons.

First of all, the geographical distribution does not provide any insight.

Secondly, it would require unnecessary effort to find an operator for projecting a complex geographical structure to human head and where it would not convey meaningful information for the user. Hence, we agreed that we should place the younger in the front of the head while placing the elder in the back of the head considering the fact that human hair is generally longer on the back of the head. For informative reasons, we did not want to leave any space on the grid so we used density in the real sense, eliminating some columns in the grid while trying to represent a lighter population.

4.2 Further Improvements

For further developments, the first issue to be handled can be the distribution of the hair. Instead of just placing hair strands on the top of the head, improvised version can place the hair on all places of the head.

Secondly, a trustful data source can be found for getting better results. Instead of just relaying on two countries like Japan and Turkey, many countries can be compared with their unique hairstyles.

5 Conclusion

In this project, which is implemented for the fulfillment of an undergraduate course, *CS 450: Arts and Computing*, at Sabanci University, we aimed to view data visualization in a different aspect. In our research phase, we analyzed present data visualization projects over the web and many of them were simply graph representations of different data. We thought that it would be interesting to look at the concept from a different artistic perspective. With this motivation, we created Hairograph to highlight how small pieces of data serve for the big picture [3].

References

[1] J. Woodring. H. Shen. Multi-variate, Time-varying, and Comparative Visualization with Contextual Cues. In *IEEE Transactions on Visualization and Computer Graphics '06*, pages 909-916, 2006

[2] D. Dorling. A. Barford. M. Newman. WORLDMAPPER: the world as you've never seen it before. In *IEEE Transactions on Visualization and Computer Graphics '06*, pages 757-764, 2006

[3] Hairograph Video. http://graphics.sabanciuniv.edu/cs450-projects/hairograph/Hairograph.avi

CAT: A Hierarchical Image Browser Using a Rectangle Packing Technique

Ai Gomi*, Takayuki Itoh*, Jia Li**

* Graduate School of Humanitics and Sciences, Ochanomizu University

** Computer Science and Engineering, The Pennsylvania State University

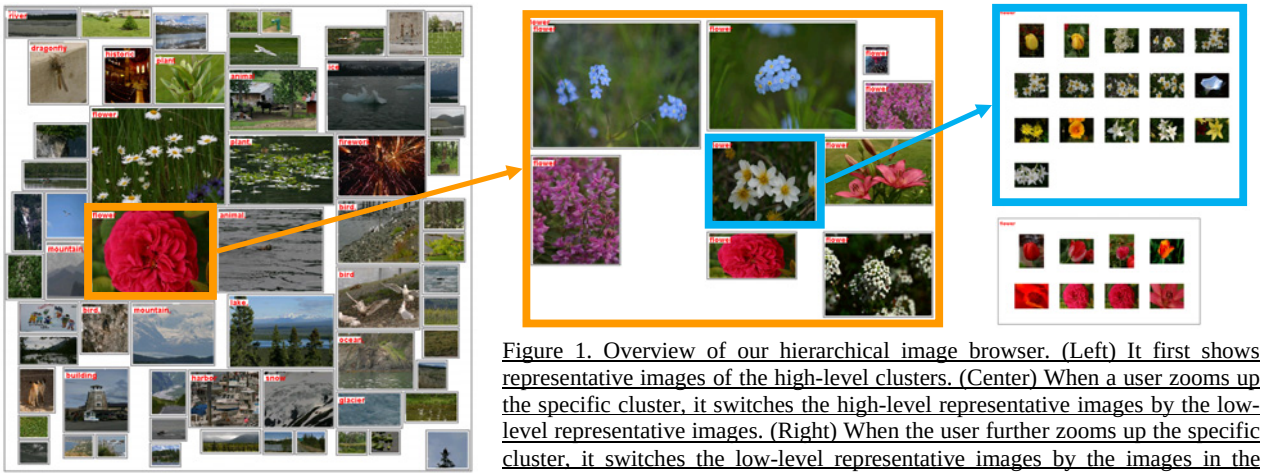


Figure 1. Overview of our hierarchical image browser. (Left) It first shows representative images of the high-level clusters. (Center) When a user zooms up the specific cluster, it switches the high-level representative images by the low-level representative images. (Right) When the user further zooms up the specific cluster, it switches the low-level representative images by the images in the clusters.

ABSTRACT

This poster proposes CAT (Clustered Album Thumbnail), a technique for browsing large image collections, and its interface for controlling the level of details. This new system clusters images according to their keywords and pixel values, and selects representative images for each cluster. It then visualizes the clusters by applying a hierarchical data visualization technique, which represents the hierarchical clusters as nested rectangular regions, and arranges individual images in grid layout inside the regions. Interlocking to the zooming operation, it selectively displays representative images by mapping onto the nested rectangular regions while zooming out, or individual images while zooming in. We argue that such an operation is friendly for users to explore and search for specific images from huge image collections, because the users are familiar with the Graphical User Interfaces (GUI) for file systems in a top-down manner. The poster also presents several evaluation methods for the technique.

CR Categories and Subject Descriptors: H.5.2 [User Interface]: Graphical User Interface (GUI); I.3.6 [Computer Graphics]: Applications.

Additional Keywords: image browser, hierarchical data visualization, rectangle packing, level-of-detail control.

1 INTRODUCTION

Technique for browsing larger collection of images is an interesting topic, and there have been novel works, such as PhotoMesa [1].

Here, we suppose that it is not always necessary to show all the images at the beginning of browsing, if we would like to browse thousands of images. We therefore focus on the development of a hierarchical image browser, which first shows the representative image of each image cluster, and then users can manually explore each cluster to see each image in the clusters. We focus on the hierarchical image browser for the following reasons.

We suppose that the adequate number of images to be shown in a display space depends on vision capability of users and resolution of displays, and in many cases it may be lower than a thousand. Hierarchical image browser is a good approach because it first shows a smaller set of representative images. User interface for hierarchy exploration seems intuitive for users, since they are already used to explore hierarchical information, typically by using file system browsers.

Another reason is performance. We do not like to spend a long time to load thousands of images from hard disk drives or servers at the beginning of browsing. If we first show only representative images, we can drastically save loading time.

This poster presents CAT (Clustered Album Thumbnail), a technique of hierarchical image browser. We assume the collection of images is large and each image is annotated by a few keywords. Our current implementation of CAT first constructs a two-level hierarchy of images; it first divides images according to distances of their keywords, and then divides again according to contents (colors and textures). CAT then selects representative images for each cluster of images. CAT selects images according to weights of keywords at the high-level clusters, and according to contents at the low-level clusters. CAT then represents the hierarchy by using our own hierarchical data visualization technique [2] which represents the hierarchy as nested rectangular regions.

When CAT completes the calculation of layout of the rectangular regions, it first shows representative images of the high-level clusters by mapping them onto the rectangular regions of the clusters, as shown in Figure 1 (Left). When a user zooms up the specific cluster, CAT switches the high-level representative

{gomiai, itot}@itolab.is.ocha.ac.jp, jiali@stat.psu.edu

images by the low-level representative images, by mapping them onto the rectangular regions of the clusters, as shown in Figure 1 (Center). When the user further zooms up the specific cluster, CAT switches the low-level representative images by the images in the clusters, as shown in Figure 1 (Right). It never spend a long time to load images, since it first loads only high-level representative images, and then loads the images in the zoomed up portion of the hierarchy.

Our hierarchical data visualization technique [2] is somewhat analogous to Quantum Treemap [3], since both technique represents leafs of trees as equally-sized icons. Our experimental tests [2] show the trade-off of the two techniques, where our technique is better in aspect ratio control and similarity among similar data. It is important to control aspect ratio of rectangular regions closer to the aspect ratio of representative images in our representation; that is the main reason why we use our own hierarchical data visualization rather than Quantum Treemap.

2 IMPLEMENTATION

2.1 Image clustering

CAT first divides images according to distances between their sets of annotation keywords, as described in [4]. It calculates distances of all possible pairs between two keywords, by using a software package WordNet Similarity to obtain similarity between two words. It then calculates the distances between their sets of keywords, from the distances between two words. It applies bottom-up linkage clustering to divide the images according to the distances. We call these clusters “high-level clusters” in this paper.

CAT then divides the images again, according to their pixel information. We simply calculates feature vectors from colors and textures of images, and applies bottom-up linkage clustering to divide the images according to cosine between the feature vectors. We call these clusters “low-level clusters” in this paper.

2.2 Representative Image Selection

CAT then selects representative images for each cluster. It calculates the sum of weights of keywords for each image, where weights are total number of images that have the keyword, for high-level clusters. It selects one of the images that have the highest sum value as the representative images for each high-level cluster. It then calculates the centers of feature vectors for each low-level cluster. It selects the images which are closest to the centers as the representative images for each low-level cluster.

2.3 Hierarchical Data Visualization

CAT then places a set of images onto a display space based on a bottom-up packing algorithm consists of the following three phases.

[Phase 1:] CAT first places a set of image thumbnails in a low-level cluster in grid layout, and encloses them by a rectangular border. It repeats this process for all the low-level clusters.

[Phase 2:] CAT then packs and encloses all the rectangles corresponding to the low-level clusters that belong to the same high-level cluster by a rectangular border. It repeats this process for each of the higher-level clusters.

[Phase 3:] CAT finally packs the rectangles of all the high-level clusters, and encloses them by a rectangular border.

Since CAT places representative images of clusters into the rectangular borders, aspect ratios of the rectangular areas should be as close as possible to the aspect ratios of the representative images. For this requirement, the technique controls the aspect ratio of the rectangular border as close as possible to the aspect ratio of the representative image of the cluster.

2.4 User Interface

CAT switches the displaying image according to wheel operation. While zooming out, it displays representative images of higher-level clusters. Zooming in, it switches to representative images of lower-level clusters, and finally to each image thumbnails. The representative images are displayed inside the rectangular borders representing the clusters. CAT stretches the representative images if the aspect ratios of the rectangular borders are not equal to those of representative images.

If the initial viewing configuration zooms out, CAT first loads only representative images from the hard disk drive into the main memory, and then loads each image thumbnails in the focused clusters on the fly, or frees memory space for image thumbnails in defocused clusters. This mechanism is effective for frame rate and memory usage.

3 EXPERIMENTS

We implemented and tested CAT on Windows XP, using an image collection which contains 2360 images of size 384x256 and stored in JPEG format. We measured aspect ratios of rectangular borders, and found that over 50% of them are between 1.2 and 1.4, where the ideal aspect ratio is approximately 1.3333.

We had the following experiments with 10 examinees, where all of them were female university students. First, the examinees played the following five variations of CAT for several minutes, and then evaluated them by ranking:

- no-cluster,
- low-level clusters with representative images,
- low-level clusters without representative images,
- high- and low-level clusters with representative images, and
- high- and low-level clusters without representative images.

We found from the statistics of the ranking that existence of representative images is very effective. In this experiment some examinees remarked that the significance of representative images is not only the level-of-detail control for displaying images at suitable sizes, but also computational efficiency in terms of both frame rate and memory usage.

Next we measured the time to search for specific images. We provided an image printed on paper to examinees, and they searched for the image from the collection displayed by the three variations of CAT. We measured the time to search for the image. We found that the time to search for the image did not improve by using “low-level cluster” version of CAT against "no-cluster" version of CAT very well. In other words, keyword-based clustering is very effective for image browsing using CAT. The result may suggest people to search for images based on semantics, rather than visual properties such as colors and textures. The "high- and low-level clusters" version therefore significantly improves over the other two versions.

REFERENCES

[1] Bederson B., B., PhotoMesa: A Zoomable Image Browser Using Quantum Treemaps and Bubblemaps, Symposium on User Interface Software and Technology, pp. 71-80, 2001.

[2] Itoh T., et al., Hierarchical Data Visualization Using a Fast Rectangle-Packing Algorithm, IEEE Transactions on Visualization and Computer Graphics, Vol. 10, No. 3, pp. 302-313, 2004.

[3] Bederson B., Schneiderman B., Ordered and Quantum Treemaps: Making Effective Use of 2D Space to Display Hierarchies, ACM Transactions on Graphics, Vol. 21, No. 4, pp. 833-854, 2002.

[4] Li J., A Mutual Semantic Endorsement Approach to Image Retrieval and Context Provision, ACM SIGMM International Workshop on Multimedia Information Retrieval, pp. 173-182, 2005.

A Synchronized Tag Cloud and Timeline Visualization

Joris Klerkx and Erik Duval

Abstract—In this paper, we present a synchronized Tag Cloud and Timeline Visualization, that is created to enable users and researchers to get an insight in the evolving patterns and focus of their interests while searching the network of repositories within the Global Learning Objects Brokered Exchange (GLOBE) consortium. The visualization is an interactive SVG drawing that is synchronized with timeline visualization by using JavaScript.

Index Terms—tag clouds, timeline, visualization, attention metadata, web search history

1 Introduction

We created a web application (GlobeMash [1]) that enables users to search the network of repositories within the Global Learning Objects Brokered Exchange (GLOBE) consortium [1]. Attention Metadata, a way to represent data about the activities of users in a certain environment [3], is tracked within GlobeMash to collect the search history of end users. In this paper, we describe a browser-based application to visualize the evolution of search keywords within GlobeMash. The goal of this application is to enable end users and researchers to get an insight in the evolving patterns and focus of their interests.

We start this paper with a description of why we want to visualize the web search history of users in GlobeMash. In section 3, we discuss the visualization in detail. Section 4 gives an overview of related work and we conclude this paper is section 5.

2 Web Search History

In order to create a feedback loop that enables learning from the way people actually use the search interface within GLOBE, it is essential to track the attention and behavior of users within GlobeMash. The tracking information is needed to answer questions like

- Which keywords are used how many times and could therefore be interpreted as representative keywords for a time period?
- Which keywords are used that have many or no results?
- Which keywords have results in multiple repositories?
- Which keywords or topics have the attention of a particular user over time?
- What are the behavior and the focus of the community within the GLOBE?
- How many results does a user look at in detail after performing a query?
- When did a user perform a query and which keywords did he or she use?
- What is the evolution of search keywords over time?
- Etc.

These questions are relevant for the end users and researchers that are interested to get an insight in the evolving patterns and focus of the users' interest, or to use the users' search history for e.g. recommendation purposes or to establish social networking opportunities. Every search event in GlobeMash is therefore logged in Contextualized Attention Metadata (CAM) [3]. A visualization of this data is created to enable users to get an insight in the questions above in a flexible, efficient and browser-based way.

- *Joris Klerkx and Erik Duval are with the Dept. ComputerScience, Katholieke Universiteit Leuven, Celestijnenlaan 200A, B-3001 Leuven, Belgium*
- *Email: {Joris.klerkx, Erik.duval}@cs.kuleuven.be*

3 Visualization

Our visualization is made up of two synchronized parts – an extended tag cloud and a timeline that indicates the different events that are tracked by the Attention Metadata and the current interval of time that is visualized in the tag cloud. These parts can be seen in Figure 1. In the next paragraphs we describe these parts in detail.

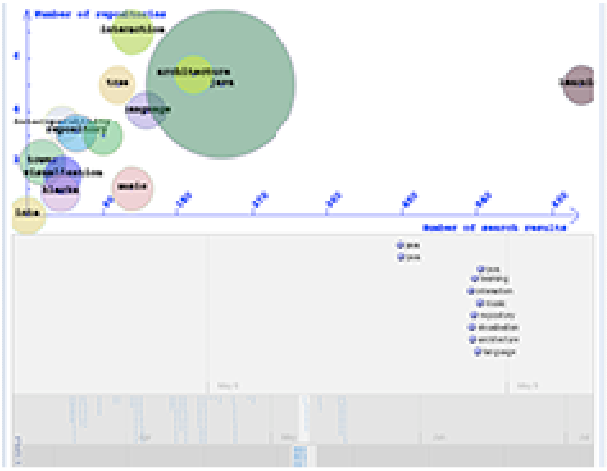


Fig. 1. A Synchronized Tag Cloud and Timeline.

3.1 Tag Cloud

Tag Clouds are visual presentations of a set of words, typically a set of tags, in which attributes of the text such as text, weight or color can be used to represent features (e.g. frequency) of the associated terms [4]. When a user clicks on a tag, the user obtains an ordered list of tag-described resources, as well a list of related tags [5]. Usually, the tags are displayed in alphabetical order.

In our visualization, every search keyword that is used in GLOBE is represented as a tag in the Tag Cloud. However, tags are not ordered alphabetically, but in a coordinate system where the X-axis represents the average number of search results and the Y-axis the number of repositories where results were found, both in relation to the performed keyword query. The scale of the x-axis and the y-axis are automatically computed from the data that has to be drawn in the visualization.

On top of that, the visualization offers the functionality to draw circles around a tag or keyword. The radius of the circle represents the frequency of the keyword used in GlobeMash over time. The color-coding of these circles can be random or it can be used to

encode the number of different users that actually used the keyword query, depending on the selection of the user.

This part of the visualization is a SVG drawing that is created with the Dojo JavaScript framework [6].

3.2 Timeline

The Tag Cloud gives an overview of all the keywords that are used in the networked repositories of GLOBE. In the bottom part of the visualization, a timeline presents the keywords at the moment they were issued. The timeline consists of three synchronized bands with different time scales: daily, monthly and yearly. The band with the daily scale shows the actual keywords. The other bands are meant to give an overview when a search event happened in GlobeMash.

This part of the visualization is created with the software that is created in the context of the SIMILE project [7].

3.3 Interaction

We synchronized the tag cloud and the timeline by using JavaScript. An end user can explore the visualization and therefore get insight in the questions of section 2 by either performing actions on the tag cloud or on the timeline. If a user clicks on a tag in our tag cloud, the timeline scrolls automatically to the latest timestamp that the keyword was used in a query. If a user clicks on a keyword in the timeline, a pop up is displayed with detailed information when the keyword was used, how many results there were and how many results were looked at in detail. If a user double-clicks on a keyword in the timeline, the timeline serves as a means to select an interval of time that should be visualized on the tag cloud. The tag cloud will filter out all keywords that were not performed in that specified interval.

3.4 Clutter

Obviously, if many queries were performed, it is possible that the tag cloud gets cluttered as it is possible that some tags overlap. In this section, we describe a number of countermeasures to avoid this problem.

A first and rather obvious way to avoid clutter is to give users the opportunity to select a small enough interval of time. A small enough interval of time means a smaller number of keywords or tags and therefore less chance of overlapping keywords. This can be done by using the timeline or by filling in a small electronic form with two dates that define the interval.

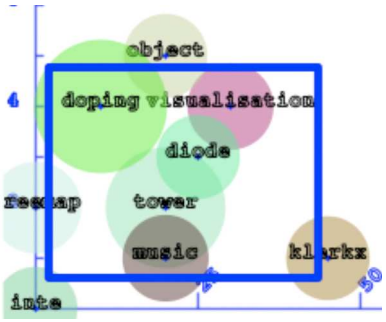


Fig. 1. Selection of an Area of Interest.

Secondly, the user is able to select an area of interest in the tag cloud after which the tag cloud is zoomed in at that area. Zooming in, has the advantage that the coordinate system can be rescaled and therefore overlapping keywords can be pulled away from each other. This can be seen in Figure 2.

A last way of reducing clutter is the notion of only showing "interesting" data [8]: only show the tags during a particular period of time that are most representative for that time period. This method is not yet implemented in GlobeMash but is certainly a worthwhile idea for future work.

4 Related Work

A Tag Cloud is a rather new interface paradigm that quickly gained popularity in Flickr [10] where it was necessary to find a way to represent vast amounts of information. Variations on tag clouds are presented in [11] where a circular form is used for a tag cloud. In [8] a visualization of tags over time is presented for the Flickr system. Finding the ideal tag cloud is a hot topic of interest, which is proved in [12] where the author starts a search for the "Perfect Tag Cloud". The history of the searches of users has been the interest of research for a long time. One example is Google Zeitgeist or its recent successor Google Hot Trends [9] that shows interesting patterns and queries of users in a list presentation.

5 Related Work

In this paper, we presented a synchronized Tag Cloud and Timeline Visualization to provide users and researchers the ability to get an insight in the evolving patterns and focus of their interests while searching the network of repositories within the Global Learning Objects Brokered Exchange (GLOBE) consortium. In the future, we will validate the usefulness and the efficiency of the visualization with log file analysis and usability tests with real users. We will also work further on creating clutter-free tag clouds as we discussed in section 3.4.

Acknowledgements

We gratefully acknowledge the financial support of the K.U.Leuven research council through the BALO project, the Interdisciplinary Institute for Broadband Technology (IBBT) through the Acknowledge project, and the European Commission through the ProLearn Network of Excellence on Professional Learning.

References

[1] "GlobeMash: A Mashup for Accessing GLOBE", Joris Klerkx, Erik Duval, accepted for publication at I-KNOW '07, KVD '07 special track on Knowledge Visualization and Knowledge Discovery Proceedings, September 5 - 7, 2007

[2] Global Learning Objects Brokered Exchange: www.globe-info.net/

[3] M. Wolpers, G. Martin, J. Najjar and E. Duval, "Attention Metadata in Knowledge and Learning Management", Proc. IKNOW'06, Graz, Austria.

[4] Martin Halvey, Mark T. Keane, "An Assessment of Tag Presentation Techniques", Poster paper, WWW 2007, Banff, Alberta, Canada

[5] Y. Hassan-Montero and V. Herrero-Solana. "Improving Tag-Clouds as Visual Information Retrieval Interfaces, Proc. InSciT 2006

[6] The Dojo Toolkit: http://dojotoolkit.org

[7] The SIMILE project: http://simile.mit.edu/wiki/SIMILE:About

[8] Dubinko, M., Kumar, R., Magnani, J., Novak, J., Raghavan, P., and Tomkins, A. 2006. Visualizing tags over time. In Proceedings of the 15th International Conference on World Wide Web (Edinburgh, Scotland, May 23 - 26, 2006). WWW '06. ACM Press, New York, NY, 193-202.

[9] Google Trends: http://www.google.com/trends

[10] Flickr: http://www.flickr.com

[11] Bielenberg, K. and Zacher, M. Groups in Social Software: Utilizing Tagging to Integrate Individual Contexts for Social Navigation. Masters Thesis, Universitat Bremen (2006).

[12] Hoffman, K, "In search Of … The Perfect Tag Cloud", white paper: http://dotnetaddict.dotnetdevelopersjournal.com/tw.htm

Judging Correlation from Scatterplots and Parallel Coordinate Plots

Jing Li and Jarke J. van Wijk, *Member, IEEE*

Dept. Mathematics and Computer Science, Technische Universiteit Eindhoven, The Netherlands

ABSTRACT

Scatterplots and parallel coordinate plots can both be used to find correlation visually [2][3][4]. In this paper, we compare these two visualization methods in two user studies. In the first study, 25 participants indicated the correlation they observed in plots with varying correlation of sample data, sample size, and timing condition. In the second study, the effect of training on parallel coordinate plots was studied. Results show that the degree of correlation is under-estimated, particularly in parallel coordinate plots, and that the observations based on parallel coordinate plots are less constant under different conditions. Moreover, training of parallel coordinate plots for moderate and low correlations might confuse users. Therefore, we conclude that scatterplots are better than parallel coordinate plots for visual correlation analysis.

Keywords: Evaluation of visualization, visual analytic task, correlation, Scatterplots, Parallel Coordinates, user studies.

1 INTRODUCTION

Scatterplots (S-plots) and parallel coordinate plots (P-plots) are techniques for the visualization of two dimensional and multi-dimensional data. S-plots are well-known and routinely used for 2D real-world cases. P-plots are popular in scientific research and particularly created and useful for multi-dimensional data. However, it is unclear which method better supports user analytic tasks [1]. An important task here is discovering correlations in multi-dimensional data.

We compare S-plots and P-plots for two dimensional data in the studies, since correlations of multi-dimensional data are just combined correlations of two-dimensional data. Statistical measures are used as the comparative baseline. Sample data are generated using two controlled parameters: the correlation of the sample data and the sample size. Another controlled parameter is the time of displaying the plots. Under different settings of these parameters, the observed correlations in S-plots and in P-plots are compared. The baseline of comparison is Pearson's correlation coefficient of the sample data. Beside that, for P-plots we also test the effect of training.

2 EXPERIMENT DESIGN

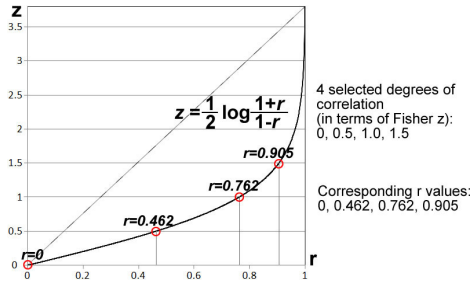


Figure 1. Fisher z transformation of Pearson's Correlation

j.li@tue.nl, vanwijk@win.tue.nl

In the experiment, we selected 7 different correlations. Our selection is based on the Fisher Z-transformation (z) of Pearson's correlation coefficient (r) (Figure 1). Fisher z is approximately normally distributed with mean r, which can reduce the skew of the sampling distribution of Pearson's r and stabilize variance.

The 7 different correlations in terms of z values are -1.5, -1.0, -0.5, 0, +0.5, +1.0, +1.5, which correspond to Pearson's r as -0.905, -0.762, -0.462, 0, +0.462, +0.762, +0.905. The sample sizes (n) used in the experiment are small (n=10), medium (n=40), and large (n=160). Figure 2 shows the resulting plots for different correlations, sample sizes and plot techniques.

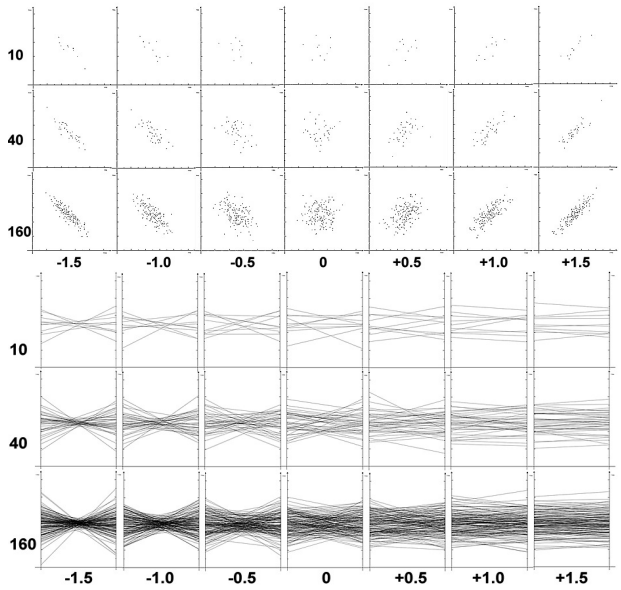


Figure 2. S-plots (top) and P-plots (bottom) with controlled correlations (in columns, defined by Fisher's z) and sample sizes (in rows).

The two time conditions are: limited, plots are displayed one second, and unlimited, plots are displayed as long as users want.

S-plots and P-plots are displayed under both time conditions resulting in 4 sessions in the experiment: ***S-l***, ***P-l***, ***S-u***, and ***P-u***. For each session, we use 42 data sets, generated by 2 random series of 21 pairs of z and n (7 correlations×3sample sizes=21 pairs). Users scored correlations (u) at five levels as strong negative, negative, not correlated, positive, and strong positive (u= –2, –1, 0, +1, +2).

Participants were given a small tutorial on both visualization methods before the test sessions. For each session, plots of the 42 data sets are displayed one by one in a random order. 25 Participants input their observed correlations into our test program. Each of them passes through all test sessions, which is a within-subject experimental design.

3 RESULTS

The influences of the three controlled parameters on users' inputs for S-plots and P-plots are compared for the following aspects:

- The possible linear relationship between ***u*** and *z*.
- The under-estimation of correlations.
- The agreement and variance among users.
- The difference of ***u*** for positive and negative correlations.
- The difference of ***u*** under the three values of ***n***.
- The difference of ***u*** under the two time conditions.

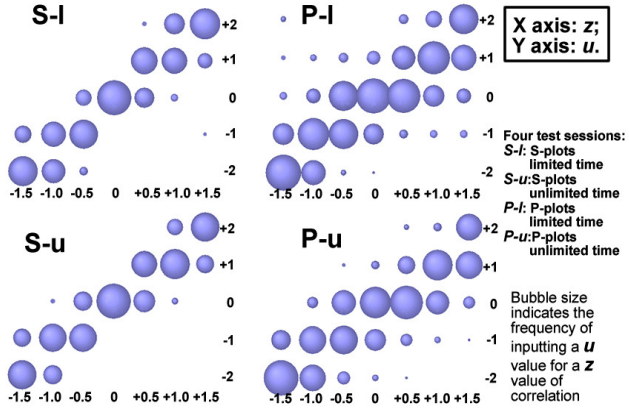


Figure 3. Distributions of users' inputs in the 4 test sessions.

For both plots, strong linear relationships exist between ***u*** and *z* (figure 3, Table A&B). However, the linear trend for P-plots is weakened when excluding strong correlations (*z*=+1.5, -1.5) and moderate correlations (*z*=+1.0, -1.0): the linear regression correlation coefficient drops from 0.82 to 0.46.

Table A - Scatterplots

Pearson correlation \ Sample size / Time condition	10	40	160	all n
limited	0.89	0.93	0.95	0.92
unlimited	0.90	0.94	0.96	0.93
all time conditions	0.90	0.94	0.96	0.93

Table B - Parallel-coordinate plots

Pearson correlation \ Sample size / Time condition	10	40	160	all n
limited	0.72	0.85	0.84	0.80
unlimited	0.83	0.86	0.88	0.85
all time conditions	0.78	0.85	0.85	0.82

If we transform *z* back to ***r***, we find that the degree of correlations is under-estimated. For scatterplots, this result is known [4]. Here, the underestimation seems to be especially true for P-plots (Figure 3 and Figure 4). This suggests that the weaker the correlations in P-plots, the more difficult to interpret them correctly. Particularly, we find that for P-plots, users tend to interpret weakly positive correlations as no correlation and no correlation as negative correlations (Figure 4).

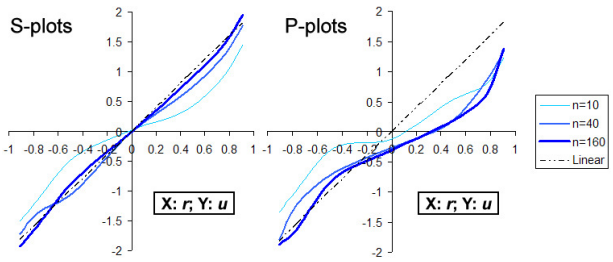


Figure 4. Means of users' inputs (***u***) for different correlations (***r***) in S-plots and P-plots

The agreement among users in ***P-l*** and ***P-u*** (Fleiss kappa=0.31, 0.35) is lower than that in ***S-l*** and ***S-u*** (Fleiss kappa=0.58, 0.61). This suggests that observations from different persons are less constant for P-plots.

Furthermore, ***u*** of positive correlations is significantly different from –***u*** of negative correlations for P-plots, but not for S-plots. This suggests that the different images that result from positive and negative correlations in P-plots lead to different judgements.

For P-plots, a significant effect of samples size ***n*** was found, but not for S-plots. This is due to the larger difference between ***n***=10 and ***n***=160 for weak correlations and no correlation in P-plots (Figure 4).

Finally, the observed correlations for P-plots are significantly different under the two time conditions, but not for S-plots. This is because under unlimited time condition, the under-estimation in P-plots is corrected for negative correlations. It seems that more time to learn leads to better judgment.

4 Training Effect for P-plots

The purpose of the test is to investigate whether training corrects the serious under-estimation of correlations and reduce the variance for P-plots.

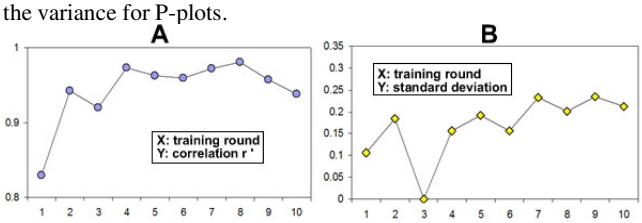


Figure 5. Training effect (**A**: Correlation between ***u*** and ***r***; **B**: Standard deviation of inputs for ***r***=0.)

We provided 10 rounds of training to one participant, and required numeric feedback (u_n) on ***r***. In Figure 5-A, we show the correlation coefficients (***r'***) of u_n and the true ***r*** of sample data in each round. The peak of ***r'*** is at the eighth round and then it falls in the end. In Figure 5-B, the standard deviation of input scores (when ***r***=0) reduces in the second round, but remains high after the third round.

5 Conclusion

Our first study shows that users can get useful information in S-plots and P-plots, but the quality of the information is lower in P-plots. P-plots cause more serious under-estimation, and decisions based on P-plots are less constant under different conditions: judged by different persons, varied positive and negative correlations, varied sample size and different time of display.

Our second study shows that training helps to correct under-estimation in P-plots. However it may confuse users since the variance of inputs remains high after the early rounds of training.

According to all of the above, we conclude that scatterplots are better than parallel coordinate plots for visual correlation analysis with two dimensional data. Therefore, for multi-dimensional data, scatterplots might also lead to better analysis results.

The project is supported by the VIEW program of the Netherlands Organization for Scientific Research (NWO) under research grant no. 643.100.502.

References

[1] R. Amar, J. Eagan, and J. Stasko. Low-level components of analytic activity in information visualization. *IEEE Symposium on Information Visualization*, pages 111-117, October 2005.

[2] L. A. Best, A. C. Hunter and B. M. Stewart. Perceiving Relationships: A Physiological Examination of the Perception of Scatterplots. D. Barker-Plummer et al. (Eds.): *Diagrams 2006*, Springer, pages 244 –257, 2006.

[3] H. Sirrtola. Direct manipulation of parallel coordinates. *IEEE Proceedings of Information Visualization,* Pages 373–378, 2000.

[4] A. Inselberg and B. Dimsdale. Parallel coordinates: a tool for visualizing multidimensional geometry. *Proceedings of Visualization*, pages 361–378, 1990.

[5] Erlick, D.E., Mills, R. Perceptual quantification of conditional dependency. *Journal of Experimental Psychology 73*, pages 9-14, 1967.

Exploring and Visualizing Patterns in Text Collections with FeatureLens

Anthony Don[1,2], Catherine Plaisant[1], Loretta Auvil[3], Tanya Clement[1], Elena Zheleva[1], Machon Gregory[1], Sureyya Tarkan[1], and Ben Shneiderman[1]

1-University of Maryland, 2- University of Bordeaux, 3-University of Illinois at Urbana Champaign

ABSTRACT

Historically, most analytic search tools had the ability to search for terms, but the selection of terms depended solely on users, who worked from memory to determine them. Innovations in knowledge discovery tools have since enabled researchers to generate equally as interesting and potentially more comprehensive pattern sets. We briefly describe the text analytics used, then the user interface and visualizations developed to explore the results. FeatureLens was developed to support literary scholars, but can be used with any text collection.

CR Categories and Subject Descriptors: H.5.2. User Interfaces; I.2.7 Natural Language Processing.

Additional Keywords: text mining, digital libraries, information visualization, visual analytics, knowledge discovery.

1 INTRODUCTION

In the Humanities, software has been used to access text documents but rarely to support new ways of reading them [1]. With the development of digital libraries, researchers gained the ability to easily retrieve and search large bodies of texts, images, and multimedia materials online. The ability to search for keywords or phrases in a text or collections is now widespread, but such searches only marginally support discovery. Researchers must still rely on their notes, files, and their own memories to find "interesting" facts that inspire them to elaborate upon or diverge from existing readings and theorizations.

Text mining tools are improving researchers' search options by suggesting potentially interesting patterns to examine. Scholars are able to accept genuinely interesting patterns and reject those that aren't. Unfortunately, text mining algorithms typically return a large number of patterns that are difficult to interpret out of context. FeatureLens [2] was designed to enhance scholars' ability to interpret text mining results by providing tool to visually explore text collections and their patterns. See figure 1.

2 ANALYTICS

FeatureLens's data mining components search for two types of patterns: frequent words (as shown in Figure 1) and frequent itemsets of n-grams, which capture the repetition of exact or similar expressions in a text or collection.

The conversion of unstructured text to patterns is performed using the D2K (Data to Knowledge) environment with the T2K (Text to Knowledge) components created at NCSA/UIUC. D2K is a rapid, flexible data mining and machine learning system that integrates data and information visualization tools with analytical data mining methods for prediction, discovery, and deviation detection. It offers a visual programming environment that allows users to connect programming modules together to build data mining applications and supplies a core set of modules, application templates, and a standard API for software component development.

FeatureLens' development team created several new modules to export the data and patterns into a database. Patterns are determined using the CLOSET algorithm [4]. The D2K and T2K environment provides a number of components that allow for flexibility in the types of patterns created. For instance, a literary scholar asked us to search for repetitive patterns, keeping all words in their existing state. We created 3-grams items (i.e., sequences of 3 words each). As we evaluated the resulting patterns, it became clear that relaxing the original constraint allowed for similar patterns to be discovered. For instance, by applying stemming, verbs were converted to their roots, and thus different patterns evolved, ones that would not have been obtained by keeping verb tenses as they existed in the document.

Figure 2 shows the D2K environment used to process the data for FeatureLens. Additional metrics were calculated using Ruby.

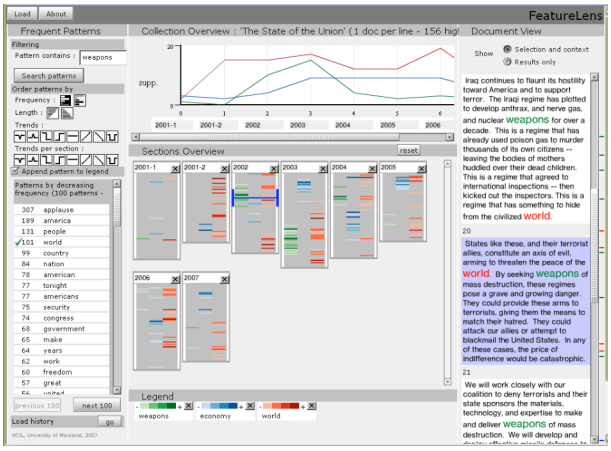


Figure 1. FeatureLens with a collection of State of the Union speeches from George W. Bush. Three patterns (words) have been selected; their usage in the text can be compared.

3 VISUALIZATION

FeatureLens aims at integrating a set of text mining and visualization functionalities into a powerful tool, one which provokes new insights and discoveries. It supports discovery by combining the following features: a visual overview of the entire collection, pattern sorting by frequency or length, usage distributions of patterns across texts or collections, side-by-side comparison of patterns, and pattern displays in the context of the text where they appear. See figure 1.

Catherine Plaisant, HCIL/UMIACS, Univ. of Maryland
College Park, MD 20742. plaisant@cs.umd.edu

A visual overview of the entire collection is shown by the set of columns in the middle of the display. The collection is divided in some manner based on xml tags or directory structure (shown as Sections), so that the overview provides a meaningful display. Each set in figure 1 shows one of President Bush's speeches.

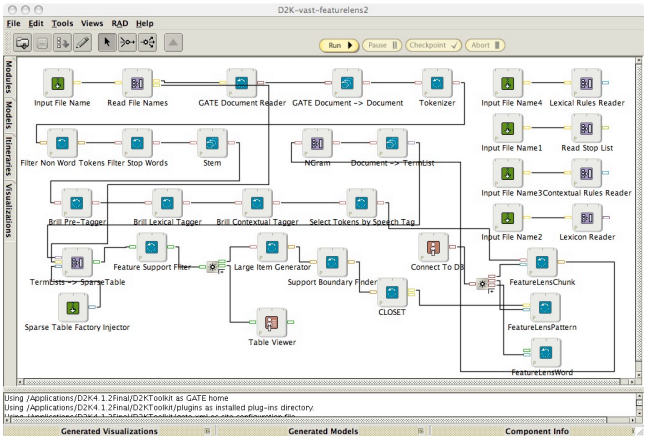


Figure 2. D2K demonstrating the processing of text to create patterns for exploration in FeatureLens.

Patterns are shown in a list on the left. Sets of buttons above the list control what is shown. For instance, shown in figure 1 are the most frequent occurring patterns (in this case, words). When a user clicks on a pattern, its usage in the collection is shown. Multiple patterns can be shown at the same time (as the above image demonstrates). By brushing the patterns in the list, we show the complete pattern as well as an example (see figure 3). Users are able to see where patterns occur across the different sections.

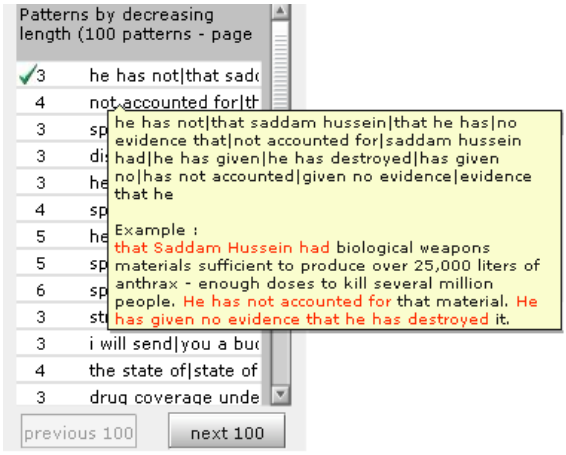


Figure 3. The list of patterns can be sorted by length, revealing long patterns of co-occurring 3-grams in the collection. A tool tip gives an example of usage.

An additional analytics layer shows users the distributions of patterns in each section (as demonstrated with the line chart). This analytics shows the trends across sections, letting users look for patterns that have usage distributions with certain characteristics (e.g. increasing, decreasing, spikes, etc). This may further assist users in discovering interesting patterns.

4 Examples of Use

This work started with a literary problem developed by Tanya Clement, a doctoral student from the University of Maryland English Department. Her work deals with the study of *The Making of Americans* by Gertrude Stein, which consists of 517,207 words—5,329 of them unique. In comparison, Herman Melville's *Moby Dick* consists of only 220,254 words—14,512 of them unique. Stein's extensive use of repetitions renders *The Making of Americans* one of the most difficult books to read and interpret. Literary scholars are developing interpretive hypotheses on the purpose of this text's repetitions. We are conducting a longitudinal case study with this expert user to understand the potential benefits of FeatureLens for literary analysis.

To date, FeatureLens has been used with four collections and a pilot test was conducted with eight users [2]. Additional information can be found online at www.cs.umd.edu/hcil/textvis/featurelens.

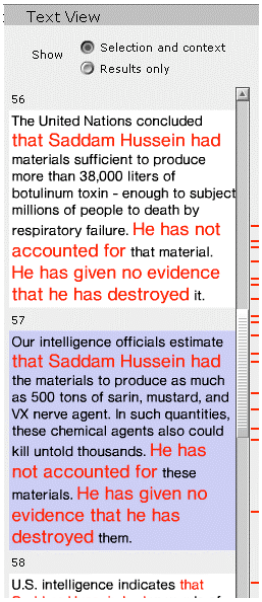


Figure 4. In the Text View panel, users can read the text where the patterns occur. Tick marks indicate the location of all other complete or partial matches next to the selected paragraph.

5 Acknowledgement

Support was provided by the Andrew Mellon Foundation as part of the Monk project (see www.monkproject.org).

References

[1] Plaisant, C.; Rose, J.; Yu, B.; Auvil, L.; Kirschenbaum, M. G.; Smith M. N.; Clement, T.; & Lord, G. (2006). Exploring erotics in Emily Dickinson's correspondence with text mining and visual interfaces. Proceedings of the 6th ACM/IEEE Joint Conference on Digital Libraries, JCDL, 141-150.

[2] Don, A.; Zheleva, E.; Gregory, M.; Tarkan, S.; Auvil, L.; Clement, T.; Shneiderman, B.; & Plaisant, C. (2007). Discovering interesting usage patterns in text collections: Integrating text mining with visualization. Proceedings 2007 ACM Conference on Information and Knowledge Management (to appear).

[3] Clement, T.; Don, A.; Plaisant, C.; Auvil, L.; Pape, G.; & Goren, V. (2007). 'Something that is interesting is interesting them': Using text mining and visualizations to aid interpreting repetition in Gertrude Stein's *The Making of Americans*. Digital Humanities Conference, 40-44.

[4] Pei, J.; Han, J.; & Mao, R. (2000). CLOSET: An efficient algorithm for mining frequent closed itemsets. ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery, 21–30.

Interactive Poster: CGV – Coordinated Graph Visualization

James Abello*
DIMACS
Rutgers University

Hans-Jörg Schulz†
Institute for Computer Science
University of Rostock

Heidrun Schumann‡
Institute for Computer Science
University of Rostock

Christian Tominski§
Institute for Computer Science
University of Rostock

ABSTRACT

Visualizing large hierarchized graphs is such a challenging task that it is hardly possible to represent all relevant information within a single comprehensible image.

To address that challenge, we pursue multiple linked view visualization. We report on a visualization framework for exploring hierarchized graphs that focusses on a modular architecture based on the model-view-controller (MVC) concept. MVC helps us in coordinating views as users explore and navigate the data. Several interaction facilities further support users in applying the framework for their data and their task at hand.

Keywords: Graph visualization, model-view-controller, interaction, lenses, dynamic filtering.

1 INTRODUCTION

Visual analysis of graph structures is a hot topic in many domains, as for instance system biology, network security, or sociology. In many of these domains, we are facing graphs that are too large and complex to be represented by a single visualization. In such cases, creating a graph hierarchy helps to drive interactive visual analysis [2]. The challenge here is that a graph hierarchy imposes certain requirements on the visualization. In our view, it is mandatory to represent the following items:

- clustering hierarchy H,
- anti-chains (abstractions of the underlying graph G), and
- data attributes associated to nodes and edges.

It is hardly possible to visualize all these aspects with a single technique. As demonstrated in previous work, multiple views are a solution to this problem [1]. So far, hard-wired view composition has been the chosen approach. We advocate here the use of an architecture where a more flexible set of views can be easily integrated and customized (e.g., to address particular needs of an application scenario). We report on how the model-view-controller pattern (MVC) can be utilized to achieve a flexible system architecture for interactive graph visualization.

2 FRAMEWORK ARCHITECTURE

The model-view-controller (MVC) concept is widely applied in scenarios where data (M) needs to be represented (V) and are subject to interaction (C). The strict separation of data, views, and interaction is the major benefit of MVC. Architectural separation of views and interaction does not necessarily mean that users are not allowed to interact with the views directly. Adhering to MVC allows for flexible architectures where components can be plugged in on demand. In particular, we instantiate MVC as follows.

The main data structure (M) contains all necessary elements to model a graph hierarchy (see [1]). It integrates the underlying graph G, the clustering hierarchy H, and associated anti-chains (i.e., abstractions of G) with their corresponding nodes and edges. It also takes into account (multivariate) data associated with nodes.

A view (V) is responsible for representing the data model. What aspects of the data are represented depends on the specific implementation of a view. We explicitly incorporate multiple views in order to represent different data aspects. In Sect. 3, we give examples of views currently available in the system.

Views are not allowed to modify the data (read-only access). However, the interactive exploration of a graph via anti-chains requires manipulation of the data model, in order to switch between different levels of abstraction. To guarantee that the data model is consistent at all times, even when represented by multiple views, manipulation requests are propagated to a controller (C). In turn, the controller notifies all views of the pending manipulation. If no view has an objection against the manipulation, the data model is altered. Then all views are notified about the change. It turned out to be a good solution to first notify the view that issued the manipulation request and then inform all other views. That gives users immediate feedback at the place where they performed the interaction. An overview of available interaction techniques is given in Sect. 4.

3 VIEWS IN CGV

MVC lays the ground for multiple view visualization. Multiple views help in communicating different data aspects and in performing different visualization tasks. This assumes that all views are arranged in a way that really supports the task at hand. Since the preferred arrangement depends on users and their tasks, we embed the views in a docking framework. This allows users to create arrangements different from the default setting, and furthermore, to store and reload customized arrangements.

Based on the arrangement used in [1], we designed the default setup of the framework as shown in Fig. 1. In particular, we provide the following views:

- *Graph view* provides at all times a node-link view of the current visible anti-chain, where nodes and edges are aggregations of the underlying graph. The purpose of this main view is to visualize structure and clusters, and to highlight a selected data attribute by means of color. The graph layout can be generated by a variety of algorithms, as for instance LinLog or a classic spring embedder.
- *Textual tree view* also visualizes the current anti-chain, however the represented edges are edges of the clustering hierarchy H. This view's purpose is to show the structure of H, labels, and to drive easy navigation. Enhanced visual and interaction capabilities support the user [4].
- *Graphical hierarchy view* gives an overview of the entire clustering hierarchy H. Additionally, a user-chosen data attribute can be color-coded.
- *Matrix view* provides a density visualization of the macrograph determined by the current anti-chain. This view enables users to spot dense graph clusters that may be used as triggers for further exploration.

*e-mail: abello@dimacs.rutgers.edu
†e-mail: hjschulz@informatik.uni-rostock.de
‡e-mail: schumann@informatik.uni-rostock.de
§e-mail: ct@informatik.uni-rostock.de

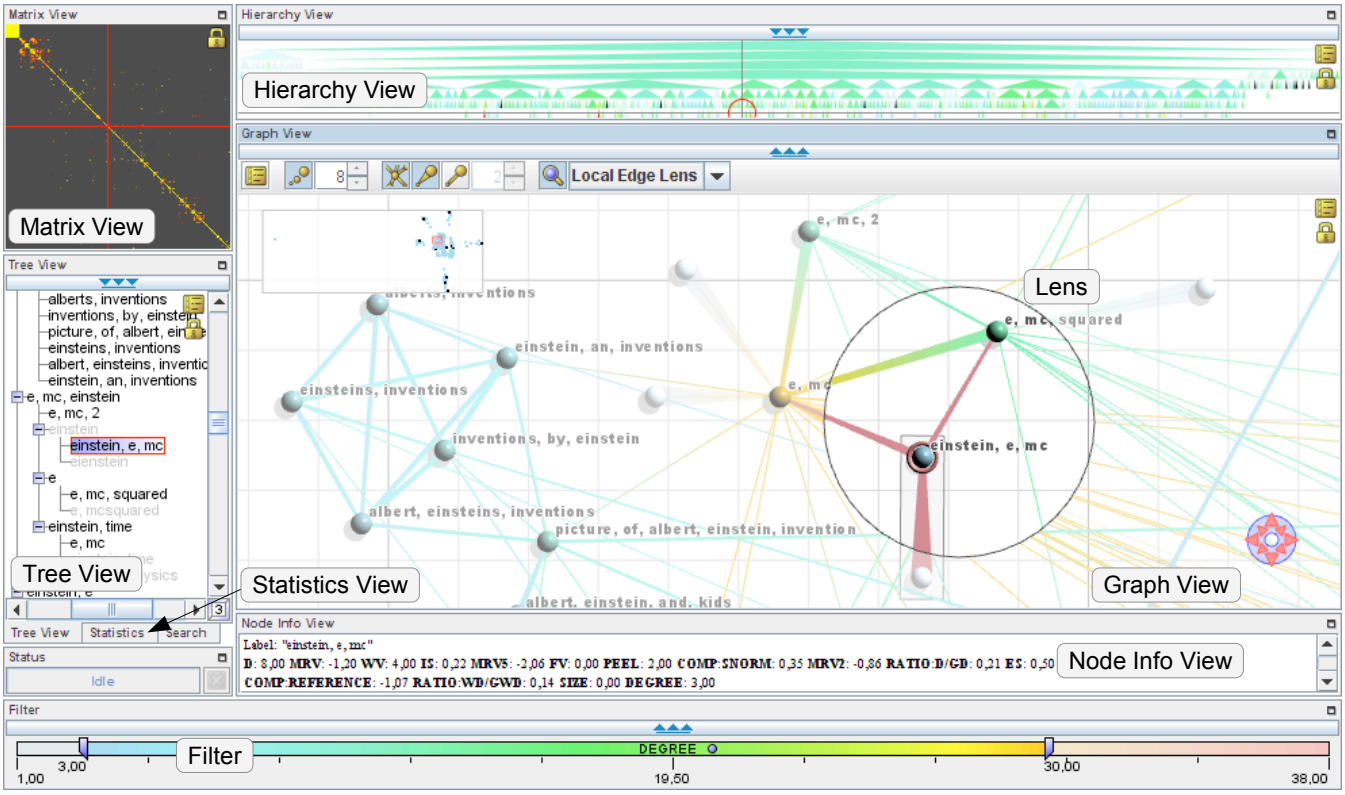


Figure 1: CGV – Coordinated graph visualization with multiple views.

- *Statistics view* represents meta information (e.g., size of the current anti-chain or the number of nodes and edges currently visible) in textual form, which is very helpful when exploring an unknown data set (not visible in Fig. 1).
- *Node Info View* represents meta information for the currently focussed node.

4 Interaction Facilities

To facilitate easy data exploration, we provide common interaction techniques, such as zoom&pan or scrolling. To allow users to switch between different abstractions of the hierarchized graph, nodes can be expanded and collapsed. Additionally, users can select and focus on nodes. All views follow consistent common interaction policies (e.g., double left click on a node in any view will result in expansion of that node). Recall that views do not alter the data model, but propagate requests to the controller. This decouples the specific physical action (e.g., click) from the effect (e.g., node expansion) and enables us to provide a basic undo/redo mechanism.

Beyond the aforementioned interactions, the system offers several novel interaction facilities:

- *Edge-based navigation* aims at supporting the common task of path navigation. For that purpose, we make the edges of a focussed node interactive. Clicking on such an edge navigates to the respective neighbor and then focusses on that node automatically.
- *Lenses* are helpful tools to support locally restricted visualization tasks [4]. They can help to tidy up edge clutter (task: "Which edges connect to a node?"), or can perform local transformations on the graph layout to bring possibly distributed nodes of interest close together (task: "What are the neighbors of a node?").
- *Dynamic filtering* is provided to support users in exploring the data for interesting nodes or selecting nodes relevant for a particular task at hand. Our filtering mechanism allows for logical combinations of basic filters, which operate on node attributes (e.g., quantitative values or categorical labels). We utilize a sieve metaphor to ease the interactive composition of filters. Through automatic fading or omitting filtered nodes, users get immediate visual feedback on the filtering result.

5 Conclusion

This work illustrates the use of the model-view-controller pattern (MVC) for coordinated graph visualization. This is exemplified in a system termed CGV (available at [3]). The proposed framework offers several common linked views and some novel interaction facilities (lenses and edge navigation). We are currently exploring the usefulness of other visualization techniques to be integrated into the framework. This is work in progress towards the end goal of facilitating user- and task-dependent setups (i.e., selection of techniques and their screen arrangement).

References

[1] Abello, van Ham, and Krishnan. ASK-GraphView: A Large Scale Graph Visualization System. *IEEE Transactions on Visualization and Computer Graphics*, 12(5), 2006.

[2] Herman, Melançon, and Marshall. Graph Visualization and Navigation in Information Visualization: A Survey. *IEEE Transactions on Visualization and Computer Graphics*, 6(1), 2000.

[3] Tominski. CGV prototype. `http://www.informatik.uni-rostock.de/~ct/CGV/CGV.html` (accessed June 2007).

[4] Tominski, Abello, van Ham, and Schumann. Fisheye Treeviews and Lenses for Graph Visualization. In *Proc. IV'06*, London, 2006.

Interactive Poster: Visualization of Gene Combinations

Christian Tominski* Clemens Holzhüter Andrea Unger† Heidrun Schumann‡

Institute for Computer Science
University of Rostock

ABSTRACT

The analysis of microarray data is a key to understanding the influence and role of genes. Visualization is one way to support analysts in finding potentially important genes. However, most visualization tools focus on representing single genes, important gene combinations are not always easy to spot and compare with existing approaches.

What we propose here is the novel idea of making gene combinations visually explicit, and thus making them easier to grasp and understand. We describe a tool that integrates several visual and interaction methods to help users gain insight into their microarray data. The tool further provides an interface to plug in analytical methods, which are important to filter relevant gene combinations out of the massive number of theoretically possible ones.

Keywords: Microarray, visualization, gene combination.

1 INTRODUCTION

In recent years, microarray analysis has paved the way to understanding the interplay of genes. Visual methods play an important role in the process of gaining insight from microarray experiments. Today, a variety of useful tools is available, which mostly help in understanding the impact of single genes. However, still much knowledge and experience is required to spot not only important genes, but important combinations of genes. The situation worsens if the analyst has to assess similarities between different gene combinations.

In this work, we describe concept and implementation of a tool that aims at explicit visualization of gene combinations. For that purpose, we make a switch from data items in form of single genes to data items that accommodate gene combinations. Apparently, since theoretically any combination of genes can carry important information, such a switch implies a massive increase in the volume of data to consider for the analysis. We will briefly describe how plugable filters can help to cope with these vast data volumes. The main contribution of this work is an interactive tool that integrates several visual concepts to visualize combinations of genes. We extend the classic heatmap approach in order to emphasize on the representation of gene combinations. The representation is enhanced with additional visual clues to support comprehension of dis/similarities between different combinations of genes.

2 CONCEPT & IMPLEMENTATION

Let us now be more specific in terms of how combinations of genes can be analyzed and visualized.

2.1 Approach Outline

Microarray data contain information on the expression of genes G for several samples S (e.g., time steps), which can be formalized as a function $exp_g : G \times S \rightarrow \mathbb{R}$. Classic heatmap visualizations represent exactly that relationship between genes/samples and corresponding expression. In other words, the visualization focusses on genes themselves. What we pursue is the visualization of gene combinations. This will allow us to assess not only the relationships between single genes, but also the interplay of combinations of genes. A gene combination is a set $GC \in \mathfrak{P}(G)$. In order to represent the expression of gene combinations, we need to aggregate the expression of those genes participating in a gene combination, which can be modeled as a function $exp_{gc} : GC \times S \rightarrow \mathbb{R}$. That is, the aggregated expression value determines the level of regulation of an entire gene combination. Usually, we want to use the average, but it is also possible to consider other aggregates if needed for a particular application.

It is obvious that the step from genes to gene combinations increases the data volume by magnitudes. And indeed, we are not able to represent that huge data volume. However, since we are not interested in all theoretically possible gene combinations, but only in biologically interesting and relevant ones, we can apply the following three-step approach to achieve our goal: 1) Generate gene combinations; 2) on the fly filter out biologically less relevant combinations; 3) visualize only gene combinations that passed step 2).

Step 1) is relatively straight-forward to implement as a serial permutation generator. Explanations on step 2) and 3) will be given in the next paragraphs.

2.2 The Need for Filters

In order to practically visualize gene combinations, it is mandatory to filter out biologically less relevant gene combinations. Commonly, expert knowledge is required to assess which combinations are relevant and which are not. Therefore, we provide two basic options to drive the filtering process.

The first option is to restrict the number of genes to consider for the analysis. That is, the (expert) analyst selects (out of the many genes in a data set) only those that are relevant with respect to the task at hand. This results in significant, though coarse reduction of the data volume.

Secondly, in order to fine-tune the filtering, analytical methods are applied [2]. This step further crystallizes possibly relevant gene combinations. Parameters to control this analysis step are exported to the user interface. Hence, it is easily possible to steer the filtering process interactively.

The result of the described filtering procedure is a set of potentially significant gene combinations, which are passed to the visualization step. Indisputably, the term "biologically significant" depends on many factors. Therefore, we designed the filtering procedure as a modular component, which allows for integration of task and data specific filter implementations.

2.3 Visualization of Gene Combinations

The first question to ask when visualizing abstract data – in our case gene combinations extracted from microarray data – is how to layout information on screen. Since heatmaps are widely accepted among biologists, we have decided to extend that approach for gene combinations. Heatmaps are based on a matrix-like display and commonly use a red-black-green color scale to visualize gene expression (red: up-regulation; black: no regulation; green: down-regulation). We extend that approach as follows.

*e-mail: ct@informatik.uni-rostock.de
†e-mail: aunger@informatik.uni-rostock.de
‡e-mail: schumann@informatik.uni-rostock.de

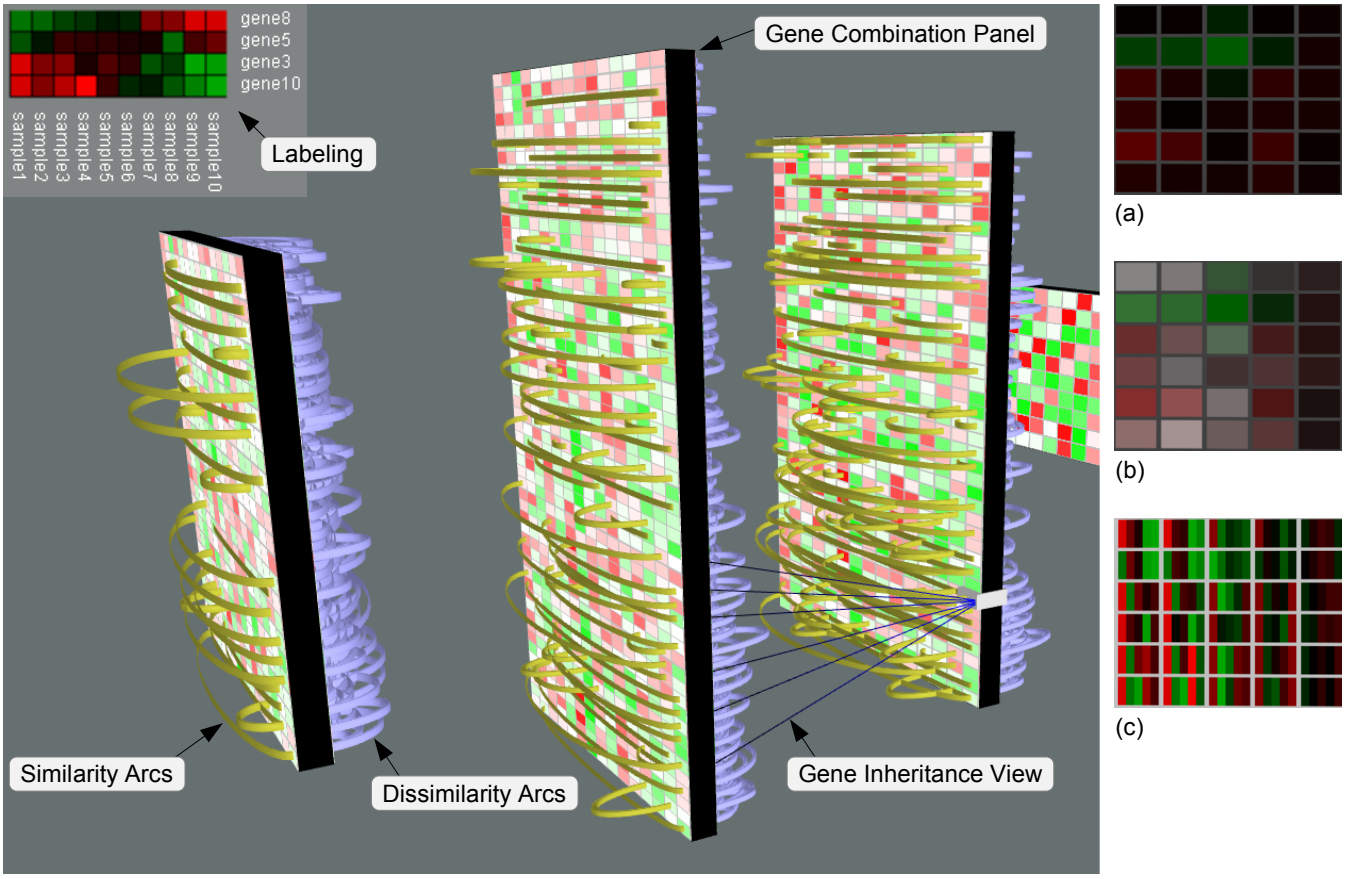


Figure 1: ViGeCo – Visualization of Gene Combinations.

First, we generate multiple panels – one for each possible size of gene combinations. That is, gene combinations consisting of two genes are represented by one panel, combinations of three genes by another panel, and so forth. These panels are arranged in a 3D presentation space as shown in Fig. 1.

Secondly, we need a way to represent aggregated data values. These values indicate the regulation of a gene combination as a whole. The problem is that the classic coloring scheme of heatmaps (see Fig. 1 (a)) can not be applied as is. This becomes clear when we consider a gene combination that consists of one up-regulated and one down-regulated gene. The aggregated expression value would be represented by a color close to black. The analyst is likely to interpret this as a gene combination with no regulation, which could be a wrong conclusion depending on the task at hand. Therefore, we can optionally use the brightness channel of colors to encode the average difference in the expression value of genes. This reduces the chance of misinterpretation, because darker colors are only created for gene combinations that are really not or less regulated (see Fig. 1 (b)). Alternatively, users can switch to a "small multiples" representations, for which each cell of a gene combination panel is subdivided to accommodate a color representation of the original (non-aggregated) expression values (see Fig. 1 (c)).

Using the gene combination panels it is possible to get an overview and to spot gene combinations that are active for all samples (saturated red or green color). The panels are also useful to find samples for which certain gene combinations exhibit similar behavior; this is expressed by similar colors. In order to further facilitate the task of finding similarities, we provide additional arc displays on demand. Arc displays are helpful in making relations between visual elements more explicit (see [4] or [3]). As such, arcs help us to make dis/similarities of the regulation of gene combinations more clear to the analyst. We provide two kinds of arcs: yellow similarity arcs and blue dissimilarity arcs (see Fig. 1). The latter arcs were requested by collaborating biologists, because for them it is also interesting to see "negative" similarity.

The users of our prototype also requested a view to see which genes contribute to which combination. This is achieved by labeling and an additional visual cue called gene inheritance view. This view links gene combinations that share common genes. (see Fig. 1).

3 Summary

We presented a novel approach to visualize microarray data. The novelty of the concept lies in the emphasis of gene combinations. We presented several extensions over the classic heatmap approach. Our concept has been implemented as a component for the microarray analysis framework Mayday [1]. The prototype is highly interactive in terms of adjusting the visualization as well as the provided analysis methods.

References

[1] Dietzsch, Gehlenborg, and Nieselt. Mayday - a Microarray Data Analysis Workbench. *Bioinformatics*, 22(8), 2006.

[2] Drăghici. *Data Analysis Tools for DNA Microarrays*. Chapman & Hall/CRC, 2003.

[3] Neumann, Schlechtweg, and Carpendale. ArcTrees: Visualizing Relations in Hierarchical Data. In *Proc. EuroVis*, 2005.

[4] Wattenberg. Arc Diagrams: Visualizing Structure in Strings. In *Proc. InfoVis*, 2002.

Visualizing very large layered graphs with quilts

Benjamin Watson, David Brink, Matthias Stallmann, Ravi Devarajan, Matthew Rakow, Theresa-Marie Rhyne and Himesh Patel

SAS Institute Inc. and North Carolina State University

ABSTRACT

Traditional node-link depictions of layered graphs such as flow charts and process or genealogy diagrams are in widespread use. Layers emerge from applied context (e.g. process stages or familial generations), or are inserted to improve visual clarity. However, for many applications these diagrams quickly lose their utility as graph complexity grows. We introduce quilting, an interactive, matrix-based depiction for very large layered graphs that remains useful even when optimized node and link depictions have become unintelligible. We demonstrate quilting using an activity-based management (ABM) application that must depict layered graphs with thousands or even hundreds of thousands of nodes. Unlike node-link depictions, quilts depict 500-node graphs quite clearly. On typical desktop displays, quilts depicting larger graphs must be summarized.

CR Categories and Subject Descriptors: H.1.2 [Models and Principals]: User/Machine Systems – Human information processing.

Additional Keywords: graph drawing, layered graphs, crossing minimization, matrix depiction.

1 INTRODUCTION

Layered graphs such as structure charts, process diagrams, and flow charts have wide-ranging applications. In these graphs, nodes are grouped into layers defined either by the application context, or introduced to increase visual clarity.

Traditional node-link depictions of layered graphs arrange members of a layer into a line [1][4]. *Proper* links connect nodes on adjacent layers; we call remaining links *skip* links. Crossing minimization algorithms reduce the intersection of proper links, and can improve legibility. Nevertheless, as the number of links grows, these depictions can become quite muddled, with viewers having trouble understanding graph connectivity (Figure 3).

To address the scalability problem for layered graphs, we introduce quilts (Figure 1), a new depiction that uses matrices to visualize layered graphs. Below, we describe the meaning and construction of quilts and show how quilts might be used in one application.

2 THEORY

The quilt depiction of a layered graph is a simple adaptation of the matrix depiction for unlayered graphs [2]. We represent proper links with dots in an achromatic matrix, and chain these matrices together with additional colored levels (rows or columns of cells) representing layers (Figure 1). Each level cell corresponds to an individual layer node. To distinguish levels from one another and from matrices, we assign a unique chroma and saturation to each level. We assign each level cell a unique luminance, effectively making the color of every graph node unique.

Each level acts as the source level for the following matrix, and the destination level for the previous matrix. Nodes on odd-numbered levels are lined up horizontally while those on even-numbered levels appear vertically. A proper link from an odd to even level makes a left turn whose corner is the cell representing that link. Similarly, a simple link from an even to odd level makes a right turn through the cell for that link.

We depict skip links with colored cells at the end of the row or below the column that follows level nodes. The color of each skip link is the same as its destination level node (recall that the color of each node is unique). To improve clarity, we sort the skip links using level number as the primary key and cell number (within the level) as secondary key.

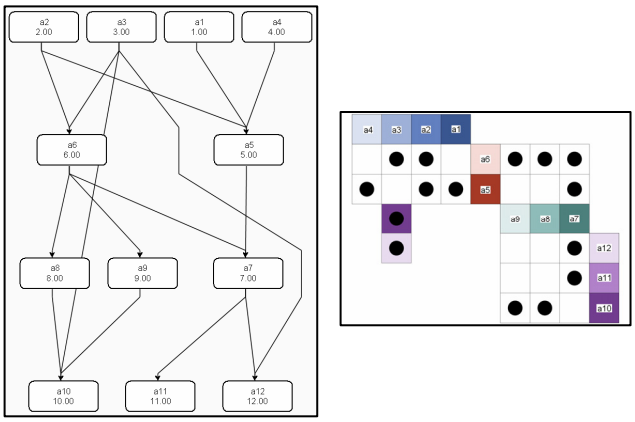


Figure 1. Node-link diagram and corresponding quilt diagram.

2.1 Interactivity

To ensure good scalability, quilt cells are typically too small to support display of application-assigned node properties, especially textual labels. Viewers can reveal a node or link's properties by hovering over its corresponding cell (Figure 3).

As noted by Ghoniem et al. [3], one of the weaknesses of a matrix depiction is path following. We address this shortcoming by allowing viewers to highlight graph paths in the quilt by clicking on node or link cells (Figure 3). The first such click highlights the cell itself, as well as immediately adjacent nodes. Each additional click highlights nodes that are one more link distant. We call this click-through. A backward click-through, removing highlights from the nodes last reached, is available by clicking on the node while the control key is pressed. Pressing the shift key while clicking on a node highlights all the nodes reachable from the clicked cell. Clicking on any portion of the quilt not containing a node or link removes the highlight.

- *Benjamin Watson is with NC State Univ., Email: bwatson@ncsu.edu*
- *David Brink is with the SAS Institute, Email: david.brink@sas.com.*
- *Matthias Stallmann is with NC State Univ., Email: matt_stallmann@ncsu.edu*
- *Ravi Devarajan is with the SAS Institute, Email: ravi.devarajan@sas.com*
- *Matthew Rakow is with NC State Univ., Email: marakow@ncsu.edu*
- *Theresa-Marie Rhyne is with NC State Univ., Email: tmrhyne@ncsu.edu*
- *Himesh Patel is with the SAS Institute, Email: himesh.patel@sas.com*

To provide viewers with the strengths of both quilts and the node-link depiction, we couple these depictions in an interactively linked view (Figure 3). Clicking in either depiction will highlight the appropriate graph path in the other depiction.

2.2 Summarization

When graphs contain thousands of nodes or more, the corresponding quilts can require summarization, since they will not fit in a typical display. To summarize quilts, we cluster nodes.

Each summarized level cell except the last represents *s* unsummarized nodes. Each summarized matrix cell represents up to *s* x *s* links. In applied settings, we find that summarized matrix cells represent far fewer links. We map the number of links in summarized matrix cells to cell luminance, with darker cells indicating more links (Figure 2).

Hovering over a summarized level cell indicates the number of cells being summarized, as well as statistical summarizations of any application properties in the summarized set of cells. Hovering over summarized link and skip link cells has similar functionality.

Selecting a summarized cell highlights summarized level and matrix cells in a manner much like that in unsummarized quilts.

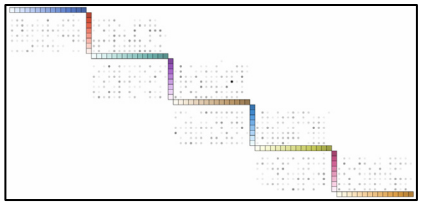


Figure 2. Summarized quilt, 16 nodes per summarized node.

3 Discussion

Scalability. Quilts remain legible even as graphs grow to contain hundreds of nodes. In contrast, node-link depictions with only a few dozen nodes can be difficult to understand.

Property visibility. Quilt scalability comes at the price of some property visibility, with node and link properties and labels hidden inside small cells until the user hovers over the cell. This can make finding nodes or links difficult. However, property visibility in node-link depictions also suffers as graphs grow in size.

Link visibility. Following a series of links in small node-link depictions is simple, since all links begin and end directly at the linked nodes. Links in similarly sized quilts require the viewer to trace the matching row and cell across the matrix that joins them or examine the color of the skip links. In larger graphs this is less of a disadvantage, as following links in node-link depictions becomes more difficult.

Skip links. Node-link and quilt legibility suffers as the ratio of skip to proper links grows, since layers become a less useful visual grouping. To aid in skip link legibility, hovering over a skip link draws a line to its corresponding destination node. Perhaps the most effective depiction of such irregularly connected graphs is a single matrix: because it does not depict layers, it does not require special cells to depict skip links.

Crossing minimization. In node-link depictions, preserving legibility as graph size moves from small to moderate requires crossing minimization – an NP-hard problem. Quilt depictions remain legible even without crossing minimization, though such minimization (which would concentrate link dots along matrix diagonals) may still prove beneficial.

3.1 An Applied Example

We developed the quilt in the applied context of SAS's Activity-Based Management application (ABM). ABM is an analytical application that models an organization's processes to determine accurately the cost and profitability of products and customers. ABM models the interactions between corporate groups and assigns revenue to those responsible for the products or services.

SAS's ABM system uses a directed graph to model the interactions within an organization. Large organizations and their processes yield complex models often containing hundreds of thousands of vertices and millions of links. While answering specific cost or revenue questions using these models is fairly simple, realizing the data's true wealth requires visualizing them to understand hidden trends. Unfortunately, most of our ABM graphs overwhelm traditional node-link depictions.

Figure 3 shows a small ABM graph depicted using linked node-link and quilt views. The graph models the cost processes involved in an airlines' flight catering service. Here we have highlighted the food prep group in our linked views. By following the incoming/outgoing links we can see the objects that contribute to/receive contributions from this group.

Even with highlighting, in the node-link depiction it is difficult to discern exactly how highlighted nodes are connected to the food prep group; in the quilt this connectivity is much clearer. On the other hand, in the quilt we cannot see the labels or attributes of any nodes except the one selected.

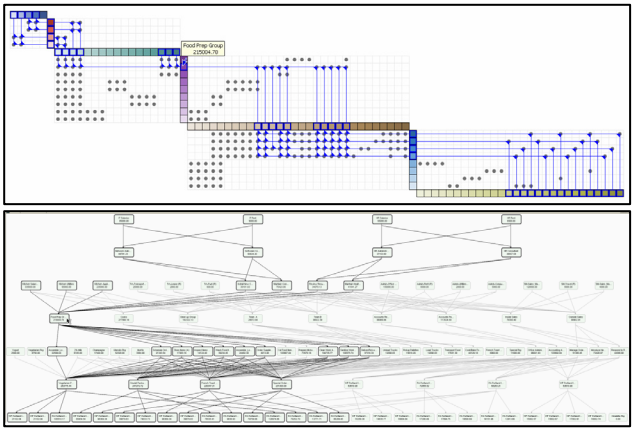


Figure 3. Example with ABM data.

4 Conclusion

We have presented a new depiction of layered graphs, which we believe remains legible even when the depicted graphs have several hundred nodes. We plan to investigate layer and crossing optimization, clustering and zooming in summarization view, interactive editing of the quilt, new ways to represent OLAP data, and user interface improvements. We would also like to conduct a user study to compare the effectiveness of quilts in comparison to node-link depictions of layered graphs.

References

[1] Battista, G.D., Eades, P., Tamassia, R. and Tollis, I.G. 1999. *Graph Drawing: Algorithms for the Visualisation of Graphs*, PrenticeHall.

[2] Bertin, J. 1967. *Sémiologie Graphique : Les Diagrammes - Les Réseaux – Les Cartes*. Editions de l'Ecole des Hautes Etudes en Sciences: Paris, France.

[3] Ghoniem, M., J.-D. Fekete, P. Castagliola. 2004. A Comparison of the Readability of Graphs Using Node-Link and Matrix-Based Representations. Proc. IEEE Information Visualization.

[4] Herman, I., Melancon, G., Marshall, M.S. 2003. Graph Visualization and Navigation in Information Visualization: A Survey. *IEEE Trans. Visualization and Computer Graphics*, *6*, 24-43.

Maestro: 3D Calendar Visualizer

Billur ENGIN
Sabanci University

Mehves CETINKAYA
Sabanci University

ABSTRACT

A 3D calendar Visualizer (Maestro) is developed for coordinating multiple schedules, with a new entry representation concept is introduced with this project, when compared to other calendar applications dealing with the management of the data only. Spotting the relevant information in order to avoid a complicated demonstration is the most challenging issue. This is the main reason why a recognizable, primitive shapes for the data expression were preferred and aimed to design a software scaleable according to the number of calendars.

Keywords: Calendars, time sheet, schedule, 3D Organizer, Information Visualization, Graphical Representation, Interactive Animation, 3D Graphics, Cylinder

1 INTRODUCTION

Although some people still choose to use traditional paper-based calendars, computer-based calendars are widely preferred nowadays. People have one, two or maybe more electronic calendars. Presentation of separate calendars in a manner to reveal their relevance is a complex task.

Another requirement for an interactive calendar is detecting the common free times of a group. Especially in places, where people work in groups, synchronizing the calendars of individuals and setting a time for a meeting or group work is always a major problem.

Maestro is created to simplify the visualization of multiple schedules of a user at a time (or single schedules of different users belonging to a group) and to enable a group to see their common free times of a day without difficulty. The user can both see the weekly and the daily state of her agenda, and get a quick idea about the upcoming events at a glance.

2 RELATED WORK

Computer-based calendars are mostly 2D and designed for single user. These calendars are adequate to store and manage the data but they are identical with the paper-based calendars used in offices. Mackinlay [1] tried different methods for 3D representations of calendars like Spiral Calendar and Time Lattice. Especially Time Lattice has a similar attempt, like the one which will be introdued in this paper, combining calendars of different users in a group. It introduces “Translucent Shadows” to display relationships between users. As reported by Macinlay this technique produces comple objects and involves challenging interactions. This issue is covered particularly in our study.

Showing the "load" of a calendar has also been a common goal in previous designs. Like Tessler said [2], "Busy-ness" level is shown with length-changing bars, or sometimes with number of primitives. One can understand busyness intuitively from the layout with Maestro.

billurengin@ su.sabanciuniv.edu
mehvesc@su.sabanciuniv.edu
Sabanci University, TURKEY

3 SYSTEM

Maestro imports the users' agendas and illustrates these data with well-known demonstrations, as the "pie chart figure" stands for daily schedules, a "cylindrical figure" stands for weekly schedule. Thus, every user can easily see the relationship between users and finding common times becomes an effortless action.

Google Calendar and iCal are two computer-based calendar programmes where the user also can have multiple organizers and can find the common free times across these calendars. But these approaches are still conventional and are not scalable. Maestro can unite any number of calendars created by the user in one place. The user will have the ability to look at all of the calendars at once or individually. The main strength of Maestro lies in its scalability to large number of calendars

One of Maestro's important features is that it ensures the members of a group to find their common free times. Maestro can compare the calendars of the members in its system and show the common free times. This will help the user in any occasion like family meeting, work conferences etc.

3.1 Monitoring Mode

In monitoring mode Maestro shows the activities of a week. By switching into this mode, user can get a general idea about the load of the upcoming week at a glance. Each schedule is represented with a unique colour. The pie in the front is the present day and the other days are placed behind each other forming a cylindrical shape.

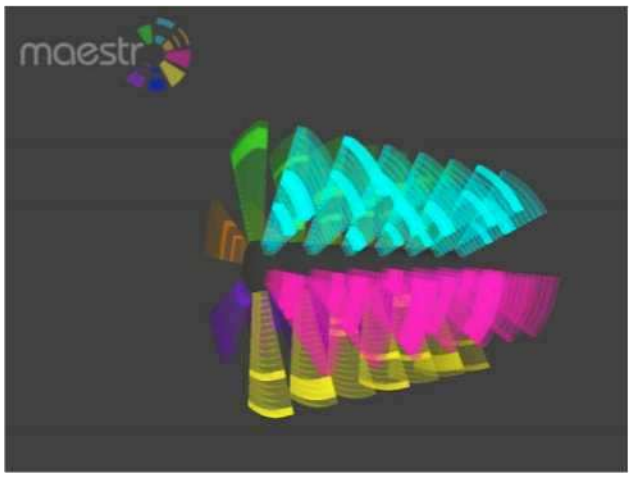


Figure 1. Monitoring Mode

Maestro's monitoring mode can be used as a screen-saver or can be displayed on an individual screen, so that each user in a

group will have the chance to see the density of the week or the day at a glance.

3.2 Interactive Mode

By turning on this mode user can access the details of her schedules for the entire week. There are two menus designed in Maestro: first menu enables users to view the activities day by day and the second menu enables users to view the weekly program of a selected schedule.

In this mode the user can also screen the details of an entry by clicking on the related slice.

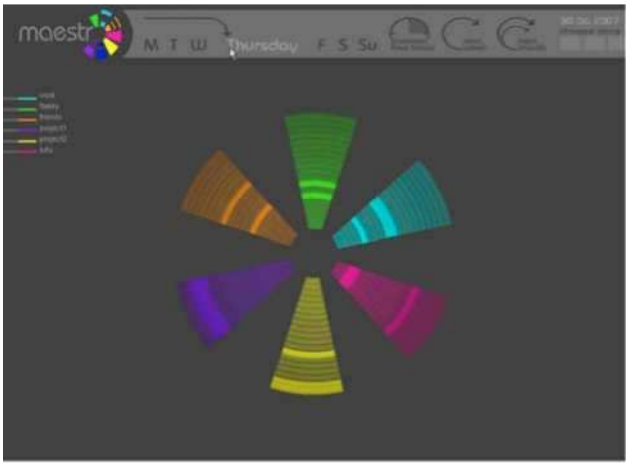


Figure 2. Interactive Mode

As the daily agenda is displayed on the screen, by enabling the "common time " option the user can highlight the available hours of the day.

Figure 3. Common Free Time Mode

The demo we have prepared, provides an insight to functioning of Maestro. [3]

4 Discussion & Conclusion

Like some former approaches [1], Maestro tries to visualize time in 3D space but it also lets the user display multiple time spans [2] at a glance in Monitoring Mode. In Monitoring Mode, users can not only see the load of a week, but can also figure out the busyness of each day in that week. One major strength of Maestro is that it displays diverse schedules of a day on one screen, to make it possible for the user to unite the different levels of data and compare them.

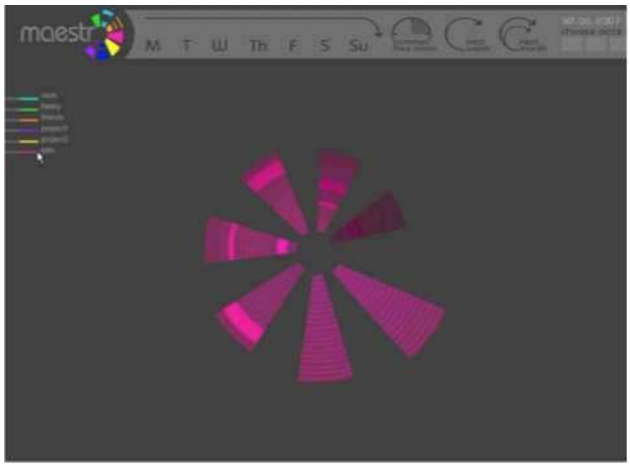


Figure 4. Weekly Display of a specific Schedule

Since Maestro is a term project of the "CS450: Arts and Computing" course offered in "Sabancı University", there should be some further improvements on it. First of all, a reminder option to alert some activities, the switching mode between users etc. may be added. Animation in Monitoring Mode can also be improved.

References

[1] Jock D. Mackinlay, G. G. (1994, November 2). Developing Calendar Visualizers for The Information Visualizer. *UIST*,pp. 109-118.

[2] Tessler, F. N. (1993, Oktober 7). Desktop Calendars.*MacWorld,* pp. 104-109.

[3] http://graphics.sabanciuniv.edu/cs450-projects/maestro/ maestro video.avi

Visual Clustering in Parallel Coordinates

Hong Zhou†* Xiaoru Yuan§† Baoquan Chen§‡ Huamin Qu†§

Department of Computer Science & Engineering, Hong Kong University of Science & Technology, Hong Kong†
Department of Computer Science & Engineering, University of Minnesota, MN, USA§

1 Introduction

Parallel coordinates [2] have been widely used for the visualization of high dimensional and multivariate datasets. In parallel coordinates, a point in the original high dimensional space is plotted as a polyline connecting all dimension axes. Parallel coordinates can be considered as special node link diagrams, in which the nodes are on the parallel axes. The edges are those polylines linking the nodes of two neighboring axes.

Although parallel coordinates have been proven to be an effective tool in general, edge clutter prevents the revealing of underlying patterns of a medium to large sized multivariate data set. The main cause of the visual clutter comes from the super density of the polylines. Most of the existing clutter reduction efforts are mainly data centric, i.e., data are clustered before they are plotted. Instead, here we perform clustering by geometrically deforming and grouping polylines while they are plotted. Our clustering is achieved through analyzing geometric relationships between polylines rather than the data itself. Specifically, an optimization system is designed to minimize the curvatures of edges and maximize the parallelism of adjacent edges through an energy function. By adjusting the weight of each term in the energy function, the user is capable of controlling the level of visual clustering according to her/his preference.

2 The Visual Clustering Algorithm

This section describes our visual clustering algorithm for parallel coordinates based on energy minimization and a color and opacity enhancement scheme.

2.1 Modeling the Visual Interaction Between Lines

Since it is difficult to directly measure the visual clutters of a parallel coordinates drawing quantitatively, we develop a simplified model to represent the interaction between the individual lines of the parallel coordinates.

We model the parallel coordinates as a system with force interaction between lines. The force is defined towards reducing the visual interference between edges. The status of the system can be described as the energy level of the whole system. By allowing edges to be curved and their shapes adjustable, visual clutter can be reduced. After computing a system status with minimized energy, the optimized configuration of the parallel coordinates can be obtained. Effectively, our algorithm achieve visual clustering without data classification.

*e-mail: zhouhong@cse.ust.hk
†e-mail: xiaoru.yuan@gmail.com
‡e-mail: baoquan@cs.umn.edu
§e-mail: huamin@cse.ust.hk

In our model, the total energy of the edges can be divided into two major terms and represented by the following equation:

$$\mathbf{E} = \alpha_c \mathbf{E}_{curvature} + (1 - \alpha_c)\mathbf{E}_{gravitation} \quad (1)$$

where $\mathbf{E}$ is the total energy of the whole system, $\mathbf{E}_{curvature}$ is the energy term representing the bending of lines, $\mathbf{E}_{gravitation}$ is the energy term representing the interactions between neighboring line pairs, α_c and $(1 - \alpha_c)$ are the weighting coefficients of the energy term $\mathbf{E}_{curvature}$ and $\mathbf{E}_{gravitation}$ respectively.

In the following text, we describe each energy term and relate its definition to the visual clustering effect we intended to achieve through energy minimization. As a design guideline, we intend to maintain a linear system so that it can be solved efficiently through linear programming.

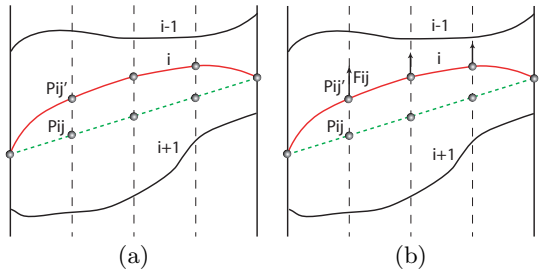


Figure 1: Energy Terms: (a) Curvature energy term; (b) Gravitation energy term.

Curvature Energy Term:

In our model, we allow edges to be bent to reduce excessive intersections between lines. We set a curvature energy term that prevents lines from being bent too much. For a given drawing, we assume there are n data points (lines in the graph). The curvature energy term is defined as:

$$\mathbf{E}_{curvature} = \sum_{i=1}^{n} \sum_{j=1}^{m} |P'_{ij} - P_{ij}| \quad (2)$$

where P'_{ij} is a control point on a curved line i (the red line in Figure 1(a)). Altogether there are m control points sampled for each line. Point P_{ij} is the corresponding point on a straight line connecting the same two end points as P'_{ij} is associated with (the green line in Figure 1(a)). The more bending the curve, the larger the curvature, and the longer the distance between the P'_{ij} and P_{ij} pair, therefore, the higher the energy contribution from the curve.

Gravitation Energy Term:

We use another term called gravitation energy term to model the interaction between neighboring lines. Again for the purpose of minimizing excessive intersections between lines, it is desirable to have neighboring lines as parallel as possible and parallel lines are pulled as close as possible. For

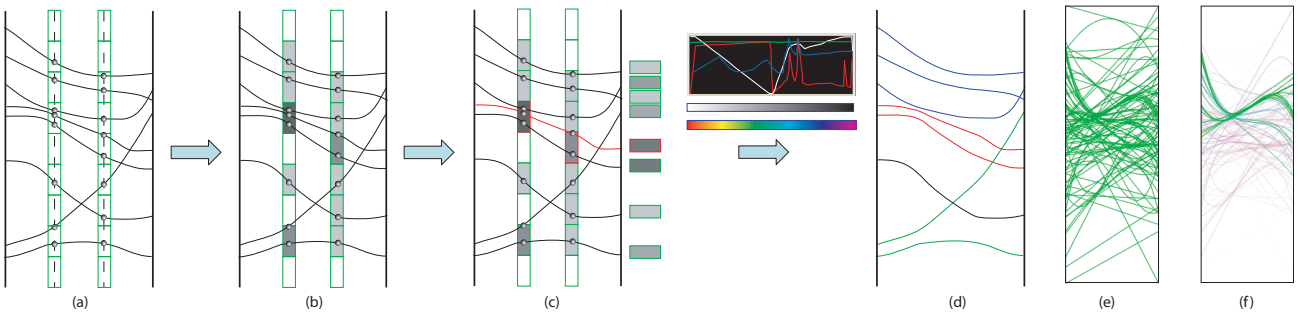


Figure 2: Applying color and opacity based on the line density: (a) Density bins for each control point column; (b) Accumulation of line density for each bin; (c) Computation of the local density for each line by averaging the density values of all its control points; (d) Applying color and opacity based on the user specified transfer function; (e) Parallel coordinates before applying color and opacity; (f) Parallel coordinates plot after applying color and opacity.

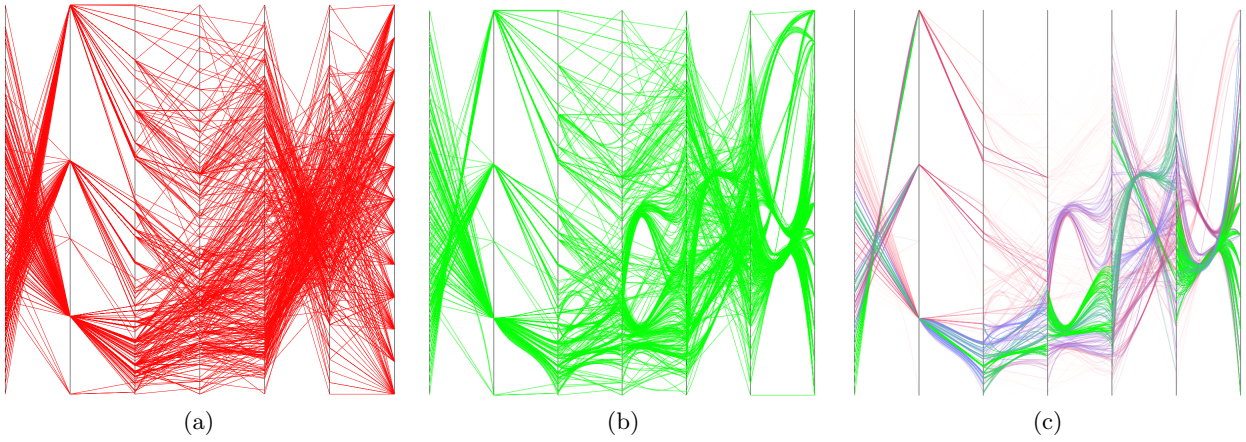


Figure 3: Application of visual clustering on a data set with 7 variables. (a) original plot; (b) after visual clustering optimization; (c) with a transfer function applied. Data from *http://lib.stat.cmu.edu/datasets/cars.data*.

the n data points (lines in the plot), the gravitation energy term can be modeled as:

$$\mathbf{E}_{gravitation} = \sum_{i=1}^{n}\sum_{j=1}^{m} -F_{ij} \cdot (P'_{ij} - P_{ij}) + E_{ij} \qquad (3)$$

where F_{ij} is the force computed based on the initial state of the neighboring edge arrangement, $P'_{ij} - P_{ij}$ is the moving distance of the control point ij along the force direction as illustrated in Figure 1(b). Note that the negative sign is to let the energy becomes lower when the control points move following the force directions. E_{ij} is defined to keep the relatively vertical order of control point ij with non-intersecting edges. If point ij is still "in order", $E_{ij} = 0$. If point ij crosses over the corresponding points of the nearest non-intersecting edges, $E_{ij} = infinite$. The force F_{ij} for each control point is computed as the summation of its interaction with all the neighboring edges.

2.2 Color and Opacity Enhancement

Applying transparency to the parallel coordinates drawings can highlight different aspects of the data [3]. In our work, to further improve the visual effectiveness, we assign color and opacity to the plot lines. After the optimization, lines with similar properties are already aggregated together into bundles. Therefore, we assign opacity and color to each line according to its local density which is much easier to be computed than the data frequency and density information [1]. The computation of the local line density and the application of color and opacity is illustrated in Figure 2. We design an interface similar to transfer function specification in volume rendering to assign the color and opacity values based on the local line density. Users can interactively manipulate the transfer function and selectively enhance line bundles according to their local density values. Figures 2(e) and (f) are plots before and after the application of a transfer function respectively.

3 Results

We demonstrate our algorithm on a popular car information data set with 7 variables. Figure 3 shows our visual clustering results. Figure 3(b) is the visual clustering result over the original plot shown in Figure 3(a). As illustrated in Figure 3(c), a transfer function is applied to filter out clusters with lower line density, thus reducing their visual prominence.

Acknowledgements

This work was supported by RGC grant CERG 618706.

References

[1] A. O. Artero, M. C. F. de Oliveira, and H. Levkowitz. Uncovering clusters in crowded parallel coordinates visualizations. In *INFOVIS '04*, pages 81–88, 2004.

[2] A. Inselberg and B. Dimsdale. Parallel coordinates: a tool for visualizing multi-dimensional geometry. In *VIS '90*, pages 361–378, 1990.

[3] J. Johansson, P. Ljung, M. Jern, and M. Cooper. Revealing structure within clustered parallel coordinates displays. In *INFOVIS '05*, pages 125–132, 2005.

Concept Relationship Editor: A visual interface to support the creation of relationships between taxonomic classifications

Paul Craig*, Jessie Kennedy†

School of Computing, Napier University

ABSTRACT

This paper describes the Concept Relationship Editor, an interactive visualisation tool designed to support the specification of relationships between hierarchical taxonomic classifications. The tool operates using an interactive space-filling adjacency layout which allows users to expand multiple lists of taxa with common parents so they can explore and add relationships between two classifications. Whenever selected lists contain too many items for them to be legible within the restrictions of available screen space the user can alleviate the problem by either operating in 'lens mode' or 'scroll mode'. In 'lens mode' the layout is configured so that both of the classifications and all the relationships are completely visible on-screen. Here a fish-eye lens type distortion effect is applied under the cursor to allow taxa names with less assigned space to be made legible. In 'scroll mode' the layout assigns sufficient space for the labels of all expanded taxa lists to be legible and scroll bars can be used to navigate across the hierarchy of either classification. While the 'lens mode' provides context and allows for more direct comparison of relationships throughout the classifications, 'scroll mode' tends to allow for relationships to be added more efficiently between smaller groups of similarly classified taxa.

Keywords: Interaction, Applications of InfoVis, Hierarchy visualization, Focus+context

1 INTRODUCTION

The science of Linnaean taxonomy classifies specimens according to shared characteristics, into hierarchies of taxonomic concepts (taxa) of differing ranks. A typical classification includes anything from around 200 to 1,500 taxa spread over about four to fifteen ranks. These taxa are given scientific names e.g. Apium graveolens L. for the species commonly known as celery. Over time the relative importance of particular characteristics used in differentiating organisms changes, resulting in alternative classifications being proposed. This along with the rules of Linnaean taxonomy which control the naming of taxa has the effect of the same taxonomic name having differing definitions. These definitions are described in field guides used by biologists for identifying organisms.

When comparing data containing the scientific names of organisms recorded by biologists using different field guides there is often ambiguity over what a particular name means; this can cause errors in data analyses [1]. To resolve this problem, an increasingly important task for taxonomists and ecologists is to understand the relationships between alternative taxa so they can be taken into account when comparing data sets. An increasingly common approach being taken to define the relationships between taxa in alternative classifications is to use a set-based notation. This states the relationship between taxon pairs [2]. These relationships may be defined when a taxonomist is undertaking a new revision of a classification by relating the new taxa to previous taxa or by a third party taxonomists examining existing classifications.

Allowing a taxonomist to visualise the taxonomies in a familiar manner, to explore them and easily create relationships between them is a significant challenge. For the efficient creation of relationships, taxonomists must not only be able to effectively navigate the hierarchical structure of classifications but also be able to have different groups of similarly classified taxa clearly identifiable. The fact that groups of interrelated taxa are often dispersed differently in different classifications makes it necessary to have multiple different groups of taxa from each classification visible at the same time. The layout for individual classifications must also be such that when relationships are added they do not impede the display of taxa names and the information needed to add further relationships.

While there are many effective visualisation techniques for large scale hierarchical data [3-9] and a number of visualisations that also allow the user to explore relationships between or within hierarchies [3, 6, 7, 10], there are none that display different varieties of relationship and allow a user to create relationships. The one visualisation that does display multiple relationship types between hierarchies [11] is primarily designed to identify overlap between hierarchies with the same name. Synonymy relationships are only displayed when they exist between a selected taxon and the limited visible portions of each other hierarchy. When displayed these relationships also have a tendency to occlude taxa names so the interface is not suitable to be adapted for the addition of new relationships.

1.1 Concept Relationship Editor

A screenshot of the Concept Relationship Editor interface for exploring and editing relationships between taxonomic classifications is shown in figure 1. The main component is the classification panel. This has representations of each selected classification on each side of the display. Taxa are represented using labelled rectangles with adjacency used to indicate positions in the hierarchy. Relationships between taxa are represented using curved lines between the related taxa. These have symbols at either end to indicate the relationship type. Relationships not attached to a selected taxon are greyed out and relationships in the process of being formed (as in figure 1) are represented using a red dotted line.

Each classification is colour-coded, with taxa currently selected for focus (together with their ancestors in the hierarchy) coloured a lighter hue than their classification. The overall layout of the classifications attempts to make, where possible, the labelled names of taxa readable. Here priority is given first to selected taxa, then the children of selected taxa, then the siblings of selected taxa and finally the siblings of ancestors of selected taxa. These nodes require priority for navigation and relationship addition. When a user selects a new taxon, priority is

*e-mail: p.craig@napier.ac.uk

†e-mail: j.kennedy@napier.ac.uk

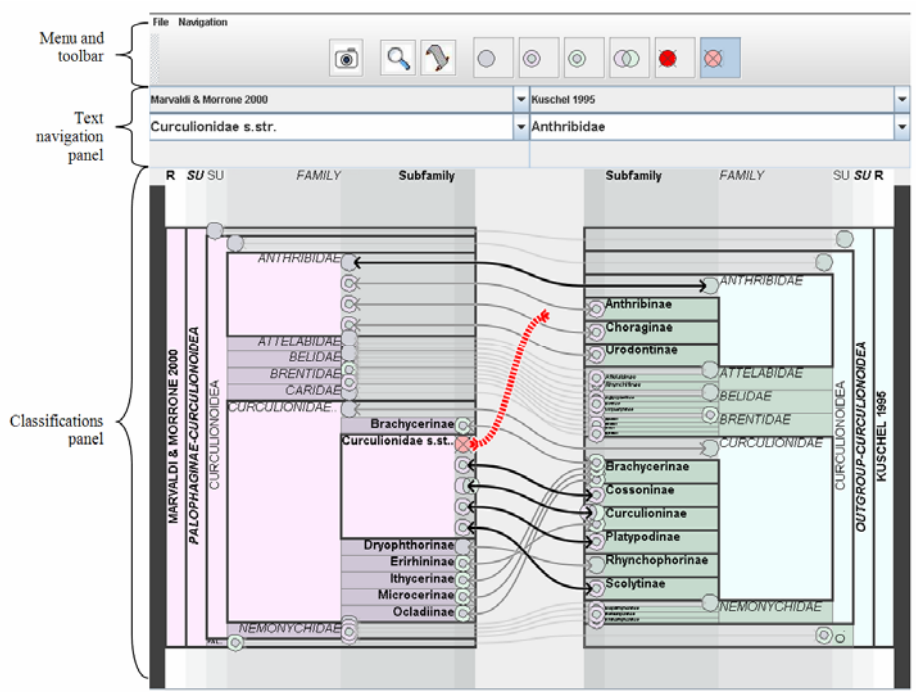


Figure 1. Screenshot of the Concept Relationship Editor interface.

appropriately reassigned and the layout changes. During the transition, animation is used to provide visual cues and aid reinterpretation.

When the layout algorithm cannot assign sufficient space to adequately display the labels of the prioritized taxa, the lens mode or scroll mode take effect to make taxon labels otherwise accessible. In lens mode a fish-eye lens [12] type effect is applied to the vertical axis under the mouse cursor. Here, the magnification is calculated to be just enough to expand the taxon directly under the cursor and make its label readable. Alternatively, in scroll mode, priority nodes for navigation are all sized above a threshold for readability and the user may be required to operate a scroll-bar to determine which taxa occupy the screen space. Whenever possible, scrolling is automated pre-empt the users actions .

2 Conclusion

We have developed an information visualisation technique for exploring and editing relationships between taxonomic classifications. The technique applies a novel focus + context adjacency layout for hierarchies that allows multiple groups of nodes to be expanded with their labels readable in horizontally stacked lists. The technique can be differentiated from other visualisations in that it allows users to edit and view different types of relationships between two hierarchies. During an informal evaluation of the interface users found it easy to navigate the hierarchy of classifications and add relationships between taxa.

References

[1] Kukla R and Paterson T. Scientific names are ambiguous as identifiers for biological taxa: their context and definition are required for accurate data integration. . 2nd International Workshop on Data Integration in the Life Sciences 2005 (San Diego, USA); 80-95.

[2] Berendsohn WG. The concept of "potential taxa" in databases. Taxon 1995; 22: 207-212

[3] Jeong C-S and Pang A. Reconfigurable Disc Trees for Visualizing Large Hierarchical Information Space. IEEE InfoVis '98 1998 (Research Triangle, North Carolina, USA), Computer Society Press; 19-25.

[4] Johnson B and Shneiderman B. Treemaps: A Space-Filling approach to the visualization of hierarchical information structures. IEEE Visualization '91 1991 (San Diego, California, USA), IEEE Computer Society Press; 284-291.

[5] Lamping J, Rao R, and Pirolli P. A focus + context technique based on hyperbolic geometry for visualizing large hierarchies. ACM CHI '95 1995 (Denver, Colorado, USA), ACM Press; 401-408.

[6] Munzner T. Exploring Large Graphs in 3D Hyperbolic Space. IEEE Computer Graphics & Applications 1998; 18(4): 18-23

[7] Neumann P, Schlechtweg S, and Carpendale S. ArcTrees: Visualizing Relations in Hierarchical Data. EUROVIS 2005: Eurographics / IEEE VGTC Symposium on Visualization 2005; 53-60.

[8] Sifer M. Filter co-ordinations for exploring multi-dimensional data. Journal of Visual Languages & Computing 2005; 17(2): 107-125

[9] Wang W, Wang H, Dai G, and Wang H. Visualization of large hierarchical data by circle packing. SIGCHI conference on Human Factors in computing systems 2006 (Montréal, Québec, Canada), ACM Press; 517-250.

[10] Holten D. Hierarchical Edge Bundles: Visualization of Adjacency Relations in Hierarchical Data. IEEE Transactions on Visualization and Computer Graphics (TVCG; Proceedings of Vis / InfoVis 2006) 2006; 20(5): 741 - 748

[11] Graham M and Kennedy J. Extending taxonomic visualisation to incorporate synonymy and structural markers. Information Visualization 2005; 4(3): 206-223

[12] Bederson BB. Fisheye Menus. ACM UIST 2000 2000 (San Diego, California, USA), ACM Press; 217-226.

Interactive Poster: Effective Display of Conserved Domains on a Multiple Sequence Alignment

Andrew D. Lindeman*, Susan M. Bridges†, T.J. Jankun-Kelly‡
Department of Computer Science and Engineering and the Institute for Digital Biology
Mississippi State University, MS 39762

ABSTRACT

Multiple sequence alignment (MSA) is used to explore the similarity of several related protein sequences by providing a near optimal alignment of the characters in each sequence. Biologists require effective visualization of these alignments as part of their analysis. Although tools such as Jalview have been developed that provide a detailed view of different aspects of the alignments and metadata such as conserved domains, these tools do not automatically download the metadata via the web and the alignment cannot be viewed at varying levels of detail. We have developed a prototype MSA visualization application that focuses on simultaneous display of characteristics of the alignment and automatically downloaded metadata such as conserved domains, and allows the user to view the information at different levels of detail to enable easy recognition of interesting patterns for further analysis.

CR Categories and Subject Descriptors: H.5.2 [User Interfaces]: Graphical user interfaces (GUI), User-centered design; J.3 [Life and Medical Sciences]: Biology and genetics

Additional Keywords: information visualization, bioinformatics, multiple sequence alignment, conserved domains

1 INTRODUCTION & MOTIVATION

Proteins, DNA, and RNA are the basic molecules of life and each can be represented computationally as a sequence of characters. Of these, proteins are especially important because they are typically the molecules that carry out functional processes within cells. Proteins are made of chains of 20 different amino acids; therefore a protein sequence can be viewed as a string of characters from a 20 letter alphabet. For understanding proteins, multiple sequence alignment (MSA) has proven to be one of the most widely used bioinformatics methods because it allows biologists to analyze the similarities and differences of related proteins. Figure 1 shows a portion of a multiple sequence alignment of DNA methyltransferase proteins from 10 different organisms. The canonical representation of an MSA has each protein sequence on a separate line with matching characters aligned in columns and spaces inserted where necessary to improve the alignment. A series of one or more spaces is called a gap and is represented by dashes. The similarities and differences highlighted by multiple sequence alignments can lead to conclusions about the evolutionary history of the organisms, as well as information pinpointing functional parts of the sequences of each organism.

Biologists often want to investigate the "functional domains" of proteins. These are the sections of the protein's sequence that enable it to serve a particular biological role. Because these sections tend to be evolutionarily conserved (they remain the same in related organisms), they are also called conserved domains. After a sequence has been identified as a functional conserved domain experimentally or using predictive methods, a computer model of the domain can be generated (often using a Hidden Markov model). That model can be used to identify the domain in sequences from other organisms. There are many online databases that take a protein sequence as a query and return matching domains. For this application, the Conserved Domain Database (CDD) from NCBI [1] was used.

Jalview [2] is the tool that is most closely related to ours. Jalview displays an alignment, as well as many kinds of associated information including conserved domains. However, with Jalview, it is not possible to effectively view more than a few domains on an alignment, especially if the conserved domain matches overlap on a part of a sequence. Furthermore, it is not easily possible to get an overall view—across the entire alignment—of where each conserved domain lies.

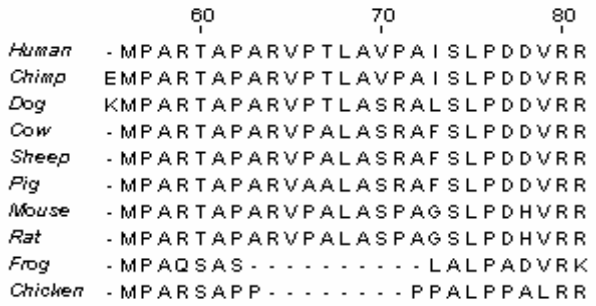


Figure 1. A view of a portion of an MSA in Jalview

Our application addresses this specialized problem by creating a new way to view an MSA concurrently with all conserved domains that match over the alignment.

2 METHODOLOGIES & USER INTERFACE

Our application was developed in Python, using wxPython for 2D graphics and BioPython for manipulating and managing the alignments and conserved domains.

Upon the input of a multiple sequence alignment file, the application queries the online NCBI CDD database for each sequence and parses the results. The first view presented is an overview of the alignment and the domains present as shown in Figure 2.

2.1 Alignment Overview

The initial overview alignment presents a condensed view of the entire alignment and associated conserved domains. In the example shown in Figure 2, the alignment spans amino acid indices 1 to 1709.

Unlike other systems for viewing alignments, this application gives an overview that initially focuses on each conserved

* e-mail: ADL91@msstate.edu
† e-mail: bridges@cse.msstate.edu
‡ e-mail: tjk@cse.msstate.edu

domain, rather than each sequence. The alignment is displayed separately for each domain and each domain is represented by a horizontal block. The background is a cream color whose saturation represents the strength of the alignment based on sum-of-pairs scoring at each position. A more saturated color represents a stronger match among the sequences at each column; a completely unsaturated color typically represents a column where many gaps were inserted.

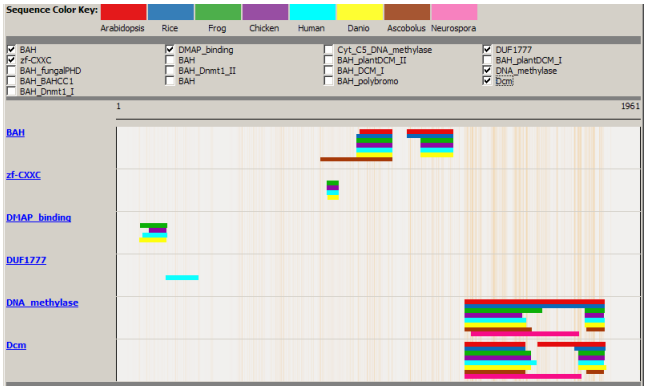


Figure 2. The initial overview of an alignment with 6 conserved domains displayed.

Each sequence in the alignment is assigned a color, as shown in the key at the top of the screen. In the overview, a bar of color is drawn above the background whenever the respective conserved domain is present in that sequence across a specified portion of the alignment. In this example, it is easy to tell that the DNA_methylase domain (second from bottom) is present in every sequence near the end of the alignment. Similarly, the DUF1777 domain (fourth from top) near the beginning of the sequence is clearly only present in human. Furthermore, even though the last two domains overlap, it is still easy to see where they lie on the alignment since they are displayed on separate tracks.

This overview can be useful for drawing general conclusions about some parts of the alignment. Often, however, this view will motivate the user to look more closely at a specific part of the alignment. The application allows the user to click on the blue, underlined hyperlinks to view a single conserved domain's relationship on the alignment in greater detail.

2.2 More Detailed Views

Figure 3. Overview of a single conserved domain.

This detail view appears below the overview, so that both are shown at the same time (a cropped view is shown in the figure). Initially, this view is very similar to the overview, in that it spans the entire alignment and has the same cream color background representing areas where the specific conserved domain is not present. However, in this view, the saturation of the colors that represent each sequence is adjusted in the same way the background is. This means the alignment strength can be visualized in conjunction the conserved domain. It may be important to compare the strength of the alignment in a domain.

To zoom in on a certain area of the alignment, a user can click and drag a rectangle around the area of interest. Upon releasing the mouse, the alignment window will redraw with only the selected area. When the user selects an area small enough to make drawing the individual amino acids characters viable, these are also displayed.

Figure 4. Close-up view of the alignment showing individual amino acids.

Figure 4 shows a portion of the alignment where most of the sequences have the DNA_methylase domain present. In this view, a user can see each individual amino acid in every sequence, in addition to the strength of the matches represented by the saturation of the colors. In this example, it is easy to see that at position 1511 (the first position displayed), all of the amino acids match in every sequence, so the color is highly saturated. However, in the middle of the view as the alignment strength lessens (gaps are inserted and columns are mismatched), the colors become less saturated.

Users can move up and down the alignment using the scroll button on their mice, as well as the arrow keys on the keyboard. Additionally, users can click and drag the sequence labels on the left to move that sequence higher or lower in the stack, for closer analysis of a subset of the sequences, while still keeping information about all the sequences available.

If the user zooms in on a specific portion of the alignment, a selection box is drawn on the overview to represent which section of the alignment the user is viewing in relation to the whole alignment, as well as the other domains. If a user selects another domain, a new detail view will appear for that domain; additionally, the zoom factor will stay the same. Users can easily zoom back out to the full view by clicking the right mouse button.

3 Discussion & Conclusions

Our prototype application demonstrates a new method for displaying multiple sequence alignments and conserved domains at different levels of detail. We plan to investigate methods for displaying additional layers of information on the display, such as predicted DNA binding sites. We also plan to conduct user studies with biologists to evaluate the effectiveness of the display.

Acknowledgments

This research was supported by an NSF EPSCoR award, grant number EPS-0556308.

References

[1] Marchler-Bauer, A., Anderson, J. B., DeWeese-Scott, C., Fedorova, N. D., Geer, L. Y., He, S., Hurwitz, D. I., Jackson, J. D., Jacobs, A. R., Lanczycki, C. J., Liebert, C. A., Liu, C., Madej, T., Marchler, C. H., Mazumder, R., Nikolskaya, A. N., Panchenko, A. R., Rao, B. S., Shoemaker, B. A., Simonyan, V., Song, J. S., Thiessen, P. A., Vasudevan, S., Wang, Y., Yamashita, R. A., Yin, J. J. and Bryant, S. H. CDD: a curated Entrez database of conserved domain alignments. In *Nucleic Acids Research*, volume 31, 383-387, January 2003.

[2] Clamp, M., Cuff, J., Searle, S. M., Barton, G. J. The Jalview Java alignment editor. In *Bioinformatics*, volume 20, number 3, 426-427, February 2004.

Interactive Poster: Plugable Lenses for Interactive Visualizations

Georg A. Fuchs* Conrad Thiede† Heidrun Schumann‡

Institute for Computer Science
University of Rostock

ABSTRACT

The size of today's typical data sets calls for visualizations that abstain from showing maximum detail, focusing instead on relevant information. One approach is to filter the data with lens techniques, i.e. spatially confined filters controlled interactively by the user. Lenses can also be 'stacked' to create complex filters.

Usually, lenses are tightly integrated into the interaction component of the visualization system. The utility of a given lens, however, depends on the data being examined. Different data sets may require diverse lens functionalities. We report on a framework that allows to define such lenses as independent plug-ins rather than as integral parts of the interaction logic, and that also provides a method to detect possible conflicts when combining lenses.

Index Terms: H.5.2 [Information Interfaces & Presentation]: User Interfaces—Interaction styles I.3.6 [Computer Graphics]: Methodology and Techniques—Interaction techniques

1 INTRODUCTION

The size of today's typical data sets increases steadily. Consequently, when visualizing such data not the maximum detail is displayed. Instead, only information relevant to a specific problem are presented. There exist a number of approaches to filter the data accordingly, among them the so-called lens techniques. A lens allows the user to examine the current region of interest by interactively positioning it over a visual representation that provides a general overview of the data.

The lens acts as a spatially confined filter, bounded by the lens region, that adapts the representation locally. The type of adaptation is determined by the lens function. Lenses can be broadly categorized into graphical and sematic lenses [4] depending on whether they modify only the graphical representation or also the information content itself, respectively. Fig. 1 and [4] give examples for both types of lenses.

Another intriguing aspect is the possibility to combine, or stack, multiple lenses to apply several local modifications to the visualization process simultaneously [1, 4]. However, this gives rise to the question how to determine which types of lenses are combinable in a useful and conflict-free way, and if so, in which order to apply the respective modification functions. For example, a conflict occurs if two lenses are stacked that both modify the color mapping.

The challenge is to integrate arbitrary lenses with a given visualization technique in an effective way. Currently, visualization systems that offer lens-type interaction rely on a tight integration of the required functionality (i.e. it is usually compiled statically into the system), see e.g. [3, 4]. This limits the flexibility to adapt the system to new types of data. Opposed to this, a plug-in architecture would allow to define new lenses suited to different data sets. In this poster we report on a new approach that addresses this challenge.

*e-mail: georg.fuchs@uni-rostock.de
†e-mail: conrad.thiede@uni-rostock.de
‡e-mail: heidrun.schumann@uni-rostock.de

2 GENERAL CONCEPT

Our approach is based on Chi's Data State Reference Model (DSRM) [2]. It describes the visualization process as a pipeline of four stages (data states) and operators on the data. Stage operators work within a single stage, while transformation operators transfer data from one state to another [2]. Raw data (1st stage) is transformed into analytical abstractions (2nd stage), e.g. by calculating statistical moments, which are then further mapped to visual abstractions (3rd stage), e.g. Scatterplots or 3D surface representations. Finally, the rendering process generates the graphics primitives that constitute the image data (4th stage).

Furthermore, the data states can be divided into two groups [2]. The first two states define the (abstract) data space, whereas the last two describe data in view space. This corresponds to the distinction between semantic and graphical lenses [4]. In fact, we can understand the lens function as data operators in the DSRM.

Integration into the Visualization Process

There are basically two ways to integrate a lens into the visualization process [3]:

- Two separate pipelines are used, one for creating the underlying visualization and one for the lens region.
- The entire image including the lens region is produced using a single pipeline. The lens function alters data on the different stages directly.

Although with the first approach it is easier to modularize it can not be used for stacking lenses, as their pipelines process the data independently unless elaborate interprocess communication is used. Therefore, we opted to implement a single visualization pipeline that is modified directly by all active lenses.

A lens can be parameterized by its position and the region (around that position) in which it affects the underlying visualization. Since an interactive lens is always positioned by the user in view space, the lens region is also given in view space (screen coordinates). However, the lens function may also operate on data from previous stages. Therefore, a description of the data stages and a mechanism to define the lens region for each are needed.

Data Stage Description

Chi observed in [2] that the dependency of the operators on the data characteristics gets more pronounced the closer an operator is to the raw data stage of the pipeline. For example, modifying pixel colors can be done regardless of the underlying data structure, while parsing a file requires far more information on the raw data format.

In order to avoid explicit dependencies between the semantics of the data and the description of the lens functions, we use a set theoretical approach. Each of the four stages in the DSRM is represented by a set A_i of attributes $a_{ik}, 1 \leq i \leq 4, 1 \leq k \leq |A_i|$ that are associated with the data elements at the corresponding state. Elements in A_1 correspond to raw data values, whereas elements from A_4 represent either vertex data (3D case), or pixel attributes (2D case). Therefore, a lens can be applied to multiple visualizations as long as the attribute sets affected by the lens are present in both visualizations, i.e. both have 'compatible' attribute sets.

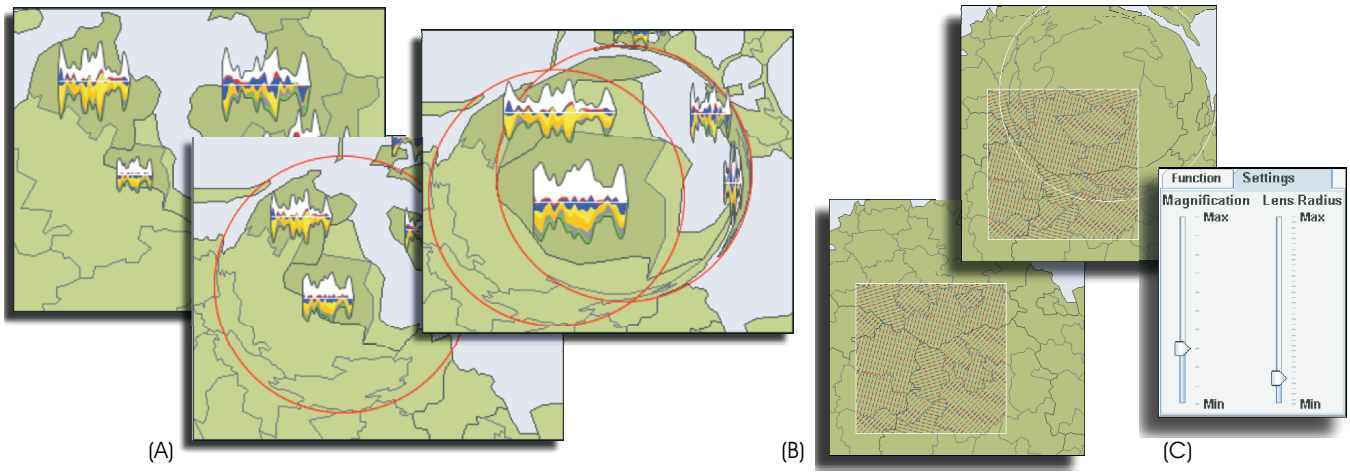


Figure 1: Examples for lens combinations: two graphical fisheye lenses (A), fisheye over texture lens (B), lens-specific configuration dialog (C).

Lens Region Definition A method to project the lens region into the data space is required for lenses that affect data elements from previous pipeline stages A_1 to A_3, i.e. semantic lenses. Thus, the definition must be:

- valid on all stages of the visualization pipeline,
- able to describe arbitrary shapes and
- capable to determine, for each data element, if it is inside or outside the lens region.

In our prototype, the lens region is therefore defined by applying a threshold operator to the attribute values for each (relevant) attribute in the set A_i. This corresponds to an intersection of half spaces.

One major challenge with this, however, is the necessary 'back-projection' of the lens regions to lower stages of the pipeline. While this is easy for the view space (basically a primitives picking problem), the often complex mapping processes that transform raw data to analytical and on to visual abstractions may not be readily invertible. In fact, in our implementation currently only simple mappings such as value-range to color scale or primitive sizes are used. We are currently investigating how to best model more complex mappings, e.g. non-invertible projections by means of lookup tables.

Sorting and Conflict Detection Two or more lenses stack when their (projected) regions overlap on at least one data stage (cf. Fig. 1). Since we work on a single visualization pipeline, the associated lens operations must be applied to the data sequentially, and ambiguous (conflicting) mappings must be prevented.

An operation f transforms data elements with associated attributes from the source S_n^f (n^{th} stage) to a target set T_m^f (m^{th} stage):

$$S_n^f \leq \bigcup_{1\leq i\leq n} A_i, \quad T_m^f \leq \bigcup_{m\leq j\leq 4} A_j$$

$$f : S_n^f \rightarrow T_m^f; 1 \leq n \leq m \leq 4$$

The lens operations are executed in the order of their source set index i, i.e. attributes are only used as elements of an input set after all modifying operations have completed. Two operators f, g ($g : S_p^g \rightarrow T_q^g; 1 \leq p \leq q \leq 4$) may principally write data to the same target stage. A conflict only occurs if $T_m^f \cap T_q^g \neq \emptyset$, i.e. at least one data attribute a_{ik} is modified by both. For some of these conflicts, viable solution strategies exist, however a detailed discussion is beyond the scope of this abstract.

3 Implementation Details

A prototype of the proposed lens plug-in framework was implemented based on a system for the visualization of demographic and health data over geographic maps (cf. [4]). The user can select a base visualization (e.g. icons, see Fig. 1) and a lens that can be interactively positioned using the mouse. To stack lenses, the current lens can be 'locked' in place, after which another, possibly different, lens technique can be attached to the mouse cursor (Fig. 1).

At the core of the plug-in architecture is an abstract 'BaseLens' class. It provides the necessary hooks into the visualization pipeline that allows input data retrieval and write back of the transformed result for every data stage, and common notification methods to control when a lens function is processed. Concrete lens implementations derive from this class by overloading methods for the appropriate stage(s). Furthermore, an XML manifest file is provided that specifies the attribute subsets S_n, T_m on the source and target stages, respectively, that are used by the framework for execution sequencing and conflict detection.

Lenses can also supply custom settings/configuration dialogs that are embedded into the system's GUI. The prototype uses a simple XML notation that allows the declaration of mutable parameters together with the value range and the type of GUI element (e.g. slider, drop-down list) that is used for manipulation.

4 Conclusion

We presented a new approach to use lens techniques as plug-ins to a visualization framework prototype using a set theoretical description of the DSRM. Future work will include studies on how to define 'compatible' attribute sets (cf. Sec. 2) by means of suitable metadata descriptions, and the extension of the lens region's back-projection for more complex mapping operators.

References

[1] E. A. Bier, M. C. Stone, K. Pier, W. Buxton, and T. D. DeRose. Toolglass and Magic Lenses: The See-Through Interface. *Computer Graphics*, 27(Annual Conference Series):73–80, 1993.

[2] E. H. Chi and J. T. Riedl. An operator interaction framework for visualization systems. In *Proceedings InfoVis'98*, pages 63–70, 1998.

[3] G. Ellis and A. Dix. Enabling automatic clutter reduction in parallel coordinate plots. In *IEEE Transactions on Visualization and Computer Graphics*, volume 12, pages 717–724, 2006.

[4] H. Griethe, G. Fuchs, and H. Schumann. A classification scheme for lens techniques. In V. Skala, editor, *Proceedings WSCG'2005*, pages 89–92, 2005.

ThisStar: Declarative Visualization Prototype

Joseph A. Cottam*
Computer Science Department, Indiana University

Andrew Lumsdaine†
Computer Science Department, Indiana University

ABSTRACT

Library-based and pre-compiled visualization tools incur many penalties that hinder the adoption of visualization as a technique for many fields. Libraries necessitate familiarity with the data structures and control flows that are incumbent in traditional programming, but not central to visualization. Task-specific visualization applications alleviate these needs, but induce users to move data between applications as their needs change. An ever expanding tool chain and corresponding context switches are inefficient. We propose a generative programming approach to visualization tool construction based on domain specific languages. Through it, we provide the flexibility of a general purpose programming language that abstracts out many of the control flow and data structure issues. Our initial prototype, ThisStar, creates star-maps based on these general ideas. ThisStar demonstrates the viability of these concepts and illuminates opportunities.

Keywords: Star Map, Visualization, Domain Specific Language, Declarative Language, Generative Programming.

Index Terms: I.3.4 [Computer Graphics:]: Graphics Utilities—Software support I.2.2 [Computing Methodologies:]: Automatic Programming—Program synthesis D.3.2 [Programming Languages:]: Language Classifications—Specialized application languages H.1.2 [Information Systems:]: User/Machine Systems—Human information processing

1 INTRODUCTION

Information Visualization has encountered many roadblocks to its adoption beyond academic research or very isolated applications. Common practice follows one of two routes: 1) Employ or train an expert to handle the visualization task; 2) Purchase off-the-shelf software. These two methods incur penalties that limit their utility beyond research applications or very specific problems as they both imply high up-front costs (e.g. employing experts or evaluating/purchasing off-the-shelf applications) and high maintenance costs (e.g. retaining experts or a lengthening tool chain as more task-specific applications are purchased). in short, there is no way for visualization problems to 'played with' without either committing substantial resources or limiting their scope in early stages.

We believe that a declarative programming language that holds close to the core concepts of information visualization will ease the adoption and, combined with a generative programming style, will ease maintenance difficulties. The need for flexible, user extensible visualization has been recognized before in [5], but our system goes a step further by allowing the visualizations themselves to be specified, not just the coordination between them. The need to improve the programming of visualizations was recognized by [3], but the Processing system retains a traditional language structure while simplifying other parts of the process. To expand beyond this prior work, we describe the embodiment of our concepts in an abstract Tuple-Space Mapper (TSM) and a preliminary implementation of them in ThisStar, an for the creation of star maps.

*e-mail: jcottam@cs.indiana.edu
†e-mail: lums@cs.indiana.edu

2 TUPLE-SPACE MAPPER

The Tuple-Space Mapper aims to reduce the problems associated with adoption of visualization. There are three central concepts in the TSM. First, most visualization is mapping entity features to visual attributes. Second, a domain specific language (DSL) provides a better conceptual fit than a general-purpose language but need not be less flexibile. Third, modification of the visualization should be accessible for rapid prototyping and extension. (A more complete description of the TSM can be found in [1].)

The first concept leadss to the formulation of tuple streams. Tuples are an elemental data representation that can be used to express many data types, and arbitrary attributes on them. By using streams of tuples, we alleviate the programmer of the burden of understanding the specifics of the data storage (a requirement and major obstacle when using contemporary libraries directly in a traditional language). This lets the programmer focus on the core operation of mapping the incoming tuple characteristics to visual characteristics on corresponding glyphs. This presents a challenge for the TSM system as it must infer proper data storage from a minimal description of the input data. This is not simple, but aided by the restriction that all programs generated by the TSM are visualization focused.

To expose these conceptual abstractions, we employ a DSL. This allows the tuples, mapping actions and selection actions to be directly expressed a straightforward manner. To provide the full flexibility of a general purpose language, we provide a mechanism to execute arbitrary code from the host application. Our implementation is patterned after the ad-hoc syntax-directed transformations of compiler compilers (such as YACC and ANTLR). This mechanism should remain constant as we migrate the TSM application to different host environments (regardless of the original host language). It also resembles the 'advanced coordination' option of the declarative coordination language presented in [5]. We employ a declarative language style as it generally incurs a low conceptual overhead and (combined with the simple data model) makes it easy to 'play' with a visualization specification. Since the TSM language implies no control structures, these must be inferred through analysis in the TSM Generator. Since data structures are also generated by the TSM system, this is simplified but remains non-trivial.

The TSM system does not generate complete visualization applications; instead it is a lightweight tool that generates components used in larger applications. These components can be based on any existing visualization library (see Figure 1). It is the responsibility of the TSM generator to convert the abstract mapping actions into concrete data structures and control flows. The resulting components conform to a straightforward interface, allowing the components generated from successive iterations of a specification to be interchanged (provided the characteristics of the incoming tuple streams remain congruent). The accessibility of the language permits non-experts to modify the visualization and the common interface allows new revisions to be simply incorporated into existing TSM software. However, since the TSM generator is only used when programming a visualization, it is not part of the tool chain durring analysis operations. Presenting a consistent interface despite a variety of inputs and potential outputs is another major implementation challenge for the TSM system.

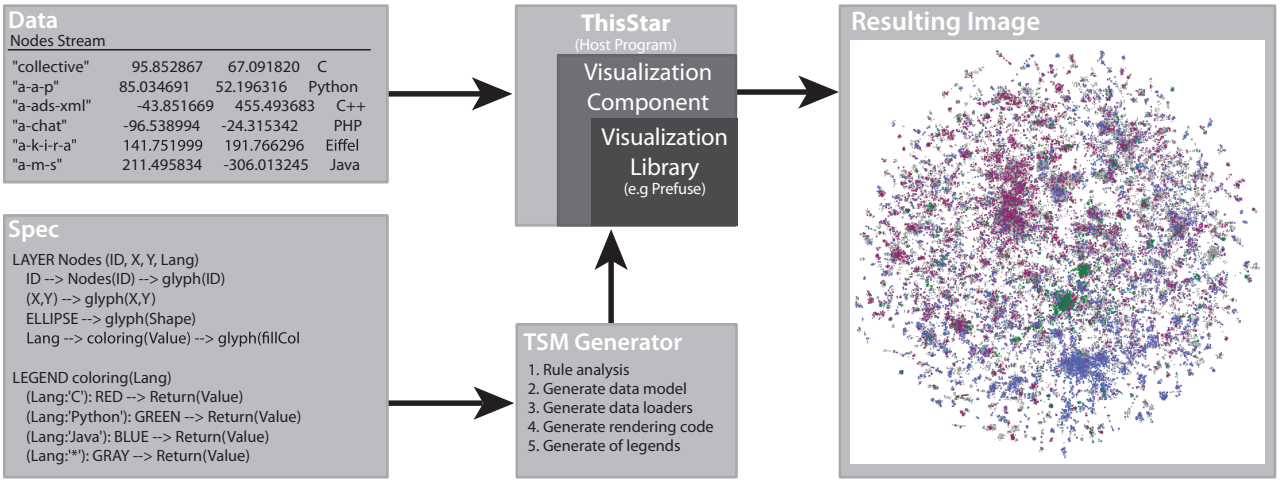


Figure 1: General architecture of a TSM application.

The concepts do three things to avoid the problems encountered in contemporary methods. First, they present a low barrier to entry by keeping visualization specific concepts in the foreground. Second, they provide a high degree of customizability. This allows applications built on the TSM to be extended, reducing the need to introduce new applications into a tool chain (the TSM generator is either orthogonal to the analysis tool chain or incorporated into an existing application). Finally, the simple data model facilitates chaining TSM-derived applications to existing tools when required by having a very simple, but general data structure.

3 ThisStar

ThisStar is an application to generate star maps. It implements several of the ideas presented above by providing: A simple DSL/generation system encapsulating the core concepts described above and a host application. The DSL engine configures a Prefuse [4] back-end to visualize data contained in delimited text files. Our selection of Prefuse means the TSM applications can access libraries written in Java ad-hoc fashion (e.g. for analysis). Specifications, given as a whole, are used to configure a run-time visualization system (no program restart is required, though this is technically a compile step it is most similar to dynamic compilation [2]). The host-application portion of the ThisStar system creates the tuple streams from files and coordinates the replacement of the visualization as the specification is changed.

4 Results

The initial system was used to visualize the interactions of open-source software development projects. The TSM language subset implemented in ThisStar was simple enough that small permutations to rules could be automatically generated and fed into the program. This automated the process of visualizing many similar but distinct aspects of our data. Results from the VxOrd layout application were easily presented as tuples to ThisStar and integrated with information from a database. Images from all rule permutations were then reviewed by hand for features of interest. Final composite images were generated in ThisStar using hand-modified rules to accent interesting points.

ThisStar is a success for several reasons. First, we were able to automatically generate the visualization rules required to show the data set (enabled by the simplicity of the DSL). Second, we were able to easily blend data from several sources (enabled by the simple data model). Finally, we were able to modify the initial rules in a straight-forward, interactive fashion to produce our eventual visualizations (enabled by the declarative DSL and the ability to iteratively the declaration).

5 Future Works

The results of ThisStar were generally successful, but highlighted areas for improvement. It is clear that better facilities for interaction with the resulting visualization are necessary. This will be part of our next iteration, representing the mouse and keyboard as tuple streams. Another avenue for improvement is the inclusion of reflexive constructs to the TSM language to allow visualizations to be self-referential in declaring mappings and conditions. These improvements will expand the range of applications that can be generated from a TSM-derived language. To demonstrate that the TSM concepts expand beyond Prefuse, we are currently working on generators to target Piccolo and Processing. We eventually plan to target non-Java libraries.

6 Conclusions

Our results from ThisStar are encouraging. Declarative languages have been successful in broadening the adoption of technologies in the past, and the success of ThisStar makes us believe that the same may apply for visualization in the future.

Acknowledgements

This work was supported in part by a grant from the Lilly Endowment.

References

[1] J. A. Cottam. Tuple space mapper: Design, challenges and goals. Technical Report TR648, Indiana University, Bloomington, IN, June 2007.

[2] K. Czarnecki and U. W. Eisenecker. *Generative Programming: Methods, Tools and Applications*. ACM Press/Addison-Wesley Publishing Co, New York, NY, 2000.

[3] B. Fry. *Computational Information Design*. PhD thesis, Massachusetts Institute of Technology, 2005.

[4] J. Heer, S. K. Card, and J. A. Landay. Prefuse: a toolkit for interactive information visualization. In *CHI '05: Proceeding of the SIGCHI conference on Human factors in computing systems*, pages 421–430, New York, NY, USA, 2005. ACM Press.

[5] C. L. North. *A User Interface for Coordinating Visualizations Based on Relational Schemata: Snap-Together Visualization*. PhD thesis, University of Maryland, College Park, May 2000. Chair-Ben Shneiderman.

Indexing Similarity Visualization over the Medial Subject Headings (MeSH)

Haixia Du*
Sterne, Kessler, Goldstein & Fox, P.L.L.C.

Terry Yoo†
National Library of Medicine, National Institutes of Health

ABSTRACT

We present an interactive visualization system for the evaluation and comparison of medical literature indexings across the Medical Subject Headings (MeSH) structure in a **radial tree** layout. Users can explore the MeSH structure in different level of details by selecting any index term as the center of the radial tree surrounded by other terms in its hierarchy. It also displays indexing results of MEDLINE documents in the form of MeSH term collections along with similarity measurements between different classifications in 2D color coding and 3D height field over the MeSH layout. This permits the evaluation of the automatic indexing system, Medical Text Indexer (MTI), compared with human indexers. The visualization of MeSH structure along with indexing collections provides a visual analytic tool for MeSH related data retrieval and analysis in the biomedical community.

Index Terms: E.1 [DATA STRUCTURES]: Graphs and networks,Trees—, H.3.1 [INFORMATION STORAGE AND RETRIEVAL]: Content Analysis and Indexing—, H.4.3 [INFORMATION SYSTEMS APPLICATIONS]: Communications Applications – Information browsers—, H.5.2 [INFORMATION INTERFACES AND PRESENTATION (e.g., HCI)]: User Interfaces—, J.3 [LIFE AND MEDICAL SCIENCES]: Medical information systems—

1 INTRODUCTION

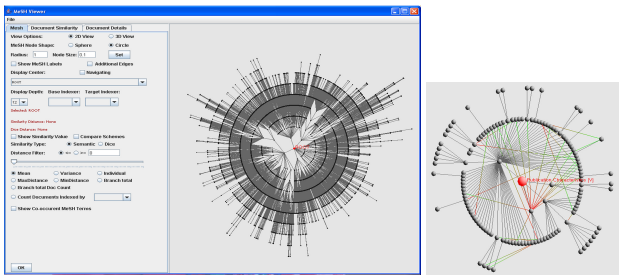
The Medical Subject Headings (MeSH) thesaurus is a controlled vocabulary in a tree structure produced by the National Library of Medicine (NLM) for indexing, cataloging, and searching for biomedical and health-related information and documents in reference databases such as MEDLINE (http://www.nlm.nih.gov/mesh/). MeSH descriptors/terms are arranged from most general to most specific in hierarchical levels. Each MeSH term appears in at least one place in the tree, and may appear in as many additional places as may be appropriate. In order to efficiently update MEDLINE frequently with index entries of increasing numbers of incoming documents, NLM initiated the development of an automatic indexing tool, Medical Text Indexer (MTI)[3]to identify index terms of a given article that can be recommended to human indexers. To evaluate and improve the performance of MTI, the indexing results consisting of MeSH term collections provided by human indexers and MTI are compared using similarity measurements [7] for an overall assessment.

For better understanding and easier exploration of the MeSH structure and indexing collections, we present an interactive visualization system that uses radial tree layout for the MeSH structure and displays indexing similarity measurements of different indexers for MEDLINE database at both the term and document levels. The indexing results and similarity values are represented in different colors and/or heights over the MeSH layout so that users can easily obtain neighboring and hierarchical information of the index terms for an intuitive estimation of similarity measurements. Such visualization tools can help researchers compare classifications of medical literature conducted by different indexing systems and techniques.

*e-mail: haixiadu@acm.org
†e-mail:yoo@nlm.nih.gov

2 RADIAL-TREE-BASED MESH VISUALIZATION

To render and explore the information space of biomedical literature built upon the MeSH thesaurus, we need a graphical layout for the MeSH structure. Among several visualization techniques for drawing graphs and hierarchies[1, 2, 4, 5, 6, 8], we chose the radial tree layout that consists of a set of concentric circles and treats the nodes with equal importance. For the MeSH hierarchy, the center is the root node, a pseudo term linking the 16 MeSH categories together. Its immediate child nodes, i.e., Category A, B, C, etc. form the smallest circle surrounding it, and the immediate child nodes of these category roots form the second smallest circle from the center and so on. For terms with multiple appearances, we assign them to the first branches where they appear and use additional edges linking them and their other parents. The system further allows users to choose any MeSH term in the tree as the center node to view the hierarchical structure with a specified number of levels started from that term. Fig. 1 shows the MeSH radial tree layout.



(a) (b)

Figure 1: MeSH radial tree layout. (a) The screen shot of the user interface with the entire MeSH tree; (b)the radial tree layout of Category V *Publication Characteristics* with additional edges showing terms with multiple appearances. The color of an additional edge changes from red to green indicating its direction.

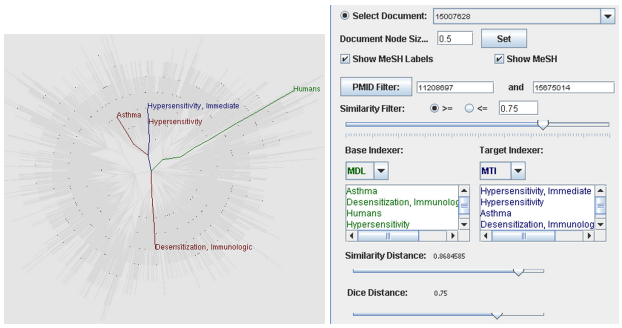
3 VISUALIZING INDEXING SIMILARITY

One application of the MeSH radial tree layout is to help researchers evaluate the performance of indexing systems such as the automatic Medical Text Indexer (MTI) compared with human indexers. The input includes indexing results for MEDLINE citations, which consist of a set of PMIDs and corresponding MeSH terms produced by different indexers, and similarity values by comparing these results[7]. Then the indexings and similarity values are displayed at either document level or term level.

3.1 Visualizing Indexing Results at Document Level

Once the MeSH layout is established, users can view indexing terms for any given documents and compare results from different

indexers. For a selected document, the terms produced by two comparing indexers can be classified into the shared terms and two distinct sets of terms for each indexer. By highlighting the index terms for a given document, the system provides a way to directly compare different indexers by examine the term difference in the MeSH structure. Users can also evaluate whether the similarity value is measured appropriately with hierarchical and neighboring information displayed in the graphical layout. The similarity values are displayed at the side panel for users' reference. The document set can be interactively filtered by similarity value thresholds to help users focus on a certain range of data. Refer to Fig. 2 for an illustrating example.



(a) (b)

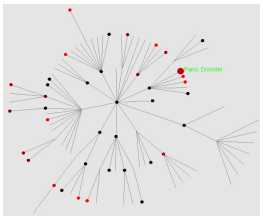
Figure 2: Viewing indexing results at document level. (a) The indexing results from two different indexers for a MEDLINE citation; (b) the corresponding user interface for document selection.

3.2 Evaluating Indexing Results at Term Level

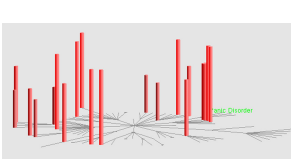
For a given term of an indexed document, it may be picked by the human indexer, MTI, or both. By counting how many times each MeSH term appeared in the indexing results from different indexers, we can divide MeSH terms into four categories, mainly picked by the human indexer only, MTI only, both, or neither. For example, in our sample indexing document set, the term **Panic Disorder** was picked 3 times by the human indexer, 4 times by MTI, and among them, there are 3 times it was picked by both of them. So it fells in the category of mainly picked by both indexers with percentage value of 75%. Since our goal is to evaluate the indexers' performance, we only consider the first three categories and display them using different color schemes in 2D and height functions in 3D. With the interactive MeSH navigation, it allows users to investigate regions in the MeSH structure where the indexers have strong or weak performance. Fig. 3 has examples in both 2D and 3D.

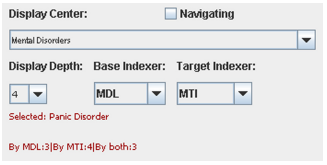
4 Conclusion and Future Work

This paper presents an interactive visualization framework for the exploration and evaluation of the indexings of biomedical references in the MEDLINE database. This framework is built upon a radial tree layout for the Medical Subject Headings (MeSH) thesaurus that portrays its large complex hierarchical structure. The interactive user interface allows users to explore MeSH in any level of detail. The MEDLINE indexings consisting of collections of MeSH terms are then displayed over the MeSH layout for exploration and evaluation. As an example application, we analyze a subset of MEDLINE references classified both by the automatic indexing system (MTI) and human indexers and create a visualization of similarity measurements of these two collections of MeSH terms to assess the performance of MTI. The ability to interact with indexing results through our framework helps users identify the strength



(a)





(b) (c)

Figure 3: Indexing evaluation displayed at term level. (a) Displaying the percentage value for the category of mainly picked by both indexers for the subtree **Mental Disorders**; (b) 3D view of the same input data; (c) the corresponding user interface display.

and weakness of indexers quickly in a visually convincing way. As part of future research, users suggested us to display term-term similarity measurements at document level when comparing the two sets of indexing results. We also plan to introduce the visualization of MeSH structures to other MeSH related research topics such as biomedical citation classification and clustering.

Acknowledgments

This research is supported in part by an appointment of Haixia Du to the NLM Research Participation Program sponsored by the National Library of Medicine and administered by the Oak Ridge Institute for Science and Education.

References

[1] G. Battista, P. Eades, R. Tamassia, and I. Tollis. *Graph Drawing: Algorithms for the Visualization of Graphs*. Prentice Hall, 1999.

[2] B. Johnson and B. Shneiderman. Treemaps: A space-filling approach to the visualization of hierarchical information. In *Proceedings of IEEE Visualization '91 Conference*, pages 284–291, 1991.

[3] W. Kim, A. Aronson, and W. Wilbur. Automatic meSH term assignment and quality assessment. In *Proceedings of AMIA Annual Symposium*, pages 319–323, 2001.

[4] J. Lamping, R. Rao, and P. Pirolli. Laying out and visualizing large trees using a hyperbolic space. In *Proceedings of UIST '94*, pages 13–14, 1994.

[5] T. Munzner. H3: laying out large directed graphs in 3d hyperbolic space. In *Proceedings of the 1997 IEEE Symposium on Information Visualization (InfoVis '97)*, page 2, 1997.

[6] T. Munzner, F. Guimbretiere, S. Tasiran, L. Zhang, and Y. Zhou. Treejuxtaposer: scalable tree comparison using focus+context with guaranteed visibility. *ACM Trans. Graph.*, 22(3):453–462, 2003.

[7] A. Neveol, K. Zeng, and O. Bodenreider. Besides precision & recall: Exploring alternative approaches to evaluating an automatic indexing tool for MEDLINE. In *Proceedings of AMIA Annual Symposium*, pages 589–593, 2006.

[8] G. Robertson, J. Mackinlay, and S. Card. Cone trees: Animated 3d visualizations of hierarchical information. In *Proceedings of the Conference on Human Factors in Computing Systems (CHI '91)*, pages 189–194, 1991.

Teaching Science in Virtual Reality with a Freehand 3D Illustration

Jadrian Miles*
Computer Science
Brown University

Daniel F. Keefe*
Computer Science
Brown University

Daniel Acevedo*
Computer Science
Brown University

Fritz Drury†
Illustration
Rhode Island School of Design

Sharon M. Swartz‡
Ecology and Evolutionary Biology
Brown University

David H. Laidlaw*
Computer Science
Brown University

Index Terms: K.3.1 [Computers and Education]: Computer Uses in Education—Computer-assisted instruction (CAI); I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Virtual Reality;

Keywords: Visualization, human-computer interaction, virtual reality, non-photorealistic rendering, illustration, education, teaching.

1 Introduction

We present the design of an immersive virtual reality (IVR) illustration and its use in teaching a biology lesson on the active structure of a termite mound. Educators at all levels use illustrations in textbooks and lectures to enhance students' comprehension of new information because they present material in an intuitive and interesting fashion [2]. Computer graphics has a rich history of application in conveying information to novice audiences, including the early use of IVR in architectural prototyping walkthroughs [1] and the interactive, 2D educational illustrations of the Exploratories project [8]. Our freehand termite mound model combines the advantages of IVR and nonphotorealistic rendering to effectively communicate a difficult scientific concept and improve upon the traditional mode of instruction through pictures.

The mound constructed by the savannah-dwelling African termite *M. bellicosus* is a fascinating example of artificial environmental regulation by animals. A clay construction two meters tall perforated with internal air channels, its three-dimensional structure provides thermoregulation and waste gas exchange for the nest at its base [6]. This structure and its interactions with the environment can be difficult to convey in text and 2D illustrations. We used CavePainting [4], a freehand modeling program that runs in the CAVE virtual reality environment [3], to design a simple, non-photorealistic cutaway model of the mound that includes convection currents and gas exchange. Since the model was intended to be used for instruction, its design and the design of the accompanying lecture proceeded in parallel. Once both were complete, a class of undergraduate biology students was brought into the CAVE in groups of six to eight for fifteen-minute lectures.

2 Presentation

As the students entered the virtual environment, they saw in front of them a life-size model of a termite mound, its peak just higher than their heads. Many viewers can be in the CAVE simultaneously, their views all controlled by a single user. The lecturer had this control and navigated to the opposite side of the mound; here a section had been cut away to reveal the nest at the base and the air channels

*e-mail: {jadrian,dfk,daf,dhl}@cs.brown.edu
†e-mail: fdrury@risd.edu
‡e-mail: Sharon_Swartz@brown.edu

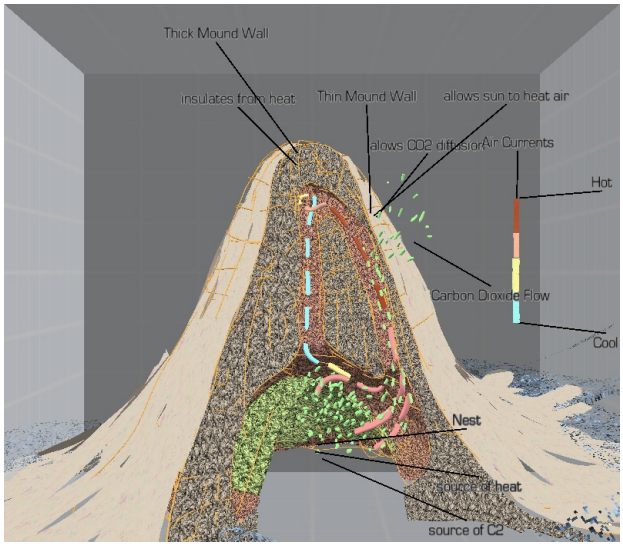


Figure 1: An external view of the mound revealing its structure, a solar-driven convection current, and diffusing carbon dioxide. Note the false coloring of the nest at the base and the reduced detail of the outside of the mound.

Figure 2: The lecturer and two students riding the convection current as it ascends the mound. Several viewers may be in the CAVE simultaneously, sharing one view, which allows a lecture to be delivered to many students at once with the accompanying illustration.

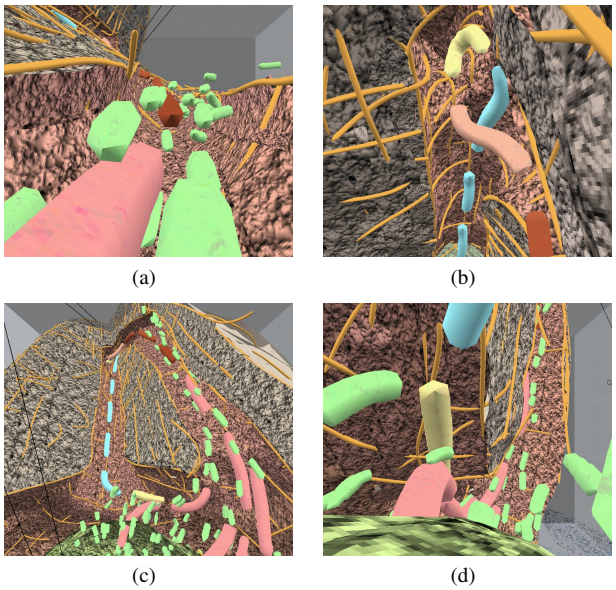


Figure 3: Virtual reality offers complete flexibility of viewing angle and scale to observe all features of a single model.

in the mound above. As the lecturer explained first the structure and then the function of the mound, he activated further portions of the illustration that show a cyclic convection current and the carbon dioxide exchange that it facilitates (figure 1).

To give the students a sense of the scale of this structure, the lecturer then expanded the model, shrinking all the users down to the size of termites. Now the students could see the mound from within and follow the path of the air like a roller coaster as it cycles (figure 2). The lecturer repeated the material from before, now with an intense visual experience accompanying it that caught the students' attention. We expect this exciting presentation improved comprehension and future recall beyond what would have been expected from a traditional lecture with 2D illustrations [2].

3 Evaluation

Using a freehand virtual reality illustration to augment a science lecture facilitates learning for three reasons. As mentioned above, illustrations in general assist the educator by reinforcing concepts in an intuitive, memorable, and appealing way. Additionally, IVR presents models of 3D phenomena in their natural space, making them easier to understand for novices. Furthermore, the nonphotorealistic rendering (NPR) community has argued that simplified visual depictions have many advantages over strictly realistic models when used for instructional illustrations. Such images emphasize important features while de-emphasizing others and also convey an intrinsic sense of generality [7]. CavePainting supports the creation of painterly and organic models in the virtual space where they will be presented to the audience, whereas CAD programs emphasize precise modeling and do not allow the illustrator to experience the model in its intended virtual space. CavePainting therefore gives illustrators the beneficial tools of NPR and lets him or her engage in a design process fueled by constant visual feedback [5].

In addition to the theoretical advantages of this style of illustration, we can also confirm that it enhanced the educational experience of the students, who were enthusiastic about the lecture and the subject matter. One student, a junior concentrating in Evolutionary Biology, told us that despite prior "trouble visualizing the link between the physical structure of the mound and its capacity to cool the air within it", she came to "appreciate the mound's formation and function in a way that would have required a much greater time investment without the lecture in the CAVE. It was an eye-opening experience!" Sharon Swartz, the professor for the class, told us that a "fifteen-minute CAVE demonstration probably gave students better understanding than they would have gained from an hour-long in-class lecture."

4 Conclusion and Future Work

Illustrations enhance learning in all contexts, but computer tools allowed us to create a single illustration that conveyed more information in a more intuitive and intriguing way in support of an educational goal. The immersive, interactive 3D experience of the CAVE allowed students to better understand the spatial structure of a termite mound, and the visually simple but exciting presentation allowed the illustrator/lecture designer to focus each student's attention on pertinent details and impress them thoroughly in his or her memory.

We noted that the process of creating the illustration itself enhances the designer's understanding of the material. Students are sometimes assigned to draw diagrams as part of their work, and using CavePainting to design an illustrative model could similarly prove to be a self-educational experience. In addition, the idea of using IVR illustrations as educational tools could be advanced by combining freehand and data-driven models; for example, a volumetric model of a skull could be augmented with annotations and freehand strokes to better convey a desired idea. Such an approach would allow for models to be based on pre-existing material, minimizing the effort needed to create them, while still reaping the benefits of an NPR-style illustration.

References

[1] F. P. Brooks, Jr. Walkthrough: a dynamic graphics system for simulating virtual buildings. In *Proc. SI3D '86*, 1987.

[2] R. N. Carney and J. R. Levin. Pictorial illustrations still improve students' learning from text. *Educational Psychology Review*, 14(1), 2002.

[3] C. Cruz-Neira, D. J. Sandin, and T. A. DeFanti. Surround-screen projection-based virtual reality: The design and implementation of the CAVE. In *Proc. ACM SIGGRAPH*, 1993.

[4] D. F. Keefe, D. Acevedo, T. Moscovich, D. H. Laidlaw, and J. LaViola. CavePainting: A fully immersive 3D artistic medium and interactive experience. In *Proc. ACM Symposium on Interactive 3D Graphics*, 2001.

[5] D. F. Keefe, D. B. Karelitz, E. L. Vote, and D. H. Laidlaw. Artistic collaboration in designing VR visualizations. *IEEE Comput. Graph. Appl.*, 25(2), 2005.

[6] J. Korb. Thermoregulation and ventilation of termite mounds. *Naturwissenschaften*, 90(5), 2003.

[7] L. Markosian, M. A. Kowalski, D. Goldstein, S. J. Trychin, J. F. Hughes, and L. D. Bourdev. Real-time nonphotorealistic rendering. In *Proc. SIGGRAPH '97*, 1997.

[8] R. M. Simpson, A. M. Spalter, and A. van Dam. Exploratories: An educational strategy for the 21st century. In *ACM SIGGRAPH '99 Conference abstracts and applications*, 1999.

Visualizing the Eclipse Bug Data

Michael Ogawa*
VIDI
University of California, Davis

Kwan-Liu Ma†
VIDI
University of California, Davis

Zhendong Su‡
Center for Software Systems Research
University of California, Davis

ABSTRACT

We apply the treemap technique to visualize bug data from the Eclipse open source software project. We are able to show an overview of the bug data which exposes interesting features in the Eclipse project and features of software evolution in general. Our plan for the poster includes these preliminary results as well as a new view of the data which is currently in development. Interested readers can try out the demo at http://vis.cs.ucdavis.edu/ ogawa/eclipse/.

1 INTRODUCTION

Software engineering researchers are interested in the process of software evolution. They analyze version control repositories, bug databases, and developer interaction to gain insight into how software evolves. Their findings can help to predict and correct bugs and make development more efficient. Visualization can be used by software researchers to give them a better overview of their data. They may see patterns which are not apparent in their more formal computational analyses.

Our visualization is based on treemaps [2], a well-known information visualization technique. We examine the bug data from the Eclipse[1] project.

2 THE ECLIPSE BUG DATA

Eclipse is an open source integrated development environment. The bug data was mined by the software engineering department at Saarland University. The data is free to download at http://www.st.cs.uni-sb.de/softevo/bug-data/eclipse/.

In this section we give an overview of the dataset. A more thorough and technical description of the data can be found in [1]. Their technique is outlined below, taken from their website:

> The data was obtained from the Eclipse bug and version databases; in essence, we automatically determined for each bug report in the bug database the associated fix in the version database and hence could determine for each bug where it was fixed and likewise, for each component, we could tell the defects that occurred.
>
> The major challenge for this task was to map bug reports from the bug database to compilation units in the version archive. To this end, we used two techniques:
>
> 1. Via text analysis of the commit messages, we identified fixes in version archives (in contrast to other changes). Typically, fix messages contain links to the bug reports in the bug database by stating a the identification number of a bug report.
> 2. By checking the latest changes of the person who closed a bug report, we could identify further changes.
>
> Furthermore, we needed to obtain the version of each defect. We did so by using the version field provided by the bug database. Note that the first reported version was used.

The dataset is provided in three XML files, each corresponding to versions 2.0, 2.1 and 3.0. The data is structured in a hierarchical fashion after Eclipse's Java package organization. Each package contains its files, and each file contains the number of its bugs reported before (pre) and after (post) the release.

3 TREEMAP INTERFACE

The hierarchical nature of Java packages lends itself to a hierarchical visualization. We chose the treemap [2] visual paradigm for its easy comprehension and inherent overview of the data. Specifically, we were inspired by the Map of the Market[2] [3] to create an easy-to-use map of the software bugs.

Each rectangle represents one file in the project source. The rectangles are grouped according to the Java package hierarchy. The size of each rectangle indicates the relative amount of bugs associated with the file, both pre- and post-release. This is an indicator of the activity occurring in debugging. Details about each file (such as name and number of pre- and post-release bugs) can be seen in a mouse-over popup. Within the treemap visual paradigm, we have created two color schemes: Absolute and Relative coloring.

3.1 Absolute Coloring

It is also useful to software engineers to see the *absolute* number of post-release defects within each package. Thus, we map one-color intensity to the number of bugs. The result for Eclipse 3.0 is shown in Figure 1. Black means there were no bugs post-release and a bright orange color means there were more bugs.

3.2 Relative Coloring

In the relative color scheme, we use a double-ended color map. A red color means a file has more defects after the release. A green color means a file has less defects after the release. The results are shown in Figure 2.

4 RESULTS

We can immediately see in Figure 2(a) which package is the most prone to bugs (org.eclipse.jdt.core, the large area of red in the lower-right). There are also several packages with no color, indicating stability.

Comparing across versions in Figure 2, we see a trend of decreasing greenness and increasing redness. This means that, as the software evolves, there are more bugs being found than being fixed.

*e-mail: msogawa@ucdavis.edu
†e-mail: ma@cs.ucdavis.edu
‡e-mail: su@cs.ucdavis.edu

[1] http://www.eclipse.org

[2] http://www.smartmoney.com/marketmap/

5 Future Work and Conclusion

The software mining team at Saarland University released a new version of the Eclipse bug data. This new version includes software complexity metrics, bug report ID's, and CVS information such as revision number and committer of each bug fix. We would like to either incorporate these new data into our existing treemap system or design a new system.

Our visualizations have produced results interesting to software engineering researchers studying software evolution. Readers interested in using our visualization can visit http://vis.cs.ucdavis.edu/ ogawa/eclipse/.

Acknowledgements

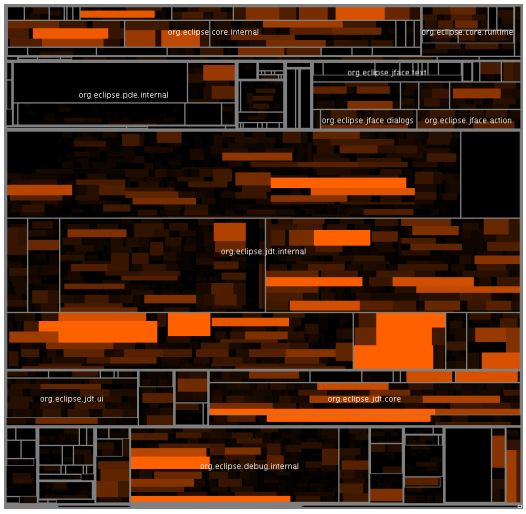
The authors wish to thank Andreas Zeller, Thomas Zimmermann, Rahul Premraj and Adrian Schröter for providing the dataset.

References

[1] A. Schröter, T. Zimmermann, R. Premraj, and A. Zeller. If your bug database could talk.... In *Proceedings of the 5th International Symposium on Empirical Software Engineering, Volume II: Short Papers and Posters*, pages 18–20, September 2006. Available at http://www.st.cs.uni-sb.de/softevo/.

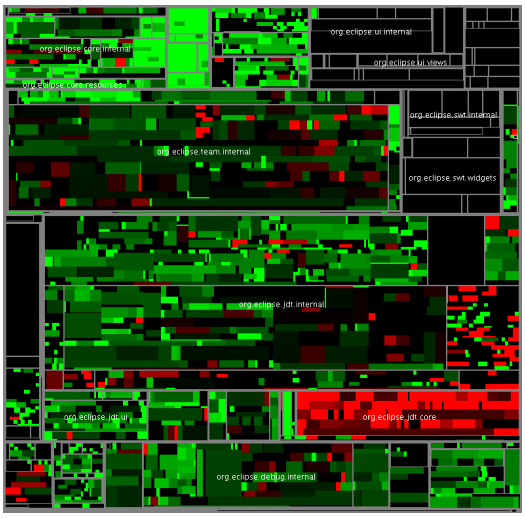
[2] B. Shneiderman. Tree visualization with tree-maps: 2-d space-filling approach. *ACM Trans. Graph.*, 11(1):92–99, 1992.

[3] M. Wattenberg. Visualizing the stock market. In *CHI '99: CHI '99 extended abstracts on Human factors in computing systems*, pages 188–189, New York, NY, USA, 1999. ACM Press.

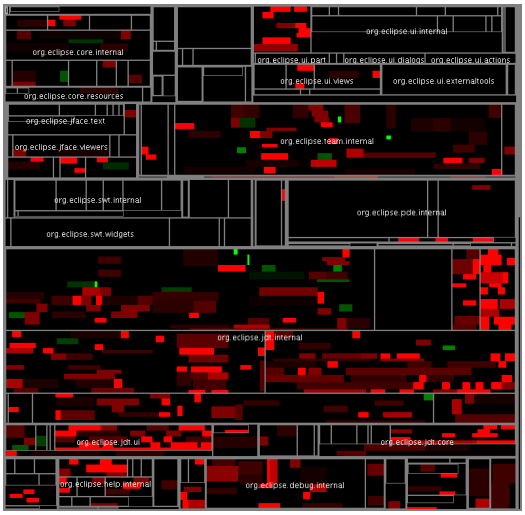


(a) Eclipse 3.0

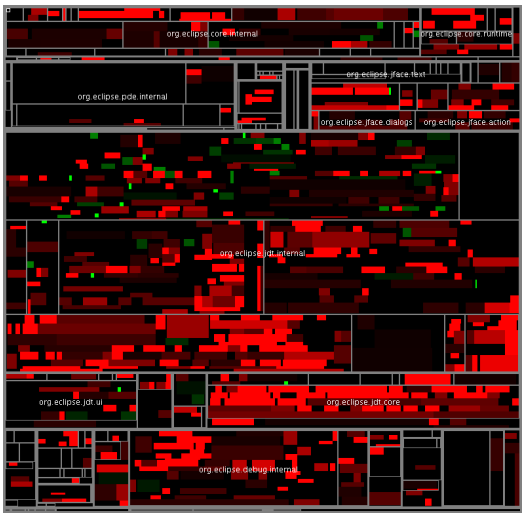
Figure 1: Absolute coloring of Eclipse 3.0. More intense orange color corresponds to more post-release bugs.



(a) Eclipse 2.0. Its mostly-green color indicates that many bugs were fixed after release. However, one package in particular stands out as extremely red.



(b) Eclipse 2.1



(c) Eclipse 3.0.

Figure 2: Relative coloring. Green means a file has less bugs after the release and red means a file has more. There is an increasing trend towards redness, which indicates that more bugs are being discovered than being fixed.

Treemap Based Graph Layout

Chris Muelder Kwan-Liu Ma

1 Introduction

Applications in many fields employ graph visualization to present data to the user. These visualizations are frequently used to show inherent patterns in the data. In order to emphasize these patterns, many algorithms have been developed that determine how best to place the nodes on the screen. While there are special cases such as trees or directed graphs for which shortcuts can often be used or directional constraints taken into consideration, probably the most commonly used graph layout algorithms are force directed layouts such as those presented in [4, 6, 9]. These layouts are versatile because they can be applied to any general graph, they can take edge weights into consideration, and they tend to make very aesthetically pleasing layouts. They are good at showing patterns such as clusters since vertices in the graph that are strongly connected get pulled together by the layout algorithm. While there have been recent advances in rapid force directed layouts [5], they are still generally computationally intensive and do not scale well as the size of the graph increases. Also, force directed layouts have a problem with local minima in the energy function where the graph can get stuck in a non-optimal layout. As a result, it is not uncommon for the layout algorithm to take many extra iterations to separate clusters from each other, or for a node in a cluster to be separated from the rest of its cluster and have its edges stretching across another cluster of nodes. There are also algebraic layout algorithms that work with the matrix of edges between nodes directly to very quickly generate a layout [5]. However, these approaches often have the problem of mapping nodes to the same locations, so that parts of the graph are obscured.

The treemap based layout algorithm presented here produces results somewhat similar to force directed approaches while attaining the speed of algebraic approaches. This is done by calculating a hierarchical clustering of a graph, applying a treemap to this hierarchy, and placing nodes in their respective regions of the screen. Since the treemap layout uses operations that are relatively easy to calculate, it is not as computationally expensive as force directed layouts. Because of this and because it does not take multiple iterations, it provides results more quickly than force directed approaches. It also avoids the problems of nodes being separated from their clusters or being placed on top of each other since the treemap forces clusters into separate regions of the screen.

2 Related Work

This work draws upon several existing techniques in both the fields of graph visualization and treemap visualization. Many graph layout algorithms have been developed, and there are several variations of and extensions to treemaps. The concept of using the two together has also been explored before.

Sometimes a graph has an intuitive layout where the vertices themselves contain positional information that can be used, such as geographical locations. However, most graphs do not have such information, thereby requiring that the positions of vertices be derived. For special cases such as trees or directional graphs, algorithms are used that exploit certain properties of the graphs. But for general graphs, more generic algorithms must be used, which generally fall into two classes: force directed and algebraic. *Force Directed Layouts* [4, 6, 9], position graphs by iteratively refining the positions of vertices in order to minimize an energy function based on edges and distances between nodes. Variations of these algorithms often reduce the amount of computation heuristically by limiting the number of calculations per iteration [5]. While these variations are faster, the heuristics can lead to suboptimal layouts or require exorbitant amounts of memory. *Algebraic Layouts* [5] are layout algorithms that directly manipulate the adjacency matrix with linear algebra techniques in order to produce an effective layout. While not very intuitive, these algorithms can very quickly produce layouts that are similar to force directed layouts. However, these algorithms can fail to produce a good layout in some cases, since they can frequently place multiple nodes at the same coordinates, which obscures the structure of the graph.

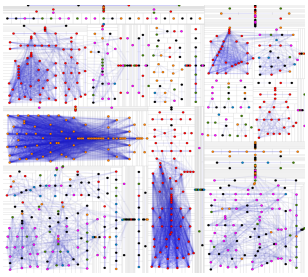


Fig. 1. A graph of the relationships between network scans, laid out by means of applying a treemap to a clustering hierarchy.

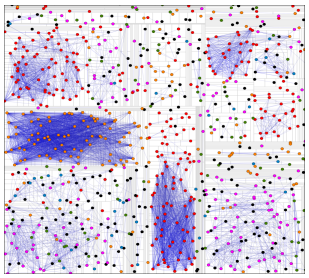
Treemaps [10] have become a common method for representing hierarchical data, with many derivative variations and improvements. While the majority of these works use a treemap to directly show a hierarchy, in several works, treemaps and graphs have been used together. In [11], both graph diagrams and treemaps are used interchangeably to represent parts of a tree more efficiently. A node-link diagram is overlaid on a treemap in order to represent a tree in [8]. Similarly, in [3], a treemap is used to represent the spanning tree of a graph, and then the remaining edges are overlaid in a traditional graph form. Finally, in [1], a treemap is used to interact with the clustering hierarchy of a graph. However, none of these actually use the treemap to simply lay out a general graph. Instead, they focus on using a treemap to represent portions of specialized graphs or to directly represent hierarchy.

3 Approach

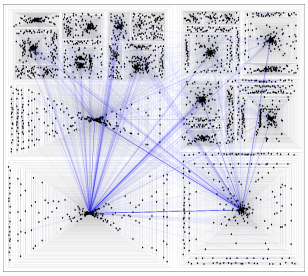
This work presents a novel method of laying out a graph by generating a clustering hierarchy, applying a treemap to the clustering hierarchy in order to allocate regions of the screen that are associated with individual vertices in the graph, and finally placing each node inside its region of the screen. An example of a graph laid out this way is presented in Figure 1. This particular graph presents the similarities between network scans (Figures 1,2(a), $|\mathbf{V}| = 878$, $|\mathbf{E}| = 385003$). It is a complete graph with edge weights between 0 and 1, but for clarity, edges with weights less than a certain threshold are not shown. The other graph used here is a non-weighted graph of links between search results for the word "California" (Figures 3,2(b), $|\mathbf{V}| = 6107$, $|\mathbf{E}| = 15160$, [7]) .

For the hierarchies in the examples presented here, single linkage

- *Email: muelder@cs.ucdavis.edu*
- *Email: ma@cs.ucdavis.edu*

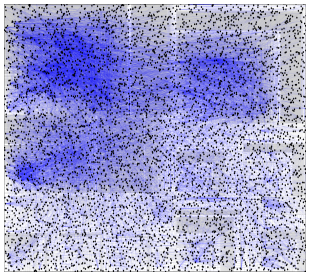


(a) Randomized placement

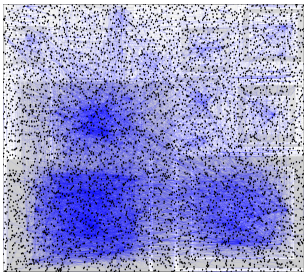


(b) Nested regions to show clusters

Fig. 2. Enhancements that alleviate difficulties due to treemaps



(a) Naive splitting



(b) Intelligent splitting

Fig. 3. Intelligently splitting regions shortens the average edge length and reduces edge crossings, but increases complexity.

was selected to work with weighted graphs because it is very simple and quickly calculated. For unweighted graphs, Clauset, Newman, and Moore's "Fast Modularity" community structure inference algorithm [2] was chosen, since it is very fast, generates a binary hierarchy, and is good at clustering small world networks. While the treemap layout algorithm can be applied to any hierarchical clustering, the ones used here make a binary tree, which keeps the treemap process simple.

Applying a treemap to the hierarchal clustering of a graph generates a division of the screen where each region of space corresponds to a node in the graph, and nodes that are clustered together end up next to each other. Once this treemap has been created, the graph can be laid out by placing each vertex somewhere in its associated region of the screen. This places each vertex in the graph near to regions of space corresponding to other vertices in its cluster, which results in a set of positions for the nodes that satisfies several of the properties that are desirable in a graph layout. Namely, the nodes are placed such that strongly weighted edges are kept relatively short compared to weakly weighted ones, screen space is not wasted, and important features such as clusters and outliers are readily discernible.

4 Enhancements

One potential issue with this approach is the frequent occurrence of collinear nodes. The problem with this is that edges between such nodes will not only cross but completely overlap so that they become indistinguishable from each other. Another common issue with treemaps is that regions can end up being very narrow. When this happens in the treemap graph layout, several nodes in the graph end up in a line so close together that they frequently overlap. In particular, this usually occurs with nodes that are outliers, so many outliers get placed closer to each other than tightly clustered nodes. One simple solution to both these problems is to randomize the position of nodes in their regions, as is shown in figure 2(a) Doing so makes the distribution of nodes spread out across skinny regions and vastly decreases the probability of collinear connected nodes. While this approach sacrifices the deterministic properties of the layout, it does solve both problems.

Randomizing the placing of the nodes in their regions has a side effect of making the nodes have an overall uniform distribution over the entire screen. While this is efficient in terms of screen space, it is not the most aesthetic view. Clusters abut each other, with no white space separating them. This problem is trivially solved by nesting the treemap regions. In order to have greater separation between clusters than inside clusters, the nesting is done proportionately to the size of the region. The results of applying this to the graph of search results for "California" are shown in Figure 2(b).

Another issue occurs with edges that go between two nodes that are in different clusters. It is easily possible for these nodes to be placed within their clusters such that they are far apart. The problem with this is that the edge between these two nodes now stretches completely across one or both of the involved clusters or even over the entire graph. This is the case in Figure 3(a). When possible, it would be better if these nodes could be placed within their clusters such that they end up near each other, so the cross cluster edge is not crossing over as many of the other edges in the clusters. This can be done by arranging the children during the splitting process according to an energy function of the edges into and out of the child regions. The result of doing this is shown in Figure 3(b) This substantially increases the computational cost, since it considers each edge, but it is still a much faster then force directed layouts.

5 Conclusions

The approach presented in here has several advantages when compared to existing algorithms, and some limitations. It performs very well in terms of speed, and it does not take multiple iterations the way force directed layout algorithms do. Further, the space filling property of the treemaps means that the graph takes up more of the screen space than many other layouts, such as force directed ones. It is also quite flexible, since any hierarchical clustering algorithm can be used. But, the treemap layout is dependent on the hierarchy, so the choice of clustering algorithm can have a large effect on the results, and an unbalanced hierarchy can lead to poor results. And since the treemap layout uses treemaps, some of the difficulties of treemaps also directly apply. So, further investigation into alternate clustering algorithms and treemap variations is planned.

References

[1] J. Abello, S. G. Kobourov, and R. Yusufov. Visualizing large graphs with compound-fisheye views and treemaps. In *Graph Drawing*, pages 431–441, 2004.

[2] A. Clauset, M. E. J. Newman, and C. Moore. Finding community structure in very large networks. *Physical Review E*, 70:066111, 2004.

[3] J. Fekete, D. Wang, N. Dang, A. Aris, and C. Plaisant. Overlaying graph links on treemaps. In *InfoVis03, Poster Compendium (Aug. 2003)*, page 8283, 2003.

[4] T. M. J. Fruchterman and E. M. Reingold. Graph drawing by force-directed placement. *Software - Practice and Experience*, 21(11):1129–1164, 1991.

[5] S. Hachul and M. Jünger. An experimental comparison of fast algorithms for drawing general large graphs. In *Graph Drawing*, pages 235–250, 2005.

[6] T. Kamada and S. Kawai. An algorithm for drawing general undirected graphs. *Inf. Process. Lett.*, 31(1):7–15, 1989.

[7] J. Kleinberg. 'california' search results graph, `http://www.cs.cornell.edu/Courses/cs685/2002fa/`.

[8] Q. V. Nguyen and M. L. Huang. Enccon: an approach to constructing interactive visualization of large hierarchical data. *Palgrave Macmillan (online)*, 2005.

[9] A. Noack. An energy model for visual graph clustering. *Lecture Notes in Computer Science*, 2912:425–436, Mar. 2004.

[10] B. Shneiderman. Tree visualization with tree-maps: 2-d space-filling approach. *ACM Trans. Graph.*, 11(1):92–99, 1992.

[11] S. Zhao, M. J. McGuffin, and M. H. Chignell. Elastic hierarchies: Combining treemaps and node-link diagrams. In *IEEE Symposium on Information Visualization 2005*, 2005.

Comment Flow: Visualizing Communication Along Network Paths

Dietmar Offenhuber*
MIT Media Lab

Judith Donath†
MIT Media Lab

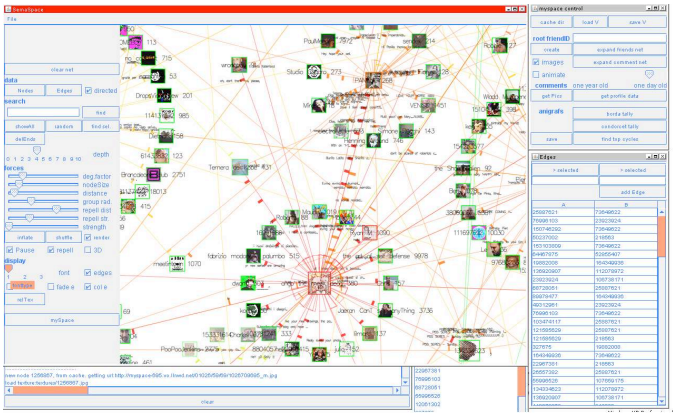


Figure 1: Screenshot of the Comment Flow application

ABSTRACT

Social networks are abstract organizational structures that help us understand the relationships among a group of interconnected individuals. Much recent research has focused on understanding the structure of these networks, whereas our approach focuses on the activity that flows along the network paths: the support offered, the information given, the gossip exchanged. This activity is invisible in traditional network depictions. We have designed and implemented a flexible tool for the content driven exploration and visualization of a social network. The underlying structure of Comment Flow is a traditional force-directed network visualization depicting nodes (people) and edges (their ties). With this as a foundation, it shows the communication between nodes, depicting the temporal pattern of exchanges and the content of the interaction.

Keywords: Aesthetics in visualization, Social visualization, Interaction.

1 INTRODUCTION

Social networks are the tangled web of personal acquaintance that ties human societies together. Much recent research has focused on understanding the structure of these networks, identifying patterns such as bridges, structural holes, etc. and on developing visualizations of these often complex entities. The "network" is a conceptual topology. It is the activity that flows along the network paths that actually forms the relationships: the support offered, the information given, the gossip exchanged. This activity is invisible in traditional network depictions.

Comment Flow is the tool we have designed for exploring and visualizing a social network and the activity that occurs within it. We have used it to explore a segment of the online social networking site MySpace. This project illustrates our approach to the general problem of social network visualization and provides an example of the practical uses of such representations.

The underlying structure of Comment Flow is a traditional force-directed network visualization depicting nodes (people) and edges (their ties). With this "skeleton" of network connectivity [2] as a foundation, it shows the communication between nodes, depicting the temporal pattern of exchanges and the content of the interaction.

For the mySpace instantiation, the nodes are profiles, the edges are friend links, and the communications are postings in comment boxes. This site has a wide range of users, from ordinary people keeping up with their acquaintances, to software agents masquerading as such to surreptitiously deliver spam, to celebrity profiles amassing huge number of fans. For the users of this site, the Comment Flow visualization can provide immediate and useful information to help navigate the complex social grounds of this and similar environments.

As Donath and Boyd described [4], most traditional social networking sites suffer a sometimes problematic simplification - the links are unnuanced and explicit, no distinction is made between a close relative and a complete stranger. All friend links are equal, and for the most part equally visible to public. Extra-network information is required to acknowledge the highly differentiated nature of human relationships. Feld used the term 'focus' for the different, sometimes incompatible partitions of social relationships[6].

As a result, people browsing public profiles of strangers in order to meet new people do not know whether the interesting profile they encounter represents a real person at all - it might as well be a commercial disguised as a person. The users have little awareness of the nature of the social the space they move through, no means of understanding who these other people are they meet or making sense of the space and the people they meet there.

One way of solving this problem is by focusing on the communicative activities taking place in the social network. It helps understanding who these strangers are by looking at their flow of conversation. Most social networking sites offer a feature similar to a guest book, where friends and sometimes strangers can leave comments.

2 RELATED WORK

The visualization of social networks has a long history dating back to Jakob Moreno's work sociometry in the 1930ies[7]. There are essentially two ways for the graphical representation of social networks: first, the arrangement of the participating actors in the rows and columns of a square matrix, its cells showing values that correspond to the relationship between the actors. The second and more popular way is the graphical representation of the social ties between the participating actors in the form of a graph. A large variety of layout strategies have been developed for the display of the structure of social networks. Beside radial and circular graph layouts force based layout algorithm are commonly used. A number of toolkits have been developed for this purpose, JUNG[10], Touchgraph[11], GUESS [1] are systems for the realtime animated display of graph structures. Jeffrey Heer and Danah Boyds Vizster toolkit [9] can handle large graphs and has been applied for visualizing the community structures social networking in the Friendster network. Other tools such as Pajek [3] offer a complete set of tools for social network analysis tasks. However, in spite of the number

*e-mail: dietmar@media.mit.edu
†e-mail:judith@media.mit.edu

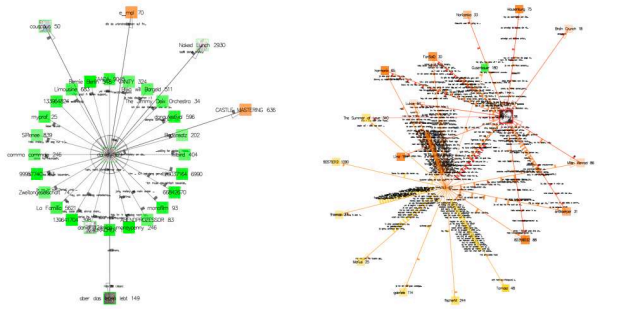


Figure 2: The visual pattern of one-way and two way message exchange. On the left side we see a band profile with a number of incoming posts, but very few replies. On the right side we see a group of profiles using the comment section for extensive conversation. The circularly clustered nodes represent instances of one-way communication.

of existing tools, very few of them are designed for the interactive manipulation of very large graphs at reasonable frame rates, due to sophisticated but computationally expensive layout algorithms, or the lack of support for hardware accelerated graphics. Comment Flow was designed as a compact tool for exploring, editing and displaying large networks, and thus fills the gap between extensive modular frameworks and compact but limited graph drawing applications.

3 Conversations in MySpace

Conversations on MySpace take place in the participant's comment area. They are thus dispersed across numerous pages, with the remarks directed to a particular person showing up on their page. This disconnected, asynchronous format is too disjointed for complex discussion; rather, it is designed for "social grooming" [5]. Leaving a comment on someone's page is a way of showing that you are thinking of them. Comments that reference private jokes or shared plans and experiences show the bonds among the linked individuals. These accumulating public comments also show an individual's social capital: someone who receives numerous personal comments appears to be well connected - to be someone whom others are willing to make the extra effort to maintain active ties with. The strength of particular ties can also be noted from the comments: a pair who leave notes for each other on a daily basis appear to have a strong bond, while those who seldom do so appear more weakly connected [1]

3.1 Usage Scenario

Many users made the experience of accepting friendship from a stranger and ending up with hundreds of spam messages on their comment boards. Naturally, before accepting an offer, one wants to know whether this person is a real person or a spam bot. Our tool can be used to visually assess communication behavior. Three parameters seem especially meaningful:

- The temporality of the network - the age of the messages, the frequency of communication. Is a profile constantly updated?
- One vs. two-way communication - is it a conversation or is it one way broadcasting? could provide clues do people really know each other?
- The quantity of information - is it a one time greeting of a newly added friend or actually a conversation?

The age of individual messages is expressed through their opacity, whereas the directionality and quantity between profiles can be established through the marks along the edges, each of them representing a single comment, and the grouping of the linked nodes. If desired, they can be animated along the edge in order to clarify their direction in a very dense network. The node color is primarily used for displaying topological distance, facilitating readability of dense networks.

[1] Any assessment of bond significance from online usage must be noted to be tentative. People have numerous ways of communicating and close friends may see each other so frequently and communicate in other ways that they have little use for online comments. Yet within the bounds of the mySpace community, the frequency of comments is significant because is both functional - people use it as a medium to communicate with each other - and display - people use it to make a public record of whom they are in frequent contact with.

4 Design and Implementation

The user of the Comment Flow software would start with an URL of a profile page. The system then would construct an egocentric network based on this profile by extracting and visualizing the posted comments, parse the profile pages of their authors and repeat the process. By matching the temporal sequence and determining directionality of conversation, a differentiated visual map of the communication between profile pages is constructed.

Comment flow was written in java with support for OpenGL accelerated graphics. It employs a force directed layout algorithm, a simplified and computationally less expensive version of the Fruchterman/ Rheingold spring embedder[8].

5 Conclusion

The visualizations generated by "Comment Flow" serve as a meaningful map of the social activities within online social networks. It creates a more differentiated picture of individual relationships than the list of uniform "friendships" offered by social network platforms. We believe that this is a step towards the next generation of social network visualization, where nodes and links are highly differentiated and associated with diverse media formats. It shows a viable path to the general problem of representation of online identity and its context in social network visualization.

please take a look at the video `http://web.media.mit.edu/~dietmar/files/comment_flow.zip`

References

[1] E. Adar. GUESS: a language and interface for graph exploration. *Proceedings of the SIGCHI conference on Human Factors in computing systems*, pages 791–800, 2006.

[2] A. Barabási and R. Crandall. *Linked: The New Science of Networks*, volume 71. AAPT, 2003.

[3] V. Batagelj and A. Mrvar. Pajek-Program for Large Network Analysis. *Connections*, 21(2):47–57, 1998.

[4] J. Donath and D. Boyd. Public Displays of Connection. *BT Technology Journal*, 22(4):71–82, 2004.

[5] R. Dunbar. *Grooming, Gossip, and the Evolution of Language*. Harvard University Press Cambridge, Mass, 1996.

[6] S. Feld. The Focused Organization of Social Ties. *American Journal of Sociology*, 86(5):1015, 1981.

[7] L. Freeman. Visualizing Social Networks. *Journal of Social Structure*, 1(1):4, 2000.

[8] T. Fruchterman and E. Reingold. Graph Drawing by Force-directed Placement. *Software- Practice and Experience*, 21(11):1129–1164, 1991.

[9] J. Heer and D. Boyd. Vizster: Visualizing Online Social Networks. *InfoVis 2005 IEEE Symposium on Information Visualization*, 2005.

[10] J. O'Madadhain, D. Fisher, S. White, and Y. Boey. The JUNG (Java Universal Network/Graph) Framework. Technical report.

[11] A. Shapiro. TouchGraph Project. *Sourceforge. Net Open Source Repository*, 2002.

Developing Colour Sequences for High Dynamic Range Data

Matthew Tobiasz, Amanda Henderson, Sheelagh Carpendale, Alan Dunning, and Paul Woodrow

Abstract—We consider the problem of developing colour sequences that offer the best dynamic range when used for pseudo-colouring. We began with nine colour sequences currently suggested for visualization and informally compared them to discover which was subjectively seen to have the most pleasing visual aesthetic. This comparison indicated that different colour sequences are more appropriate for different data ranges and led to the development of three new colour sequences. Again consulting our large group of collaborators we consider which of twelve colour sequences had the greatest dynamic colour range.

1 Introduction

For many types of data choosing an appropriate colour range is important for effective visualization [2, 4, 6]. We present an exploration of several existing colour sequences [3] and introduce three new sequences with the goal of most clearly revealing the greatest dynamic range of pseudo-coloured astrophysics data. This work is part of a research initiative into the visualization of large astronomical data sets, conducted in collaboration with computer scientists, astronomers, and media artists. Here we focus on our initial findings into pseudo-colouring this data. Our research concentrated on visualizing data from galactic spectral line emissions, gathered as part of the Canadian Galactic Plane Survey (CGPS) [5]. We used a relatively small subset of the entire CGPS data, measuring 879 megabytes in size.

Book Rdx is the software we developed to visualize this CGPS data set. This spectral line emission data is a 3D data set where spectral intensity, measured as velocity, is a function of galactic longitude, galactic latitude, and frequency range. Since all three axis contain contiguous values, the complete data set can be conceptually modelled as a single 3D volume. By filling the 3D volume with spectral intensity values. The contents of the data cube can be seen by cutting along a plane, aligned with either of the data cubes two primary axes and rendering only the area on the cutting plane. As in "Interactive Video Cubism" by Fels et al. [1], by sequentially moving this cutting plane up and down through the data cube and interpolating between contiguous values, a fluid animation can be made.

2 Pseudo-Colour

During visualization, scalar CGPS spectral intensity values are first normalized and then transformed to colour vectors, using the GPU for efficiency. Our investigation into the use of pseudo-colouring began with an assessment of this transformation. Here we considered globally and locally optimized colour, using a set of nine colour sequences as suggested by Levkowitz et al.[3]: nonlinearized gray, linearized gray, heated object, optimal colour scale, linearized optimal colour scale, rainbow, blue to cyan, blue to yellow, and magenta.

These sequences were first compared to discover which was subjectively seen to have the most pleasing visual aesthetic. We then introduced three additional colour sequences and this time considered which sequence revealed the greatest dynamic range.

2.1 Globally and Locally Optimized Colour

In globally optimized colour the complete data set is normalized relative to all values in the data set. This allows for a direct visual comparison of all resulting renderings since the scaling parameters to the normalization function are common to all values, providing a consistent colourization transformation. An assigned colour always maps to a constant scalar magnitude, no matter were this value resides in the data cube.

When the CGPS data set is normalized globally, the fundamental limitation of the resulting visuals is a pronounced lack of image contrast. This situation is due to the drastic difference in the mean spectral intensity values between cutting planes versus that of the global mean. The global colour would be appropriate if the complete CGPS data set were to be rendered in a single instant and presented in its entirely, however this is not the case since only a small slice of the data set is seen at a given instant of time. The fluctuations in the mean value of a cutting plane slice result in scalar to colour vector mappings which produce a poor utilization of a rich colour sequence.

In locally optimized colour, normalization of scalar spectral intensity values occurs on a per cutting plane basis, relative only to the values on the current cutting plane. By normalizing exclusively to those values to be visualized, the mapping of scalar to colour vector changes between positions of the cutting plane in the data cube. This means that direct visual comparison cannot be done between images created at discrepant discrete locations in the data set. However the result of the colourization process yields a vastly superior dynamic colour range for a given plane. Figure 1 illustrates the prominent difference between the global and local optimization. Local optimization was chosen for visualizing the CGPS data as it accentuated the subtle changes that occur within individual subsets of the data.

2.2 Considering Colour Aesthetics

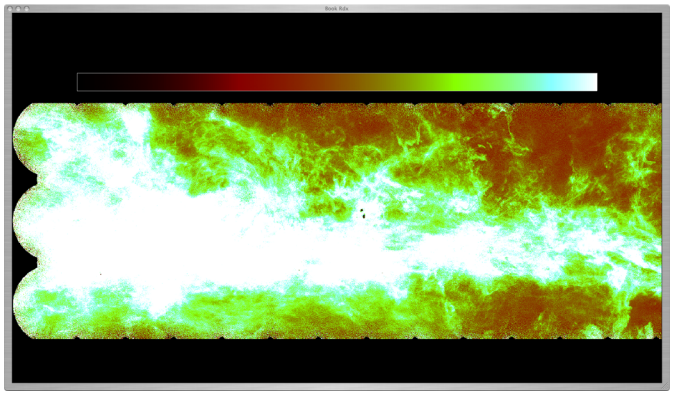
Particularly for the artists in our collaboration, attention to the visual aesthetic of the final rendering was of concern. To evaluate the aesthetics of our visualization, we consulted with our larger group of collaborators who had not been part of the colour choices up to this point in time. The people's declared backgrounds were: four from art, seven from computer science, and one undeclared. We asked which were the three most aesthetically pleasing colour sequences, rated from highest to lowest, and which was the single worst colour sequence aesthetic.

The results indicated an interesting dichotomy between the preferences of people with art and computer science backgrounds. Monochromatic colour sequences where chosen exclusively by those with an artistic background. This same group also unanimously chose the rainbow colour sequence as the most displeasing aesthetic. In contrast, none of the people with a computer science background though the rainbow colour sequence was the single worst. The linear grey colour sequence was most favourable by all, with heated object and magenta as the next runners up.

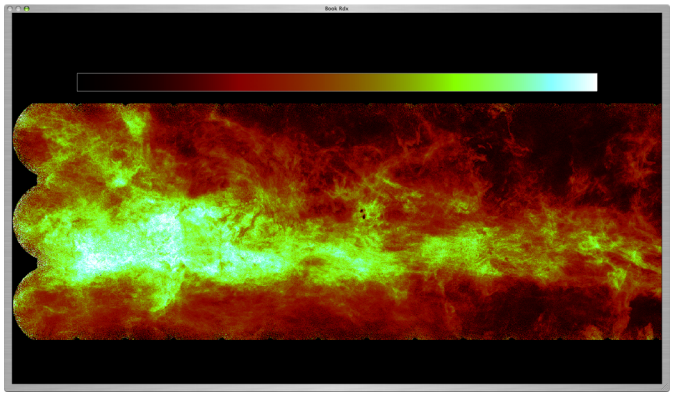
2.3 High Dynamic Range Colour Sequences

Since different colour scales applied to identical data sets accentuate discrepant ranges within the data, one can imagine an ideal generic colour scale would have no bias towards any particular range within its gamut yet offer the greatest ordinal dynamic colour range. With this goal in mind, we developed three additional colour sequences to augment the existing nine. These three new sequences are shown in Fig-

- *Tobiasz, Carpendale: Department of Computer Science, University of Calgary: tobiasz@ucalgary.ca and sheelagh@cpsc.ucalgary.ca.*
- *Henderson, Dunning: Department of Media Arts and Digital Technology, Alberta College of Art and Design: amanda.henderson@acad.ca and einsteins-brain-project@shaw.ca.*
- *Woodrow: Department of Art, University of Calgary: ebp@shaw.ca.*



(a) Globally optimized colour.



(b) Locally optimized colour.

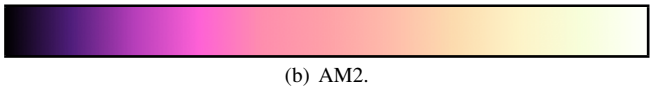
Fig. 1. Comparison between the globally and locally optimized colour on identical sets of scalar values, both using the linearized optimal colour scale.

ure 2. The first two sequences, AM1 (an example of its usage shown in Figure 3) and AM2, were artistically inspired. The third sequence, Flow, was created based on colour sequence design recommendations by Ware, in which each colour in the scale "provides a kind of upward spiral in color space; each color is lighter than the preceding one" [6, page 132].

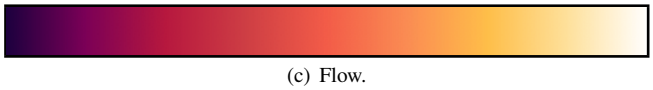
2.4 Improving the Colour Sequence's Dynamic Range

With the additional colour sequences, we once again consulted our larger group of collaborators, with the intent to evaluate the three best colour sequences from the group of twelve (again rated from highest to lowest) that have the greatest dynamic colour range; and again also selecting the colour sequence perceived to have the single worst dynamic colour range. Thirteen people where involved, all of whom had a background in computer science. The newly introduced colour sequences AM1 and Flow were highly favoured for the produced dynamic colour

(a) AM1.



(b) AM2.



(c) Flow.

Fig. 2. Additional colour sequences.

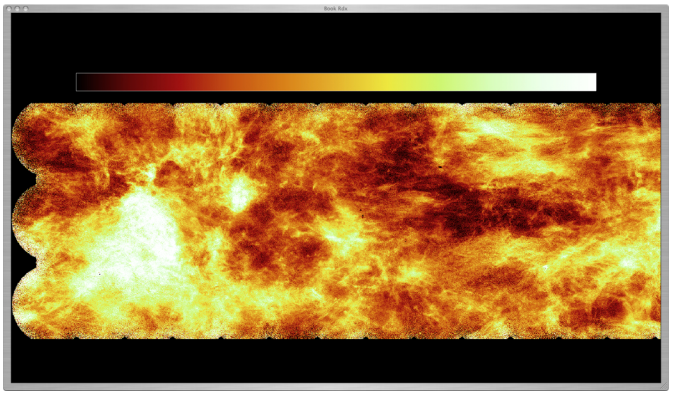


Fig. 3. Example of AM1 colour sequence.

range with the CGPS data set, which was encouraging positive feedback. The feedback also indicated that the optimized colour scale was seen to perform most poorly for this task of revealing dynamic range, which was a surprise.

3 Future Work

The next step in this research will be to formally evaluate the colour sequences, possibly a survey consisting of a large and diverse population sample. Interesting aspects include considering characteristics of colour sequence that best reveal the dynamic range locally, within a given sequence as well as globally across a large data set. Furthermore applying and evaluating our developed colour sequences to visualizations outside of the astrophysics is of interest.

Acknowledgements

The authors would like to thank Maureen Stone, and all the instructors and students of the 2006-07 *Art Science and Technology* course, offered conjointly by the University of Calgary, Alberta College of Art and Design, and The Banff Centre.

References

[1] S. Fels and K. Mase. Interactive video cubism. In *Proceedings of the 1999 Workshop on New Paradigms for Interactive Visualization and Manipulation (NPIVM '99)*, pages 78–82, New York, NY, USA, Computer Science 1999. ACM Press.

[2] C. G. Healey. Choosing effective colours for data visualization. In *VIS '96: Proceedings of the 7th conference on Visualization '96*, pages 263–ff., Los Alamitos, CA, USA, 1996. IEEE Computer Society Press.

[3] H. Levkowitz and G. T. Herman. Color scales for image data. *IEEE Comput. Graph. Appl.*, 12(1):72–80, 1992.

[4] M. Stone. *Field Guide to Digital Color*. A. K. Peters, Ltd., Natick, MA, USA, 2002.

[5] A. R. Taylor, S. J. Gibson, M. Peracaula, P. G. Martin, T. L. Landecker, C. M. Brunt, P. E. Dewdney, S. M. Dougherty, A. D. Gray, L. A. Higgs, C. R. Kerton, L. B. G. Knee, R. Kothes, C. R. Purton, B. Uyaniker, B. J. Wallace, A. G. Willis, and D. Durand. The canadian galactic plane survey. *The Astronomical Journal*, 125:3145–3164, 2003.

[6] C. Ware. *Information Visualization: Perception for Design 2nd Ed.* Morgan Kaufmann, 2004.

FanLens: Dynamic Hierarchical Exploration of Tabular Data

Xinghua Lou, Shixia Liu, and Tianshu Wang
IBM China Research Lab
{louxh, liusx, wangtsh}@cn.ibm.com

ABSTRACT

Tabular data is a very popular format for storing information from various domains. However, as the data grows in size, it becomes increasingly difficult to discover its intrinsic structure or see the comparison among cells from the traditional column-row presentation. To address the problem, we propose an enhanced technique based on traditional radial, space-filling visualization (e.g. Sunburst) named FanLens, which helps users to explore tabular data by dynamically specifying the hierarchies and then visualizing them. Our work is an improvement upon existing approaches in terms of flexibility, context preservation and interaction.

Keywords: Tabular data visualization, dynamic hierarchy specification, radial space-filling visualization, fisheye interaction.

1 INTRODUCTION

Tabular data is usually a matrix-like structure of columns and rows containing data cells and is one of the most popular formats for data storage. Although the table is the most straightforward way to present the tabular data, it becomes awkward when data volume increases, since there is no emphasis on its organization. [1].

Here we present FanLens, which enables dynamic exploration of tabular data with an incremental, radial space-filling visualization technique. Compared with traditional radial space-filling visualization (e.g. Sunburst [2]), our technique provides more flexibility and better preserves the context.

2 DATA TRANSFORMATION

Generally speaking, the tabular data can be divided into two categories, categorical data and quantitative data. Naturally, the hierarchy can be structured by breaking down the tabular data in order of its categorical data. FanLens further improves this feature by allowing the users to bin the quantitative data into different ranges and impose the results on structuring the hierarchy, and by allowing to group the categorical data by the instance of its value.

FanLens also supports dynamic visual data transformation, namely two separate dimensions of data can be mapped to the angle and color of the slices in the visualization, respectively. The source of the mapping could be one attribute or a mathematical expression of some selected attributes.

3 VISUALIZATION DESIGN

3.1 Incremental Layout

Incremental layout is the primary feature of FanLens, which follows two principles:

- *High-level start-up*

  When visualizing the hierarchy, our method does not lay out the entire hierarchy initially but displays only several high levels showing the summarization information (Figure 1(a)).

- *Expanding/collapsing mechanism*

  Users can drill down into lower levels by expanding one branch from the higher level. The newly expanded branch will be incrementally laid out around the periphery of its parent slice, radially, and is regarded as the focus (Figure 1(b)(c)(d)).

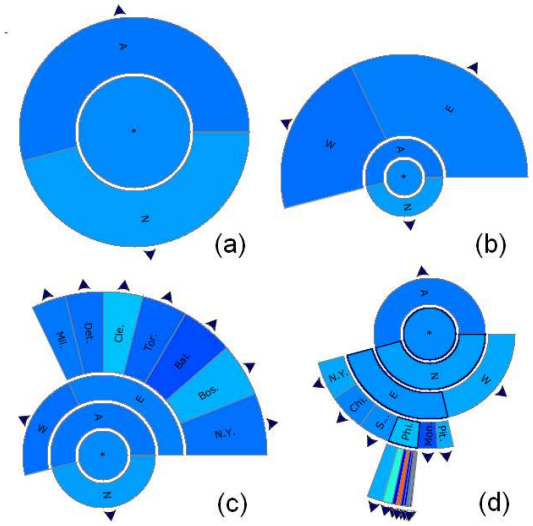


Figure 1: Examples of layout principles. (a) High-level start-up; (b)(c)(d) Expanding/collapsing mechanism.

This incremental layout firstly brings flexibility. Users can focus on the branch of interest and also see the overview by redefining the base levels to cover the entire hierarchy (which creates the classic Sunburst visualization). The readability is also improved because of the expanding/collapsing mechanism which offers the users with a clear view of the exploration path and structure of the focus.

3.2 Zooming and Picking

Zooming is used to deal with the thin-slice problem and is implemented by enlarging the sweep angle of the focus so all the thin slices in it are enlarged as well. Former designs [2, 3] applying this method have difficulties with preserving the context. Our solution follows the same idea but, benefiting from the expanding/collapsing mechanism, preserves the context better because the focus is enlarged where it is and need not be repositioned (Figure 2).

The goal of our picking solution is to ensure that users can have a clear view of this current selection. The transformation algorithm is prepared when the mouse is moving within the focus, but is only executed when it gets close to or into a thin slice (Figure (3)).

4 CASE STUDY

We use the statistics of NBA players for season 2005-2006 to evaluate the effectiveness of FanLens.

Overall Evaluation

Figure 4(a) shows the result of structuring the data in order of Conference, Division, Team and Player and defining the base levels

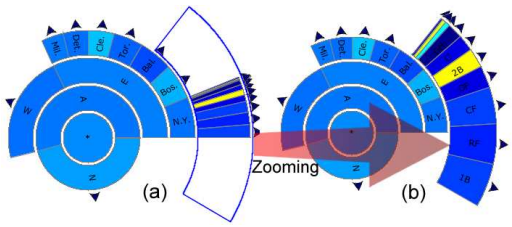


Figure 2: An example of focus zooming.

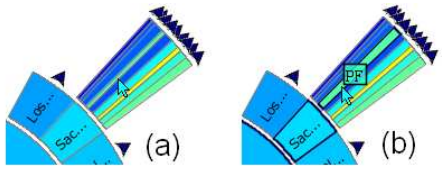


Figure 3: (a) Picking without fisheye transformation. (b) Picking with fisheye transformation.

to cover the entire hierarchy. The slice angle and slice color are mapped to the player's offensive ability and defensive ability, respectively. This overall evaluation indicates that, even though the players' offensive and defensive abilities vary a lot, the league is quite balanced in all levels of Conference, Division and Team. That is one important reason why NBA game is usually exciting because it is always a close matchup.

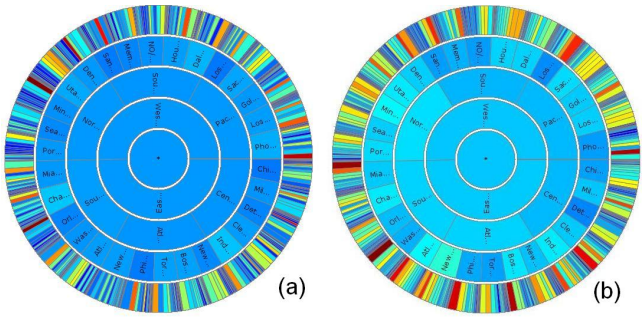


Figure 4: Evaluation. (a) Evaluate the balance of the NBA league; (b) Evaluate scoring and turnover.

Special Pattern Discovery

Figure 5 shows the same hierarchy but is dedicated to analyze the 3-point shooting ability. The angle is mapped to 3PM (3-Points Made per game) and the color is mapped to 3P% (3-Point shooting Percentage). The large angle of Pacific Divison guides us to explore it and find the team Phoenix Suns which is actually wild about shooting 3-points. In addition, a special pattern is noted that most of its players have close 3-Point shooting percentage which explains why they dare shooting so many 3-point baskets.

Hypothesis Testing

Figure 4(b) is supposed to analyze player's scoring ability (PPG) and mistakes (TO, Turnover). The angle and color are mapped to the player's scoring ability and turnover, respectively. This visualization is not really intrinsic but guides us to hypothesize that players with high scoring ability also have more turnovers. To test this hypothesis, we specify the hierarchy by ranging the players into several categories according to their PPG (see the following table) and visualizing the new hierarchy with the same visual presentation configuration, as shown in Figure 6. This new visualization proves our hypothesis that players with stronger scoring ability also give more turnovers.

PPG	Category
$25.0 < \text{PPG}$	super
$18.0 < \text{PPG} \leq 25$	strong
$10.0 < \text{PPG} \leq 18.0$	regular
$5.0 < \text{PPG} \leq 10.0$	low
$\text{PPG} \leq 5.0$	poor

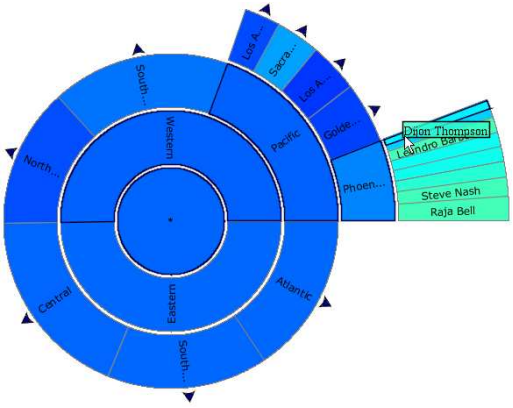


Figure 5: Explore the 3-point shooting abilities.

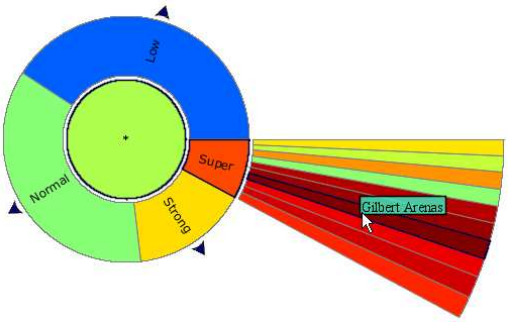


Figure 6: Testing the hypothesis on players' scoring ability and turnovers.

5 Conclusions

We have introduced the FanLens, an approach for dynamic and hierarchical exploration of tabular data. Our primary contribution is an incremental, radial space-filling visualization technique which better preserves the context, provides extra flexibility and has an unique fisheye transformation supported picking interaction.

References

[1] G. Chintalapani, C. Plaisant, and B. Shneiderman. Extending the utility of treemaps with flexible hierarchy. In *Proceedings of the Eighth International Conference on Information Visualisation*, pages 335–344, 2004.

[2] J. Stasko and E. Zhang. Focus+context display and navigation techniques for enhancing radial, space-filling hierarchy visualisations. In *IEEE Symposium on information visualisation*, pages 9–12, 2000.

[3] J. Yang, M. O. Ward, and E. A. Rundensteiner. Interring: an interactive tool for visually navigating and manipulating hierarchical structures. In *IEEE Symposium on Information Visualization*, pages 77–84, 2002.

Trammel Map: Providing a Clear View of the Enterprise Social Network

Shixia Liu, Nan Cao, Paul Moody*, and Tianshu Wang
IBM China Research Lab, *IBM Watson Research Center
{liusx, nancao, wangtsh}@cn.ibm.com, *paul_moody@us.ibm.com

Abstract

In this paper, we designed a Trammel Map to display the organization and network structure embedded in the enterprise social network. The basic idea of this design is to display the organization structure as a Treemap and graph links as curved links overlaid on the Treemap. Nodes from different organization units are put into separated squares. The proposed method also places nodes by their quantitative attributes. The most influencing person and organization unit are posited on the most noticeable place. This method enables a quick exploring and better understanding of the enterprise social network. The user can understand the collaboration and communication among diferent organization units in a straight way.

Keywords: Enterprise social network, Treemap visualization, Graph visualization, Interaction techniques

1 Introduction

People and their social networks are the most important assets in organizations. The informal network in the organization can promote change or stifle it. It can augment or disrupt the structure that the organization hierarchy is attempting to create. Thus it is important to study the enterprise social network under certain organization structure. In enterprise, there are a lot of social networks built from large scale communication and collaboration data, such as co-authoring, Email data, chatting, forum. Being able to visually analyze these enterprise social networks and understand their social dimension among different organization units, can help us to understand and exploit these networks more effectively.

Most existing social network visualization methods focus on visualizing the network structures [2]. These methods have been effectively used in the analysis of domains such as e-mail communication [1], online social networks [2], and co-authorship networks in scientific publications [4]. However, existing work mainly focuses on the layout of the unweighted network structures, they treat each person as equal when calculating the layout. Especially for the collaboration and communication networks in enterprise, there is no apparent approach to show the relationships between people under the organization structure. And some important questions, e.g. who is the more important persons of the social network, might be a little hard to answer. Furthermore, little work has been done on weighted enterprise social networks, in which each person has at least one quantitative attribute to indicate its importance in the network. In this paper, we desinged a Trammel Map to display the organization structure and collaboration and communication acts between the people involved in the enterprise social network.

2 Visualization Method

In the enterprise social network, relations are given between persons, and an organization hierarchy is defined on these persons as well. Thus the key of the visualization design is to show the hierarchical structure and relations at the same time. We designed a Trammel Map to visualize such kind of structures.

The basic idea of this design is to use a hybrid social network (link/node style) and organizational map (square Treemap style) to display hierarchical and link structure of an enterprise social network. Such kind of representation provides a number of insights into which persons are contributing to the organization units and which ones are more important. More specific, our design is driven by the following key qualities. For convenience, we denote the person with the highest quantitative value as the highest contributor.

1. The placement of the organization structure.

   The rule for the Treemap layout is to place the block with the highest contributor(s) at the center position. The next highest contribution block goes below the highest contributing block. Continuing down the list of the non-leaf nodes of the hierarchy with the highest contributions, they go to the left, then to the right of the center positions, and continue by going lower and then left and right.

2. The placement of the nodes within their blocks.

   The major goal is to centralize the key person, thus the person with the strongest network ties or highest quantitative value should go to the center of that block.

Both nodes and edges are size-coded. The nodes can be sized based on any quantitative attribute of interest; this could correspond either to measured data (such as the age of a person in a social network) or derived statistics (such as the difference between node in-degree and out-degree). Edges are also sized by the quantitative attribute of interest. In Trammel Map, we use the thickness of the edge to indicate the frequency of connections between two nodes. For example, in a social network, we can use the thickness of the edge to indicate the Email reply frequency of two persons.

3 Interaction

For effective information acquiring and understanding, navigation and user interaction are as important as presentation. We have embedded our visual presentation described above in an interactive system for the analysis of enterprise social networks.

To facilitate network exploration and observation, we adopt the highlighting techniques. The basic idea is to highlight nodes based on connectivity in the network. When the mouse hovers over a node, its direct connections with other nodes are highlighted and a tool tip is shown to display the detail information of the hovered node (See Figure 2).

To help the user find information more easily, Trammel Map combines the overview and detail technique with dynamic queries to facilitate the searching and pruning of large trees and complex links between nodes. The technique allows ranges of depth dependent attributes and ranking values to be specified to prune the tree and links dynamically. To reduce visual clutter caused by large amount of data, the following two filters are adopted.

1. Filter by the depth of the organization structure.

   When the user adjusts the slider on the depth of the organization hierarchy, the depth of the Treemap will change accordingly. The higher the adjusted value is, the more the low-level

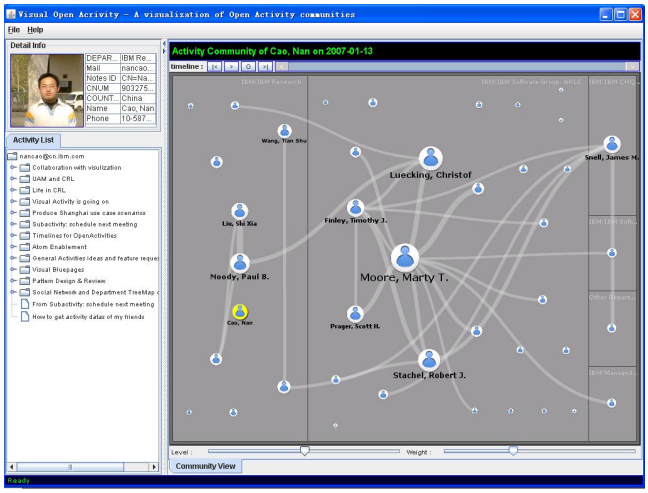


Figure 1: Filter function of Trammel Map

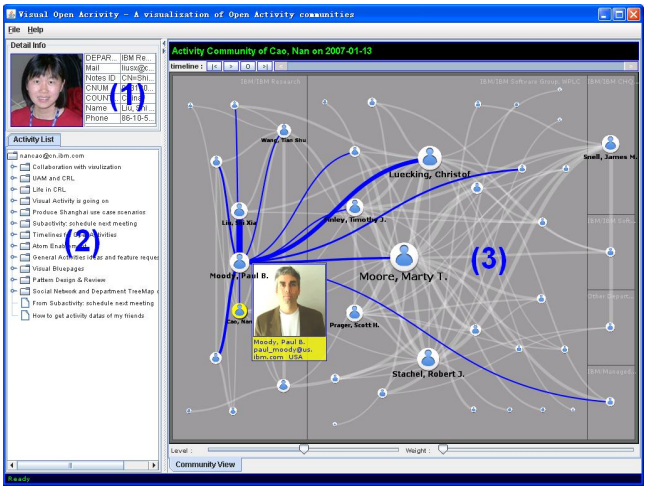


Figure 2: The Visual Activities. 1)the information view; 2)the activity tree view; 3)the social network view

nodes are shown in the treemap. Furthermore, the nodes belong to the sibling treemap blocks will gradually aggregate to one single node when the slider is adjusted to a lower value. This allows the user to drill down from summary data to detailed data continuously, thus encourage further exploration.

2. Filter by the weight of the links or nodes.

   In many applications, the node and/or the link of the enterprise social network have some quantitative attributes (weights) to indicate their importance. To reduce the visual clutter caused by the large number of links and/or nodes, we allow the user to adjust the weight slider to filter some unimportant nodes/links and help him focus on the more important ones. Figure 1 shows an example of filtering the unimportant links.

We apply this method to the Activities component of IBM Lotus Connections (http://www-142.ibm.com/software/sw-lotus/connections) for analyzing the collaborative network. Activities component is designed to help users organize and work with all their tasks, both individual as well as group projects, in a single place. Activities are collections of items (memos, files, links and tasks). It allows users collaborate in activities with shared messages, tasks, web links, and files and tag content for better organization. Generally speaking, there are two kinds of relationships among the Activities users. "Reply to" is the most common kind of relationship. For example, when the user B replies to the user A's topic, a direct relation will be generated between B and A. "Tag" is another aspect which is used for generating the relationships. When user A and user B using the same tag in different topics, whether they reply to each other or not, there is an implicit connection between them to indicate both of them have the same interests. We integrate all kinds of relationships together to generate a social network. Here, we use the data corpus acquired from an IBM internal Activities website. In this data corpus, since all the Activities users are the IBM employees, besides the implicit relations generated by the above rules, there also exist a organization tree to organize these users. Intergrating the organization structure with the social network together generates a hierarchical graph. It can be well presented by using the Trammel Map.

We designed a Visual Activities tool based on the above data analysis. It helps visually disclose the network built from Activities data. As shown in Figure 2, three views have been designed in this tool. The social network view is used to visualize the organization structure and network structure built from the Activities data. It overlays the network on an organization map. The activities' owner is highlighted as a yellow node in this view. As mentioned before, the foreground of this view is a graph visualization of the constructed social network. The node sizes corresponds the quantitative attributes of persons. Each line between two nodes is a relationship between two persons. The thickness of the line indicates whether the relationship between two persons is strong or not. The blocks in the background stand for IBM organization units. The users are laid out inside their corresponding organization blocks. The layout design mentioned before ensures that the most active person is always at the center of the view as the focus which is surrounded by other persons. This tool helps to disclose the collaboration and communication patterns among different organization units. Furthermore, it helps to visually manage and expand personal social capital, enables users to find people with specific knowledge or skills in extended enterprise social networks.

4 Conclusion and Future Work

We have designed a Trammel Map to provide a clear view of enterprise social networks. The main novelty of the proposed method is to use a hybrid graph layout and an improved Treemap to display the link and hierarchical structures in the enterprise social network. Customized interactions are designed for the proposed visualization method.

We regard the work presented as initial, there are improvements to be made as well as many directions to pursue. First, we are planning to design more interaction techniques to enhance the usability of the developed toolkit. Second, we need to design an temporal visualization method to help the user to keep his mental model when analyzing the evolution of the enterprise social network. And finally, further user experiments have to be performed to gain insight in the practical usage of our technique.

References

[1] D. Fisher and P. Dourish. Social and temporal structures in everyday collaboration. In *Proceedings of SIGCHI conference on Human Factors in computing systems*, pages 551–558, 2004.

[2] J. Heer and D. Boyd. Vizster: Visualizing online social networks. In *Proceedings of the 2005 IEEE Symposium on Information Visualization (InfoVis05)*, pages 33–40, 2005.

[3] Kleinberg JM. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5):604–632, September 2006.

[4] M.E.J. Newman. Co-authorship networks and patterns of scientific collaboration. In *Proc. of Natl. Acad. Sciences*, pages 5200–5205, 2004.

IEEE Visualization Conference,
IEEE Information Visualization Conference and
IEEE Symposium on Visual Analytics Science and Technology

October 28 - November 1, 2007
Sacramento, California, USA

InfoVis Panel

Panel: The Impact of Social Data Visualization

Robert Kosara* UNC Charlotte
Brent Fitzgerald† Swivel
Hans Rosling‡ Gapminder
Warren Sack§ UC Santa Cruz
Fernanda B. Viégas¶ IBM Visual Communication Lab

1 Introduction: Setting Information Free by Visualizing It (Robert Kosara)

In his famous talk at TED 2006, Hans Rosling [7] used animated and interactive graphics to get his point across. Rosling's energetic style and interesting message certainly helped, but the means he used were an incredible demonstration on the power of visual communication. From the perspective of current visualization research, the displays used were simple and one has to wonder if they would be accepted for publication at the InfoVis conference. Perhaps we need to rethink our criteria for evaluating visualizations and consider a broader set of goals than we currently do.

Information – some argue – wants to be free. Certainly data that was collected with the support of public funds should be freely available, and a lot of it is, at least in principle. But there are many different kinds of access, from easily usable data files in a common format to obscure data encodings and querying interfaces to printed tables in locked PDFs.

Availability of data therefore does not equate meaningful access. Examples like the 2000 US Census data [8] show that just making the data available is not enough if the format is obscure and the interface for accessing it is overly bureaucratic. Even simple visualizations like gCensus [4] and a project by Juice Analytics [5], which both create layers for Google Earth [3], finally provide access to that data years after it was published. In a similar way, TheyRule [6] shows data about board members of large American companies that is available in principle, but that needed to be collected in one place and shown visually to reveal connections that were hidden in plain sight.

An interesting problem that arises from the growing availability of useful data visualization tools for the general public is that while making data available so far did not necessarily lead to that data being actually used for investigation, be it by researchers, journalists, or just interested individuals. Will less data become available for fear of data mining by the general public? Will the possibility of connecting many different data sources make breaches of privacy possible that will lead to tighter restrictions on what data can be published at all?

Of course, all of the above is more a question of general data processing than visualization. Freeing data from an obscure format does not require or even involve visualization, and neither do many kinds of analysis. What makes visualization so powerful is its compelling visual nature. Seeing a graph display numbers like unemployment or, recently, climate change, is much more interesting to look at and more impressive than reading a table with numbers.

Visualization, therefore, is an agent of change, a powerful tool that needs to be understood not just in terms of mathematics or perception, but in terms of the impact it can have on the views and opinions of people.

*e-mail: rkosara@uncc.edu
†e-mail: brent@swivel.com
‡e-mail: Hans.Rosling@ki.se
§e-mail: wsack@ucsc.edu
¶e-mail: viegasf@us.ibm.com

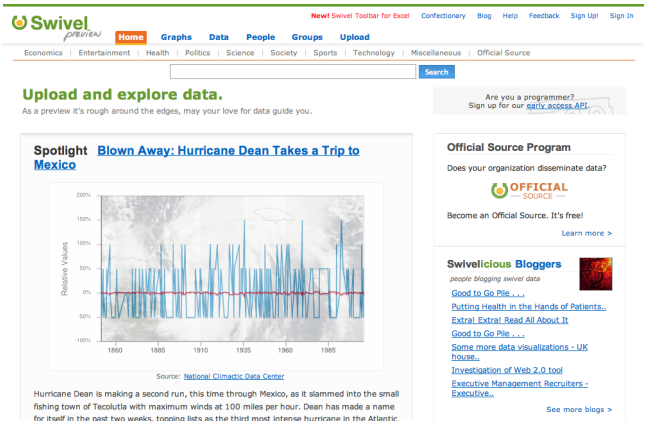


Figure 1: Swivel: http://swivel.com/

2 Empowering the Public (Brent Fitzgerald)

Numbers affect our every day life. They are in newspaper articles, they serve as justification for policies, they help guide our decisions and support (or refute) our opinions. But finding them is no small task. The data behind the quotes seem to be ephemeral. Despite the huge leap in access to information enabled by the web, the focus has been primarily on narrative text. Without access to those data, consumers of information are left with only half of the picture. We are forced to take claims at face value rather than develop alternative hypothesis of our own. This leads to a society where often facts are neither understood fully nor considered critically. This is a problem.

A solution to this problem is that any individual whether they are in Bangladesh or in Boston would have equal access to all relevant data both for their own decision-making and for research purposes. To make this widely accessible, implementation would need to utilize the technologies and methods of the social web. Swivel's online community and suite of features enables users to compare and contrast indicators from vastly different data sets on different topics for a variety of purposes – which is what is needed as a start to get people to explore data and consider its implications on their own lives.

The ability to visualize data is an essential piece to the puzzle of data exploration and use, and can no longer be reserved for academics. The fears data providers have of having their data trivialized, misunderstood or misrepresented also apply to the process of making visualizations, even simple ones. We believe that a strong community supported with data and tools will help advance the role of both data and visualizations in positive directions.

Visualizing data changes how data are understood and increases interest in data generally, which will encourage more and better data to be developed. Making data more visual exposes patterns in data not otherwise apparent. Putting visualizations on the web further encourages insight by allowing a broad audience to access, comment on and discuss what they see.

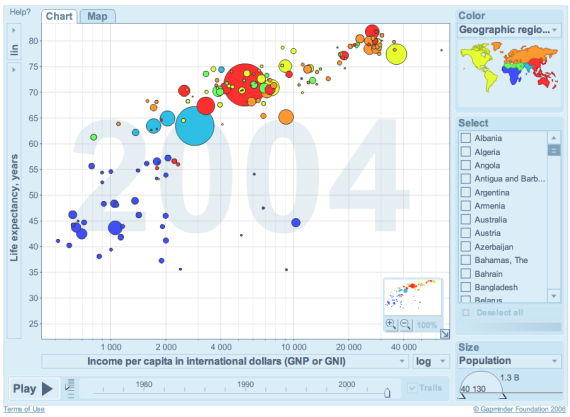


Figure 2: Gapminder: http://tools.google.com/gapminder/

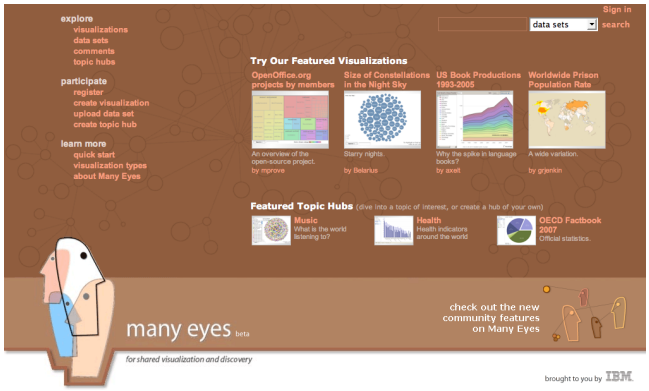


Figure 3: Many Eyes: http://many-eyes.com/

3 The Difference between Visualization and Animation of Data (Hans Rosling)

I am old enough to remember how impressed I was the first time when at the age of five my uncle read Donald Duck Cartoons for me. But I do not remember anything of the story, only how nice it was to sit on his lap and look at the colorful visuals. When I saw the first animated Donald Duck film at the age of seven, I was stunned! More than half a century later I can still recall the storyline of the animated film but I do not remember where I saw it. Films and stories are more powerful than images. It is my impression that the positive reactions to Gapminder's graphic presentations of data are due to the fact that they are animated and not just static visualizations. We have mainly used animation to let the movement represent the passing of time and all humans directly understand movement as a representation of time. To accept a horizontal X-axis as a representation of time you have to pass through many years of formal schooling and integrate this graphic representation in your visual literacy. But when we let the bubbles race in the Trendalyzer graphics, even well-trained minds understand the passage of time better. The use of movement to represent time also liberates one dimension of the visual area to represent another indicator. Furthermore, story telling linked to time series becomes natural with time as movement. I therefore think we should distinguish between visualization and animation.

A major challenge in replicating animation is that movement has to be carefully designed, and that minute differences in the animation have quite different impact on the interpretation. The conversion of data into graphics requires a number of different modalities of visualization and animation, and that will require many inventors to experiment and develop new ideas. In order for publicly funded statistics to benefit from such a force of innovation, the statistical agencies need to provide access for young inventors to the databases. A sad experience is that Gapminder's main problem in the innovation process was not technology, ideas or funding – but to get access to tax funded databases. Some new innovations are coming from staff in the agencies, but the bulk of the innovating forces are outside those agencies and perhaps it is the most innovative minds that have the most difficulties to pay the fees for commercialized data. In order to develop new visualizations and presentations, small pieces of free data are not enough: we need bulk download access. Only then will data understanding and access grow and flourish.

4 Overcoming Cognitive Bias (Warren Sack)

Data concerning groups or networks of people, i.e., "social data," are prone to a number of misunderstandings frequently referred to as "social biases" or "attributional biases," but more generally termed "cognitive biases" in the fields of cognitive science, social psychology, and elsewhere. For example, the false consensus effect is well known in public opinion polling: people tend to overestimate the number of people who agree with their own opinion. Pluralistic ignorance is another example of social bias: it occurs when almost everyone rejects a group norm, yet believes that most other group members accept it. False consensus and pluralistic ignorance are symptomatic of the fact that people have a hard time estimating the distribution of others opinions about some given topic. Other kinds of cognitive bias affect related areas of cognition and perception. For example, a number of biases make people prone to errors in estimating probabilities. Another set of cognitive biases influence memory and, unfortunately, make eye witness testimony in a court of law error prone. Some of these forms of cognitive bias can be explained by noting that people tend to employ heuristics rather than algorithms or scientific methods to make decisions. Conversely, we can understand various forms of scientific methodology as procedures for overcoming these cognitive biases (e.g., the employment of random statistical sampling to overcome ones naturally tendency to rely too heavily on personal experience and anecdotal evidence).

Following in a longer tradition of technological development (e.g., Douglas Engelbarts project of "Augmenting Human Intellect"), many in the field of information visualization have sought to design and evaluate visualization technologies according to their ability to amplify or augment our cognitive abilities [1]. The general form of the argument is that computers can be used to build upon and strengthen our cognitive and perceptual abilities. Unfortunately, this is an unworkable approach in certain domains such as within the realm of social data because our cognitive "abilities" are liabilities prone to error and bias. Or, in other words, while seeing is believing, we should not believe everything we see because our eyes can deceive us. Consequently, I propose a complementary methodology of design an evaluation in which we judge social data visualization technologies according to their powers to help us overcome our cognitive biases. I will present results from an NSF-funded project in which my students and I are working to design an interface and search engine appropriate for deliberative democracy in which the cognitive bias of false consensus is minimized.

5 Many Eyes: Democratizing Visualization (Fernanda Viégas)

Historically visualization technology has been accessible only to the elite in academia, business, and government. But recent years have witnessed internet-based visualizations ranging from political art projects (e.g. Theyrule [6]) to New York Times stories (Faces of the Dead [2]). These displays are seen by thousands and have

brought visualization to a new, large audience. Unfortunately, the revolution is not yet complete: while lay users can view many sophisticated visualizations, they have few ways to create them.

In order to "democratize" visualization, we have built Many Eyes, a web site where users may upload data, create interactive visualizations, and carry on discussions. The goal is to support collaboration around visualizations at a large scale by fostering a social style of data analysis in which visualizations not only serve as a discovery tool for individuals but also as a medium to spur discussion among users.

To support this goal, the site includes novel mechanisms for end-user creation of visualizations and asynchronous collaboration around those visualizations. Traditionally, visualization builders have focused on systems that support a single user or small groups, while seeking ways to present huge graphs, very high dimensional data sets, and tables with billions of rows. Many Eyes embodies the reverse perspective: Instead of scaling the size of the data, we scale the size of the audience. What happens when we design for massive public visualization?

Since the site launched earlier this year, users have uploaded data and created graphics on everything from DNA microarray data, to co-occurrences of names in the New Testament, to Senate testimony of a top White House official. Apart from data analysis, our preliminary results show that Many Eyes is used for goals ranging from journalism and advocacy to personal expression and social interaction. We contend that these findings suggest a growing role for visualization as an expressive medium–and the power of putting visualization into the hands of the people.

6 Biographies

Robert Kosara is an Assistant Professor of Computer Science at the University of North Carolina at Charlotte (UNCC). He received his M.Sc. and Ph.D. degrees in Computer Science from Vienna University of Technology in Austria. Robert considers the visualization of data not only an interesting topic in itself, but wants to see visualization widely adopted in practice and in the real world.

Brent Fitzgerald is the chief designer at Swivel.com. He graduated from MIT's Media Lab, where his research focused on designing and developing applications to support creative collaboration and virtual markets, especially lightweight, socially enforced micro-contracts.

Hans Rosling is a Professor of International Health at Karolinska Institute in Sweden and the driving force behind Gapminder. While serving as doctor in Moçambique 1979-81, he discoverd konzo, a new epidemic paralytic disease. He later co-founded Médecins sans Frontierès Sweden, started courses and wrote a textbook on global health. Hans co-founded Gapminder to unveil the beauty of statistics by turning boring numbers into enjoyable animations that make sense of the world.

Warren Sack is a media theorist and software designer. He has exhibited work at the ZKM—Center for Art and Media, Karlsruhe, Germany; the Walker Art Center in Minneapolis; the New Museum for Contemporary Art in New York City; and, on the Artport website of the Whitney Museum of American Art. Warren earned his B.A. from Yale College and his Ph.D. from the MIT Media Laboratory. He currently teaches in the Digital Arts & New Media M.F.A. program and in the Film & Digital Media Department at the University of California, Santa Cruz.

Fernanda B. Viégas is a research scientist in IBM's Visual Communication Lab. Together with Martin Wattenberg, she created Many Eyes. Her work addresses the social and collaborative aspects of data visualization, focusing on representations of online communities to support identity, collective memory, and story-telling. Her visualization-based artwork has been exhibited in galleries in New York, Los Angeles, and Boston.

References

[1] S. K. Card, J. D. MacKinlay, and B. Shneiderman, editors. *Readings in Information Visualization: Using Vision to Think*. Morgan Kaufmann Publishers, 1999.

[2] G. Dance, A. Pilhofer, A. Lehren, and J. Damens. Faces of the dead in Iraq. http://www.nytimes.com/ref/us/20061228_3000FACES_TAB1.html.

[3] Google Earth. http://earth.google.com/.

[4] I. Haque. gCensus. http://gecensus.stanford.edu/gcensus/.

[5] Juice Analytics. Census data visualization. http://www.juiceanalytics.com/weblog/?p=119.

[6] J. On. They rule. http://theyrule.net/.

[7] H. Rosling. Talk at TED. http://video.google.com/videoplay?docid=2670820702819322251.

[8] U.S. Census Bureau. http://www.census.gov/.

IEEE Visualization Conference,
IEEE Information Visualization Conference and
IEEE Symposium on Visual Analytics Science and Technology

October 28 - November 1, 2007
Sacramento, California, USA

InfoVis Contest

Exploring Meta-Data Associations with Bungee View

Mark Derthick*

Human Computer Interaction Institute, Carnegie Mellon University

ABSTRACT

Bungee View is designed to support non-technical users in familiar document searching and browsing tasks while introducing the unfamiliar task of discovering patterns in the documents' meta-data. Two expert-oriented features were added for the InfoVis contest to support hypothesis testing and discovery of patterns involving multiple attribute values. Most of the contest questions were answered in the negative, or with only weak associations. Stronger associations of potential interest were observed in the course of analysis. The accompanying video documents the analysis, while this document explains the design decisions behind Bungee View.

Keywords: information visualization, exploratory data analysis.
Index Terms: H.5.2 [Information Interfaces and Presentation]: User Interfaces---Graphical user interfaces, Interaction styles, Screen design; H.3.3 [Information Storage and Retrieval]: Information Search and Retrieval---Information filtering, Query formulation; H.2.8 [Database Management]: Database Applications---Data mining, Image databases;

1 BUNGEE VIEW DESIGN

The name "Bungee View" evokes Shneiderman's Information Seeking Mantra [4]: Overview first, zoom and filter, then show details on demand. It also emphasizes that viewing a dataset from a high level and diving into ever smaller subsets is a cycle with many bounces. Bungee View (BV) is an open-source (GPL) Java Web Start application available at BungeeView.com, built with the visualization toolkit Piccolo [1]. While it includes a result list and a summary of the selected document, the InfoVis contest asks about meta-data patterns, and so these features are little used.

The focus is BV's display of associations among meta-data attribute values (called *features*). Associations are deviations from independence, and can be positive or negative. Users bounce through the data by adding and deleting filters on features, which determine an evolving set of selected movies.

The analysis in the video found that R-rated movies are much less likely to have a high box office value than PG or PG-13 movies. To explore this negative association further, in Figure 1 the universe of movies in which to show associations has been restricted to those that grossed at least $70 million (with the Restrict button). Then the rare R-rated movies within that universe were selected. The R bar and label are colored bright green to show that this filter is in effect. (Bright red is used for negative filters.)

Each row of bars is a Mosaic Display [2] for one attribute. In the figure, the Genre attribute is selected, so its Mosaic Display is magnified and labeled. For other attributes, feature names and counts are shown on rollover at the bottom right of the screen.

Bar heights encode the percentage of movies with each feature that are selected, and vary from zero for Animation and Family to 78% for War. That is, 78% of high-grossing War movies were rated R. By definition, a feature is independent of the filters if its selection percentage is the same as that for the whole database. The height of the gray background encodes this expected

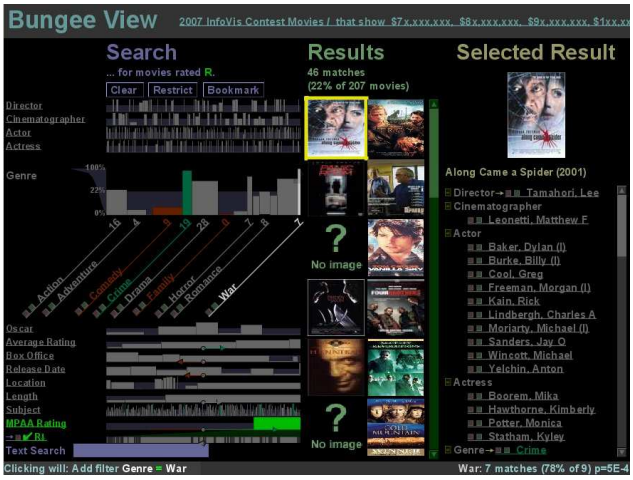


Figure 1. Associations between an R rating and each genre for high-grossing movies.

percentage (22%). To emphasize variation around this value, heights are scaled non-linearly to place it in the middle. Bars higher than this represent positive associations, and vice versa.

Bar widths show the total number of movies having a feature. Thus [distorted] area is proportional to the number of selected movies with the feature. Width controls for the unfiltered distribution, leaving height to show associations.

Experimental subjects using BV sometimes included overly specific features in their hypotheses. For instance, a user might find that Best Pictures tend to have a high box office, when *all* Oscar categories have this property. To reduce this problem, BV now considers only movies having the parent feature in the independence test. The background rectangle now represents the average percentage *for each attribute*. This makes the significance test hierarchical; it tests whether a feature is unusual among its siblings, rather than in the whole database. This doesn't make much difference for an attribute that almost all movies have, like Genre, but it can be important for Oscar or for a nested feature like Location → Canada → Alberta.

BV also minimizes unwarranted conclusions by coloring bars dull green (orange) to show whether a bar's height is significantly higher (lower) than the background, according to a Chi-squared test. Bars that are colored (statistical significance) *and* noticeably higher or lower than the background (practical significance) should be given the most attention.

BV uses Bonferroni correction of the χ^2 significance threshold with the goal of reducing the chances of finding any false positive associations during an analysis to 5%. It assumes a user will check all bars on the screen for color, and that she will view 5 different sets of bars over the course of analysis. For the example in the figure, the resulting threshold is 5% / 1500 bars / 5 sets = $7 \cdot 10^{-6}$. BV settles for a conservative correction procedure in the belief that making any association salient is a bonus compared to other search and browse tools, and that failing to point out less strong but potentially interesting associations is a relatively minor failure. With moderately selective filters, p-values are often very

small (10^{-50} or less), and corrections on the order of 10^4 have negligible effect.

2 Expert Features Added for Contest

For this contest there are specific a priori hypotheses to be tested, so being more precise about significance is important. Therefore the unadjusted p-value for each Chi-squared test is now added to the usual rollover feature summary, allowing an expert to do his own adjustment in his head if desired. In Figure 1, War has the highest bar, but is not colored. Rolling over it shows a p-value of $5 \cdot 10^{-4}$, which is greater than the $7 \cdot 10^{-6}$ computed above. If we hypothesized that Genre is associated with an R rating before looking at any data, the significance threshold could be based on only the 19 Genre bars, giving $0.05 / 19 = 3 \cdot 10^{-3}$ and making the War association significant.

BV only shows associations between the evolving set of filters and individual features. However the contest asks for *teams* of people associated with high-grossing movies. To address this, a command was added to actively search for sets of conjunctive features associated with the current filters. The top 100 of these clusters of features are listed in order of p-value, and can be added as filters just like single features. This sort of data-mining (market-basket analysis) often finds obvious or otherwise useless clusters. BV addresses this problem by only looking for positive associations, and by allowing users to specify features to ignore. It is relatively quick to click on features in uninteresting clusters to remove them, as shown in the video.

3 Related Work

Flamenco [6] supports the same kind of filters on hierarchical attributes as BV, and this capability is now commonly seen on shopping and other web sites. However the reason for filtering is primarily searching and browsing. The number of filtered objects having each feature is shown, but not the baseline frequency needed to judge association.

The Relation Browser (RB) [3] shows both filtered and unfiltered counts, but uses conventional equal-width histograms with a linear scale. It is difficult to compare percentages between bars of different lengths in order to infer associations, and impossible when queries are even moderately restrictive and the filtered bars are less than one pixel high. Both Flamenco and RB show all feature labels all the time, which may require scrolling in Flamenco, or unreadably small fonts in RB.

In the commercial product InfoZoom [5] the simplest mode, Overview, is similar to BV. The most common (in their papers and demos) behavior when filters are applied is to zoom in on the selected objects in order to maximize space for their bars and labels. This loses the baseline distribution, and hence the ability to see associations. BV has less need to expand the selected objects because space is dynamically allocated to a selected attribute, labels don't have to fit inside bars, and a greedy algorithm draws labels with the largest filtered count first, which are expected to be the most relevant. The InfoZoom behavior is available in BV with the Restrict button if you want to ignore associations with the current filters and show those in a restricted universe. InfoZoom supports derived attributes, e.g. percentage of movies with a high box office, which can support visualization of associations as height differences. This is a tradeoff of power versus simplicity, and derived attributes have almost unlimited power.

None of these applications show significance, like BV's bar colors, nor do they integrate automated search with user-guided exploration as does the Cluster command.

Mosaic Displays (MDs) [2] do show significance with color. They can show associations among multiple hierarchical attributes, but quickly become difficult to interpret. BV uses multiple MDs each with one n-ary variable (an attribute) and one binary variable (selection). For binary variables the height and color of the positive value uniquely determine those of the negative value, so nothing is drawn to represent the non-selected movies. This makes BV's MDs look more like histograms. This special case is easy to interpret, but cannot show general multivariate associations (though BV does show associations between each feature and the Boolean combination of applied filters). BV also scales heights non-linearly, and adds Bonferroni correction. Attributes are primarily treated as nominal, though multiple selection supports range restrictions as well.

4 Future Work

There is ongoing tension between informing and overwhelming users. In Figure 1, the rollover summary is in the lower right corner and very terse. In order to make a previous version of the video understandable, arrows had to be superimposed to show where to look. The final video uses more verbose popup summaries, which are indeed salient, but can also be annoying.

5 Summary

Bungee View was designed to be simple. Users can't choose what variables to display or how to display them. They can't sort, group, or nest. Yet with the addition of p-values and clustering, an expert user was able to answer all contest questions, while finding additional interesting associations along the way.

The simplified Mosaic Displays encode unconditional attribute distributions with width, conditional distributions with area, and deviations from independence with height. They support alphabetic search by their left to right order, and cardinality search based on width or area. Especially tall or short bars, as well as colored bars, can be picked out quickly even for the unselected attributes. Thus there is room for the 13 movie attributes and thousands of their values on the half of the XGA window devoted to meta-data.

The video answered the contest questions as follows:

- Best Actress is weakly associated with Drama.
- No Oscar category is significantly more or less likely than another to be a box office winner, though there is a strong association with Oscar winners in general.
- The six most bankable people are actresses Cameron Diaz and Halle Berry, cinematographers Oliver Wood and Don Burgess, and actors Clyde Tull and Tom Cruise.
- No multi-person teams are significantly associated with box-office winners due to the large Bonferroni correction.

References

[1] B. B. Bederson, J. Grosjean, and J. Meyer, *Toolkit Design for Interactive Structured Graphics*. IEEE Transactions on Software Engineering, 2004. **30**(8): p. 535-546. http://www.cs.umd.edu/hcil/piccolo/learn/Toolkit_Design_2004.pdf

[2] Michael Friendly, *Visualizing Categorical Data*. 2000: BBU Press.

[3] Gary Marchionini, Carol Hert, Liz Liddy, and Ben Shneiderman. *Extending User Understanding of Federal Statistics in Tables*. in *Conference on Universal Usability*. 2000. Washington, DC: ACM. http://www.ils.unc.edu/~march/CUU/tables.pdf

[4] B. Shneiderman. *The eyes have it: A task by data type taxonomy for information visualizations*. in *Proceedings of the IEEE Symposium on Visual Languages*. 1996: IEEE Computer Society Press.

[5] Michael Spenke and Christian Beilken. *InfoZoom - Analysing Formula One racing results with an interactive data mining and visualisation tool*. in *Second International Conference on Data Mining*. 2000. Cambridge University, United Kingdom. citeseer.ist.psu.edu/spenke00infozoom.html

[6] Ping Yee, Kirsten Swearingen, Kevin Li, and Marti Hearst. *Faceted Metadata for Image Search and Browsing*. in *Proceedings of the SIGCHI conference on Human factors in computing systems*. 2003. Ft. Lauderdale, Florida, USA: ACM Press. http://bailando.sims.berkeley.edu/papers/flamenco-chi03.pdf

Bring your Popcorn and Enjoy the Show!

Heike Hofmann, Dianne Cook, Ulrike Genschel, Hadley Wickham, Michael Lawrence,
Barret Schloerke, Spencer Bradley

Abstract—This paper summarizes the most interesting findings made on the movies data provided for the IEEE Info Vis 2007 challenge. The original 2000-2006 data was supplemented with data from as far back as the 1800s and other information from IMDb and The-Numbers web sites. Findings include expected patterns such as the summer and christmas increase in releases, the oscar timed-releases, in addition to surprise findings such as that Tuesdays are likely to see scary releases, that George W. Bush is a major actor, and that war movies are popular in war years.

1 Introduction

What exactly makes a movie? The first movie to be copyrighted in the United States shows a man taking a pinch of snuff and sneezing. Its premier (and probably only showing) was on April 14 1894. This movie has a length of 5 sec and is available for purchase on the internet, but is probably not what we would think of as a feature film today. One of the main points our team discussed initially was the question of "what makes a movie a movie?". For the purpose of this paper, our criteria for a "proper" movie was the existence of at least 1000 ratings by (re-)viewers, settling once and for all the question whether a movie is a movie if nobody has ever seen it.

One of the issues we encountered with the data quality concerned the matching from different sources. While scripts helped to match, for example, budget numbers and gross income from The-Numbers website to the IMDb data, the famous last percent had to be aligned manually. Most of the titles concerned involved numbers, abbreviations or punctuation that differed between the data sources.

The software packages that we ended up using most were mainly interactive visualization systems that are being developed "in house", i.e. we had some say in making necessary changes, if not performing the changes ourselves. These systems are GGobi (www.ggobi.org), ggplot (had.co.nz/ggplot/) and Manet (www.rosuda.org/manet). We also used Mondrian (www.rosuda.org/mondrian), Many Eyes (http://services.alphaworks.ibm.com/manyeyes/home) and R (www.r-project.org).

All data sets that we used throughout this paper can be found online on our website at http://had.co.nz/infovis-2007/

2 The Storylines

The main focus of our contribution is the exploration of the Movie and Actor Database. Besides the questions provided to us by the organizing committee of the IEEE InfoVis 2007 Contest, we have found and answered a set of other interesting questions. Sit back and enjoy!

Oscar Hopefuls Movie releases peak twice throughout the year; once in Spring and then in Fall (see figure **??**). Oscar nominated movies are typically released very late in the year or just before the Academy Awards ceremony in March or April (typically with pre-release dates at the end of the previous year). Out of the summer nominations only a handful of movies ever made it out as winners, such as *Crash, Gladiator, Pollock, Adaptation, Little Miss Sunshine,* and *Road to Perdition*. Are those surprise nominations, or are they trying to beat the odds deliberately? It'd be nice to know. On a different note, we can say that all but one of the Oscar winning movies of the last years were declared as dramas, while 94% of all nominated movies were dramas. This gives definitely some guidelines for our next home video releases!

Scary Tuesdays Figure **??** shows a barchart of the release day of the week, a histogram of movie ratings and a spineplot of horror movies (0 = no, 1 = yes). Ratings seem to be slightly skewed left with a peak between 6-7. Releases on different week days show the same pattern, i.e. do not seem to have an influence on the ratings - except for

All authors are with Iowa State University, E-mail: {dicook, ulrike, heike, hadley, lawremi, sbradley}@iastate.edu, schloerke@gmail.com.

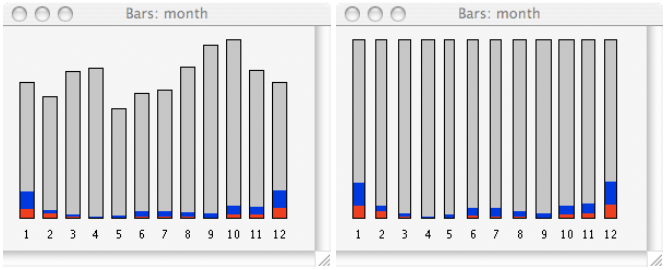


Fig. 1. *Movie releases peak in Spring and Fall (left). Movies with nominations for an Academy Award are painted in blue, Oscar winning movies are shown in red. The spineplot (right) emphasizes the conditional probability of for nominations and wins.*

movies released on Tuesdays. Tuesday releases tend to get somewhat lower ratings with a center somewhere around 4. At the same time, horror movies tend to be released on Tuesdays. Horror movies make up 18.5% of all Tuesday releases - a scary connection!

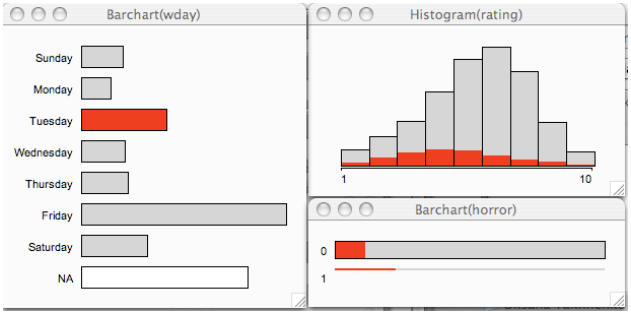


Fig. 2. *Horror movies released on Tuesdays seem to be double scary because of both content and ratings.*

Box Office Flops and Surprises Figure **??** shows a scatterplot of the relationship between box office earnings and budget. Luckily for investors, a higher budget typically implies higher earnings. Some of the relatively most successful movies are *Meet the Fockers* and *My Big Fat Greek Wedding*, both have a small to medium budget but very high earnings. A few of the big-budget movies might be considered flops because box office, although big, was less than the budget, e.g. *Sahara (2005)*, and *Van Helsing (2004). Superman Returns (2006)* and *Hulk (2003)* earned close to their budgets, but did not draw in the 6% additional earnings an average movie is expected make beyond its budget.

There is a seasonal relationship in the box office: late summer and Christmas movies have higher earnings. This is slightly lagged with the number of movies released, where the peaks are late spring and late fall. Oscar winning movies are never among the top box office earners, but they completely dominate the market in January.

Bankability: And the winner is ... ... Orlando Bloom! - at least, if we take participation in successful movies into account. Orlando Bloom was involved in both the box-office busting *Lord of the Rings* trilogy and the *Pirates of the Caribbean* sequel, making him the winner of the

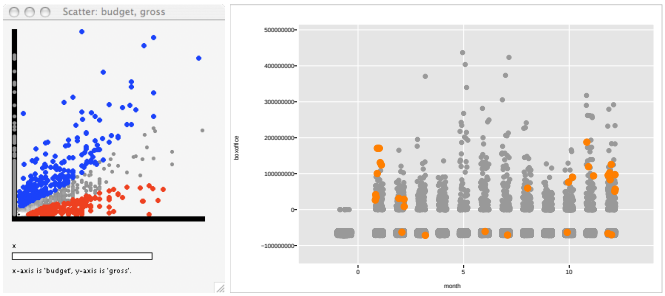


Fig. 3. *Box office earnings versus movie budget (left). Blue movies are doing surprisingly well, whereas red movies are flops. Christmas movies and late summer movies earn more money (right). Here, Oscar winners are painted orange.*

good choices award. To get this result, we extended the movie-person data base by budget numbers and collapsed over person. This way we get a list of the most "bankable" people in the business.

To drive the idea of bankability further, we selected all pairs of actors/directors from the list of movies done by the top 150 bankable people. We had to impose the additional constraint of each pair having done at least two movies together. The results are shown in the networks on our website at http://had.co.nz/infovis-2007/. Some of the pairs we found are well known and established, such as the Clooney/Soderbergh power connection. The two of them produced and directed several films together, such as *Ocean's Eleven, Ocean's Twelve, The Good German, Syriana, Good Night and Good Luck,* and *Solaris*. Other connections are based on sequels, such as the Rodriguez/Banderas team (*Spy Kids 1,2,* and *3*) or the Peet/Willis team (*The Whole Nine/Ten Yards*) - but maybe there's a pair coming to watch out for ...

Some larger networks between actors also show up. We could call the first one the "Sandler" clique (for obvious reasons). The second one stands out because we can find the current President, George W. Bush, united with the former President, Bill W. Clinton. This pair has a respectable three joint movie appearances: *Bowling for Columbine, Bush's Brain*, and *Enron: the smartest guys in the room*.

The Frat Pack The largest connected network of top bankable actors is shown in figure **??**. The person with the most connections is Rick Kain, a stunt double, and with 25 movies between 2000 and 2006, the busiest person of all.

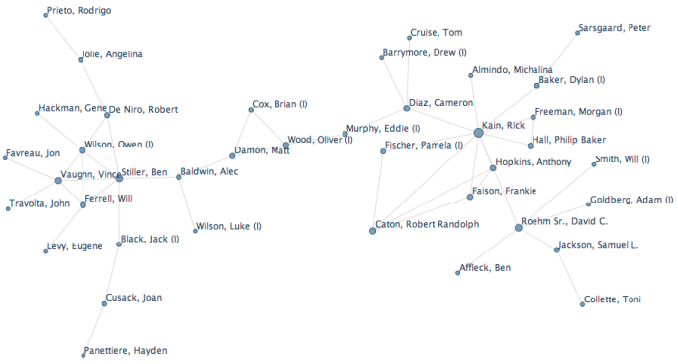


Fig. 4. *Largest connected network amongst 150 most bankable actors.*

In the whole graph there is only a single K4 – a full four-way connection – between actors. Look for it on the left hand side of the figure. Owen Wilson, Ben Stiller, Will Ferrell and Vince Vaughn all have made at least two movies with each of the others. In *Starsky & Hutch* they even appear all side by side. They are part of a tight group of actors jokingly called the "Frat Pack" (as a reference to their joint movie *Old School*). Other members of the group, such as Luke Wilson and Jack Black can be found in the network in the immediate surroundings of the main four. Please note, that the other K4 involves Rick Kain, who is a stunt double, not an actor.

A link to the full network with interactive features can be found at http://had.co.nz/infovis-2007/

Hail to the Chief In the same way as studying "bankability", we can look at other criteria for success, e.g. Academy Awards (where Clint Eastwood was the clear winner) or number of appearances in movies. Here, we received a little surprise - among the top achievers in this category is none other but the current President, George W. Bush! This is due to the way appearances are counted: length of appearance is neglected, but only separate appearances are counted, which means that a 20 min dialogue counts as much as a 5 sec news clip from the archives.

Genres over Time What happens next? This answer can usually be answered most reliably by looking at the history. For that, we have pieced together genres of movies since 1888. Figure **??** shows an excerpt of the development of individual genres. Generally, genres stay overall fairly level over time, as can be seen e.g. for the last couple of decades of mystery movies top left in figure **??**. There are interesting exceptions, though. War movies interestingly show a huge correlation to times of war. We can easily recognize the World Wars, Korea, Vietnam and, to a lesser degree, the Gulf Wars in the peaks (top right). Westerns seem to have dropped out of favor - their numbers are in a steady decline. We've only seen the tip of the iceberg with Reality TV, it seems!

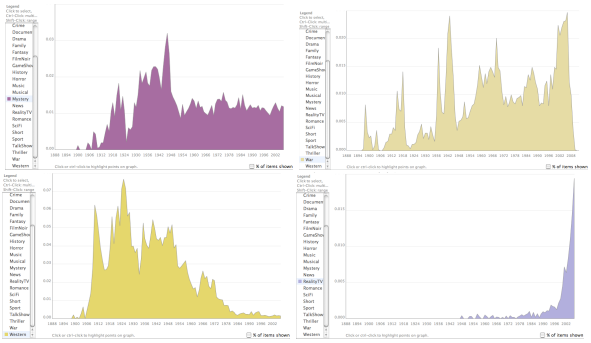


Fig. 5. *Development of Genres over time (1888-2006). Most genres show the same steady percentages as 'mystery' (top left). The other genres show interesting exceptions.*

A link to the full graphic with interactive features is available at http://had.co.nz/infovis-2007/

3 Conclusions

This analysis required an enormous amount of data cleaning and processing. More than half of the movies are characterized to be short films, and the majority of the movies have less than 1000 users rating them. These "movies" are probably not what most people consider to be movies, and hence using these samples will likely produce spurious findings. The findings we've reported were made using very careful dissection of the data, subsets, and scaling in different ways to examine it at multiple resolutions.

Our toolbox contains a plethora of software and hacked together code, which allowed us to extract many storylines from the motion picture data. The findings reported here are a small subset of these. There are more revealed in the accompanying video, and more detailed information on methods on the web page.

Acknowledgements

Thanks to Robert Kosara, T.J. Jankun-Kelly and Eleanor Chair for providing the original data and organizing the IEEE InfoVis 2007 Contest. Further thanks to Martin Wattenberg and Fernanda Viégas for providing Many Eyes and pointing us towards it. Special thanks to Martin for the inspiration to the title! This work was supported in part by National Science Foundation on grant # 0706949.

Interactive Exploration of the Movie DB on a Semantical Level

Thorsten Liebig* Ulm University, Germany
Olaf Noppens† Ulm University, Germany
Timo Weithöner‡ Ulm University, Germany

ABSTRACT

The IMDb can be seen as an ontology made of a schema and a huge network of individuals. This allows to reason about movie data and to define filters in terms of declarative descriptions. We believe that many interesting queries about movies can be answered by interactive visual exploration utilizing browsing primitives such as aggregated club views, selective expansions, or drag-n-drop filters.

1 MOTIVATION AND APPROACH

The Internet Movie Database (IMDb) is a large collection of movie information mainly compiled from user submissions. The core of an IMDb movie record consists of data about involved persons (e. g. director, cast), about the movie itself (genre, locations, awards) as well as additional details (rating, length, etc.). The two former collections have some similarities to a social network. In this regard the movie network is a structure made of different types of nodes (actors, movies, genres, etc.) that are tied by one or more specific kinds of relations (movie appearances, oscar achievements etc.).

In Knowledge Representation (KR), such a vocabulary of named nodes (typically called concepts) and relationships (called properties) along with a formal semantics is called an ontology. Having a formal semantics allows to precisely characterize the concepts and properties of the focused domain (e. g. by defining a property as functional or a concept by a logical expression). This enables to make implicit information within the ontology explicitly available with the help of a reasoning engine.

Our approach combines logical reasoning over an enriched ontological model of the IMDb with an interactive, semantically guided investigation of the manifold interconnected movie network in order to visualize and understand the larger structure of the data or to answer specific tasks. We propose an user-driven exploration strategy, making use of animated expansion steps, clustering techniques, and different levels of detail or abstraction views. During exploration, any expansion set may be narrowed by drag-n-drop of restricting criteria. Instant reasoning feedback is employed for automatic instance classification or to check for conflicting filtering criteria. This is done with help of a specially enhanced version of our ontology browsing and authoring tool ONTOTRACK [1] linked to our high performance relational reasoner U2R2.

2 THE MOVIE ONTOLOGY

An ontology typically is divided into two parts: the schema part introduces concepts and properties and gives structure to them in terms of axioms using language constructs such as sub-concept, sub-property or logical operators. The assertional or data part defines individuals and concrete relationships between those individuals utilizing the concepts and properties of the schema.

*e-mail: thorsten.liebig@uni-ulm.de
†e-mail: olaf.noppens@uni-ulm.de
‡e-mail: timo.weithoener@uni-ulm.de

2.1 Movie Schema

Within the movie domain our schema obviously contains terms such as `Movie`, `Person`, `Female`, or `Genre` and properties like `hasGenre`, `directedBy`, `cast`, or `leadingActorOscar`. The latter relates a movie to a person and is defined as a sub-property of `cast` as well as functional, which means that one particular movie can only have at most one `leadingActorOscar` (other types of oscar awards are still allowed of course). In addition, most properties have an inverse counterpart. For instance, `appearsIn` is the inverse of `cast`.

Furthermore, we have defined some advanced concepts of obvious interest within this domain. For example, an `OscarMovie` is defined as a movie who is related to a person via at least one of the `oscarAward` properties (Fig. 2 shows this definition in a logical KR notation). Moreover, a `Perfomer` is defined as either an Actor or Actress. A `PerformerOfAnOscarMovie` in turn is a performer who `appearsIn` some `OscarMovie`.

2.2 Movie Data

The corresponding assertional part covers all the data about concrete movies, persons, received oscars etc. For instance, in 2005 Cate Blanchett received an award for the best supporting actress in the "The Aviator". Therfore, the `supportingActressOscar` property holds between the individuals `Aviator_The` and `Cate_Blanchett`. As a result, `Aviator_The` will be derived as an instance of `OscarMovie` by our reasoning engine.

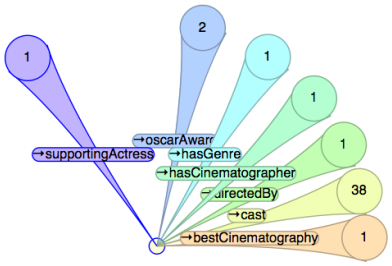


Figure 1: Clickable preview clubs for expanding fillers.

The Web Ontology Language (OWL) has been used as representation format for both parts of the movie ontology. The movie schema with its roughly 50 concepts and 20 properties has been modeled manually following the IMDb record structure. The data part was mapped into OWL syntax by a XSL transformation. To show extensibility wrt. the underlying date we added a `posterURI` to our model storing the corresponding poster URLs which we gathered from the IMDb web site.

3 INTERACTIVE VISUAL EXPLORATION OF THE IMDB

We utilize an adapted version of our ONTOTRACK ontology IDE for exploring the movie data in the following way. The schema authoring component can be used to browse through the concepts and properties as well as to specify additional concepts whenever needed. The data visualization component allows a user-directed exploration of interrelated individuals [2]. The user can start by dragging an individual from a list of all individuals onto the data

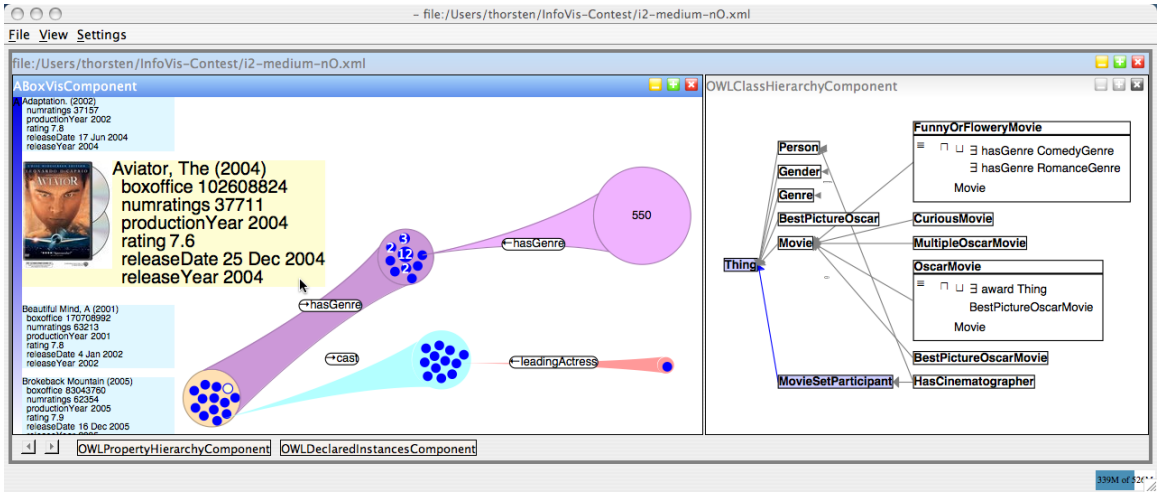


Figure 2: Application screen shot showing data pane on the left and schema pane on the right.

pane. This individual as well as all others are depicted as circles whithin the data pane. Now the user can interactively exploit the related individuals (property fillers) in a step-wise fashion. On mouse click a preview of clubs will show those properties which have fillers and their quantities with respect to the originating one. Figure 1 shows the preview for the movie "The Aviator". One can see that there are 38 fillers of the cast property, 2 fillers of the award property etc. A preview club will expand in an animated fashion when selected by mouse. After expansion, all fillers of this property are grouped within a so-called *property filler cluster* which will be drawn as a colored club (cp. Fig. 3). Now, the user can select a new or the same individual as the source of further property expansions.

To distinguish the various properties different colors are used. Alternatively it is also possible to expand the fillers of a whole cluster with respect to a property as shown in Figure 3. This can be done by choosing the club itself as expansion source. The emerging filler cluster then contains the union of all fillers and is depicted by a broader interconnection. For instance, to reveal all genres of a

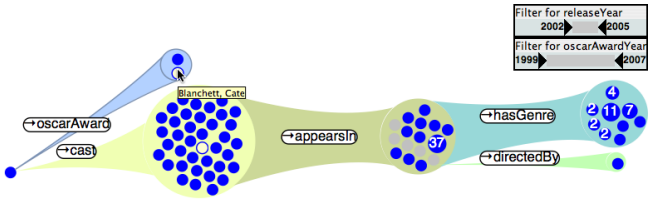


Figure 3: Expanded chain of clubs with release date constrains.

set of movies one simply can expand the union of the `hasGenre` property as shown within Figure 3 with one single mouse operation.

Typically each genre is referenced from a couple of movies. In order to visualize these quantities the diameters of the genre circles will scale proportionally with the number of related movies in the predecessor club. In case there is more than one source individual we also write the number of those on the circles. Whenever expanding a surjective property this technique allows to grasp the distribution of relationships to fillers. E. g. the accumulation of associated genres within a set of movies is depicted as shown in Figure 3.

A list with details for clustered individuals is optionally available. The list content will instantly switch to those individual set the user is hovering with the mouse. For instance, when listing movies their ratings, release and production date as well as movie poster is shown on the left hand side of the view pane. This list can be scrolled and provides a mouse over magnifying effect.

A second dimension of exploration is given by drag-n-drop operations with concepts from the schema. Dropping a concept onto the background of the data pane results in a new initial set of root individuals (the instances of this concept). A concept can also be used as a local filter when dropped onto an existing cluster. For instance, any set of movies can be restricted to oscar movies simply by dragging the `OscarMovies` concept onto this cluster. Non oscar movies will then disappear from this cluster. This is a powerful selection mechanism since concepts can be defined on demand during exploration. The underlying reasoning engine supports the user by instant classification of individuals as well as in providing feedback about nonsensical restrictions (e. g. restricting a set of actresses to male persons inevitably would lead to an empty set).

A local filter, however, may affect adjunctive or prospective clusters. Consequently, a filter will propergate its restrictions. Therefore, when restricting a set of movies to oscar movies each successive club will adapt its filler sets only to those individuals actually related to oscar movies.

Since time is an important variable within the movie domain we also enhanced our tool with user adjustable time sliders. A time slider allows to reveal the correlation of individuals to particular time points or intervals. For example, each movie has fillers for the properties `releaseYear` and `productionYear`. The slider for the latter allows to specify any arbitrary interval of years between 1999 and 2007. All those movies which do not fall into the actual interval will be instantly colored grey. Similar could be done with other concrete domains such as ratings or box office actuals.

3.1 Data Specific Tasks

The proposed approach and our system is best suited for answering qualitative queries, such as the first example from the contest web page ("Do Best Actress Oscar winners tend to come from certain genres?"). Even more complicated queries are easily answered such as "How many actresses who sometime played in a movie which received an oscar also appeared in an adult movie?".

References

[1] T. Liebig and O. Noppens. OntoTrack: A semantic approach for ontology authoring. *Journal of Web Semantics*, 3(2):116 – 131, 2005.

[2] O. Noppens and T. Liebig. Understanding Large Volumes of Interconnected Individuals by Visual Exploration. In *Proc. of the 4th Europ. SW Conf. (ESWC 2007)*, pages 799–808, Innsbruck, Austria, 2007.

Blockbuster – A Visual Explorer for Motion Picture Data

Sebastian Rexhausen* Mischa Demarmels* Hans-Christian Jetter* Mathias Heilig* Jens Gerken* Harald Reiterer*

Human-Computer Interaction Group
University of Konstanz

ABSTRACT

In this paper we introduce our visual explorer "Blockbuster" as a contribution to the InfoVis Contest 2007. The system's development followed a user-centered design process and a design rationale considering not only the pragmatic qualities of the system, but also hedonic qualities like aesthetics or "joy-of-use". Apart from briefly outlining the employed visualization techniques, we will focus on Blockbuster's interaction design, which is aimed at facilitating the selection, combination and mutual filtering of visualizations under a consistent interaction paradigm. Blockbuster thereby demonstrates the potential of information visualization for end-user-centered applications that blur the boundaries of information visualization, visual information seeking and browsing.

Keywords: Visualization, Interaction Design, Semantic, Zooming, Coordinated Views, HyperGrid

1 INTRODUCTION AND APPLICATION SCENARIO

Producing and marketing new motion pictures involves high financial stakes and remarkable risks. Therefore the motion picture data provided for the InfoVis Contest 2007 could deliver valuable and interesting insight for business decisions or the dedicated movie fan. However, the amount and complexity of the data makes it necessary to develop visual tools to facilitate exploration, hypothesis generation and decision making. For example it might be important to discover (among many other questions!) if Samuel L. Jackson has been involved in commercially successful movies during the recent years and how his recent success rate is compared to that in the past.

Blockbuster is a flexible visual exploration application providing such answers and is aimed especially at the novice user without prior visualization experience.

2 DESIGN RATIONALE

Like any user interface and interaction design process, designing information visualization should follow an end-user-centered approach, not only limited to pragmatic qualities like usability or effectiveness, but also in regard to "hedonic qualities" like aesthetics, "joy-of-use" or pleasurable design [1].

Based on the insights and experiences gained from our user research in course of the "MedioVis" project [2], we have formulated a design rationale for Blockbuster that has served as a guideline throughout system development:

Pragmatic qualities – Understanding new visualization techniques is always a demanding challenge for the user and therefore a potential threat to a system's usability. This is especially true considering the nature of visualization as a technical science, "where the technical achievement of doing something new usually outweighs any questions on how useful or necessary a technique is" [3]. For this reason we deliberately use and combine only well-proven and straightforward visualization techniques (e.g. scatter plots, network graphs, bargrams, tables), which are especially appropriate for interactive exploration and direct manipulation (e.g. through dynamic queries [4] or semantic zooming approaches [5]).

Inspired by multiple coordinated views [6] and snap-together visualizations [7] with linking and brushing interaction, the different visualization techniques in Blockbuster can be applied simultaneously on arbitrary subsets of the data and can be displayed in arbitrary combinations and screen layouts. The user can control the layout, the subset of visualized data and the employed visualization techniques with direct manipulation, e.g. by selecting filter criteria, layouts or visualizations through continuous zooming or "drag and drop".

Hedonic Qualities – The selection of colors, transparencies, fonts, font sizes, icons and animations follows a consistent style guide based on our prior experiences and is aimed at achieving high attractiveness for the user. Blockbuster tries to evoke positive emotions by offering an organic and slightly playful look and feel. Furthermore it takes up design concepts known from popular applications (e.g. Firefox) to create a feeling of familiarity and mastery for the user and to improve the system's learnability.

On the level of interaction design Blockbuster tries to create a rich and satisfying user experience by integrating external multimedia content (e.g. images, videos, maps, web pages) through semantic zooming to provide a "browsing the shelves sensation for large collections of information items" as is recommended by [8].

3 INTERACTION PARADIGM

The exploration of data in Blockbuster is based on the usage of simultaneous views, which are linked by linking and brushing behavior. This allows identifying, highlighting and selecting data sets by complementary use of different visualization techniques. Blockbuster is capable of supporting the user in gaining complex insights by iteratively applying such combinations of easy-to-understand visualization techniques to narrow down the amount of visible information and to explore it in task-specific ways.

All these visualization combinations and layouts are organized on tabs similar to the multiple-document interfaces known from state-of-the-art internet browsers. This enables the user to switch between different task-specific views and to keep intermediate results or relevant selections during the exploration process.

In Blockbuster all views within a tab are synchronized and allow mutual filtering of data. If the user filters data in one visualization (e.g. by zooming into a section of a scatter plot), the data will be instantly removed from all other views on the tab. This way the consequences of filtering in one spot can be interpreted by the user through observing the effect on the other visible visualizations.

*e-mail:{rexhause,demarmel,jetter,heilig,gerken,reiterer}@inf.uni-konstanz.de

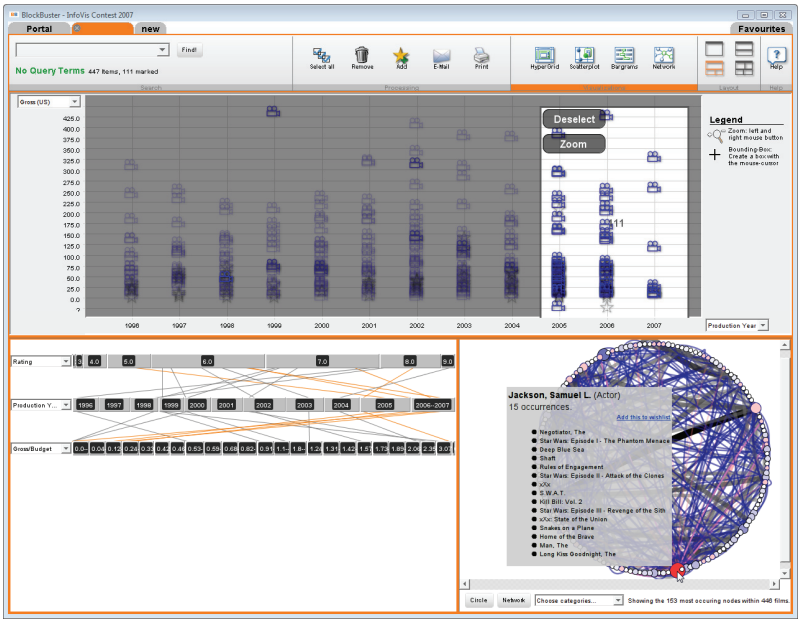


Figure 1. Data exploration with HyperScatter, parallel bargrams and social network visualization

4 Employed Visualization Techniques

Blockbuster is based on four visualization techniques, which are briefly outlined in the following.

The HyperGrid is a novel zoomable table visualization developed by the WG HCI which allows relating, sorting and comparing individual data sets as rows in a table [9]. The semantic zooming functionality of the HyperGrid enables the user to explore metadata and external multimedia information (e.g. videos, maps or content from Wikipedia) "on-the-spot" inside a sticky browser window overlaying the sizable table cells. This way a free exploration of rich external content becomes possible without leaving the table as orientation frame and conceptual model thus overcoming typical problems of the loss of orientation or change blindness.

The HyperScatter is a zoomable two-dimensional scatter plot which allows an overview and the exploration of correlations between quantitative data. The HyperScatter is effective for the selection, zooming and filtering of arbitrary rectangular subsections of the plot and therefore especially supports quantitative filtering and reasoning.

The parallel bargrams in Blockbuster follow the work of [10]. In Blockbuster a single bargram serves as a visualization of the size of data subsets with certain attributes. Multiple attributes can be specified by displaying several bargrams of different attributes simultaneously. Furthermore connecting lines between the bargrams serve as a parallel coordinate visualization of the data [11] to discover correlations and charateristic distributions of the data among attributes.

The network visualization allows analyzing the relation between key persons of different categories (e.g. actors, directors) or plot keywords. Using the JUNG open source library for graph drawing the network visualizes the most occurring persons or keywords in the data as vertices and aggregates all films connecting these vertices in the edges. The number of occurrences is mapped to the color and size of the vertices' shapes and the thickness and transparency of edges.

5 Conclusion

To conclude our description we present a screenshot of the system in figure 1 to illustrate how the answer to the question from the introduction can be given by Blockbuster. However, due to the interactive nature of Blockbuster screenshots can only give a rough idea of the data exploration with the system. Through filtering the time period with the scatter plot and focusing Samuel L. Jackson in the network, we can discover in the bargrams that Samuel L. Jackson has been the most active actor in genres of adventure and action movies from 1996-2007. While his commercial success is clearly decreasing in the last years the IMDb rating of his movies remains approximately equally distributed in a range of 5.0 to 8.0.

6 Acknowledgements

We would like to thank Werner König and Daniel Klinkhammer for their valuable input and all the contributors to the JDIC and JUNG Java open source projects which made Blockbuster possible.

References

[1] M. Hassenzahl, A. Platz, M. Burmester and K. Lehner. Hedonic and ergonomic quality aspects determine a software's appeal. In *CHI '00: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2000, pp. 201-208.

[2] C. Grün, J. Gerken, H. Jetter, W. König and H. Reiterer. MedioVis - A user-centred library metadata browser. In *Proceedings of the 9th European Conference, ECDL, Research and Advanced Technology for Digital Libraries*, 2005, pp. 174-185.

[3] R. Kosara. Visualization criticism - A new way of thinking about visualization. (last checked on 13 July 2007) http://eagereyes.org/VisCrit/VisualizationCriticism.html

[4] C. Ahlberg, C. Williamson and B. Shneiderman. Dynamic queries for information exploration: An implementation and evaluation. In *CHI '92: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 1992, pp. 619-626.

[5] K. Perlin and D. Fox. Pad: An alternative approach to the computer interface. In *SIGGRAPH '93: Proceedings of the 20th Annual Conference on Computer Graphics and Interactive Techniques*, 1993, pp. 57-64.

[6] M. Q. W. Baldonado, A. Woodruff and A. Kuchinsky. Guidelines for using multiple views in information visualization. In *AVI '00: Proceedings of the Working Conference on Advanced Visual Interfaces*, 2000, pp. 110-119.

[7] C. North and B. Shneiderman. Snap-together visualization: A user interface for coordinating visualizations via relational schemata. In *AVI '00: Proceedings of the Working Conference on Advanced Visual Interfaces*, 2000, pp. 128-135.

[8] M. Hearst, A. Elliott, J. English, R. Sinha, K. Swearingen and K. Yee. Finding the flow in web site search. In *Commun ACM*, vol. 45, pp. 42-49, 2002.

[9] H. Jetter, J. Gerken, W. König, C. Grün and H. Reiterer. HyperGrid - accessing complex information spaces. In *People and Computers XIX - the Bigger Picture, Proceedings of HCI 2005*, 2005.

[10] K. Wittenburg, T. Lanning, M. Heinrichs and M. Stanton. Parallel bargrams for consumer-based information exploration and choice. In *UIST '01: Proceedings of the 14th Annual ACM Symposium on User Interface Software and Technology*, 2001, pp. 51-60.

[11] A. Inselberg and B. Dimsdale. Parallel coordinates: A tool for visualizing multi-dimensional geometry. In *VIS '90: Proceedings of the 1st Conference on Visualization '90*, 1990, pp. 361-378.

Overlapper: movie analyzer

Roberto Theron Rodrigo Santamaria Juan Garcia Diego Gomez Vadim Paz-Madrid
University of Salamanca

ABSTRACT

The presented work aims to identify interesting relationships between persons involved in the movie world. The tool integrates different visualization techniques: parallel coordinates, scatterplots, treemaps, word clouds and graph networks, along with textual information and searchs. The visualizations are interconnected and allow overlapping through different data selections to quickly extract data analysis. Though all our visualizations reveals interesting overall information, the analysis converge to our visualization of social networks where detailed interactions are revealed. To prove the analysis power of the tool and obtain more interesting conclussions we have extended proposed data with additional information such as producers, composers, editors and writers, distribution companies, film budgets, etc.

1 INTRODUCTION

In the movie world there are a lot of close interpersonal relationships between people belonging to this environment. This work aims to explore a data set in order to find these relationships by using different visualization techniques. Visualization tools can be used to explore a huge quantity of data and find some relevant information by interacting and putting together some visualization techniques, filtering data by applying different criteria and getting as a result the information we are looking for.

Our proposal is a visualization framework with simultaneous visualizations following a model similar to the Rank-by-Feature Framework [2] that is able to overlap different data subsets in a force graph in order to find the relationships between people involved in different movies.

2 THE INFOVIS CONTEST DATA SET

The InfoVis Contest Data Set contains 20204 movies and 149397 persons, collected from http://www.imdb.com. Each movie has information about its title, year, IMDb rating, number of IMDb votes, genre(s), box office earnings in the USA and oscars won. Also it has information of its director(s), cinematographer(s) and cast. About each person, its name, role in each film and sex is collected. Only movies from 2000 to 2006 are collected, and only those made in the USA. For the cast, only the ten first actors listed as cast on IMDb and all the actors listed as rest of cast are recorded. The original data set was modified in order to include new important information as box office earnings outside USA, the budget spent in each movie, the earnings during the first weekend in USA, the total of screens where the movie was first displayed and the time of running and other awards, specifically . Furthermore, important information about other people involved in the movie was included, like composers, producers, writers, editors, designers, and the producer companies. As a result of this, our data set has been increased, getting a total of 21247 movies, 235442 persons in the cast, becoming a more real data set, even though there is information still missed.

3 VISUALIZATIONS

3.1 Scatterplot

The Scatterplot tool loads the whole data and is able to contrast two variables in a two dimension space model. The main goal of this Visualization Tool is to find a data subset focusing in all the 2-variable relationships, this is useful to find in an easy way the ”less” or the ”most” contrasting values.

For instance we are interested in those movies whose box office earnings are high and spent budget low

3.2 Parallel Coordinates

Parallel coordinates visualization tool loads the whole data and is useful to filter data using all the available variables (overall information of all dimensions together). This visualization tool helps us to evaluate all the possible combinations within whole space of variables, it means that we are able to put together all the criteria to filter data. This is developed by plotting with a line joining the points representing each variable value and leaving a trace. These wide range of traces can be modified by adjusting the limits of the values we are interested in, resulting in a filtered data subset which belongs to our data filter criterion.

3.3 Treemap

The treemap visualization tool can be built with the whole data set or with a filtered data subset. The treemap is hierarchized by levels depending on the criteria we choose, coloring leaf nodes by another criterium. This is a very eye-catching way to find things in common in a huge quantity of data, but also is useful to detect the more or the less common values in a general view of data. The next step, is to feed again our overlapping visualization tool in order to visualize in detail the information, or filtering again using another criterion.

3.4 Word Clouds

The Word Cloud technique is developed to find relevant trends in non-numerical variables as genres, distribution companies and keywords of each movie. Fig. 3 show a word cloud for genres, but even more interesting tendencies can be found by keywords, a kind of data not present in initial dataSet.

4 OVERLAPPER

All the previous visualization tools are used to filter data, due to they are not able to show information in a great detail, so our most important analyzing tool is the ”overlapper”. The main goal of this extended social network [1] is to show just a few quantity of information but in a great detail, using overlapped layers of filtered data, which let us to find relationships between people, movies or awards. This tool is developed using a modified forces graph representing people as nodes, and all the people involved in a movie are joint together in a kind of bubble (surrounded by a hull), resulting in an environment full of bubbles interconnected by people involved in more than one film. Some of this tool properties are explained below.

4.1 Edge Model

There are two edge models used, the first one is the usual model, which interconnects each node with the other ones with which it is related (i. e. other persons that appear in the same movies). The

second edge model is radial, just connecting each person node to a central node representing the movie. This radial model helps to reduce the complexity of the graph.

4.2 Pie charts

Nodes can be visualized as pie charts, divided in as much sections as movies in which the node is involved. Each of this sections is colored by the data sub-selection it belongs. For example in Fig. 2, Joaquin Phoenix has a pie chart divided in three sections, two green and one blue.

4.3 Glyphs

The shape of the node represents roles and awards for persons in each movie (see Fig. 1). Glyphs have been chosen in a way that its overlapping does not clutter information. Also, they try to give an abstract metaphor of each role. Finally, color labels represent sex and golden glyphs represent oscar winners.

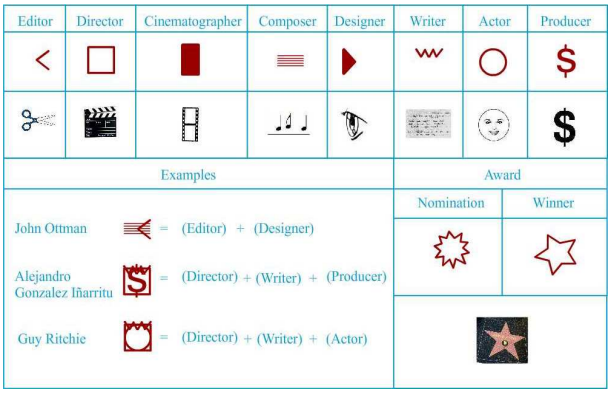


Figure 1: Glyph for person roles in movies.

5 Data Analysis Examples

5.1 Finding the 'winning groups'

As an example, we present the specific task proposed: "Do winning groups on movie sets tend to work together?". To achieve this goal, we do three different data filtering. One of them gets those movies with high box office earnings and low budget. Another one gets most awarded movies, and the last one selects most popular films according to the number of IMDb votes. These three data subsets can be obtained by different techniques (textual search, scatterplot point selection ,parallel coordinates scrolling) and visualized with them. Also, the three data sets are overlapped and displayed by the overlapper view were relevant persons on three 'winning' conditions appear tricolored. For example, as seen in Fig. 2, Christopher Lee (appearing in Star Wars prequels and Lord of The Rings trilogy) is one of the most bankable actor, but not awarded. Both Lord or the Rings' group and Star Wars' group are also 'winning' persons. Harvey Weinstein (one of the most important producers in Hollywood) is producing most of the best critized movies as 'Chicago' or 'The Departed'.

5.2 Our loved ladies

Now we want to search which actresses have been awarded and whit are the most important movie genres in these awards. Fig. 3) shows the result of selecting in scatterplot awarded movies and then making a textual search for just awards regarding actresses. Our extended dataset allows us to select not only Oscar awards but also Golden Globes, Cannes awards and Berlin bears. It's easy with a connected word cloud to see that Drama is the genre from

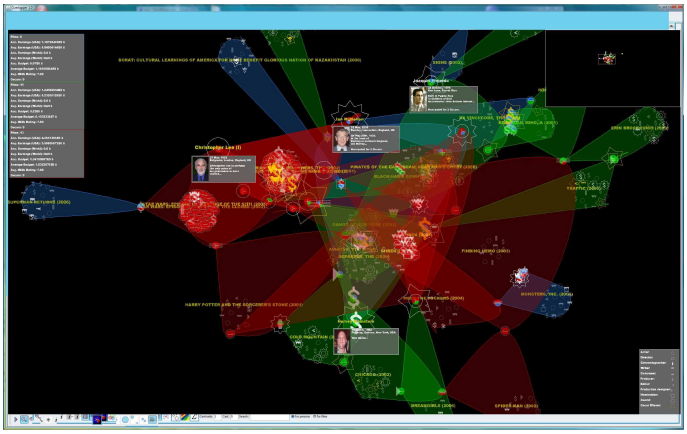


Figure 2: Relationships between people that worked in 'winning' groups. In green we can see academic awarded movies in the most important festivals (Oscars, Golden Globes, Cannes and Berlin). In red there are the most profitable movies while in blue we can see most popular movies.

which ladies win its awards. It's also remarkable how a very secondary genre as musical has regained relevance thanks to films as 'Chicago' and 'Dreamgirls'.

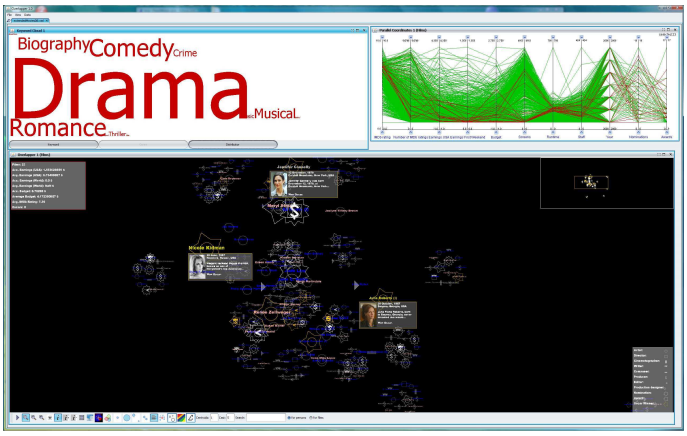


Figure 3: Actresses who have won awards as leading or supporting actresses. Some of them are Julia Roberts, Nicole Kidman, Jennifer Connely and Renee Zellweger. We can also see the profile of its corresponding movies in parallel coordinates along with genre relevance in word cloud.

6 Conclusions

A powerful analysis tool has been developed to gain insight in movie world. Lots of different searches, filtering and visualizations can be done, combining them, visualizing simultaneously and overlapping results in the overlapper. All that functionality gives the chance to find relevant person relationships under different criteria.

References

[1] L. C. Freeman. Visualizing social networks. *Journal of Social Structure*, 1(1), 2000.

[2] J. Seo and B. Shneiderman. A rank-by-feature framework for unsupervised multidimensional data exploration using low dimensional projections. *IEEE Symposium on Information Visualization*, pages 65–72, 2004.

Visual Discovery of Box Office and Oscars in Movie Data

Ying Tu*
The Ohio State University

Teng-Yok Lee†
The Ohio State University

1 Visualization Design

Like most relational databases, the information stored in the movie database is multi-dimensional. It includes multiple types of entities with their attributes, and multiple types of relations.

Although the data themselves are multiple dimensional, it is unnecessary to display all the information when we are querying some relationships among the variables. We chose the classical 2D space for our design because we believe that if carefully designed, 2D image is capable of displaying enough information for a question.

The table, matrix, and scatter plot are traditional 2D presentations. There is something in common among a cell in a table, an entry in a matrix, and a dot in a scatter plot: it is a data point with a particular value to be located in the 2D space.

To visualize the movie data, our design is based on this model. For an given query, we first decide whether tables, matrixes, or scatter plots should be used. Then we determine the meanings of each dimension, and the information a point should display.

1.1 Table

We create a table to represent the movies' attributes, as shown in Figure 2. In the table, each column is for a movie and each row is for an attribute. Whether a movie belongs to a particular movie genre is a boolean attribute. For the given contest data set, each movie has 19 boolean attributes for the genres. In order to observe the relationship between the winning movies of a particular Oscar Award and the genres, we highlight the movies' columns who won particular Oscar awards, and sort the movies by the genres, in which at least one of the movies won an Oscar Award. For the "leading actress", the genres are drama, romance, musical and crime. To stress the combination of those genres, we grouping movies by the genre combination, so that the entities are divided into several groups, and the table becomes multiple smaller tables side by side. By doing this, the distribution of this award to each group is clear.

1.2 Matrix

Matrixes are widely used to understand the relationship of two types of entities. Specifically, rows and columns can be used to represent two elements, and an entry in the matrix can show whether there is a relation between the two elements. The attributes of this relation can be encoded to the entry's representation.

To observe whether the "winning" groups tend to work together, we use the rows and columns represent movies and people, respectively. We only include the Oscar movies and the people who worked for at lease two Oscar movies. For two persons, if both of their columns have data points at the same row, they both worked for that movie. If they have multiple such rows, they often collaborate together. For such pair of columns, if they are far from each other in the matrix, we arrange the order of columns to make such pairs of columns be close to each other to reveal such a relationship.

If we check whether two rows have data points in some columns, the collaborating relationship can also be found. Rearranging the order of rows can also help. In essence, our goal of rearranging the order of rows or columns is to make the data points close to each other. Technically, we are looking for the order that can minimize the sum of the distance of any pair of two data points.

*e-mail: tu@cse.ohio-state.edu
†e-mail:leeten@cse.ohio-state.edu

1.3 Scatter Plots

We often use scatter plots to reveal the distribution of data points in terms of two variables. To discover the correlation between box-office winners and Oscar winners in various categories, based on the model of scatter plots, we distribute movies in terms of the movies' box office and release date. The representation of movies is to display the Oscar awards that a movie won.

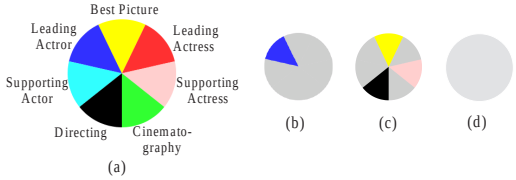


Figure 3: Illustration of the encoding for a movie.

Visualization Design for a Movie. (b), (c), and (d) in Figure 3 are examples of the representation of movies. As shown in (a), each slice represents an award, and each award has its unique color and position in the pie chart. A movie which does not win any award will be in the background color, as in (d). If a movie wins an award, the corresponding slice of the award will be drawn on the person's pie. For example, in (b) the movie wins the leading actor award, and in (c) the movie wins best picture, directing, and supporting actress awards.

To answer the question of what the most "bankable" actors, actresses, directors or cinematographers are, it is hard to use a value to summarize how bankable a person is. So we define some statistics of a person's history of movies's box office, for example the sum of box office, the average of box office, the average of the box office of the movies which are the top 2 or 5 among this person's all movies in terms of box office. Users are free to choose the scatter plots's two dimensions from any of the statistics to place a data point for a person. However, several statistic values are still far from sufficient to reveal the trend for a person's box office, thus we design a representation for a person as described below.

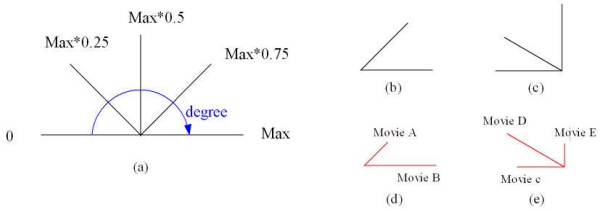


Figure 4: Illustration of the encoding for a person.

Visualization Design for a Person A person is represented as a set of rays from a center point. Each ray represents a movie in which this person took part. The angle of the ray represents the box office: the higher the box office is, the larger the angle is. The color represents the person's role in the movie, for example, actors, directors, and cinematographers use different colors. The length of a ray represents the person's billing rank in a movie: the higher the

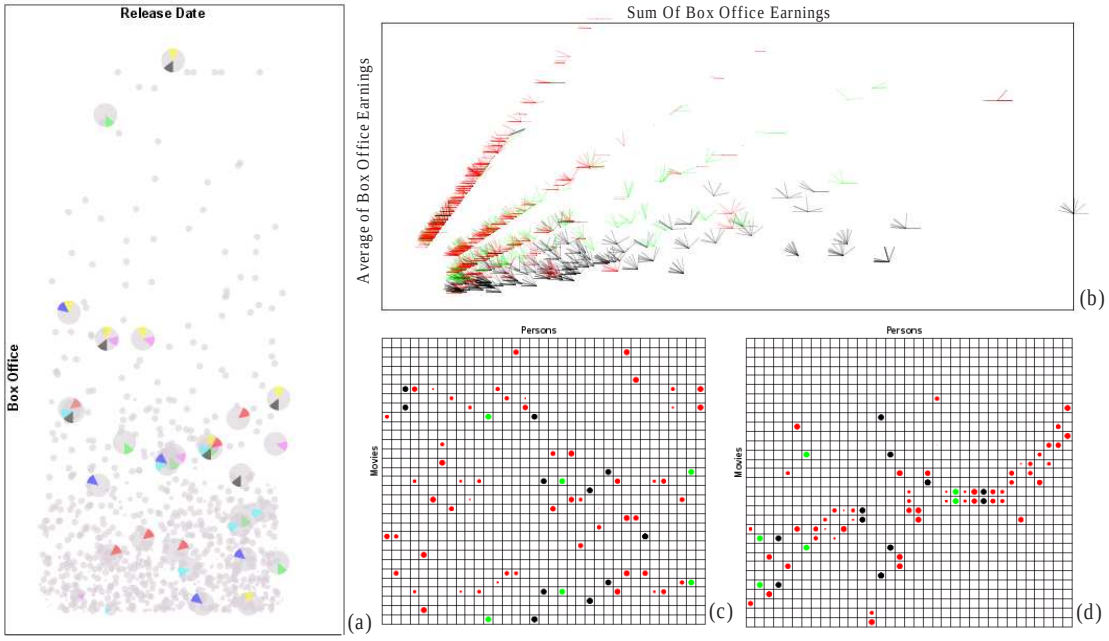


Figure 1: (a) shows the correlation of the categories of awards and the box office; each dot represents a movie. (b) shows how bankable a person is; each dot is a person. (c) and (d) shows the winning groups' cooperation relation; each dot means that the person that this column stands for took part in the movie which this row stands for.

Figure 2: A table of movies and attributes. It can show the relationship of an award category with the genres.

rank is, the longer the ray length is. For example, the person in (b) of Figure 4 took part in two movies as the director; one movie's box office is the max value and the other's box office is $Max \times 0.75$. The person in (c) took part in three movies with the box office of 0, $Max \times 0.17$, and $Max \times 0.5$, respectively. The person in (d) and (e) are actors, the rank of Movie A in (d) is not very high, but that of Movie B is the highest.

2 Data Preprocessing and Augmentation

Data filtering was applied to the original data. We noticed that there were some TV series in the given file. Because TV series and movies are not very comparable, we filtered all TV series.

In addition to the original dataset, we also considers other information that are useful for movies fans. The most important one is the billing rank of the casts for all movies. From the ranks, the contribution of each actor/actress to a movie can be estimated. The rank can be found from the offline IMDB dataset [1], which is also available in the compressed file from the contest website.

When we estimate how bankable a person is, an actor/actress's billing rank is considered to be very important information. For a actor, if the movies in which his/her billing rank is below six, he/she is not considered as this person's contribution. We count all directors, all cinematographers, and only the top six billing ranked actors/actresses, as the contributors to the movie's box office.

3 Visual Analysis

From Figure 2, we can see that all winning movies for *leading actress* is from *drama*. Specifically for some genre combination, such as *drama, romance, and crime*, even the total number of such movies is small, there is a *leading actress* winner, in *Monster (2003)*. It may indicate that this combination is favored.

Figure 1 (a) shows that movies winning *directing* or *best picture* usually has high box office. Movies, winning *supporting actress*, often comes with *best picture*, so their box office is higher than some other award winning movies.

By exploring the user interface like Figure 1 (b), we found the most bankable actor, director, cinematographer may be George Lucas, Peter Jackson, and Andrew Adamson, respectively, who are located near the right top corner and from their points, we can see they mainly work for movies of the highest box office.

From Figure 1 (d), two movies have a lot of persons working for both movies. The movies are *Lord of the Rings (2001)* and *Lord of the Rings(2003)*. The people of *the Aviator (2004)* and *the Departed (2006)* also cooperated closely.

References

[1] Infovis contest 2007 data. http://eagereyes.org/InfoVisContest2007Data.html.

From Beautiful to Useful: A Multi-Scale Visualization of Users Movie Ratings

Romain Vuillemot*
LIRIS, INSA Lyon, France

Verónika Peralta†
PRiSM, Versailles, France

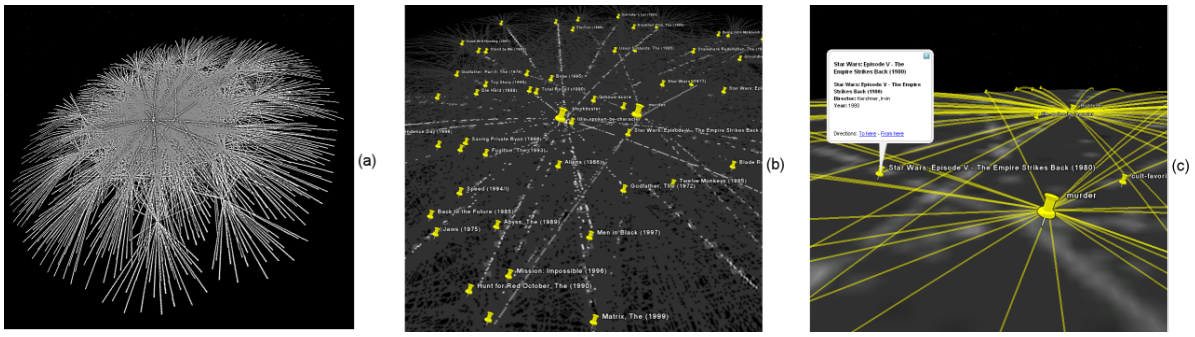


Figure 1: From left to right, from beautiful to useful: (a) movies and keywords relationships overview (with an appealing rendering), (b) movies and keywords relationships (with movies titles and keywords) and (c) movies filtering and details available by user interaction.

Abstract

Interactive environments lack of attractiveness and sex-appeal. While nowadays so many digital arts are available, they have not been included in navigation processes yet. In this paper we suggest to include both artistic and inspiring depictions of data, while proposing an interactive query environment. We present our visualization of the InfoVis 2007 contest data set, focusing particularly on movies rated by users. Our contribution is twofold. Firstly, we extend the contest data set with individual user ratings on movies and we store data in a relational database, allowing SQL-like queries. Secondly, we support multi-scale analysis of query results based on graphical representation of data. This allows high-level analysis of query results and detailed-analysis of specific results by zooming in the data of interest, filtering and getting details on demand. Our proposal incorporates post-processing visualization and hijacking geo-spatial environments in order to explore data. Our approach distinguishes two main phases: (i) construction of a relational database about movies and ratings, (ii) development of an interactive query environment that presents query results in an innovative geo-spatial-like way. Section 1 briefly describes the preparation of the data set, and then, Sections 2 and 3 present data visualization issues and solutions.

1 Data Set Preparation

The original XML-formatted contest data set has been transformed into a relational database. Such a transformation allows us to benefit from Database Management System functionalities, specifically, from efficiently answers to user queries.

We augmented the data set with other publicly available movie features (extracted from IMDb) and user ratings on movies (extracted from MovieLens). The additional movie features consist in more than forty attributes describing movies (e.g. genres, languages, keywords) and persons involved in movies (e.g. actors, directors, producers). User ratings consist in individual users' evaluations for each movie (rather than average ratings available in the original data set). We sorted a total of 52 tables and 15 aggregation views.

The challenge when studying movie databases is to find correlations between movie features and user behaviors. In that respect, we crossed movie features with the evaluations of each user obtaining a collection of preference rules of the form *attribute = value : support*; for example *genre = Action : 0.80* means that 80% of movies evaluated by a certain user are Action movies. Statistics on the obtained rules are shown in Figure 2. The attributes that resulted to be more representative of user preferences were keywords and genres. We specially considered these attributes for analyzing data. We also think that social trends come with huge quantity of data, thus we will focus on large subsets of data in our forthcoming analysis.

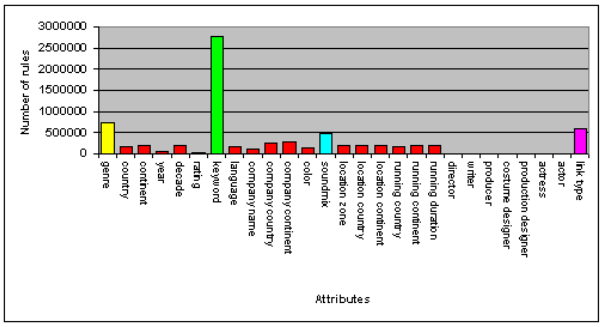


Figure 2: Number of preference rules obtained by movie feature

2 Information Visualization

The result of a query is represented as a Graph. Vertices represent nominal attributes (e.g. movies, keywords, users). Edges represent

*e-mail: romain.vuillemot@insa-lyon.fr
†e-mail: veronika.peralta@prism.uvsq.fr

relations among attributes (e.g. a user evaluated a movie). Ordinal attributes (e.g. ratings, budgets, revenues) are represented as graphical properties of edges or vertices (e.g. color). Movie ratings don't hold an explicit layout, thus we picked-up auto-organizing graph layouts based on relationships between movies and attributes of interest. For example, movies having similar keywords are placed nearly in Figure 1b.

Advantages of this representation are: (i) the possibility of seeing great amounts of data at a glance, (ii) the graphical representation of data relationships (instead of presenting long lists of tabular data) and ordinal attributes, and (iii) the neighborhood of data having similar relationships. The major drawback of such a technique is the lack of knowledge the user gets on the data location and their persistence: two layouts of an identical data set might result differently and not always optimally. We managed to circumnavigate that by means of consistent and intuitive coloring. We used LGL [1] to display data resulting from queries.

Post-Processing Image Rendering

We found mandatory to apply an extra step upon the usual visualization generating process [2], using Image Analysis techniques. Image Analysis aim to extract properties from an image, with similar models than the human eye. Among all the existing techniques, we used 2D image filtering capabilities, in order to (i) extract features to assist the user and (ii) provide an artistic looking image while still keeping the same data layout. The trick of image filtering is that every pixel of the image takes the product sum of surrounding pixels. For instance, blur is done by taking the weighted average of the current pixel and its 8 neighbors. Technically, the vicinity is integrated by means of convolution matrix such as the following (Gaussian blur filter kernel example):

$$\begin{pmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{pmatrix}$$

We found across a wide variety of filters that none of them is specifically optimal to a situation; it has to be strategized according to user's tastes. Figure 1a shows a Laplace filter effect combined to a color filling.

3 Hijacking Geo-Spatial Environments

A Geo-spatial environment provides efficient and intuitive interfaces, by means of multi-scales data exploration. Based on Earth metaphor, it doesn't require a particular learning: everybody inerrantly knows how to "handle earth", from mechanical laws to pictures and miniatures models one might have had as kid. Thus, interactions are easily discoverable even for a non-geo-spatial application. We re-used this paradigm for non-geo-spatial tasks.

2D-Visualizations are mapped into a virtual 3D-sphere, which provides both WIMP (menus, icons) and POST-WIMP (zoom, pan, rotate) interactions. Interactions are quick and permit the user to be lost or to wander randomly to explore visualizations and adopt the tool. We focus on two very interesting features: *Focus+Context* (my means of spherical distortion) and *Multi-Scale* (according to user's altitude).

Focus+Context keeps the focal point at full size and detail, while having view on the surrounding with spherical distortion. By adding a small external map, with a higher point of view we can help the user to enhance his capabilities. By enlarging the map on the globe the spherical distortion will increase.

Multi-Scale is linked to user's *altitude* and indicates how close or far is the user to the visualization. Thus, a strategy is set up coordinating appropriate visualizations at each step of view. Our three major goals are based on known techniques [3] and can be formulated as: (i) attract the user, (ii) give him good insight of the data, and (iii) let him filter data himself. These goals are achieved by three levels of data visualization, each one conveying a different kind of information but preserving data layout:

***Beautiful* Overview Layer:** is a highly abstracted visualization with structural information only, which has been post-processed to provide an appealing look. The goal is to catch the user's attention, but also providing quantitative analytics elements. Then, the user can identify clusters and areas of interest, and takes the decision to get details by lowering his altitude.

Zoom Layer: is a transitional layer which is very reactive, which starts to partially provide details. No more artistic experience here; the result is raw and some qualitative elements are available. The user is not stuck at this level range: he can go back or forth quickly, especially if he made a mistake or if he already knows what his target is.

***Useful* Details Layer:** is an exhaustive layer about details that the user can dynamically filter (with acheck box). The post-processed image is slightly blurred and becomes a background support, with some drawing on top of it to convey detailed informations. Vectorial drawings (with high resolution) are performed and coupled with pinpoints on the map.

Figure 1 illustrates a typical multi-scale analysis of movies keywords.

4 Experimentations and Conclusions

We implemented our system with client-server architecture: data visualization is performed on a distant server and the client application maps the visualization on the 3D-sphere that provides very-reactive environment and dynamic local data selection.

We evaluated our approach for several types of user queries, which are detailed in our contest web page. Our experimentations validated our intuition that graphical, multi-scale and inspiring visualization aids understanding data and provides an alternative point of view for decision-making analysis. The data selection interface has to be enhanced in order to allow users to write their own requests.

Many fields were bridged together (Databases, Computer Graphics, Information Visualization, Geo-spatial Information Systems) and we went a bit further than just a conceptual mock up, by recombining widely spread software.

Our main perspective is to augment the server with more libraries and layouts techniques, in order to provide a wide range of visualizations technique and to offer a Visualization-On-Demand service (VizOD). Another perspective is to define user *visual* profile criteria, according to user's tastes, perceptual capabilities and culture.

Acknowledgements

This research was partially supported by the French Ministry of Research and New Technologies under the ACI program devoted to Data Masses (ACI-MD), project #MD-33.

References

[1] A. T. Adai, S. V. Date, S. Wieland, and E. M. Marcotte. Lgl: creating a map of protein function with an algorithm for visualizing very large biological networks. *J Mol Biol*, 340(1):179–190, June 2004.

[2] E. H. Chi. A taxonomy of visualization techniques using the data state reference model. In *INFOVIS*, pages 69–76, 2000.

[3] B. Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *VL '96: Proceedings of the 1996 IEEE Symposium on Visual Languages*, page 336, Washington, DC, USA, 1996. IEEE Computer Society.

InfoVis 2007 Contest Entry: Cinegraph

Chris Weaver*

The GeoVISTA Center and the Department of Geography
The Pennsylvania State University

ABSTRACT

Cinegraph is an interactive visualization for exploring and analyzing the InfoVis 2007 contest data set derived from the Internet Movie Database (IMDB). By combining two complementary visual interaction techniques, *cross-filtered views* and *attribute relationship graphs*, Cinegraph supports a wide variety of general and highly-focused analytic tasks. Users can express complex lines of questions in the form of rapid sequences of simple interactions. Designed and built in a little over two days by a single visualization designer using the Improvise visualization environment, Cinegraph provides high-dimensional interactive drill-down capability into the people, genres, awards, release dates, and box office characteristics of movies described in the database, using ancilliary photographs of people, images of movie posters, and icons of movie genres to enhance the interaction process.

Index Terms: D.2.2 [Software Engineering]: Design Tools and Techniques—User Interfaces; H.5.2 [Information Systems]: Information Interfaces and Presentation—User Interfaces

1 INTRODUCTION

Improvise [4] is an information visualization environment that integrates document design and dissemination into a self-contained user interface along the lines of popular spreadsheet "productivity applications". The visualization builder and browser capabilities in Improvise are inspired by approaches developed in DEVise [1] and other earlier systems such as Snap-Together Visualization [2] that bring database queries together with coordinated multiple view visualization techniques. These approaches provide rich data browsing by translating interactions into navigation of multidimensional space and selection of data items across multiple views.

The user interface design of Cinegraph maps seven key aspects of the movies database—movies, ratings, release dates, genres, Oscars, people, roles—into coordinated multiple views laid out in seven areas of the user interface. Together, these views combine two visual techniques for exploring small webs of relationships between spatial, temporal, and abstract attributes in high-dimensional data sets. *Cross-filtered views* provide drill-down set queries across multiple tabular data columns by turning on and off brushing of arbitrary sets of attribute values, as originally developed for visual analysis of historic hotel registries [3]. *Attribute relationship graphs* are node-edge-pack diagrams that support exploration of attribute value cooccurrences across two or more columns of tabular data using a multigraph representation that can be dynamically filtered on selection of columns as well as on selection of arbitrary subsets of attribute values in those columns—essentially a fully-filterable mapping of the top two levels of an N-dimensional data cube into the nodes and edges of a graph. The graph is implemented so as to support manual and force-directed layout even while it is undergoing dynamic filtering and other coordinations.

*e-mail: cweaver@psu.edu

Using Improvise, *Cinegraph* was conceived, designed, built, and applied to the InfoVis 2007 contest data set in under a week. This time was split roughly equally between data preprocessing, actual live design of the Cinegraph visualization interface, and interactive exploration and analysis of the data. Various view and query compositions that make up Improvise visualizations, including the "reruns" calendar and the attribute relationship graph, have been isolated and saved in a reusable, data-neutral manner for incorporation into other visualizations. Reuse of such visualization "chunks" was a large factor in the speedy development of Cinegraph.

2 DATA PREPROCESSING

A principle design goal for Cinegraph was maintenance of reasonable interactivity, even during the expensive operations needed to generate the attribute relationship graph. To achieve this goal, it was necessary to perform offline preprocessing of the original contest data set. First, the provided XML document was transformed and reduced in size by converting it into four comma-delimited text files, one each for movies, people, genres, and Oscars. Second, the movie file was filtered to remove invalid box office values. Justification for this choice comes from the desire to have useful movie numbers be a integral part of the analytic process. Third, the other three files were filtered to remove references to such movies. Finally, the people file was filtered to remove people with less than two roles. This procedure reduced the size of the overall data set by a factor of about ten. Movies were reduced from 20204 to 1324, people entries from 246755 to 23372, and genres entries from 35384 to 3281. All 43 Oscar entries were preserved.

After reading in the four text files upon initialization, Cinegraph also performs an adjustable amount of additional online preprocessing controlled by two hidden interactive parameters: a number of ratings threshold that filters out references to infrequently rated movies from all four tables, and a number of roles threshold that further filters the people table. Good interactivity was achieved using thresholds of 10000+ ratings and 5+ roles, further reducing tables sizes to 433, 1439, 1324, and 33, respectively. Nevertheless, many of the contest tasks were performed with these thresholds at zero by using fast hardware and exhibiting a little patience.

3 TASKS AND QUESTIONS

A few of the tasks performed and questions expressed in Cinegraph during analysis of the contest data set include:

- On what days of the week do movies tend to be released? Are any Monday releases special in terms of holidays or Oscars? When do movies tend to be released throughout the year?
- What are the biggest release date(s) in terms of number of movies? Which movies were released on these dates? What genres are represented, who was involved, and what are the box office characteristics of those movies?
- Which genres tend to have the highest box office? Which of the highest grossing movies are top-rated? When were they released? Who was involved? Do (movies of) any genres tend to be released at unusual times (of the week or year)?

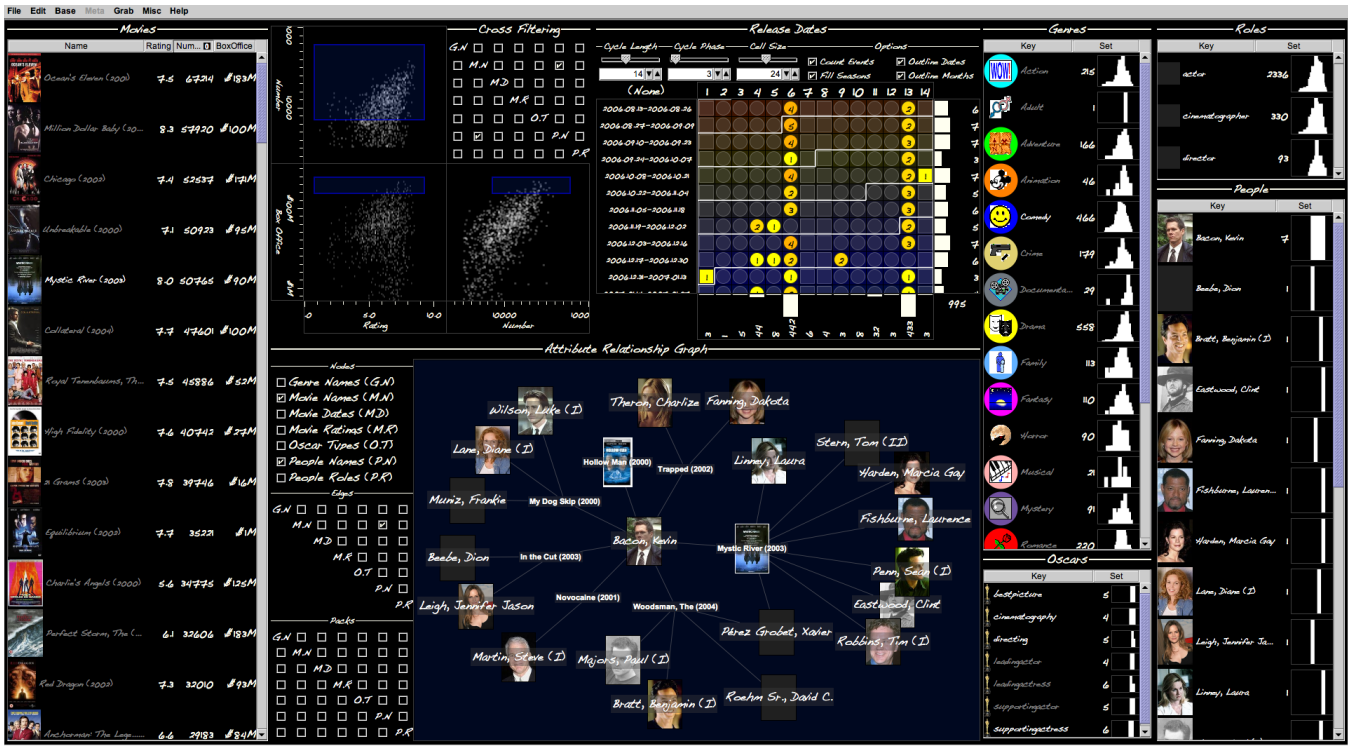


Figure 1: The Cinegraph interface, showing all people who have been involved in one of Kevin Bacon's seven movies from 2000-2007.

- Which do Oscars relate to genres? To release dates? To ratings? Is there a pattern/tendency in Oscar winning by movies, such as a tendency to win multiple awards?
- Which people tend to have roles in low-rated but high-grossing (say, more than $10M) movies? How many movies *have* the Wilson boys (Luke and Owen) been in together?
- Who are the "hub" people? Who have they been in movies with? Who is in the "N degrees of Kevin Bacon" set for the (admittedly small) subset of recent films in the data set?
- Who has been in an animation film more than once? How do animation films do at the box office and at Oscar time? Which animation movies are not classified as family movies? How do they do number-wise compared to family animation films?

Users pose complex inquiries into multi-attribute relationships in the form of simple sequences of interactions. For instance, figure 1 shows an intermediate state in an inquiry into the social network of people who have been involved in movies together, as in the infamous "N degrees of Kevin Bacon" game. The visual state in the figure is achieved by turning on movies, people, and their edges in the graph, then selecting Mr. Bacon in the people table. At this point, the graph shows $N = 0$, the movies in which he has been in with himself. Next, filter movies on people, select all visible movies, filter people on movies, then select all visible people. At this point, the graph shows $N <= 1$, the people who have been in a movie with him. Further degrees on separation are a matter of repeating this interaction sequence. For $N <= 2$, the graph contains around 400 people. With a branching factor of 20 even on such a small subset of the IMDB database, no wonder the game is so hard!

4 Conclusion

Cinegraph can be used to perform many more analytic tasks on the contest data set than could ever be described here. The single representative task described above is intended to suggest the flexibility and effectiveness of the information visualization design approaches that have been used in Cinegraph in particular, and that could be employed in Improvise visualizations in general. (The accompanying web page of contest task solutions in fact enumerates 14 tasks with subtasks corresponding to the questions listed above, each with screenshots and insights gained.)

Improvise is available as a Java Webstart application at http://www.personal.psu.edu/cew15/improvise/, along with source code released under the GPL. A web-suitable (i.e. image-free) version of Cinegraph is available with other example visualization documents at the same location.

Acknowledgements

The author wishes to thank Alan MacEachren and the rest of the gang at the GeoVISTA Center and the North-East Visualization and Analytics Center (NEVAC). Particular thanks to Anthony Robinson and David Fyfe for feedback on attribute relationship graphs.

References

[1] M. Livny, R. Ramakrishnan, K. Beyer, G. Chen, D. Donjerkovic, S. Lawande, J. Myllymaki, and K. Wenger. DEVise: Integrated querying and visualization of large datasets. In *Proceedings of the International Conference on Management of Data (SIGMOD)*, pages 301–312, Tucson, AZ, 1997. ACM.

[2] C. North and B. Shneiderman. Snap-together visualization: A user interface for coordinating visualizations via relational schemata. In *Proceedings of the Working Conference on Advanced Visual Interfaces (AVI)*, pages 128–135, New York, NY, USA, May 2000. ACM Press.

[3] C. Weaver, D. Fyfe, A. Robinson, D. W. Holdsworth, D. J. Peuquet, and A. M. MacEachren. Visual exploration analysis of historic hotel visits. *Information Visualization*, February 2007.

[4] C. E. Weaver. *Improvise: A User Interface for Interactive Construction of Highly-Coordinated Visualizations*. PhD thesis, University of Wisconsin–Madison, Madison, WI, June 2006.

IEEE Visualization Conference,
IEEE Information Visualization Conference and
IEEE Symposium on Visual Analytics Science and Technology

October 28 - November 1, 2007
Sacramento, California, USA

InfoVis Art Exhibit

Lipsticks

Stacy Greene

www.stacygreene.com

Figure 1. Four photographs from the *Lipsticks* series: *Wendy* (1993); *Lisa I* (1992), *Ellen* (1993); *Simona* (1993).

Abstract

Lipsticks is a collection of macrophotographs of lipsticks owned by a variety of women. The colors, textures and repetitively eroded shapes of these intimate artifacts densely encode rich information about their individual owner's complexion, dress habits, and spatiotemporal gestures.

CR Categories and Subject Descriptors: J.6 [Computer Applications]: Arts and Humanities – Fine Arts.

Additional Keywords: information visualization, small multiples, lipstick, cosmetics.

Statement

The *Lipstick* photographs were created as a series of 20" x 24" color close-ups of lipsticks, with each print titled with the name of the lipstick's owner, and distinguished by striking variations in form and texture arising from the owners' individual techniques of application.

\

The everyday, factory, 'ready-made' product turned into a surreal, biomorphic, subconscious image – a sculpture evolving from a private daily ritual taken for granted. A personal object/process that reveals , through colors and shapes, a relationship of imprint at the periphery of the body.

The genesis of the project came a number of years ago after walking out of the Whitney Biennial in New York City. A friend of mine, Rosie, dropped her lipstick, so I picked it up and unscrewed it for her. Rosie's lipstick excited me more than anything I had seen at the Whitney and sparked this photographic work.

Biography

Stacy Greene (stacy@stacygreene.com) is an artist and photographer based in Brooklyn, New York.

We Feel Fine:

An Exploration of Human Emotion in Six Movements

Jonathan Harris and Sepandar Kamvar

wefeelfine.org

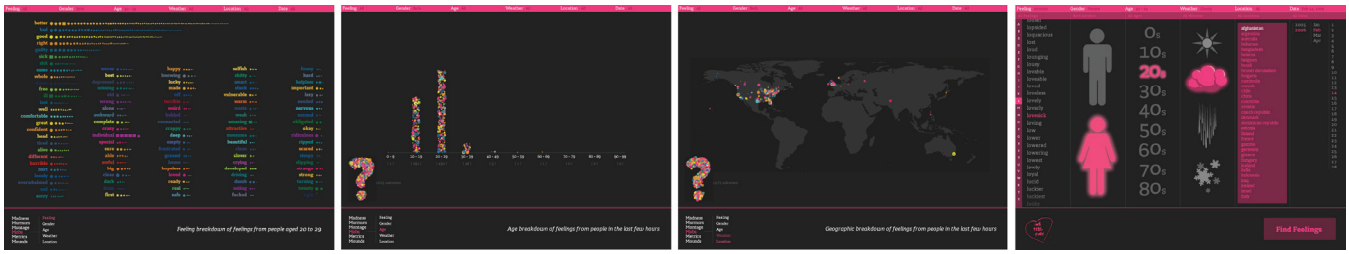


Figure 1. Interactive search and browsing interfaces from *We Feel Fine*.

ABSTRACT

Since August 2005, *We Feel Fine* has been harvesting human feelings from a large number of weblogs. Every few minutes, the system searches the world's newly posted blog entries for occurrences of the phrases "I feel" and "I am feeling". When it finds such a phrase, it records the full sentence, up to the period, and identifies the "feeling" expressed in that sentence (e.g. sad, happy, depressed, etc.). Because blogs are structured in largely standard ways, the age, gender, and geographical location of the author can often be extracted and saved along with the sentence, as can the local weather conditions at the time the sentence was written. All of this information is saved in a database that grows by roughly 20,000 new feelings per day.

Using a series of simple playful interactive interfaces, the feelings can be searched and sorted across a number of demographic slices, offering responses to specific questions like: do Europeans feel sad more often than Americans? Do women feel fat more often than men? Does rainy weather affect how we feel? What are the most representative feelings of female New Yorkers in their 20s? What do people feel right now in Baghdad? What were people feeling on Valentine's Day? Which are the happiest cities in the world? The saddest? And so on..

CR Categories and Subject Descriptors: J.6 [Computer Applications]: Arts and Humanities – Fine Arts.

Additional Keywords: information visualization, interactive online art, demographics, emotion.

BIOGRAPHIES

Jonathan Harris is a computational artist working primarily on the Internet. His work involves the exploration of humans through the artifacts they leave behind on the Web.

Sepandar (Sep) Kamvar is the technical lead of personalization at Google and a Consulting Professor of Computational Mathematics at Stanford University.

Figure 2. Photographs retrieved and compiled by *We Feel Fine* in tandem with written statements from bloggers.

Flags by Colours

Shahee Ilyas

shaheeilyas.com

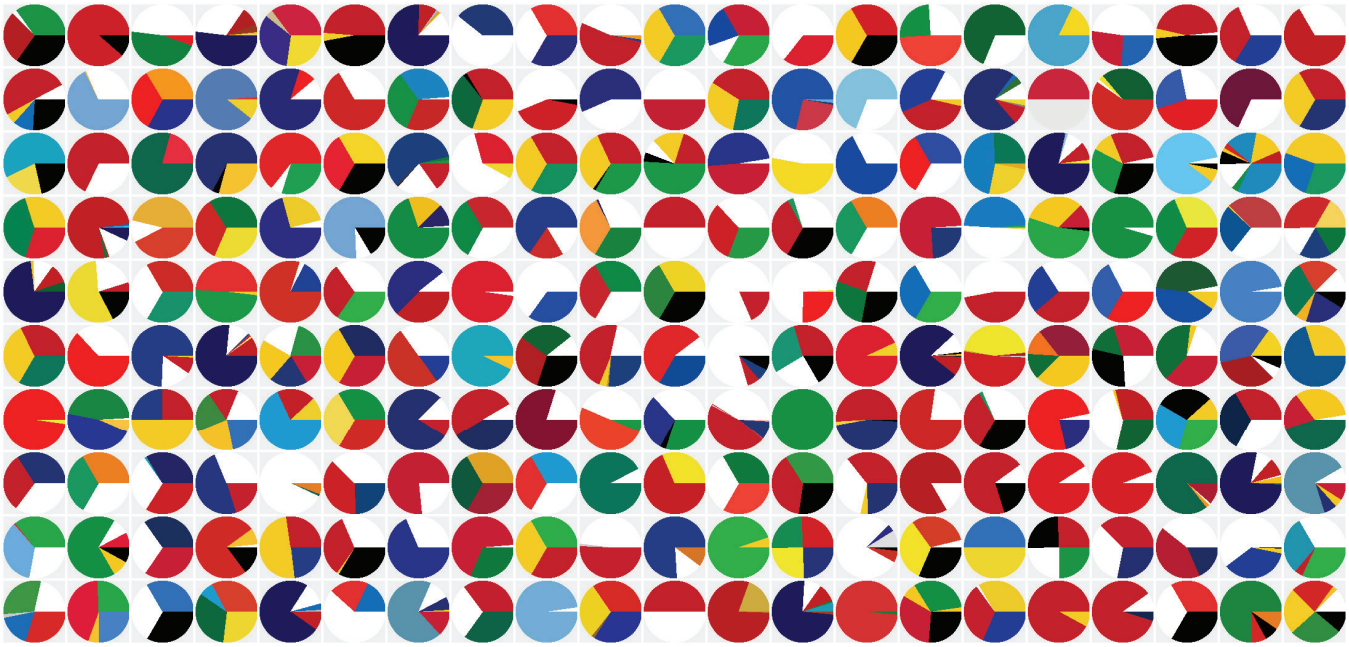


Figure 1.Flags by colours.

Abstract

Using a list of countries generated by The World Factbook database, national flags of countries fetched from Wikipedia (as of 26th May 2007) are analysed by a custom made Python script to calculate the proportions of colours for each of them. That information is then translated into a piechart using another Python script. The proportions of colours on all unique flags are used to finally generate a piechart of proportions of colours for all the flags combined. Colours making up less than 1% may not appear.

CR Categories and Subject Descriptors: J.6 [Computer Applications]: Arts and Humanities – Fine Arts.

Additional Keywords: information visualization, color, flags, pie charts, small multiples.

Biography

Shahee Ilyas (shahee@shaheeilyas.com) recently graduated with a Master of Arts in Media Design from the Piet Zwart Institute for postgraduate studies and research, Netherlands. Ilyas lives and works in Maldives, where he now develops and implements projects for clients ranging from some of the largest companies to some of the smallest grass-root level organizations.

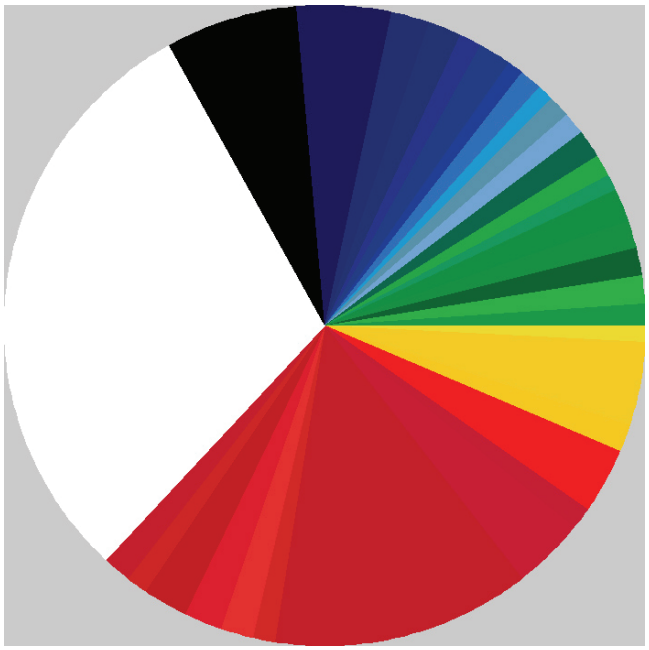


Figure 2. Aggregate color proportions for all flags combined.

Eventide

Cassandra C. Jones

cassandrac.googlepages.com

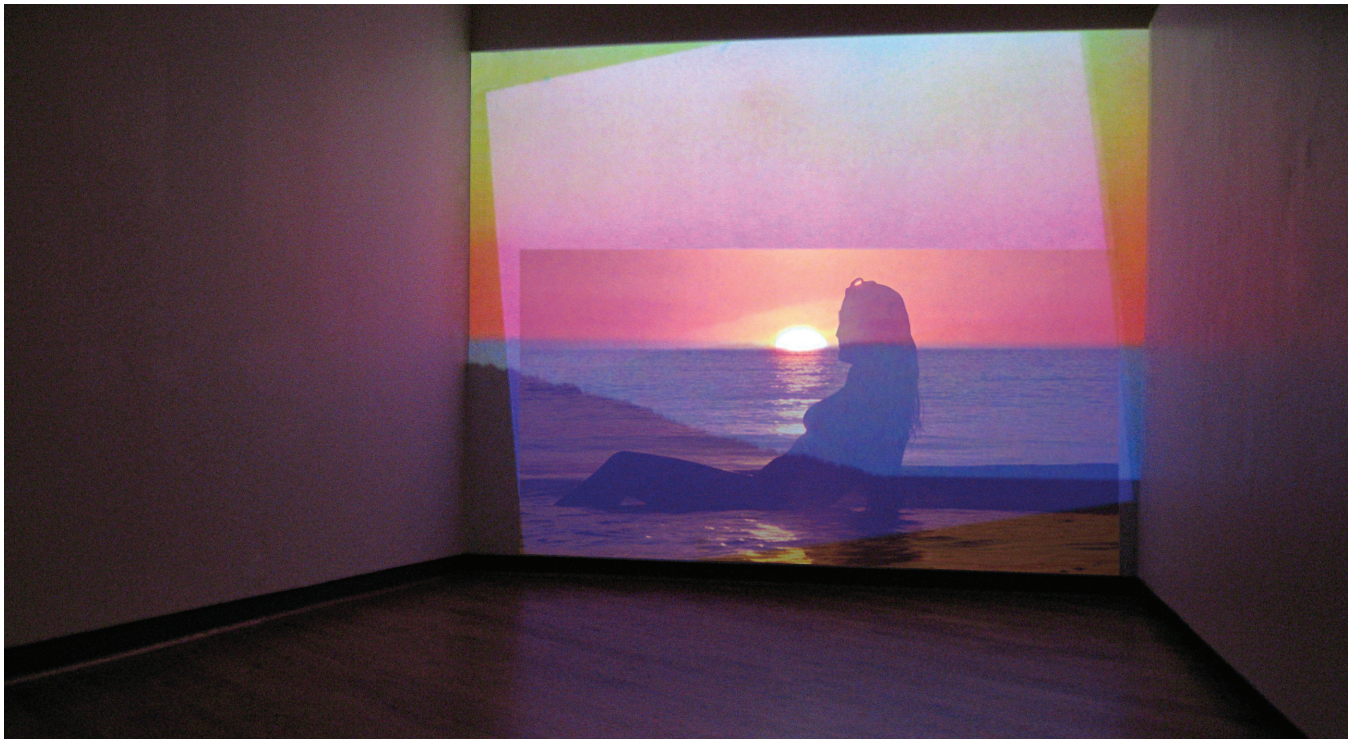


Figure 1. *Eventide* presented in an installation format.

Abstract

Eventide is a five-minute video in which amateur shapshots of sunsets have been meticulously aligned and sequenced into a coherent video of the sun setting slowly over the horizon.

CR Categories and Subject Descriptors: J.6 [Computer Applications]: Arts and Humanities – Fine Arts.

Additional Keywords: information visualization, sunset photography, collections.

Statement

Eventide is a "Snap-Motion Re-Animation" of the sunset, an icon in snapshot photography. It is a collection of 1,391 photographs that are placed in succession to reveal a story about innate aesthetics and one grand universal tie that binds us.

"Snap Motion Re-Animation" is an animation technique I have been developing over the last three years. It emerged from my unshakable belief that there exists an over-abundance of everyday photographs, taken of everyday things, in every possible position. Thus, by collecting enough of them I can place them in an order to re-invent or re-animate life.

The photographs that are included in *Eventide* came from around the world and are taken by different photographers, mostly amateur. I collect them from friends, family, colleagues, acquaintances, strangers, image banks, photo exchanges, thrift stores, libraries, private collections, want adds, eBay and the public domain archives of the US Army, NOAA and NASA.

Biography

Cassandra C. Jones received her MFA in 2004 from Carnegie Mellon University. Presently she works with photographic and video materials in Brooklyn, New York and Ojai, California, where she is represented by Nathan Larramendy Gallery and Vanina Holasek Gallery, respectively.

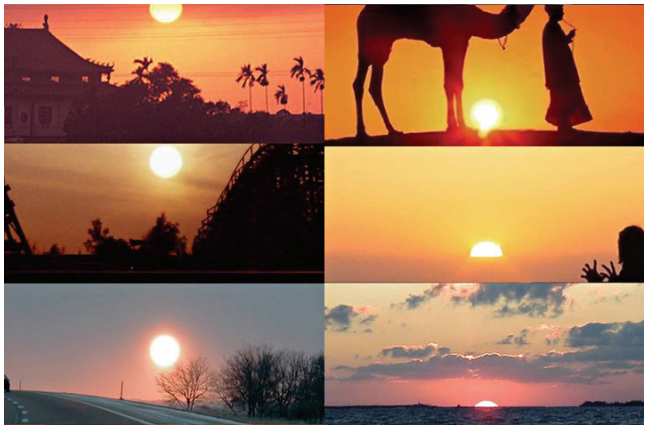


Figure 2. Stills from *Eventide*.

The Sheep Market

Aaron Koblin

Department of Design | Media Arts, UCLA

Figure 1. A selection of drawings from *The Sheep Market*.

ABSTRACT

The Sheep Market is a web-based artwork which appropriated Amazon's Mechanical Turk (MTurk) system to implicate thousands of online workers in the creation of a massive database of drawings. From one simple request, submitted to the MTurk system as a "HIT" or Human Intelligence Task, workers created their version of "a sheep facing to the left" using simple drawing tools. The artist responsible for each drawing received a payment of two cents for his or her labor.

The first 10,000 sheep collected through the system were placed online to be resold as collectible plate blocks of stamps. In addition to images of the sheep, the drawing process was recorded as an animation which can be reviewed to observe the creation process. The website `www.thesheepmarket.com` is an online marketplace and exhibition for browsing, sharing, and purchasing the sheep.

The inspiration for *The Sheep Market* project stems from the urge to cast a light on the human role of creativity expressed by workers in the system, while explicitly calling attention to the massive and insignificant role each plays as part of a whole.

CR Categories and Subject Descriptors: J.6 [Computer Applications]: Arts and Humanities – Fine Arts.

Additional Keywords: information visualization, sheep, drawings, small multiples, human-assisted information processing and automation.

BIOGRAPHY

Aaron Koblin (akoblin@ucla.edu) received his MFA from the Department of Design|Media Arts at UCLA and his BA in Electronic Art at the University of California, Santa Cruz. Utilizing a background in the computer game industry, he led a course in game design for the web at UCLA and has been working with data driven projects as a designer, artist and researcher.

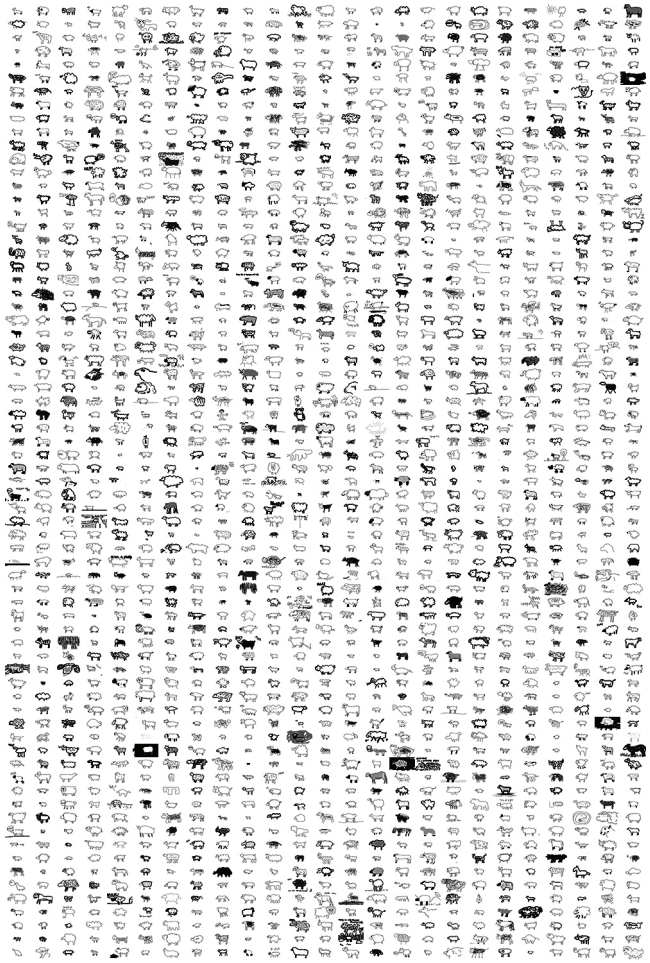


Figure 2. Additional drawings from *The Sheep Market*.

Skymall Liberation

Evan Roth

Eyebeam Atelier / ni9e.com

Figure 1. “Male versus female”, an arrangement of facial image fragments from *Skymall Liberation*.

Abstract

Skymall Liberation is a series of in-flight studies into the visual anthropology of travel magazines. Fragments from the *Skymall Catalogue* were extracted and re-arranged, without the aid of a computer or scissors, in order to explore the magazine’s target demographics.

CR Categories and Subject Descriptors: J.6 [Computer Applications]: Arts and Humanities – Fine Arts.

Additional Keywords: information visualization, travel magazines, demographics.

Statement

Here is a fun way to waste a couple of hours on your next flight. Rip out all of the faces from the *Skymall Catalog*. You will see from all of their smiling faces that they are pleased with their liberation. The images can then be used to create data visualizations of *Skymall* demographics. Return the vandalized catalog to its home in the seat back in front of you for the next passenger to stumble upon.

Biography

Evan Roth (evan@ni9e.com) is an artist and Research Fellow at the Eyebeam OpenLab, New York City, where he conducts research and development on creative technologies and open-source media projects which directly enrich the public domain.

IEEE Visualization Conference,
IEEE Information Visualization Conference and
IEEE Symposium on Visual Analytics Science and Technology

October 28 - November 1, 2007
Sacramento, California, USA

InfoVis Doctoral Colloquium

Personal information management through interactive visualizations

Florian Evequoz and Denis Lalanne

Abstract—The PhD thesis presented in this paper tackles the personal information overload phenomenon. Our purpose is to offer a set of interactive visualization techniques allowing one's personal digital memory to be organized and overviewed. This system will provide easy browsing, guiding towards the wanted piece of information and allow a free exploration of the personal information space. Three main axis of research are involved: (1) cross-media mining to generate indexes and alignments, (2) data clustering to organize personal information and (3) information visualization techniques, that should provide means to easily navigate through the personal information space. In this paper, we present our approach to these challenges using the personal email archive as an entry point into the personal information space. Indeed, we believe that emails are a particularly rich source of metadata for indexing personal information, as well as a representative subset of the whole personal information space. We finally present the work done during the first year of this PhD thesis and the roadmap for the future.

Index Terms—Personal information management (PIM), email, visualization.

1 Introduction

With all types of media becoming digital and storage devices regularly increasing in capacity, we tend to accumulate a growing amount of information that becomes "personal" as soon as we decide to keep it. However, this personal information (PI) grows very fast, challenging our natural wish for order. Thus, e-mails, pictures, videos, music, personal documents and every other pieces of information creating our individual digital memory are often stored anarchically and become hard to retrieve or correlate. The very hierarchy of file storage system is the main reason for this failure. Indeed, as Bush pointed out in [2], our mind works by association rather than by following the rules of a static hierarchy. However with current systems, no explicit links connect heterogeneous pieces of information about the same topic, related to the same people or having another characteristic in common. This implies that we often have to perform repetitive searching tasks using several different applications in order to gather the obviously correlated information we look for. All in all, the inherent or instinctive structure of our personal information is hidden, and the current searching mechanisms do not put it into evidence. It remains obscure also because we cannot get an overall view of it.

The purpose of the research work presented in this paper is to investigate means of visualizing the personal information space interactively. The accent will be put on browsing capabilities, taking advantage of the structure of the personal information and the links existing between different pieces of information. The personal email archive will be used as the main source for generating a personal information structure. Other personal information (texts, pictures, music) will then be aligned with this structure and integrated together into an interactive visualization tool.

In the following sections, we present a brief state-of-the-art of personal information management (PIM), with a particular interest for existing email management and visualization systems. Section 3 is devoted to a more concrete presentation of the work envisioned for our particular research project. Sections 4 and 5 present the work achieved during the first year of the project and the future plans.

2 State of the art

PIM research has been receiving a growing interest in the recent years, leading to the developement of several tools and methodologies, described in particular in [10]. As a complete state of the art is not the point of our discussion, we simply want to recall that previous research mainly focused on the data management perspective of PIM, trying to apply semantic modelling to PI. Some other works focused only on specific parts of PI, like emails or agendas, or offer a port of web search engines to the desktop. Our works differs from the cited ones because it focuses on finding similarities between different pieces of data and taking advantage of the links inferred from them to browse the whole PI with the help of information visualization techniques.

- *Florian Evequoz is a first-year PhD student at University of Fribourg, Switzerland. Email : florian.evequoz@unifr.ch.*
- *Denis Lalanne is a senior research assistant at University of Fribourg, Switzerland. Email : denis.lalanne@unifr.ch.*

3 Project overview

The main goal of this PhD thesis is to investigate how interactive information visualization techniques combined with cross-media mining can help face the challenges of PIM. More specifically, the purpose of the project is to provide solutions and techniques to :

Create a PI network Cross-media information mining techniques shall be used to generate indexing metadata. On top of this metadata, thematic, temporal or social-network based links shall be created, connecting information of homogeneous or heterogeneous types (documents, pictures, music ...), and thus building a PI network.

Organize PI The system shall provide means to organize PI in flexible ways, combining automatic clustering techniques based on indexes along with user-assisted clustering. This method will give the user a feeling that he controls to some extent the archiving and organization of his PI.

Navigate through PI The use of visualization suits well for navigating into a PI space. Therefore, synchronized views of personal data should offer different levels of details, provide details on demand and filtering mechanisms, and introduce browsing information as an alternative or a complement to more traditional search engines.

Support PIM Thanks to the novel access to personal information it provides, the system shall help the user elaborate new strategies to manage its PI and avoid being overloaded.

To reach these goals, we follow a user-centered approach, gathering user requirements at the beginning of the project and conducting user-satisfaction evaluation once a working system is available.

In order to generate metadata on PI, we use the personal email archive as main source of metadata. Email is indeed a rich subset of PI [9]. A single email inherently connects together people, topics and time. Therefore, a whole personal email archive contains invaluable thematic, temporal and social metadata that would be hard to obtain

with other types of PI: people knowing each other usually appear together as recipents of a message, some topics are related to particular groups of contacts, topics and relations are closely related to time periods, etc. Our purpose is to gather this metadata and retrieve clusters pertaining to the social, thematic and temporal dimensions. In the next step of analysis, the remaining PI will be aligned with the dimensional structures extracted from emails, following the approach successfully used in [7] to align multimedia data with textual documents. Once the metadata is available, relevant visualization techniques will be applied in order to allow browsing the whole PI. Our system will then be an email-centric personal information manager. More specifically, the following steps need to be performed:

1. Features or metadata extraction from emails
2. Clustering according to social, thematic and temporal dimensions, based on similarity computations
3. Alignment of the structure extracted from emails with the remaining PI
4. Visualization of the PI, based on the structure and taking advantage of similarity links

As we will not use data-driven classification of PI which requires a training set of already classified data, but statistical analysis, the role of the personal information visualization will be particularly emphasized. We plan to present to the user several visualization techniques simultaneously, in order to enable visual query refinement using known interaction methods such as on-demand filtering, link & brush, etc. Each of the visualizations proposed will focus on one of the aforementioned dimensions, or a combination of two of them (e.g. variation of themes over time, using a technique similar to ThemeRiver [5]). We believe that the combination and synchronization of several visualization techniques applied to different dimensions will help the user browse instinctively through his PI. Therefore, a goal of this thesis will be to confirm or invalidate our hypothesis that a good use of visualization can be efficient for handling PIM, without using any semantic modelling.

As an extension, we also plan to provide access to professionnal data through the PI structure, thus offering an ego-centric view of professional data, for instance meeting recordings [8].

4 Achieved work

The first research efforts were centered on emails. In a first phase, we collected personal data. It consists of (a) a personal email archive containing around 6000 emails and 3500 addresses, (b) the Enron public email archive [4] and (c) the AMI meeting corpus [3], that contains textual, audio, video and email data. A user requirements questionnaire was set up, focusing on the relationship between personal and professional information and the preferred way of accessing them. We then developed a tool for extracting email data from IMAP servers and local archives into a database and perform statistical analysis on textual content and addresses to gather relevant features. Moreover, using the similarity based on the co-occurrences of words in the subjects and contents of emails, a hierarchical clustering was performed, which aims at finding a thematic organisation of emails. As well, a social network was built based on similarity measures between email addresses. For the information visualization part, simple views of the email archive were implemented with the help of the prefuse toolkit [6]. The result of thematical clustering was fed into a treemap-visualization, while the result of the social analysis is visualized as a social network graph (see Fig. 1). Even though the two visualizations still need refinement and have not been synchronized yet, limiting the possibilities of visual querying, they show interesting trends that would not have been highlighted by traditional mail clients.

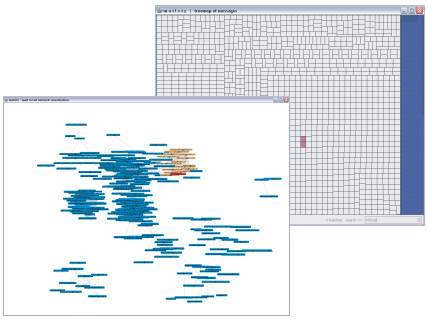


Fig. 1. The implemented views of a personal email archive. The left window presents the social-network graph, where nodes map to email addresses. A subgroup of connected addresses is highlighted. The right window shows a treemap of emails, clustered by textual similarities. Results of a search with the keyword 'infovis' are highlighted in purple.

5 Future works and applications

In the short-term future, we will investigate further data mining and clustering methods on email data, in particular including the temporal dimension. However, the main focus will be on the visualization aspects, following the directions presented in section 3, namely introducing synchronized views of different dimensions, overview capability, details on demand and filtering mechanisms. The next step of the project in the long-term will be the alignement of the email structure with the remaining personal information. Finally, a user-satisfaction evaluation will validate our approach. The deliverables of this PhD thesis will serve two different research projects. In the first place, in the scope of Hasler Stiftung's Memodules, this thesis will provide an automatically extracted PI structure [1] on top of which the user will be able to connect tangible shortcuts to his personal information. In the second place, a goal of NCCR IM2.HMI project is to develop methods for accessing recorded meetings data. In this context, our thesis will facilitate personalized browsing of huge amounts of recorded data, thanks to similarity links that can be drawn between personal and professional information, thus opening the door to ego-centric professional data browsing [8].

References

[1] Memodules homepage: `http://www.memodules.ch/`.

[2] V. Bush. As we may think. *The Atlantic Monthly*, 176(1):101–108, 1945.

[3] J. Carletta et al. The AMI meeting corpus: A pre-announcement. In S. Renals and S. Bengio, editors, *MLMI'05: Proceedings of the Workshop on Machine Learning for Multimodal Interaction*, number 3869 in LNCS, pages 28–39. Springer-Verlag, 2005.

[4] W. Cohen. Enron email dataset. Retrieved May 5, 2005, from `http://www.cs.cmu.edu/~enron/`, 2005.

[5] S. Havre et al. Themeriver: Visualizing theme changes over time. In *INFOVIS '00: Proceedings of the IEEE Symposium on Information Vizualization 2000*, page 115, Washington, DC, USA, 2000. IEEE Computer Society.

[6] J. Heer et al. prefuse: a toolkit for interactive information visualization. In *CHI '05: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 421–430, New York, NY, USA, 2005. ACM Press.

[7] D. Lalanne et al. Using static documents as structured and thematic interfaces to multimedia meeting archives. In *MLMI*, pages 87–100, 2004.

[8] D. Lalanne, M. Rigamonti, F. Evequoz et al. An ego-centric and tangible approach to meeting indexing and browsing. In *Machine Learning for Multimodal Interaction (MLMI' 07)*, 2007. Accepted for publication.

[9] S. Whittaker, V. Bellotti, and J. Gwizdka. Email in personal information management. *Commun. ACM*, 49(1):68–73, 2006.

[10] H. Xiao and I. F. Cruz. A multi-ontology approach for personal information management. In *ISWC 2005: Proceedings of the Semantic Desktop Workshop*, 2005.

Augmenting Digital Library Search Interfaces with Visual Analysis Tools

Edward Clarkson and James D. Foley

Georgia Institute of Technology

ABSTRACT

Digital libraries commonly elide hierarchical metadata that might be used more effectively. This proposal presents the ResultMap concept, a tool that leverages that metadata for digital library search facilities; an initial study of its effectiveness; the concept of applying ResultMaps to faceted metadata, allowing visual detection of implicit correlations between facets; and proposals for further study of ResultMaps in both directed search and faceted browsing environments.

CR Categories and Subject Descriptors: H.5.4 [Hypertext/ Hypermedia]: Navigation, User issues.

Additional Keywords: user studies, exploratory search.

1 INTRODUCTION

Over the past 3 years, we have built two digital libraries for educational materials: one for Visual Analytics[1] and one for Human-Centered Computing[2] (HCC) [1]. A common theme among digital repositories, including ours, is the use of either pre-structured or generated hierarchy to organize content. Hierarchies like ours may be stand-alone or part of a *faceted metadata* scheme. Faceted metadata is a way of classifying an object along multiple independent dimensions. For example, a research paper might have facets for its topic, its publication venue or its length.

We are interested in how digital repositories can use visualization to leverage these existing structures, and studying how such techniques affect users. We focus in particular on the areas of traditional keyword search and faceted browsing interfaces. We have augmented a standard search engine with what we call *ResultMaps* [2] (RMs), and we further propose to study their applicability to faceted environments.

2 MOTIVATION AND RESEARCH QUESTIONS

There has been a variety of work in the area of visualization for digital libraries (DLs) and generalized search visualization. The standard paradigm for keyword search result listings (a list of relevance-ordered document hits) does little to contextualize the results within the overall information space, and existing hierarchical classifications are ideal for this purpose. Consequently, search environments do not fully leverage the utility of hierarchy's inherent metadata.

Several factors make this proposal distinct from previous work (too extensive to mention in this format): we are interested specifically in the digital library usage environment; we deploy our tools on *live* repositories; and we propose questions which have not previously been studied in such contexts.

2.1 Directed Search

ResultMaps are based on the popular treemap layout technique [3]. They take a digital repository that uses some form of hierarchical classification and uses that structure to map each document into a treemap, and search engines can highlight query result items. Representing the entire document space provides context for individual search results, and facilitates tasks like detecting outliers and clusters within search results by them making visual instead of textual processes. Moreover, this context is consistent between successive search queries.

Both the Visual Analytics Digital Library (VADL) and HCC Education Digital Library (HCC EDL) have implemented ResultMaps as a part of their search engine facilities. Space constraints prevent us from including an illustration; we encourage the reader to visit either the VADL or HCC EDL for an example. We have used these environments to address a number of larger research questions regarding directed search tasks:

1) How do RMs affect user performance on DL search tasks?
2) How do RMs affect subjective impressions of DL interfaces?
3) Do RMs yield a greater level of knowledge about the overall content in digital library?
4) How do RMs affect query string characteristics over sequences of queries?

2.2 Faceted Environments

Faceted browsing environments do an admirable job of supporting painless transition between directed and exploratory search activity. But one problem with these environments is that it can be difficult to diagnose the effects of an action after the fact—that is, how a selection in one facet affects the distribution of remaining items across the other facets. The only information present in current tools (such as Flamenco [5] among several) that might assist such diagnosis is the raw item counts remaining in each facet category. Indeed, designers of another faceted environment suggest "…representing temporary hierarchies to the user is important for…understanding the effects of facets on each other" [4]. The relationship between two facets can be complex, but can be quantified by the correlation between their respective metadata values. For example, a dataset may contain subtle relations like the connection between medium and country of origin in a visual artworks dataset. But the frequency and magnitude of inter-facet correlations is unknown, and forms an initial exploratory research question:

5) What is the extent of facet correlation in real-world and standardized sample datasets?

Another salient observation of our ResultMap environment is that it is essentially a degenerate case of a generalized faceted system in which a library has a single hierarchical facet. As a result, we can apply ResultMaps to a general faceted environment using one ResultMap per facet—what we call *faceted ResultMaps*. Faceted ResultMaps allow users to use visual analysis to intuitively discern correlations—users may notice that a selection in one facet non-randomly culls items in another facet. But faceted ResultMaps are merely one approach to representing dataset correlations: a simpler alternative is to link visually facets

801 Atlantic Drive, Atlanta, GA 30332-0280
{edcclark, foley}@cc.gatech.edu

[1] http:/vadl.cc.gatech.edu
[2] http://hcc.cc.gatech.edu

with strong correlations. Accordingly, our research questions concern the objective and subjective effectiveness of faceted ResultMaps in digital library environments:

6) How do faceted ResultMaps and other correlative displays affect performance on exploratory search tasks?
7) How do faceted ResultMaps and other correlative displays affect subjective attitudes of faceted DL interfaces?
8) How do faceted ResultMaps and other correlative displays affect query sequences and/or facet selection paths?

3 Methodology

We address our research questions through a series of controlled and field studies. We have addressed questions 1-3 in a recent study A1; study A2 approaches 1-3 with less experimental control but more ecological validity; questions 4 and 5-8 are addressed in studies B-D. We summarize the design of each study below.

3.1 Study A1 (Questions 1-3)

The design is between subjects with two levels: ResultMap vs. non-ResultMap (non-RM). The dependent measures include task completion time, accuracy, and post-tests of users' knowledge of and subjective attitudes about the DL environment.

Subjects are presented with a series of 12 search result listings (using 10 results per page) from the HCC EDL, randomly ordered for each subject. For each result list, subjects are asked to select a document satisfying a set of constraints. The RM condition is the search engine interface in use with the HCC EDL; the non-RM interface simply removes the RM interface piece, making it similar to a standard search engine (e.g., Google). Query strings for the tasks are the same for all subjects to control for search expertise. After subjects complete all tasks, they are asked to complete a set of questions testing their knowledge about characteristics of the repository as a whole and about their subjective impressions of the library.

3.2 Study A2 (Questions 1-3)

ResultMaps are in use with each of our digital libraries as mentioned above. That installation has yielded a large (ever-expanding) corpus of log data for analysis of *in-situ* user behavior. We are implementing a semi-controlled longitudinal field replication of study A1 by randomly presenting the ResultMap or the traditional search interface to users.

3.3 Study B (Question 4)

The design is within subjects with two levels of interface type: ResultMap (RM) vs. non-ResultMap (non-RM). The dependent measures include task time and accuracy, number of query strings, average query length, query string change between successive queries, and subjective user ratings.

Subjects are asked to use the library search engine to find two series of documents meeting a series of related constraints. The series are randomly assigned to condition for each subject, and conditions are counterbalanced between subjects. Subjects are pre-tested on their query string generation skill.

3.4 Study C (Question 5)

This is a non-experimental study aimed at gathering data about correlations in real-world datasets. Assuming there are effective techniques for showing correlation data, the prevalence of correlated facets provides a baseline for how often those techniques might be employed. Examples of real-world faceted datasets to examine include eBay auctions, the Flamenco dataset examples, and faceted versions of our own digital libraries.

3.5 Study D (Question 6-8)

The design is within-subjects with three levels of interface type: RM, linked, and baseline. The baseline control condition is an unaugmented faceted environment; the linked condition is a faceted environment that draws connections between facets that are significantly correlated; the RM condition is a faceted environment augmented with ResultMaps. We use a larger, non-visualization dataset than our DLs, allowing us to control for visualization expertise and to test on multiple corpora sizes.

The dependent measures include task time, accuracy, number of facet selections and queries, query string characteristics (length, etc.), and subjective ratings. Subjects are given both unstructured and structured tasks requiring them to interleave both faceted browsing and directed search behaviors.

4 Results

We have completed Study A1 (N=20). A preliminary analysis of the data yields mixed findings:

- There were no significant differences in performance or accuracy between groups (like many other studies of visualization tools).
- RM users scored significantly higher on some parts of the repository knowledge post-test that the control group.
- RM users rated the subjective impact of the interface on task difficulty significantly better than the control group.

More detailed analysis is underway to explore any effects of result list characteristics (e.g., clustering or outliers) on performance.

5 Future Work and Conclusions

Our early results show that ResultMaps have some positive benefits for search users, though they show no statistical benefits in many cases. Unfortunately, this is typical of many studies of visualization systems. But ResultMaps do make use of hierarchical metadata typically ignored by search interfaces, which at least provides the opportunity for users to gain unsought knowledge—a situation difficult to experimentally verify or quantify—of either ancillary topics or the digital library information space as a whole.

Faceted systems also do not typically reveal the full extent of the available hierarchical metadata; that fact also makes it difficult to identify patterns or linkages between facets. As a result, the application of ResultMaps to faceted environments is an intriguing possibility for further study and refinement. Here, we describe one such path of research in this problem space.

References

[1] Clarkson, E., Day, J. and Foley, J. An Educational Digital Library for Human-Centered Computing. In *CHI '06 Extended Abstracts*, pp. 646-651.

[2] Clarkson, E. and Foley, J. ResultMaps: Search Result Visualization for Hierarchical Information Spaces. In *Proceedings of the CHI '07 Workshop on Exploratory Search and HCI*, pp. 32 - 39.

[3] Johnson, B. and Shneiderman, B. Tree-maps: A space filling approach to the visualization of hierarchical information structures. In *Proc. of IEEE Visualization '91*, pp. 284-291.

[4] Wilson, M., schraefel, m. c. and White, R. Evaluating Advanced Search Interfaces using Established Information-Seeking Models. Technical Report (2007), School of Electronics and Computer Science, University of Southampton.

[5] Yee, P., Swearingen, K., Li, K. and Hearst, M. Faceted Metadata for Image Search and Browsing. In *Proc. of CHI '03*, pp. 401-408.

Understanding Information Visualization Within the Context of Visual Representation

Caroline Ziemkiewicz*

University of North Carolina at Charlotte

ABSTRACT

Research on the theoretical foundation of information visualization has so far focused mainly on classifying methods by properties of the data or properties of the image. Although this has resulted in many useful taxonomies of visualization methods, we aim to investigate a more generalizable mapping between information and image that is independent of the data. By defining visualization methods independently of specific datasets or surface appearance, we hope to embed the theory of information visualization within the broader space of visual representation of information. We present the early stages of this work, which attempts to define visual mapping properties by drawing from aesthetic theory and properties of mathematical functions. The long-term goal of this research is to fill a crucial gap in the growing theoretical study of information visualization and to work towards more mature theories of visual understanding.

Keywords: Visualization, theory, taxonomy, classification, visual communication, visual mapping.

Index Terms: H.5.1 [Information Interfaces and Presentation]: Multimedia Information Systems—Evaluation/methodology; H.1.1 [Models and Principles]: Systems and Information Theory—Information theory

1 PROBLEM STATEMENT

Research in information visualization (InfoVis) has produced a wide array of methods and applications, but far less effort has been focused on understanding these methods in relation to each other and to the goals of the field as a whole. This kind of theoretical understanding would mature the field by focusing new methods, improving evaluation, and supporting better design decisions. What is needed is a deeper knowledge of the place of information visualization within related forms of visual representation, both to clarify its own distinct goals and requirements and to borrow knowledge from these other forms. In addition, tools for classifying methods within information visualization could help to identify new research paths and place existing work in context. Without these resources, research faces the danger of becoming unfocused and redundant.

There is a growing body of theory that attempts to solve these problems. Representative examples include Bertin's seminal work in analyzing information graphics based on the elements and layout of the image [1]; Shneiderman's task-by-data-type taxonomy of visualization methods [4]; and Tory and Möller's taxonomy based on design models of the data [5]. While these all provide important perspectives, there are some limitations to both image-based and data-based taxonomies. The former viewpoint can obscure the importance of information by focusing on the surface of the visualization, and while data-based theory and the higher-level design model theory of Tory and Möller counteract this by focusing on information, they also lose something by isolating data visualization from other forms of visual representation.

A theory that addresses properties of the mapping between information and image would provide a complementary perspective to existing work. By establishing an additional viewpoint independent of both data properties and specific visual implementations, such a theory can help to view visualization methods from the more universal standpoint of how people understand information visually. This can begin to embed information visualization within the larger context of visual arts, information design, human-computer interaction, illustration, and visual languages. These related endeavors may have important contributions to make to the understanding of visualization, and the long-term goal of my thesis research is to extend the theoretical foundation of information visualization to assist in discovering these contributions.

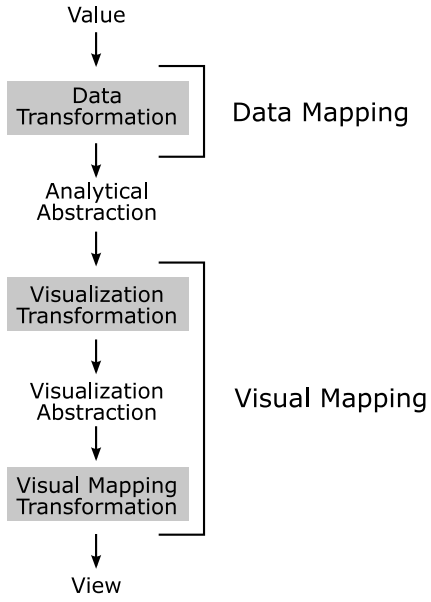


Figure 1: Our model of the visual mapping process, which is a simplified version of the data state reference model [2].

2 INFOVIS WITHIN VISUAL REPRESENTATION

The first stage in this work is to understand information visualization in terms of the mapping between information and image. Using a simplified version of Chi's data state reference model of visualization algorithms [2] as a definition of the visual mapping from data to image, Dr. Robert Kosara and I have developed a definition of information visualization in terms of four attributes of the visual mapping, distinguishing it from other visual representations (Figure 1).

Data-based. It must be derived from external data.

*e-mail: caziemki@uncc.edu

Bijective. It must represent a bijective mapping from information abstraction to image. In other words, each element generated in the data processing step must map to a unique visual element in the image. This is analogous to bijective functions in mathematics, where every element in the domain maps uniquely to a single element in the codomain.

Interactive. It must provide for interactivity. Since we take the goal of information visualization to be the exploration and analysis of information, rather than its presentation, it is necessary for the user to be able to alter the mapping dynamically so that it can provide more than a single fixed view of the data.

Syntactically notational. It must be a syntactically notational symbol system, so that information is mapped to a set of discrete and disjoint symbols.

This last requirement is derived from the work of Goodman [3], who sought to analyze the difference between pictorial images, which are understood as holistic visual representations, and notational images, which are read as visual languages. We believe that Goodman's analysis, although taken from the perspective of aesthetics, has significant implications for visualization.

3 Internal Structure of InfoVis

In addition to the definition of the boundaries of information visualization, we have extended our model to an internal taxonomy of visualization methods. This taxonomy is composed of three axes that deal with properties of the visual mapping.

Linearity. Whether the mapping is linear or nonlinear. Again working with an analogy from mathematical functions, we consider linear mappings as those in which constant changes throughout the dataset result in constant changes throughout the image. Nonlinear visual mappings are those in which the appearance of an individual element is dependent on overall properties of the dataset.

Information loss. Whether the mapping makes use of intentional information loss, as in aggregative or filter-based visualizations.

Continuity. As in Tory and Möller, whether the mapping treats the data as continuous or discrete. To formalize this, we define a discrete mapping as one which is semantically notational according to Goodman; i.e., a mapping in which the meanings of symbols are unambiguous and non-overlapping.

We have used these three axes to form an initial taxonomy of existing information visualization methods based on aspects of the visual mapping. While this initial stage of our work is completed [6], the refinement and further application of our model is ongoing.

4 Conclusion and Future Directions

This work represents a first step towards a unified model of visualization based on mapping properties. Currently, we wish to explore the model's application to more specific portions of the information visualization field, such as categorical data visualization. By narrowing the area of study, we hope to explore the applicability of our model and taxonomy in more detail.

As the visual mapping model becomes more robust with continuing research, I plan to apply it not only to visualization methods but to concepts from related forms of visual representation, as in the case of Goodman's aesthetic theory. This line of research is intended to find parallels among these varying disciplines, many of which have longer histories of theoretical study than information visualization. By using the visual mapping model as an aid in translating between, e.g., ideas from aesthetics, design, and visualization, I hope to find new ways of adding these more mature theories to the greater body of understanding of information visualization.

The goal for this model is eventually to provide a more thorough and consistent taxonomy of mappings, and to begin to identify the primitive elements of visual mapping as Bertin and others have identified the primitive elements of the image in information graphics. In time, I hope for this model to provide a complementary perspective to existing models based on data types, tasks, and image elements, making for a richer choice of perspectives from which to understand and evaluate information visualization. A variety of tools for understanding the field can lead to novel research paths, more meaningful methods of evaluation, and the discovery of connections among seemingly unrelated work.

References

[1] J. Bertin. *Semiology of Graphics*. Univ. of Wisconsin Press, 1983.

[2] E. H. Chi. A taxonomy of visualization techniques using the data state reference model. In *Proceedings Information Visualization*, pages 69–76. IEEE CS Press, 2000.

[3] N. Goodman. *Languages of Art*. Hackett Publishing Company, 1976.

[4] B. Shneiderman. The eyes have it: A task by data type taxonomy for information visualization. In *Proceedings IEEE Symposium on Visual Languages*, pages 336–343. IEEE CS Press, 1996.

[5] M. Tory and T. Möller. Rethinking visualization: A high-level taxonomy. In *Proceedings Information Visualization*, pages 151–158. IEEE CS Press, 2004.

[6] C. Ziemkiewicz and R. Kosara. Understanding information visualization in the context of visual communication. Technical Report CVC-UNCC-07-08, 2007.

Author Index