AN ABSTRACT OF THE THESIS OF

Jeremy Gragg for the degree of Honors Baccalaureate of Science in Business Administration presented on August 22, 2008. Title: Exploration and Analysis of Information Visualization Techniques Applied to the TeachEngineering Digital Library.

Abstract approved:

_____

Byron Marshall

The rapid development of computing technologies in recent decades has allowed for data to be generated at unprecedented rates. As a result, users are challenged with the task of finding the precise information that they are looking for. Visualization techniques provide the ability to view and analyze data in different ways, helping users to find what they are looking for and gain additional insights. TeachEngineering is a digital library system where K-12 educators can search for high-quality lesson plans that meet national and state-specific educational standards. Users have several different types of criteria to search by, including: grade level, cost, time required, group size, keywords, source state, and educational standards. Being able to see how individual documents compare to specific variables can help identify additional results that are worthy of user consideration. Surveying visualization techniques, specific methods (scatterplots, parallel coordinates, star glyphs and hyperbolic trees) are qualitatively analyzed with regards to how they could be applied to TeachEngineering search query results. Each visualization type has advantages and limitations, and no single technique can be considered the definitive solution. Recommendations on conducting future experiments and research are included.

Key Words: Information Visualization, Digital Library, TeachEngineering, Search Relevance, Interface Design

Corresponding e-mail address: jeremy.gragg@gmail.com

Exploration and Analysis of Information Visualization Techniques

Applied to the TeachEngineering Digital Library

by

Jeremy Gragg

A PROJECT

submitted to

Oregon State University

University Honors College

in partial fulfillment of
the requirements for the
degree of

Honors Baccalaureate of Science in Business Administration (Honors Scholar)

Presented August 22, 2008
Commencement September 2008

Honors Baccalaureate of Science in Business Administration project of Jeremy Gragg presented on August 22, 2008.

APPROVED:

_____

Mentor, representing Business Administration

_____

Committee Member, representing Business Administration

_____

Committee Member

_____

MIS Option Coordinator, College of Business

_____

Dean, University Honors College

I understand that my project will become part of the permanent collection of Oregon State University, University Honors College. My signature below authorizes release of my project to any reader upon request.

_____

Jeremy Gragg, Author

ACKNOWLEDGMENT

This project has been a long and sometimes painful process, and I have several people to thank for its successful completion. Foremost, thanks to Professor Byron Marshall, who served as my mentor on this project and helped keep me focused on the task at hand; every time we had a discussion, I came out of it with optimism and motivation. Thanks to Professor Rene Reitsma for helping me identify my thesis topic, suggesting literature to start with, and sitting on my committee. Thanks to Richard Van Winkle for taking the time and effort to sit as the third member of my committee, particularly given the long distance that you had to travel. Thanks to all three of you for working with me on an accelerated schedule, providing feedback, and challenging me to create a quality product that I can be proud of.

Thanks to the Honors College for their guidance during my five years at Oregon State University. I am truly glad to have been a part of the program and to not have quit, although it sometimes seemed much easier to do so. Nearly all of my favorite professors and courses were part of my Honors College experience.

Lastly, thanks to my mother, Jane, for her unwavering support and patience throughout the years. If there is anyone I owe my success to, on this project or otherwise, it is you.

TABLE OF CONTENTS

Page

TABLE OF CONTENTS (Continued)

Page

LIST OF FIGURES

LIST OF FIGURES (Continued)

LIST OF FIGURES (Continued)

Exploration and Analysis of Information Visualization Techniques Applied to the TeachEngineering Digital Library

INTRODUCTION

The rapid development of computing technologies in recent decades has allowed for data to be generated at unprecedented rates. Indeed, attempting to comb through the vast volumes of information has become an increasingly difficult and unmanageable task. As a result, users are faced with the challenge of finding the precise information they are looking for. To assist users in their efforts, information visualization techniques can aid in the effective presentation of data and may help identify additional information that is likely to be of interest.

Information Visualization is by no means a new field, with implementations far predating modern technology. Scottish engineer and economist William Playfair pioneered the field of information visualization when he developed the line graph and bar chart in 1786 and pie chart and circle graph in 1801 (Friendly, 2008, p. 13-14). Other early examples of information in visual form can be found in cartography; perhaps the most famous example would be the graphic developed by French engineer Charles Joseph Minard, which depicted Napoleon's disastrous Russian campaign. In it, Minard combines "statistical information (troop numbers and temperature variations) and topographical data (direction, distance and location)" (Mijksenaar, 1997, p. 30). Another more recent example includes Henry Beck's 1933 design for the diagrammatic map of the London Underground system (Friendly, 2008, p. 29); although not topographically

accurate, it is a highly effective map that has served as a model for subway systems across the globe (Mijksenaar, 1997, p. 5).

It is clear that the visual display of information can benefit the understanding of data. “Yet despite what many people think, graphical representations are less suitable for inexperienced readers. The fact is that they require a degree of intellectual training. Pictorial representations are not necessarily ‘easier’; rather, they are more concise, more compact, clearer and, when well-done, more compelling” (Mijksenaar, 1997, p. 30). Tufte uses the term *graphical excellence* to describe the well-designed presentation of interesting data through use of substance, statistics and design; further, *graphical excellence* consists of complex ideas, nearly always multivariate in nature, communicated with clarity, precision, and efficiency (Tufte, 1983, p. 51).

The information age of the last fifty years, and the rapid generation of data associated with it, has created the necessity of data mining. In attempting to analyze complex datasets, one of the significant problems involves finding useful ways to simultaneously visualize multiple dimensions. When dealing with only two or three variables, displaying information is a fairly straightforward task; more than that, and things can get messy. Much research has been dedicated to determining feasible and context-specific solutions to the question, *“How do we visualize our datasets?”*. This paper is no different, and in it I attempt to provide insight to the useful development of graphic visualization tools for the TeachEngineering digital library repository.

This paper will be presented as follows: Section II will introduce the TeachEngineering project, identify the problem, and recognize preliminary considerations. Section III will survey numerous visualization techniques and tools that

have been developed to aid end-user visualization. Section IV will consist of an exploration towards the development of information visualization techniques for TeachEngineering; methods of visualization will be specifically analyzed and evaluated for their potential use in the TeachEngineering project. Section V will provide a conclusion and recommendations for future action.

TEACH ENGINEERING

TeachEngineering exists as "a searchable, web-based digital library collection populated with standards-based engineering curricula for use by K-12 teachers and engineering faculty to make applied science and math (engineering) come alive in K-12 settings" ("TeachEngineering"). The goal of this system is to provide teachers with easy access to high-quality lesson plans that meet national and state-specific educational standards. Supported financially by the NSF National Science Digital Library, this project is a collaborative effort between the American Society for Engineering Education and the faculty and students associated with five universities.

TeachEngineering was created to address the problem facing K-12 educators of how to find well-designed lesson plans. Designed for these users, TeachEngineering represents a collection of peer-reviewed and certified lesson plans that educators can utilize to promote engineering-related disciplines in the classroom. The aim of TeachEngineering is to allow users to *explore* the available documents and find lesson plans that are of interest; regardless if the user has a specific or general idea of what they are looking for, they can use TeachEngineering to find lesson plans that meet educational standards. Thus, TeachEngineering can be an excellent resource to help educators in developing curriculum. With the focus on designing the document repository for the users' benefit, we can approach the task of actually implementing such a system.

In order to develop a useful digital library system that can help educators obtain lesson plans, there are two main issues that must be addressed. First, a method of information retrieval, involving determining which documents to return to the user and

how to order them, must be established. As a result, an effective relevance metric must be developed to demonstrate associations between search parameters and the documents within the database. In particular, there is the concern of how to map the state-specific certifications of these lessons plans for use in other locations. Educational standards vary between states; what might be appropriate as a 5th grade math lesson in one state could be used as 4th grade curriculum in another. Simply because a lesson plan was designed for use in a different state is no reason to exclude the possibility of its use. When searching, not only should documents that perfectly fit the criteria be returned, but also documents that match close enough that they ought to be considered. The objective is, "given a collection of objects each described by the values associated with a set of attributes, find the most acceptable such object or, perhaps, a small number of candidate objects worthy of more detailed consideration" (Spence, 2001, p. 77). This problem, although significant, is beyond the scope of this paper.

The second issue involves identifying and designing effective methods to display the retrieved search results, using information visualization tools and techniques. The method of presenting this data to users is critical to their ability to efficiently and intuitively discover the most relevant lesson plans from the digital library. Regardless of the specific results returned from the search query, this information can be visualized and interpreted in multiple formats, significantly impacting the ability of the user to identify useful documents.

My research has focused on addressing the second of these raised issues, with particular regards to identifying the optimal solution to visualizing multivariable search results in a graphical user interface. Drawing on fields of human-computer interaction,

design, information visualization and others, I have surveyed the available techniques, applications and methods to visualize such data. The problem of how to visualize digital libraries, in this case specifically the TeachEngineering system, is complex and without a clear-cut, definitive solution. There are numerous visualization techniques, each with various strengths and limitations; although there is no single technique that definitively solves this problem, there are several methods to visualize the data that deserve further consideration.

General Considerations & Limitations

Because TeachEngineering is a web-based tool, visualizations need to be capable of rendering quickly in various web browsers, which may exclude certain techniques from being applied. An important consideration is to keep the search result interface intuitive, in the sense that it can be clearly understood by users without specific training. Furthermore, it is important that the visualization technique can handle multiple variables, depending on the number of user specified inputs.

Database Size

As of August 8th, 2008, there are 436 activities and 240 lessons, organized into 36 curricular units and covering 13 main subject areas, listed for public review from the TeachEngineering database. It is fair to expect this amount to continue to increase over time to perhaps upwards of 10,000 documents or more. It is unlikely, however, for the database to become so large that query performance (speed) issues become a concern from a visualization standpoint. Regardless of the overall number of documents in the

database, only the queried subset of relevant documents is significant from a visualization standpoint. Visualizations of the collection as a whole may also be of some use, but will not be the focus of this paper.

Quantity of Search Results

In order to determine the finest graphical interface or visualization techniques, an important consideration is *"What is a reasonable number of search results?"*. The process of visualizing 15 relevant search results is different than doing so for 150. Having an idea of how many search results must be visualized is a key concern in finding good design solutions. Unfortunately, what constitutes a *reasonable* number of search results must be left ambiguous here, as the number of results that can be effectively displayed is directly related to the graphical technique being used. Some techniques may only simultaneously display a dozen documents well, whereas others may handle hundreds. It would be unreasonable to limit or exclude visualization methods based on the number of search results they can handle; however, the ability to handle small or large search result sets will factor into the analysis and recommendation of visualization techniques. Flexibility is beneficial, and information regarding the different visualization applications can be found in Section III.

Document Visualization vs. Information Retrieval Techniques

One important issue has to do with the difference between document visualization and information retrieval techniques. As Robert Spence succinctly notes, "… a user may be unable to say exactly what they are looking for in a collection of documents because

they may not *know* exactly what they are looking for. They may want to discover *roughly* what is available in the collection and then, by exploration, gradually refine their inquiry." (Spence, 2001, p. 179).

In the document retrieval stage, it is thus important to distinguish between filtering the database and returning only a subset that perfectly matches the search parameters, as opposed to providing documents that are a close match to the search parameters and have a high relevance ranking. If only the perfect search matches are returned, and if that is all the user is interested in, then there would likely not be many documents to show. Specifying multiple parameters can quickly lead to only a small number of compatible documents, if any at all, and these limited results could be visualized via a simple list. However, following Spence's line of reasoning, we assume that the users may be interested in search results that vary from their exact search parameters. Although not required for every search query, the existence and inclusion of a relevance metric is important as it will allow for additional forms of visualization that may provide user insight as to which documents are available and relevant.

Regardless of the specific document retrieval method being used, the visualization step merely projects the list of relevant documents into one form or another such that they can be interpreted. Many searches, regardless of the number of input parameters, simply result in a list that displays the results, ranked by relevance. In many cases this may be appropriate and perhaps the best feasible method of displaying results; however, in the process it abstracts much of the information away from the user. Other visualization techniques can provide more information about the search results and display identified

documents in a way that allows the user to determine for themselves which are the best search results.

There is a level of complexity related to the interplay between document retrieval and the resulting visualization. For example, a user may wish to search via one parameter, but show the results by comparing other attributes. Although this contrast between which attributes are *selected* on and *visualized* by is not common, it is still worth consideration. In the analysis of various visualization techniques, examples of potential user inputs will be used to demonstrate how the visualization technique might display the search results.

Variables

The TeachEngineering site breaks down its search operation into three different types: Simple, Advanced, and Educational Content Standards. The Simple search consists of a single, general search parameter; if the word is found anywhere in a lesson plan or activity, the document will be returned. The advanced search consists of several parameters including: Keywords, Words in Title, Words in Summary, Words in Engineering Connection, Grade Level (range) and Time Required (range); additionally, activities can be searched upon based on Group Size (range) and Cost per group (range). It should be noted that the parameters of words in title, summary, or engineering connection are strict; any document not complying with these criteria will not be returned, regardless of how well it fits the other parameters. Lastly, the Educational Content Standards Search can be based on the document Source, Subject, Topic, Number/Version, and Grade Level (range). Based on these criteria, an individual search

may currently contain as many as eight different parameters, and thus potentially requiring visualization techniques that can handle up to this many different dimensions.

Although not a direct, user-inputted parameter, the computed document relevance represents a search results dimension that can be used to sort, organize and visualize the list of returned documents. This relevance score, on a continuous scale from zero to one, could be in relation to how well a TeachEngineering document matches the user-inputted keywords or educational standards. The use of this relevance metric in visualizing search results is a common technique that will be explored further in Section IV.

For the sake of our investigation and analysis, we will attempt to demonstrate the use and visualization of the following variables: Grade Level, Cost, Group Size, Time Required, Relevance (to a keyword or educational standard) and State Source. The search parameters that strictly filter data (such as Words in Title) will not be addressed, as they merely limit the overall subset of data to be returned and there is no need to visualize these parameters. Looking at our list of variables, it is of note that both continuous and discrete data are being represented. The relevance metric is continuous, as values may range anywhere from zero to one. The other variables represent discrete data that are often represented via a category value (state) or range of integers (time, cost, grade and group size); in each case, there are a limited number of values that can be selected. Despite cost and time actually representing continuous data, they will be treated as discrete and only nonnegative integer values will be accepted for their representation. For example, it would not be practical to have a value of *"Grade 5.7"*; other factors such as time and cost will also be represented with whole values to reduce complexity and allow for clear graphical representations.

VISUALIZATION TECHNIQUES

In order to visualize multivariate datasets, there are two main approaches that are used. In the first approach, data is reduced to two or three dimensions through projection – in these low dimensions, a basic scatterplot can be used. In some cases, a matrix of scatterplots can be used to provide additional views of complex datasets. The second approach is to use some technique in which a large number of variables can be presented in a low-dimensional display format. Well-known techniques of this form include rearranging axes to be nonorthogonal (parallel coordinates and star plots); hierarchical approaches (tree structures and worlds within worlds) or screen-based techniques (pixel-oriented techniques and themescapes). Both these and additional techniques have shown to be successful in presenting multivariate data for certain situations. Through an examination of these different visualization techniques and how they can be applied to TeachEngineering, we can gain insight and provide recommendations on how best to visualize search query results.

Scatterplot

The scatterplot is the conventional and most commonly used approach to the visualization and interpretation of bivariate data (Voigt, 2002). It consists of a two-dimensional plot, with each individual record in the dataset being drawn as a point in the Cartesian coordinate system (Spence, 2001, p. 39). Scatterplots are particularly adept at showing correlation and relationships between two variables, and clusters of similar data points can be seen (Chiricota, 2004).

Indeed, when feasible, there are several advantages to using the scatterplot approach. Because the technique is so common, users find them intuitive and will not require any sort of specialized training. Furthermore, compared to more graphically and computationally complex visualization techniques, scatterplots generally render fast, making them ideal for interactive use via the Internet (Van Wijk & Van Liere, 1993, p. 120).

In situations where data can be reduced to a low number of dimensions (two or three), scatterplots are the preferred technique (Voigt, 2002). In addition to representing two-dimensional data, as is usually the case, scatterplots can be extended to encapsulate additional dimensions of data through the use of volume, size, icon shape and color. However, it should be noted that adding additional elements to the scatterplot raises its complexity and makes user interpretation more difficult. When using color in a scatterplot, it is suggested to use the axis coordinates to display the more important, primary variables, and other elements such as color to indicate secondary ones (Card, 1999, p. 30).

The main limitation of this technique rests on difficultly to display more than the two dimensions governed by the axes (Thai, 2008, p. 224). Although additional variables can be mapped into the two-dimensional space, the use of more than three variables makes the graphic significantly more difficult to interpret, even for users with training. In some multidimensional situations it can make sense to use dimensional reduction techniques to view slices of the data in two-dimensional space, such as with the scatterplot matrix and Hyperslice methods. Another concern with scatterplots is the issue of over-plotting, where there are so many points on the grid that they overlap and cannot

easily be individually identified; the use of color and translucence can help to limit this problem, usually associated with large datasets, although it cannot completely eliminate it (Andrienko, 2004, p. 96).

Regardless, for low-dimensional situations, it is clear that scatterplots are considered a favorable technique to visualize information, and as a result this has caused for its adaptation into numerous visualization tools.

Scatterplot Matrix

Extending the concept of the scatterplot, scatterplot matrices allow for the displaying of more than two variables at a time (Voigt, 2002). Assuming an N-dimensional data set, a scatterplot matrix is an arrangement of $(N^2 – N)/2$ pairs of two-dimensional plots in which row and columns of the matrix share common variables. Relationships between individual variables can be observed by scanning a row (or column) and analyzing how one variable is plotted against all others.

Scatterplot matrices provide simple representation of unstructured data. “An advantage is that the different dimensions are treated identically, no more or less arbitrary decision is expected from the user how the data must be structured for presentation purposes” (Van Wijk & Van Liere, 1993, p. 120).

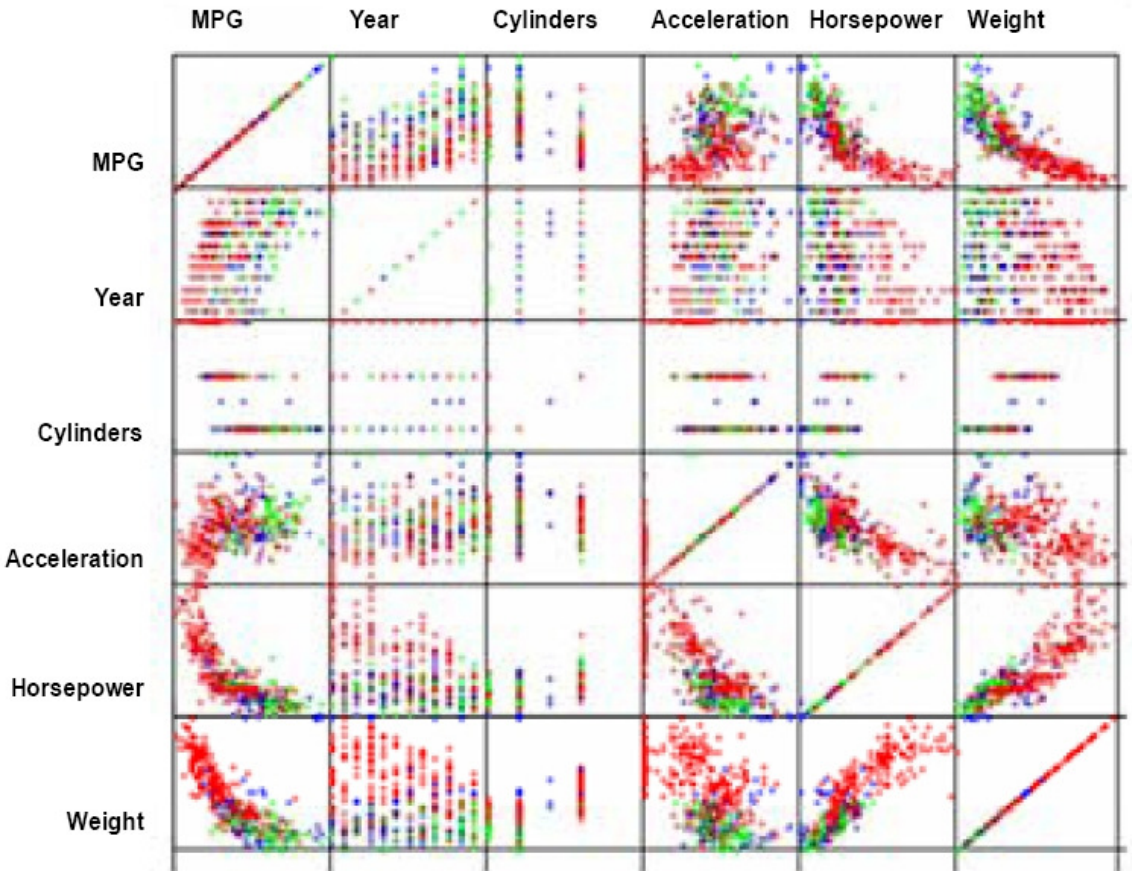


Figure 1: A scatterplot matrix displaying six variables and brushing. (Kromesch, 2005)

The scatterplot matrix allows for individual relationships between two variables to be seen in a multivariate dataset. However, it does not provide as a means for visualizing more than the normal two dimensions. Furthermore, since the matrix contains as many scatterplots as there are pairs of parameters, it can become unwieldy for more than about five parameters (Spence, 2001, p. 42). Other independent reviews have found that scatterplot matrices do not scale well to high-dimensional datasets, as they suffer from display level clutter (Yang, 2004, p. 73).

It should be noted that the scatterplot matrix includes redundant information, which can be reduced. "All off-diagonal slices $S_{i,j}$ are the same as slices $S_{j,I,}$ rotated over

90 degrees" (Van Wijk & Van Liere, 1993, p.121). Regardless of the overall dimensions being visualized, there are only N(N-1)/2 unique scatterplot combinations in the matrix; the plots along the matrix diagonal have the same variable on both axes, and are insignificant (Voigt, 2002). In most cases, the redundant and unvalued diagonal plots in the matrix can be omitted; in this case the visualization is called a triangular matrix.

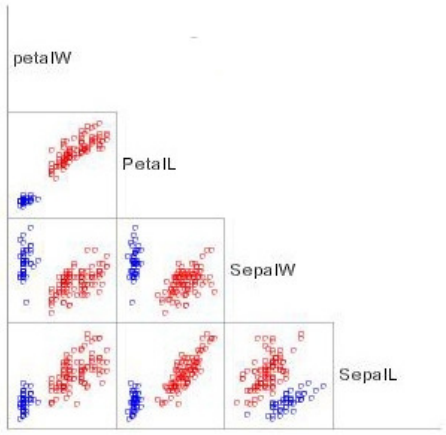


Figure 2: Triangular scatterplot containing the Iris dataset (Voigt, 2002)

The primary problem with scatterplot matrices is that the overall relationship of the entire dataset is not easily visible, and it is difficult to see patterns that are present only when three or more dimensions are taken into account (Voigt, 2002). As with traditional scatterplots, possible solutions to encoding additional dimensions within a single plot include the use of color, icons, and size.

Hyperslice

A technique very similar in design to the scatterplot matrix, Hyperslice is a method developed for the visualization of multidimensional functions into two-dimensional slices (Van Wijk & Van Liere, 1993, p.119). However, unlike the scatterplot matrix, Hyperslice is not designed for use with unstructured data but for the visualization of multivariable scalar functions. "The use of two-dimensional slices allows for faster rendering, and more important, easy interaction via direct manipulation" (Van Wijk & Van Liere, 1993, p. 120). Wong, Crabb and Bergeron suggest modifying this

technique to avoid the duplicate mirror images that appear in this matrix-based visual representation (1996, p. 75).

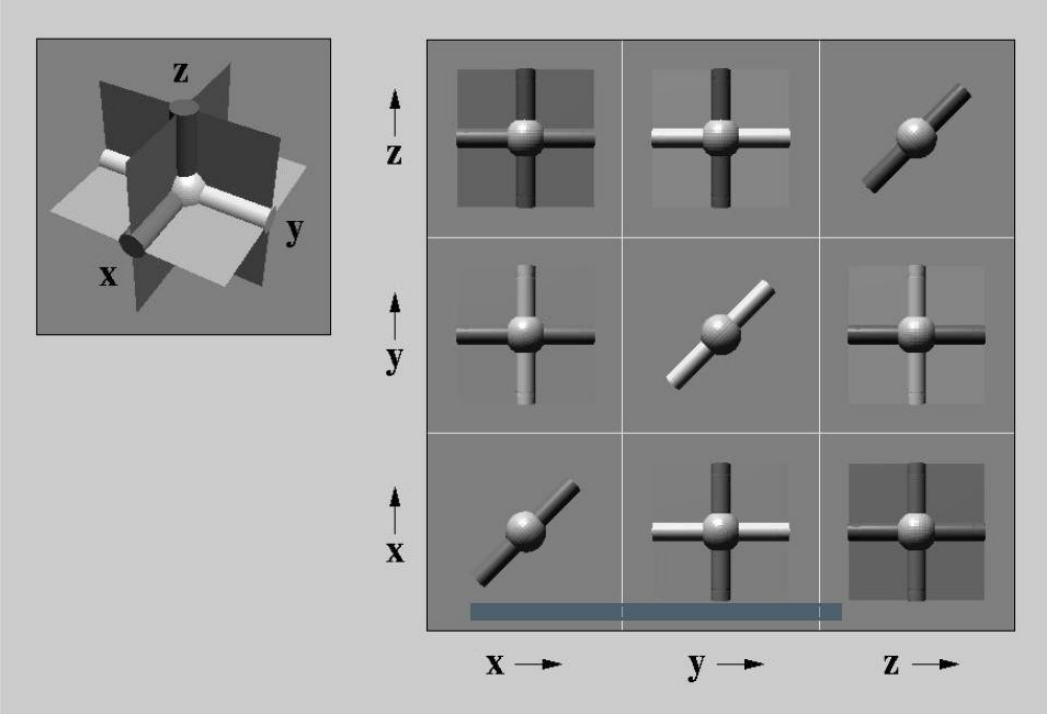


Figure 3: The concept of HyperSlice for N=3 (Van Wijk & Van Liere, 1993)

Parallel Coordinates

The concept of parallel coordinates has become a widespread technique used to visualize high-dimensional geometry and analyze multivariate data. Originally invented in the late-nineteenth century, it was rediscovered and popularized by Al Inselberg in 1959 (Voigt, 2002). The technique of parallel coordinates is realized by reconfiguring the coordinate axes to be non-orthogonal and situating them side-by-side, in parallel to each other, with line segments running between the axes (Chen, 2001, p. 111). Dataset variables are mapped to each axis on a one-to-one relationship, and thus there are as many axes as dimensions required; there is no limit to the number of dimensions that can

be projected into two-dimensional space, and it allows the user to "see more than four dimensions" (Jianxin, 2007, p. 322).

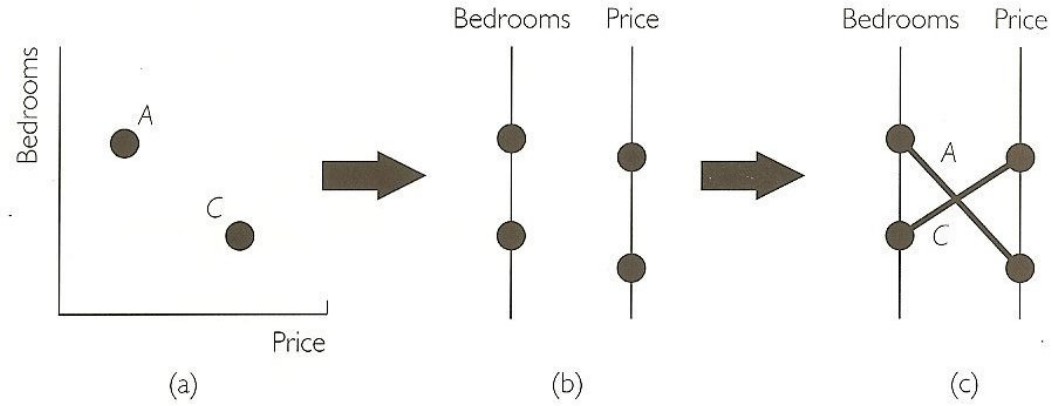


Figure 4: The principle of parallel coordinate plots (Spence, 2001)

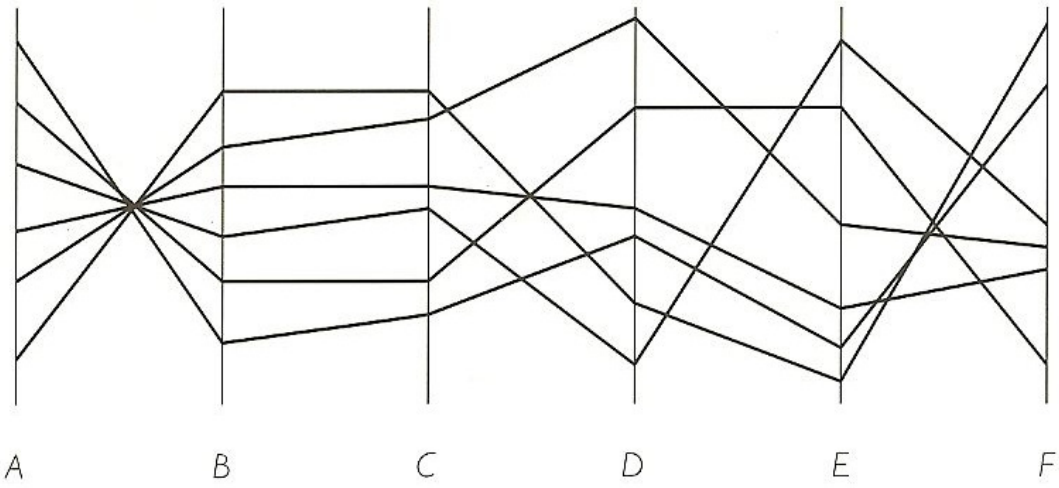


Figure 5: The principle of parallel coordinate plots applied to six variables (Spence, 2001)

Parallel coordinates are useful for demonstrating relationships between two dimensions, particularly when the axes are adjacent to each other. Although in theory parallel coordinates can support a limitless number of dimensions or data items, different

authors have suggested appropriate limitations to maintain usability of the technique. Chen and Wang argue that parallel coordinates are "limited to data sets that have only a few more dimensions than 3D" (2001, p. 111). Voigt (2002) suggests the limit of dimensions to be about a dozen, and Inselberg demonstrates the technique using twenty or more dimensions in his examples (Inselberg, 1999, p. 113). If more than a limited number of dimensions are displayed using parallel coordinates, then the visualization gets crowded. Similar logic is also applied to the size of the dataset that can be effectively visualized on the set of parallel axes. Recommendations for the number of data items that can be simultaneously viewed range from several dozen to a few hundred, depending on the screen size and implementation used. If the pattern of lines running across the axes becomes too crowded and dense, it becomes impossible to trace any individual line visually and thereby understand the characteristics of a single item. The technique of *brushing* the parallel-coordinate plot with color can help to reduce this problem, as it aids in the clustering of similar items and presents a clarified graphic. Inselberg is more optimistic regarding the flexibility of the parallel coordinates technique, and says:

> Without the proper geometrical understanding and queries, the effective use of parallel-coordinates becomes limited to small datasets. But contrast, skillful application of the methodology's strengths enables the analysis of datasets consisting of thousands of points and hundreds of variables. The intent here, is not to elaborate on the design and implementation criteria but rather to provide some insights on the 'discovery process'. (Inselberg, 1999, p.108)

In addition to the potential problems of displaying too many dimensions or too large of a dataset, there are a few other considerations of note. Because parallel coordinates is based in part on the linear combination of consecutive variables, each dimensional axis must share a similar common scale; this may limit the ability to drill-down into more informative views of the data (Moustafa, 2006). In addition, although all of the axes are lined up in parallel, research suggests that the "ease of interpretation can be strongly influenced by the ordering of the axes" (Spence, 2001, p. 47). Regardless, parallel coordinate plots treat all variables equally, which is a significant advantage of the technique, and "experts in the interpretation of parallel coordinate plots can derive a great deal of understanding from these plots" (Spence, 2001, p. 47).

In an empirical experiment to test the usefulness of the parallel-coordinates technique, 90.9% of the participants stated that they were able to get information from the visualization at first glance; this statistic may be influenced by the fact that 36.4% of participants had some level of previous familiarity with parallel coordinates (Lazenberger, 2005, p. 118). The results of the study showed that while not ideal for generating insights or conducting extended data analysis, the parallel coordinates technique is effective in quickly providing information to the user.

Andrews Plot

An extension of the parallel coordinates plot, Andrews'curves represents a Fourier transformation of the dataset. "Andrews' curves plot each N-dimensional point as a curved line using the function $f(t) = \frac{x1}{\sqrt{2}} + x2 * \sin(t) + x3 * \cos(t) + x4 * \sin(2t) + x5 * \cos(2t) + \cdots$ "(Kromesch. 2005, p. 9). The advantage of this technique is that it can represent an unlimited amount of dimensions, and through the nature of the formula,

cluster similar values. The first variables entered into the equation carry significantly more weight than the later variables, which could be considered an advantage or disadvantage with regards to what data Andrews's curves is applied to ("Statistics Toolbox"). An undeniable disadvantage is the computational time to display each n-dimensional point, especially in large datasets. Furthermore, Andrews' plots can be difficult to interpret without familiarity of the technique.

Parallel Coordinates Box-and-Whisker Plot

Although not typically associated with multidimensional visualization, I propose the idea of combining a box-and-whisker plot with parallel coordinates to provide additional insight into the average values of a dataset. Although it could be utilized in small datasets, this would specifically help to show clustering and data ranges in large datasets. By superimposing a box-and-whisker plot onto the parallel coordinate axes, the median values for each variable can be seen, which may help identify data items with strong scores across all variables. The top quartile of values for a given axis will be located above the box, and the lowest quartile below it. Essentially, this addition would help to give the user a quick way to compare where a single data line stands in comparison to the other values in the plot. In Figure 6 we see an individual data item that fits the midrange of all variables. Figure 7 presents additional data lines to demonstrate how they might interact

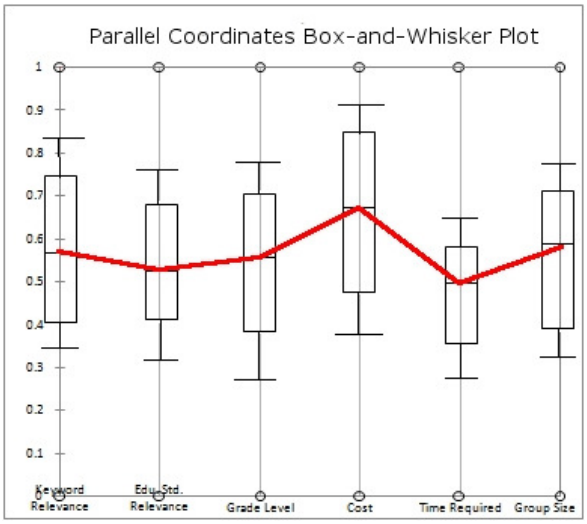


Figure 6: Box plots graphed onto parallel coordinates

with the box and whiskers. To my knowledge, this technique has not been applied to parallel coordinates, and is merely a suggestion for future parallel coordinate research.

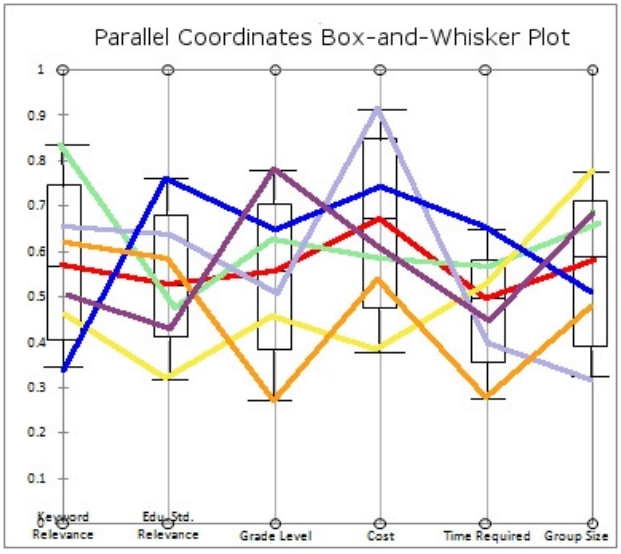


Figure 7: Six data lines plotted on parallel coordinates and represented by box-and-whisker plots

Star Plot

Also referred to as circular parallel coordinates or as star glyphs, the star plot is similar to the parallel coordinates technique, although with an arrangement of radial axes. Similar to parallel coordinates, the star coordinates allow for lossless projection for each dimension of n-dimensional space (Tominski, 2004, p.1242). These star plots range the axes on a two-dimensional circle with the origin at the center and the axes being radiated outward, usually separated by equal angles. Some work has been done, however, where the axes are not equally separated; this is often to highlight a dimension of importance or cluster similar variables that contain a strong relationship (Liu, 2007, p. 109). "The data length of a spoke is proportional to the magnitude of the variable for the data point relative to the maximum magnitude of the variable across all data points. A line is drawn

connecting the data values for each spoke. This gives the plot a star-like appearance and the origin of the name of this plot" ("Engineering Statistics Handbook," 2006).

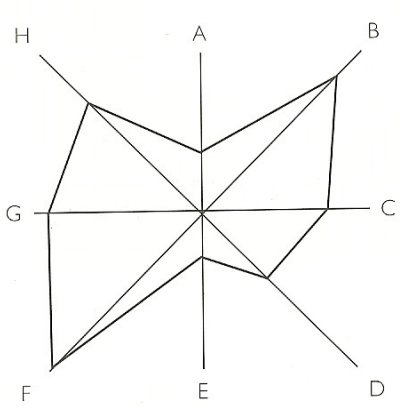


Figure 9: Example of a starplot with eight variables (Spence, 2001)

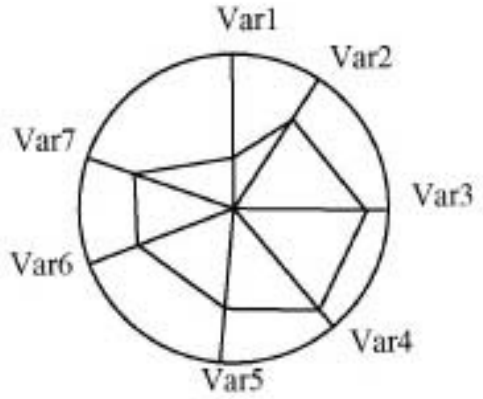


Figure 8: Star glyph with seven variables. Axes are not equal distances apart (Liu, 2007)

Similar to the parallel coordinates technique, the star plot is limited by the fact that the arrangement of axes plays a significant role in being able to interpret information, such as variable relationships. Further, if there are even a moderately high number of dimensions or data lines being represented, it can be difficult to follow individual line segments; as a result, color brushing techniques have been developed to help ease this concern (Tominski, 2004, p. 1244). In practice usually only a single data point is graphed onto a star plot ("Engineering Statistics Handbook," 2006). In this case, star plots are limited to datasets containing a few hundred plots, after which they tend to be overwhelming; it becomes too difficult to compare and interpret multiple star glyphs simultaneously (Fanea, 2005, p. 149). In an experimental study, Pillat, Valiati and Freitas found that in relation to parallel coordinates, star plots "has a more difficult layout for interpreting generic datasets. The values precise identification is not possible in this technique when only the mapping of items is considered" (2005, p. 28). In a separate

study, Lee, Reilly and Butavicius found that star glyph visualization led to slow, inaccurate responses to questions, with the participants reporting low confidence, in contrast to spatial visualizations resembling scatterplots (2003, p.7).

Despite these limitations, star plots contain several of the favorable attributes of the parallel coordinates technique, and are considered effective for providing quick impressions of data. Advanced techniques based off the star plot include the *parallel star glyph* and *Data Meadow*.

Parallel Star Glyph

Originally created as a method to address the problem of overlapping data lines in parallel coordinates, the parallel glyphs technique "integrates parallel coordinates with star glyphs by extending parallel coordinates into 3D space and unfolding them around a pivot axis" (Fanea). Each data line, from the parallel coordinates, becomes an individual dimension on the pivot axes of the star glyphs; this contradicts previous interpretations of star plots, and the authors provide explanation:

> Traditionally, Parallel Coordinates have one axis for each dimension of the data set. Consequently, the Star Glyphs emerging in our visualization have on glyph per dimension, each spike corresponding to one data item. This is not the conventional way to map the data in Star Glyphs: most related literature shows each glyph representing all dimensions of a single item. In fact, our visualization can generate this representation as well by merely switching the way the data is read from the data table: changing from columns over rows to rows over columns or vice versa. This also

> results in a modified Parallel Coordinates representation that shows dimensions as polylines and objects as axes. (Fanea, 2005, p. 151)

In the article, the authors are mainly concerned with the representation where each glyph represents a dimension, and the spokes represents the magnitude of the data line running through it, as seen in Figure 10:

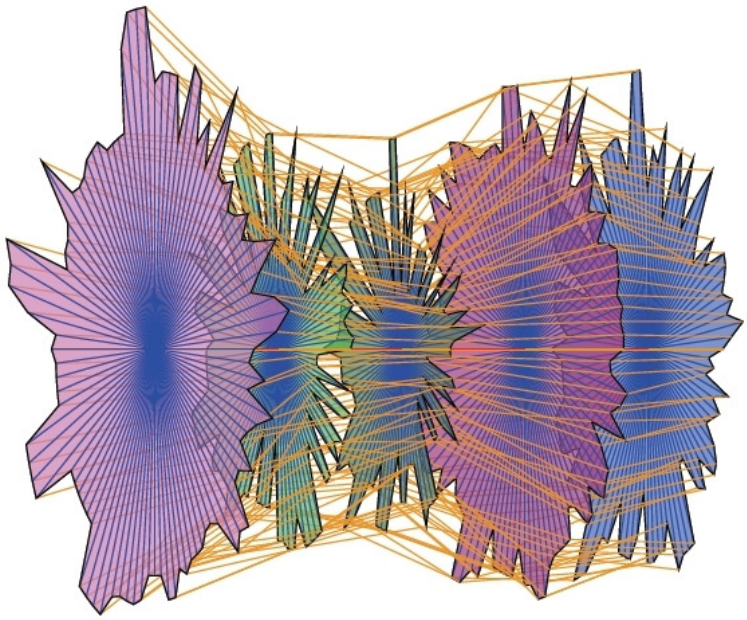


Figure 10: 3D star glyphs representing dimensions (Fanea, 2005)

Although perhaps useful in some situations, I believe it is still difficult, perhaps even more so than in regular parallel coordinates, to track a single data item, across multiple dimensions. Rather, I find the idea of switching the way the data is read, as described in the previous quotation, to be a novel and exciting approach when applied in a digital library setting. As seen in Figure 11, each star glyph represents one object as opposed to one dimension, as shown in Figure 10.

Figure 11: Star Glyphs, each representing one object as opposed to one dimension (Fanea 2005)

With regards to utilizing this representation for TeachEngineering, consider each individual glyph as a document returned from a query. Since each radial axis is mapped to a dimension, and the magnitude of the spike shows query relevance for that parameter, the largest glyphs in the visualization represent the documents that best match the search results. Glyphs could be ordered by some metric, although it certainly is not necessary as it is easy to relate and compare glyph size in this arrangement. Although color can be used to represent variable information in some visualization techniques, in the parallel glyphs technique it is used only to increase the readability of the graphic; with the glyphs so spatially close to each other, color becomes important in differentiating objects on the screen.

Data Meadow

Designed to manage and present large, high-dimensional datasets, the Data Meadow represents a powerful method for interacting (selecting, filtering and combining)

with data (Elmqvist, 2008). The main visual element is the Data Rose, a parallel coordinate star plot that displays the selected variables. In contrast to typical starplot implementations, the Data Rose contains an entire dataset, which may contain hundreds or thousands of items, on a single plot. The starplot axes are equally separated and their relative location to each other impacts the tool effectiveness; it may be useful to employ axes reordering techniques to ensure optimal ordering of variables. The data can be represented visually in three different ways: color histogram mode, opacity band mode, and standard parallel coordinates mode. For all three methods, "a single black polyline is used to show averages for each dimension. Low values are close to the origin, high values reside on the outer radius" (Elmqvist, 2007, p. 192).

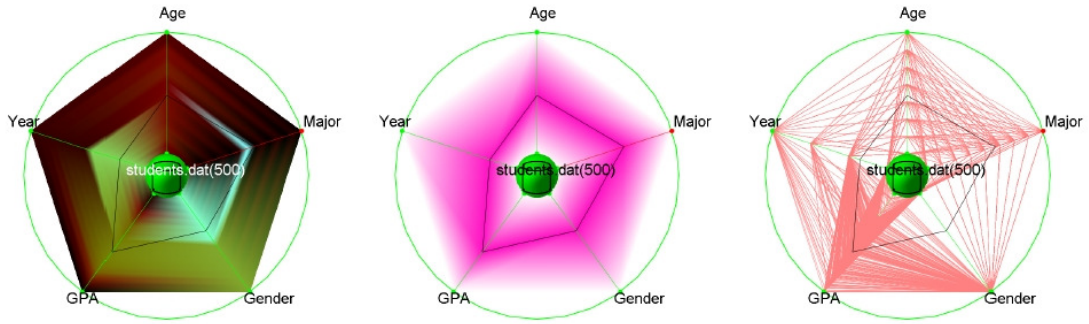


Figure 12: Sample DataRose visualizations for (a) color histograms, (b) opacity bands, and (c) parallel coordinates mode (Elmqvist, 2007)

The main advantage of the Data Meadow technique relies on its high level of user interactivity. Data sliders are positioned on each axis that allows for dynamic filtering via the data rose, and advanced queries are represented with multiple data roses on the screen. Although there are more advanced features of the Data Rose that are not being described here, its main purpose is to analyze trends in complete datasets and not identification of a single item.

TileBar

In typical information retrieval systems, queries are returned in a ranked order and visualized as a standard list; the document ranking, although controlled by some algorithm or function, is opaque and the end user cannot see why one document has a greater relevance than another (Hearst, 1996, p.394). The TileBar scheme was developed to combat this issue. This technique accepts multiple keyword topics from the user, each individually called a *termset*, and visually shows where the relevant terms (or synonyms) are located in a given document. For each search term specified, the TileBar method searches through the document and identifies each occurrence of keywords; each document is sliced via a statistical segmentation algorithm called *TextTiling*. "Rectangles correspond to documents, squares correspond to text segments, [and] the darkness of a square indicates the frequency of terms in a segment from the corresponding Term Set" (Hearst, 1995).

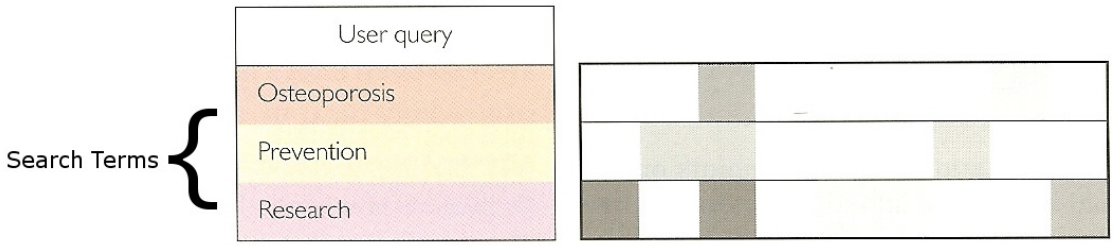


Figure 13: TileBar indicating the relative occurrence of topic words within segments of a single document (Spence, 2001)

It particular, this method helps to show where search terms are located in the document. Indeed, merely glancing at the TileBar visualization provides insight as to not only what documents are relevant, but what sections of a document contain the queried

search terms. Sliders can be used to indicate the importance that the user assigns to each search term, and the results will adjust and be ranked accordingly.

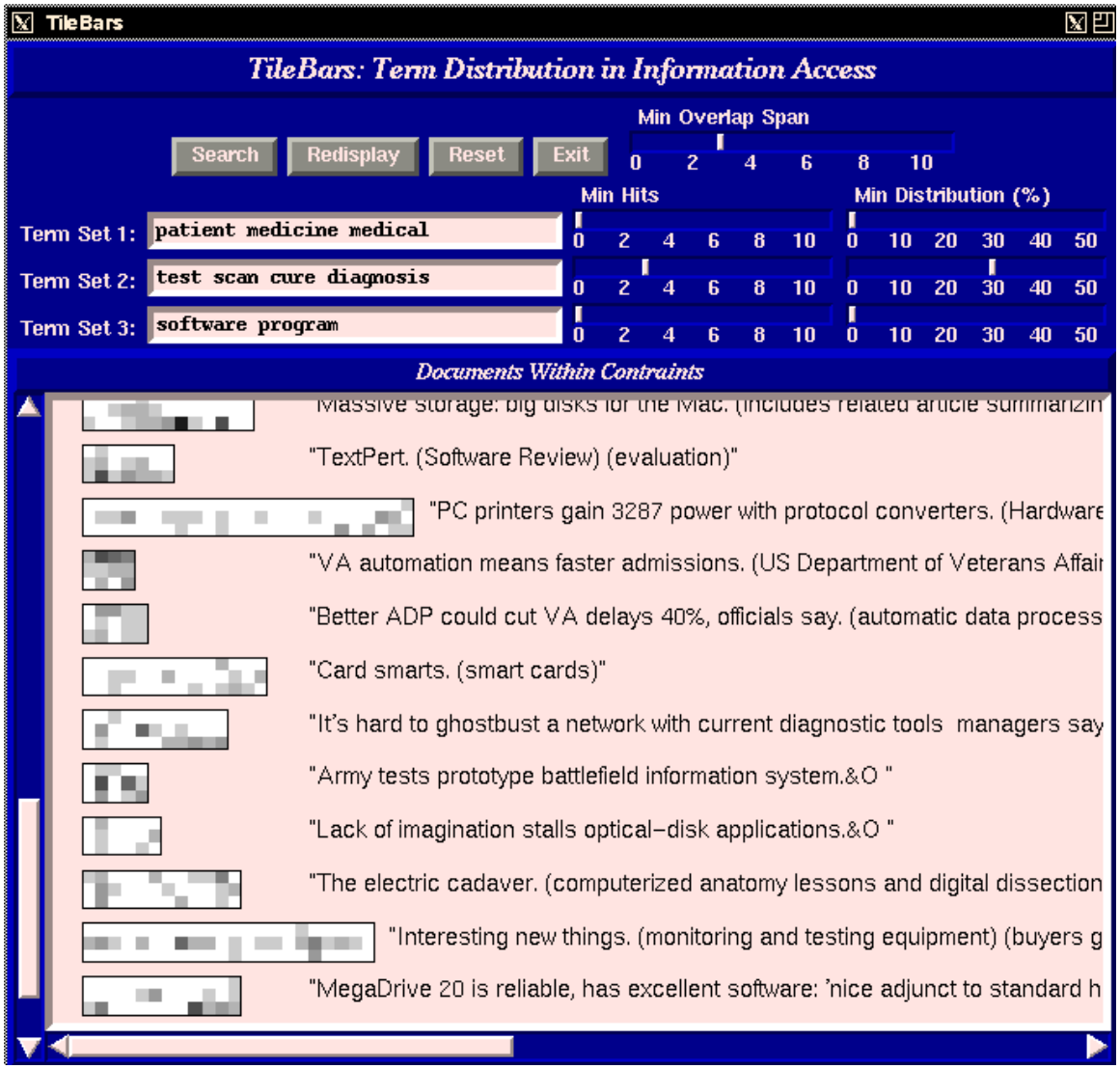


Figure 14: A typical set of TileBars for a collection of documents (Hearst, 1995)

This technique appears to be quite effective, as "the representation simultaneously and compactly indicates relative document length, query term frequency, and query term distribution" (Hearst, 1995). However, in order for this method to be effective, the entire text must be electronically available, not only the abstract or partial sections. Its use then,

seems particularly suited for digital library and journal article database situations. Although the visualization technique is straightforward and easy to interpret, the system may require some minimal training and understanding before it can be effectively used.

Although the TileBar method was originally proposed in 1995, not much additional research could be found regarding modern implementations. The issues of system performance and scalability are not addressed but are of particular concern; searching through large databases and long documents will impact the speed of the information retrieval.

A related technique to the TileBar approach is the GridVis system, which uses a similar visualization scheme to identify metadata location within a document; instead of using this technique to provide a relevance summary for several documents, GridVis focuses on analyzing a single document, and finding the most relevant sections within (Weiss-Lijn, 2002, p.50). The main importance of this tool, however, is that the authors have created a "servlet [that] will produce an HTML version of the document in which every paragraph tagged with the selected metadata is highlighted using a bold font. The browser will display the paragraph selected; thereby offering details on demand" (Weiss-Lijn, 2002, p.53).

Tree Structures

A large part of the world's information, including library cataloging schemes, are hierarchically structured and thus can be visualized using tree structures (Card, 1999, p.149). As a result, we will explore related visualization techniques such as cone maps, tree maps, and the hyperbolic browser. Information constructs in the tree are represented as nodes, which connect to each other through a parent-child relationship. The base node,

representing the highest level of abstraction, is known as the *root*, and all other information is derived from this general categorization. See Figure 15 for an example of a vertical tree structure with multiple elements spanning several generations. Tree representations are very good at displaying data relationships; "most people come to understand the content and organization of these structures easily if they are small, but have great difficulty if the structures are large" (Johnson & Shneiderman. 1991, p.152).

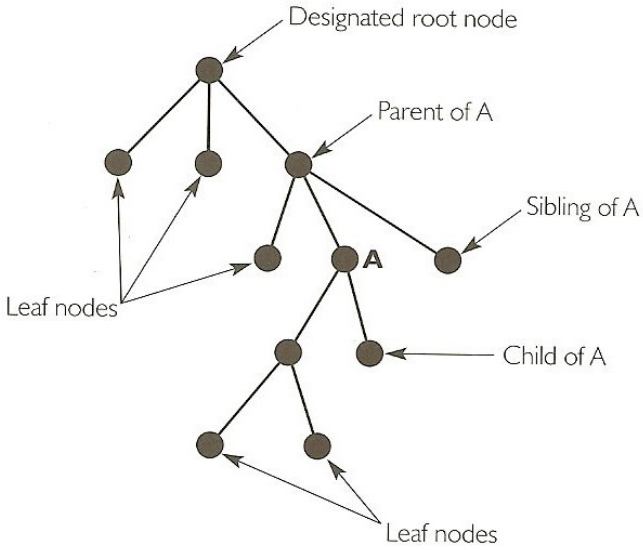


Figure 15: Terminology associated with trees (Spence, 2001)

Cone Tree

Tree graphs projected into three-dimensional space are known as cone trees due to the shape that emerges from the visualization; branches are arranged around a series of circles, as illustrated in Figure 16. "The inventors of the cone tree claim that as many as 1000 nodes may be displayable without visual clutter using cone trees—clearly more than could be contained in a 2D layout.

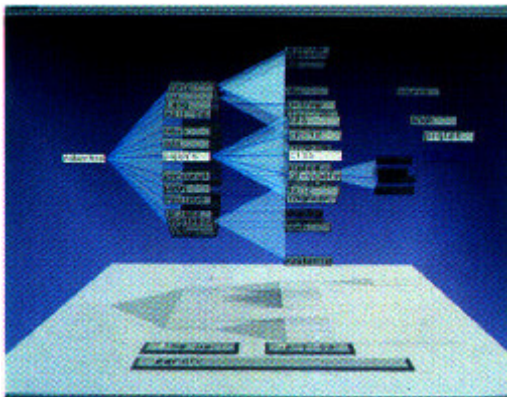


Figure 16: Cone-Tree representation of hierarchical data (Robertson, 1991)

However, 3D cone trees require more complex user interactions to access some of the information than are necessary for 2D layouts" (Card, 1999, p. 284). Due to the three-dimensional nature, some information nodes may not be easily visible; thus, user manipulation of the cone tree, such as through rotating it on its axis or other distortion techniques, must be applied to be able to see all of the data. It has been suggested that 3D hierarchical structures are easier for users to navigate and promote stronger information recall than other methods do (Calitz, 2001, p. 60)

Tree Maps

An alternative representation of a tree is the Tree Map (Johnson & Shneiderman, 1991). The inventors were happy with using traditional tree drawing layouts, particularly with small datasets, but wanted to improve the efficiency of using the screen space. "In a typical tree drawing more than 50% of the pixels are used as background. For small tree diagrams this poor use of space is acceptable, and traditional layout methods produce excellent results. But for large trees, traditional node and link diagrams cannot be drawn adequately in a limited display space" (Johnson & Shneiderman, 1991, p. 153). Building the tree map is a straightforward process: beginning with the root node, a rectangle is drawn that utilizes the full available screen space (Spence, 2001, p. 151).

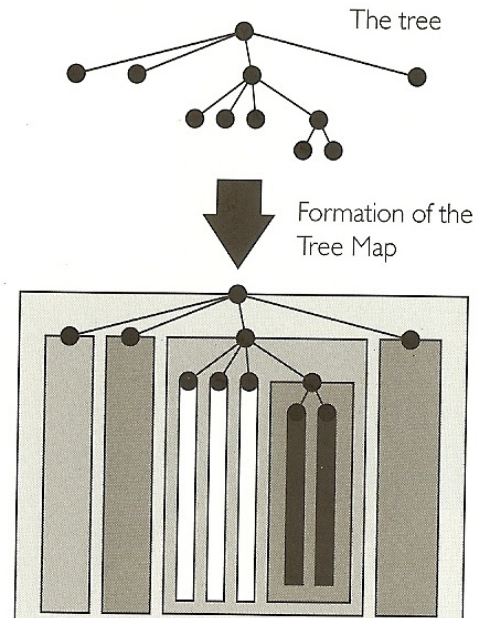


Figure 17: The mapping of a Tree Map from a tree (Spence, 2001)

Inside this rectangle are several other rectangles, each representing a subordinate, child node. This process is then repeated until all nodes have been drawn. On the tree map, the size of the rectangles is weighted to indicate the relative size or importance of the nodes. Color can also be used to denote attributes. The advantage of this method, compared to traditional tree views, is that the amount of information on each branch of the tree can easily be visualized; further, thousands of branches could be shown (Ware, 2004, p. 217). However, typically tree maps are more difficult to read because the hierarchical structure is not as clear as in other approaches. Figure 18 represents a more complex, albeit realistic use of a tree map, where it is displaying roster information for teams in the NBA.

Figure 18: Tree map displaying the hierarchy of the NBA. The large box represents the NBA, with four large vertical columns representing the divisions in the league. Horizontal boxes are then drawn to indicate each team located within a specific division. Inside each team box, players are represented by colored vertical rectangles. Data used for this graphic is unknown, but better players are represented with wider rectangles. (Spence, 2001)

Hyperbolic Browser

Another method used to visualize large hierarchies and graphs that have been converted into trees, the Hyperbolic Browser technique uses fisheye distortion to display all nodes on the hyperbolic plane.

> The hyperbolic browser initially displays a tree with its root at the center, but the display can be smoothly transformed to bring other nodes into focus. In all cases, the amount of space available to a node falls off as a continuous function of its distance in the tree from the node in focus. Thus the context always includes several generations of parents, siblings, and children making it easier for the user to explore the hierarchy without getting lost. (Lamping & Rao, 1995, p. 389).

The authors indicate that due to the context-focusing fisheye distortion, the hyperbolic browser can display ten times as many nodes as traditional tree visualizations in a similar space. When a new section of the tree is to be explored, simply clicking on or dragging a node to the center of the display results in the tree dynamically updating. The process behind implementing this technique involves the use of hyperbolic geometry.

The hyperbolic display appears to be an effective technique for projecting large tree hierarchies into two-dimensional space. It can be more intuitive to navigate than cone trees, and does not lose hierarchical focus such as tree maps. Although performance issues are of concern, particularly during animated node-focusing transitions, Lamping and Rao suggest several techniques to reduce the computational strain of the hyperbolic

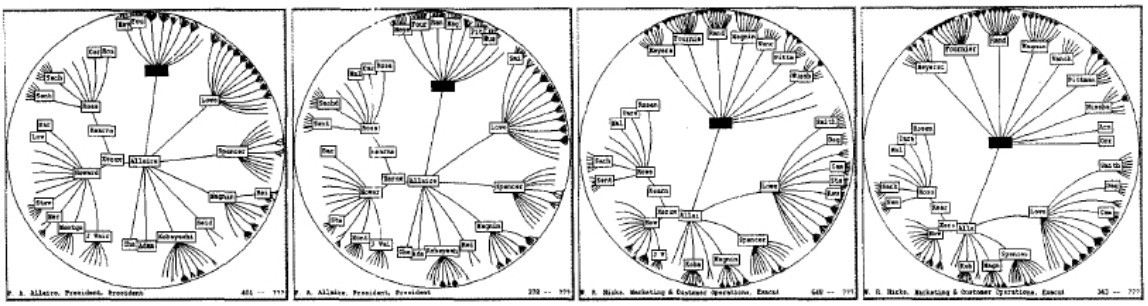


Figure 19: Results of manipulating the Hyperbolic Browser (Lamping, 1995)

approach (Lamping & Rao, 1996, p. 2). The largest roadblock to using the hyperbolic tree, however, is clearly due to Xerox having patented the software. As a result, it is illegal to use this visualization technique without licensing the technology from Inxight (an SAP company, sold off from Xerox) which has a cost upwards of $25,000. Perhaps the fee can be waived for use in an educational and research setting, such as for TeachEngineering; regardless, this still poses a limitation on the use of this method.

Brushing

Brushing generally refers to the use of color on graphics that allows subsets of data elements to be interactively highlighted (Becker, 1987). In this situation the color is not used to encode an additional variable, but rather is used to assist in the understanding of the data. This is particularly useful when displaying a large dataset that could become very confusing without color-clustering approach. Used to enhance the work

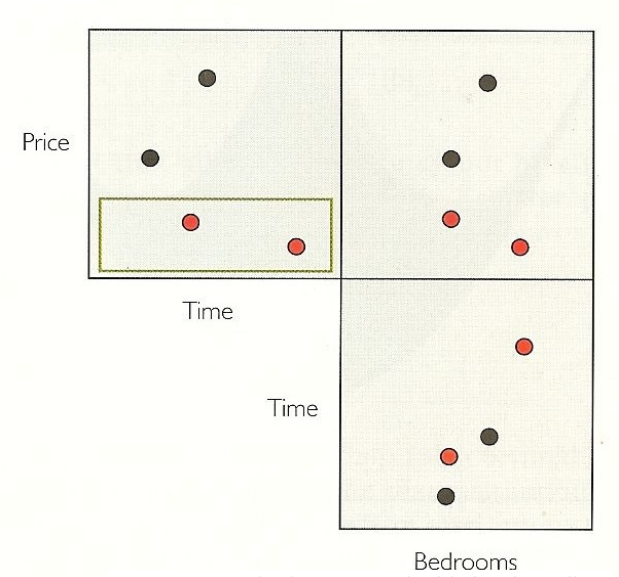


Figure 20: Brushed points are highlighted on all plots. (Spence, 2001)

of scatterplot matrices and parallel coordinates, brushing involves selecting a subset of data items and highlighting them; the user can then either delete the subset from the view or zoom in on it. In scatterplot matrices, individual plots are linked together to relate where data subsets exist in different views. Brushing is often used to cluster large datasets visualized with parallel coordinates, but can also be used to color individual data lines. Wegman (1990, p.10) suggests the use of between five and fifteen colors and the use of gradients in parallel coordinates. Heavily plotted areas can be blended with color mixes and transparencies of colored clusters.

CASE STUDY ANALYSIS

After reviewing the available literature on visualization techniques, we can now focus on how they might be applied specifically to the TeachEngineering digital library. First, we will summarize the information visualization techniques in the following table:

Technique	Dimensions	Advantages & Limitations
Scatterplot	2	Familiar, but difficulty displaying more than 3D.
Scatterplot Matrix	5	Directly compares all variables against each other, but can be difficult to read and interpret.
Parallel Coordinate	20	All dimensions are treated the same. Requires units to have a similar scale. Studies have shown users quickly obtain data from it.
Star Plot	8	Radial version of parallel coordinates, but in a more familiar layout. Quickly becomes cluttered with only a few data lines.
Parallel Star Glyph	6	Similar to parallel coordinates, but each glyph represents a data item, while spokes represent dimensions. Users likely not familiar.
TileBar	3	Identifies the location and concentration of multiple keywords within a document. TeachEngineering users likely are not looking for instances of a specific word.
Cone Tree	4*	Represents multiple dimensions, but quickly takes up screen space.
TreeMaps	10*	Takes up all available screen space to map as many variables as possible. May be difficult to read unless user has limited understanding. Similar to a Venn diagram.
Hyperbolic Browser	∞*	Scales well to show multiple dimensions. May have trouble indicating the notion of relevance. This tool, like ordinary library catalog systems, is hierarchical in nature.

* For the cone tree, tree map, and hyperbolic browser techniques, this value represents the highest number of levels in the hierarchy found. As will be explored later in this section, each hierarchy level on a tree can be mapped to a specific variable. Thus this transformation allows these methods to be compared directly with the other techniques in the table.

In the *Dimensions* column, the value indicates the largest number of variables represented by the specific technique in examples from the literature. Furthermore, the *dimensions* number does not take into account the use of elements such as color, size, or shape which could be used in various ways, such as to encode additional dimensions or to increase visualization readability.

In applying these different methods of visualization to TeachEngineering, we will utilize the following variables as potential search criterion: relevance (to an educational standard or keyword), grade level, time required, cost, group size, and state source. Using these variables, the goal is to represent documents from the TeachEngineering database that are either an exact or close match to the specified search parameters.

Scatterplot

The scatterplot is a straightforward technique that is used to compare and contrast values along two orthogonal dimensions. Users find the visualization intuitive and familiar, as it is identical to the x-y graphs used in basic mathematics. Indeed, when only using two variables, the scatterplot can be used to effectively demonstrate which documents fit search criteria, as we will demonstrate with examples. In Figure 21, the user has created a theoretical query looking for a document using the keyword *'Earth'* and the grade range of five to six:

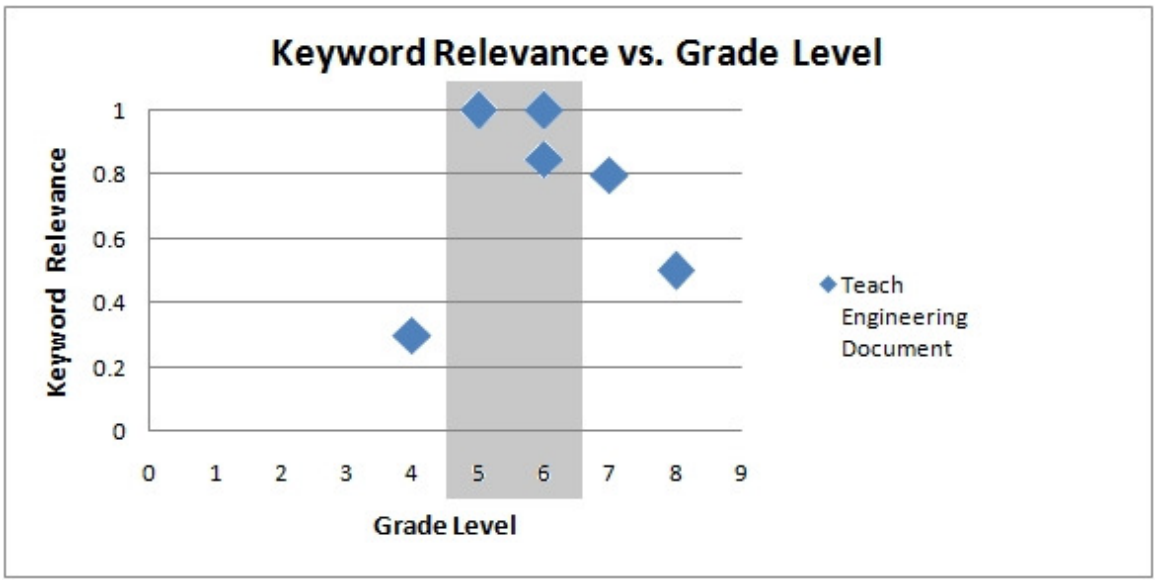


Figure 21: Scatterplot plotting keyword relevance against grade level. Queried grade range is highlighted.

Here, we see a graphical layout representing both parameters of interest that the user specified. Documents designed for grades five and six are highlighted, as they fall within the bounds; documents with the greatest relevance scores for the keyword "*Earth*" are located at the top of the chart. Thus, we can see that from our graphic example that there are two documents which fit the criteria. However, let's assume the search returned a plot that instead looked like this:

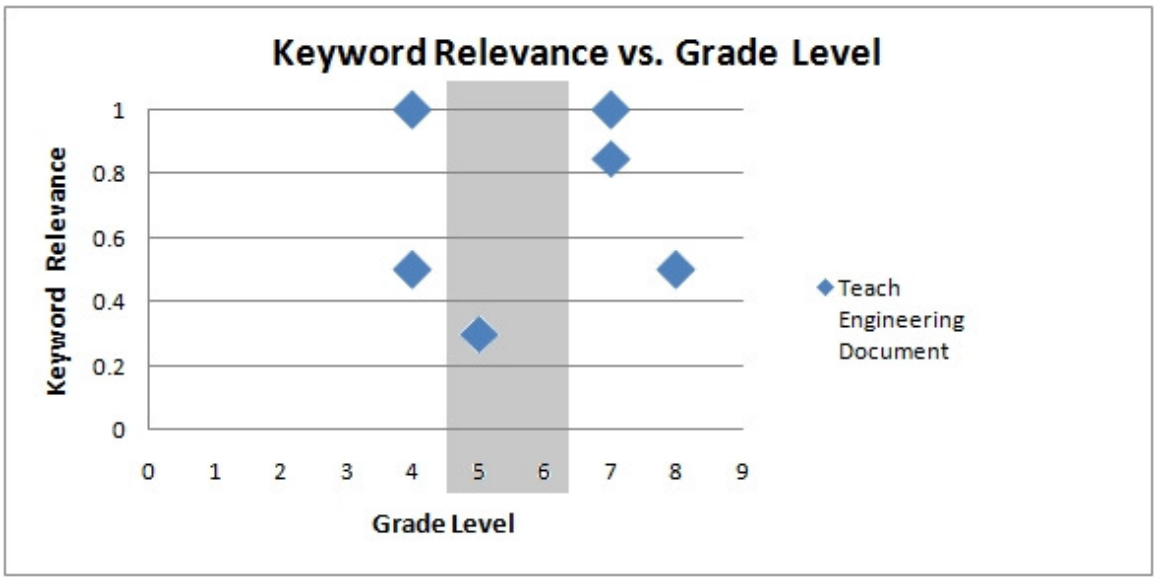


Figure 22: Scatterplot demonstrating relevant results outside of specified range.

In this scenario, there is only one document displayed in the specified grade range, but its relevance to the keyword is low. However, from our visual, we see that there are documents designed for both fourth and seventh grade that have a very strong relevance to our keyword. Although they are outside the specified age range, these documents represent lesson plans that may be of interest to the user, perhaps more so than the document that matched our criteria, despite a lower relevance score. Other methods of displaying search results would not provide these insights as to documents that are close, but not perfect fits. Assuming they even appeared in a traditional list, they would likely be ranked near the bottom, despite being what the user may really be looking for.

Extending our previous example, let us assume that the user now wishes to search against keyword relevance, grade level, and time required. In order to encode this additional variable, we will use color to indicate the value of time required for each document. This graph, although similar to Figure 21, now uses color (as specified in the legend) to indicate the approximate amount of time the activity will take.

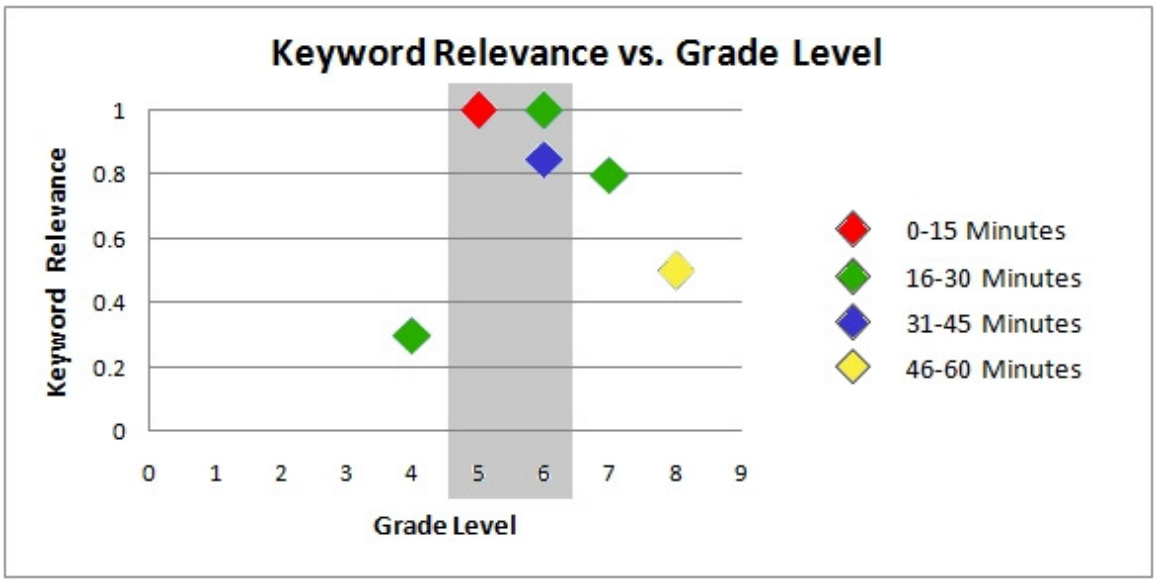


Figure 23: Brushing for additional dimensions on a scatterplot.

Although more complex, the scatterplot still appears viable as a method to visualize three variables fairly effectively. Continuing this trend, let's assume that in addition to previous parameters, the user also decides to search by group size, which will be represented by icon shape.

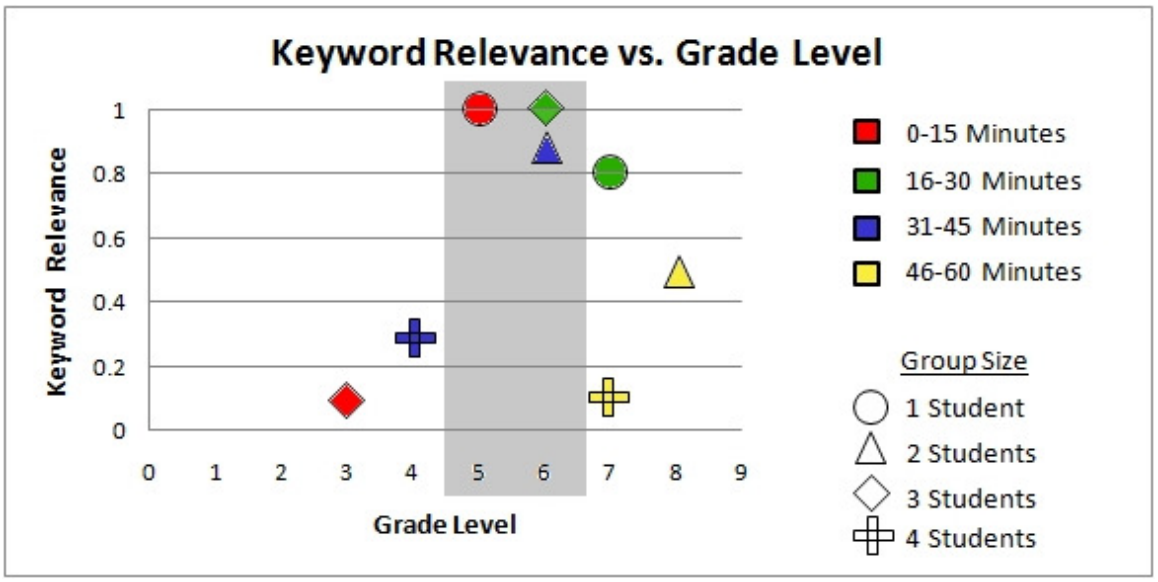


Figure 24: Icon and color used to represent four dimensions in a 2-D scatterplot representation.

At this point, the graphic becomes increasingly difficult to read, having to interpret color, shape, and location along two dimensions. For a graphic consisting of only a few data points, such as above, this may not present a problem; however, if the graphic contains more symbols, then it may be unwieldy.

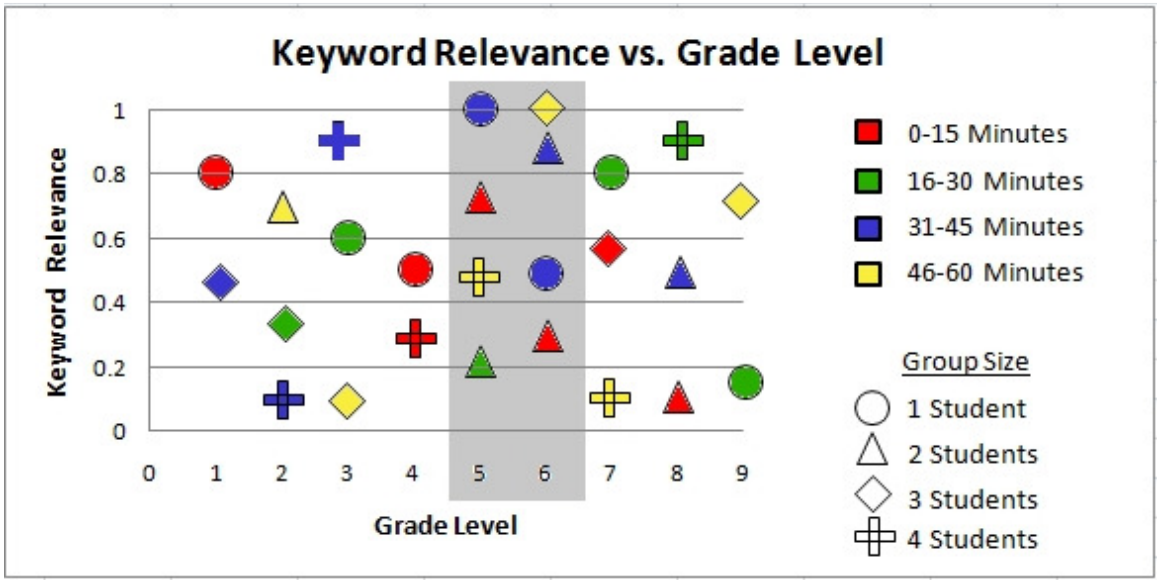


Figure 25: Clutter impacting the effectiveness of a scatterplot.

Although this 4-dimensional scatterplot does encode information for all four of the variables searched upon, it is awkward and difficult to read. One suggestion is to include a method to filter which icons are visible on the screen. In Figure 26 we see an implementation of this idea, where only results matching the checked criteria (minutes, group size) are visible. As a result, the graphic is much easier to read.

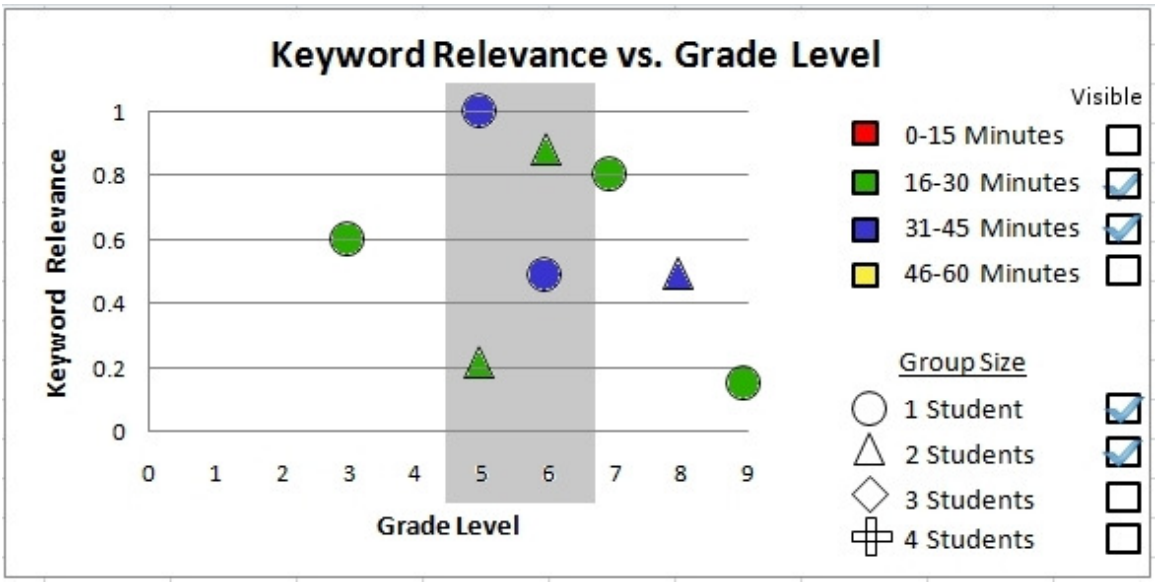


Figure 26: Filtering scatterplot results using criteria. This helps make the graph easier to interpret.

For the sake of exploration, we can attempt to add a fifth dimension, state source, to the graphic. In order to represent all of the data, however, we will replace keyword relevance with state source along the y-axis, and represent our relevance metric using the visual size of the icon on the screen.

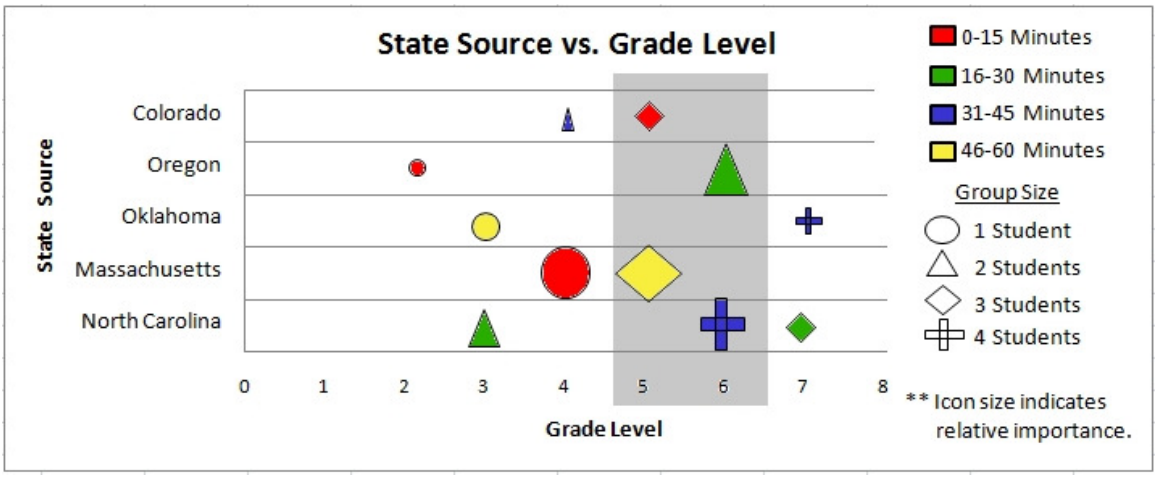


Figure 27: Attempt to encapsulate five-dimensions on a 2D scatterplot. Interpretation is difficult.

Clearly, as can be seen in the Figures 21 and 23, although the scatterplot is effective for displaying two or three variables, it becomes significantly worse for displaying more information than this. In addition to the sheer busyness of the scatterplots, there are some additional concerns. When only displaying two variables, each is assigned to an axis, and there is no problem. However, when additional variables are introduced, only two of them can be represented via the axes; choosing which to represent via the axis is not a trivial task. As mentioned by Card (1999), the axes should represent the *most important* variables; the question then arises, of which variables is the user most interested in? Users may differ concerning this issue, but it would be difficult and perhaps really unnecessary to provide them with an interactive way of manipulating which variables are represented via an axis and which through size, shape or color.

A second concern in using the scatterplot technique is the possibility of overlapping data points. Even when there are only a few points being displayed, there is a risk of documents overlapping. If, in our previous example, two documents for a single grade had a similar relevance score, their icons could overlap on the screen, perhaps completely, making individually identifying them more difficult. One solution to this issue is to use columns to represent value ranges (such as for grade, time, cost, etc.) and thus documents of similar value can be placed horizontally to each other, rather than overlapping.

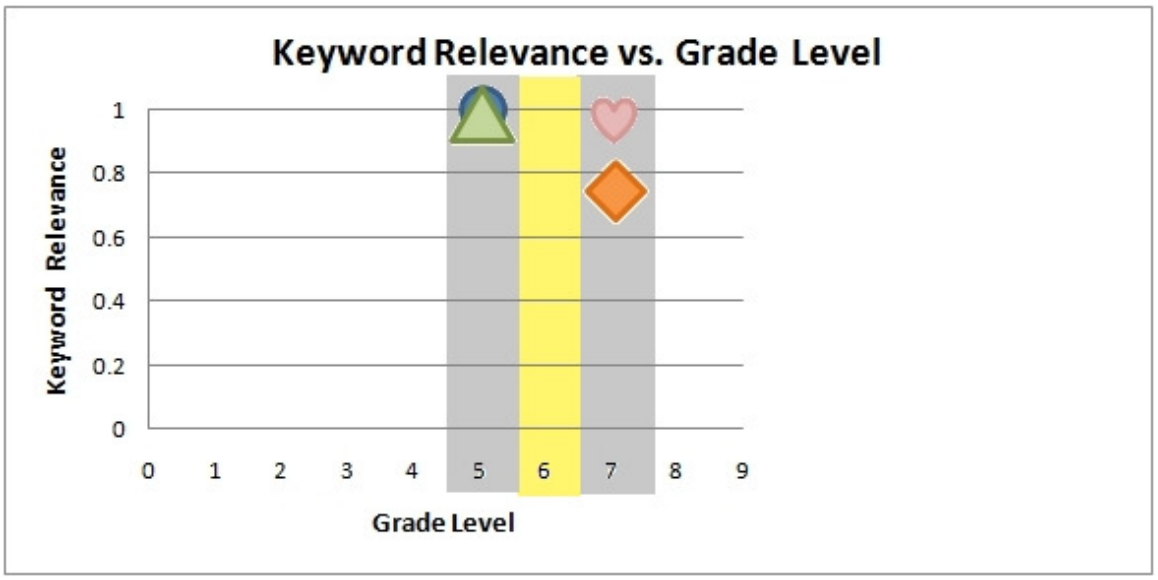


Figure 28: Overlapping scatterplot points.

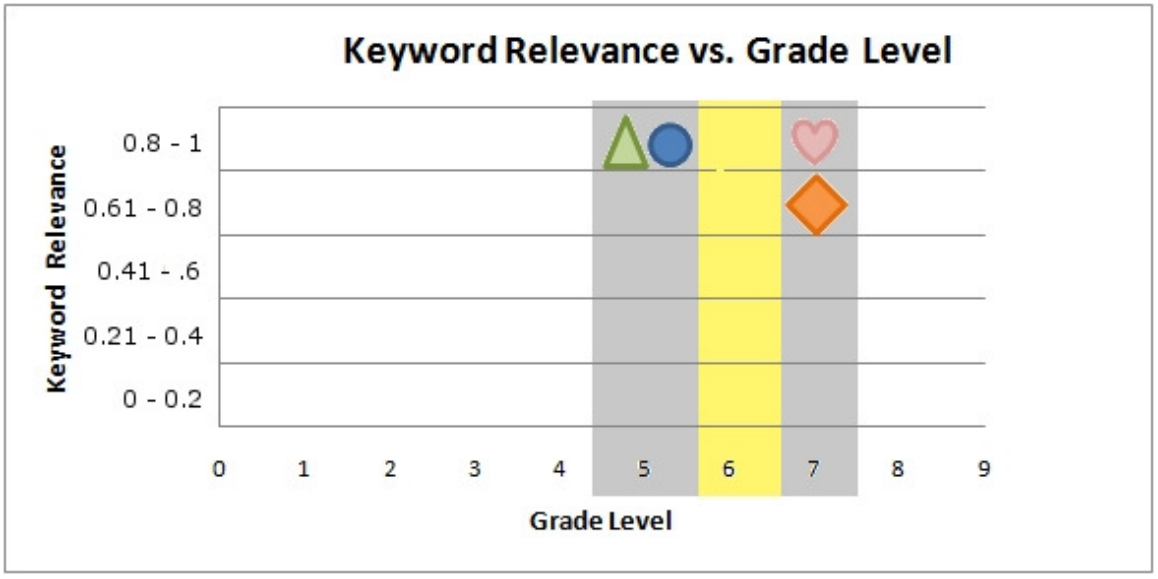


Figure 29: Adjusting overlapping points to sit horizontal to each other in a data column.

A third consideration centers around the ability to display data such as *source state* or *educational standards* on the scatterplot graphic. As TeachEngineering is currently, there are only a few of each category, but this is likely to change as more lesson plans and activities are added to the database. The idea of lining up fifty rows or

columns, to represent every state, on a graphic does not make much sense and would be cumbersome. Rather, an alternate approach could be to display relevance to a state, through a mapping scheme.

One additional suggestion regarding the use of scatterplots involves a method to visually demonstrate the appropriate grade *range* of a document; although the lesson plans usually indicate the grade level it was designed for, in many cases there is an additional range of grades that it is appropriate for. By extending *whiskers* from the document icon, the actual grade range can be visualized, as opposed to just being assumed. This technique adds additional information to the graphic, and could cause unnecessary clutter, but does present a method to visually indicate that documents not directly associated with a particular grade level may still be extremely relevant.

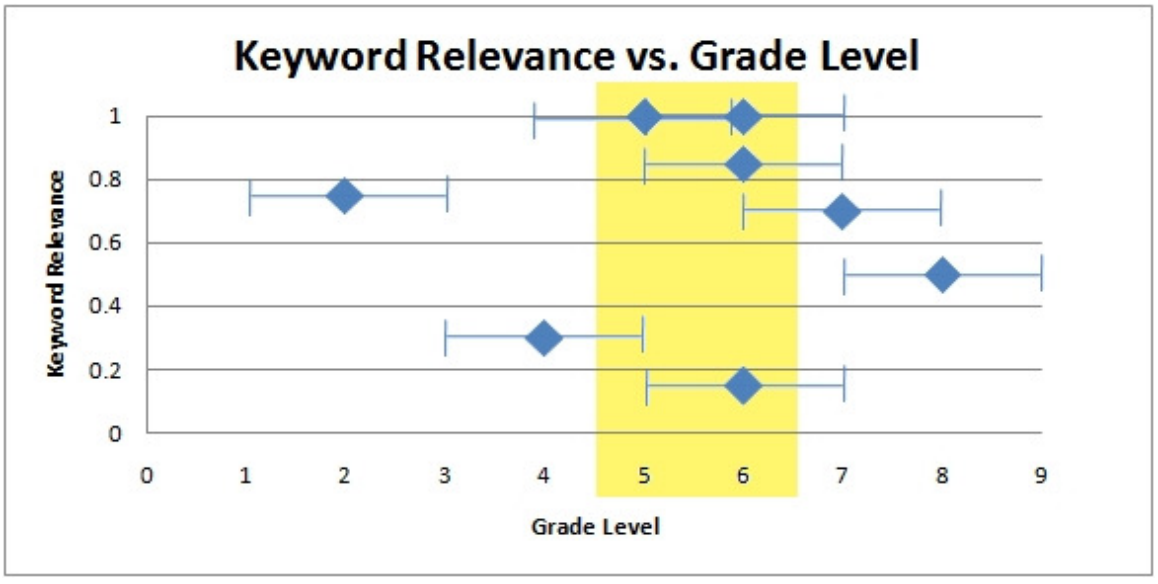


Figure 30: Using whiskers on scatterplots to specify documents' appropriate grade range.

Scatterplot Matrix

The advantage that the scatterplot matrix provides is the ability to simultaneously display multiple two-dimensional scatterplots at once, making it easy to view data relationships between every combination of variables possible. Figure 31 demonstrates the display of five dimensions in a matrix where the redundant plots have been removed.

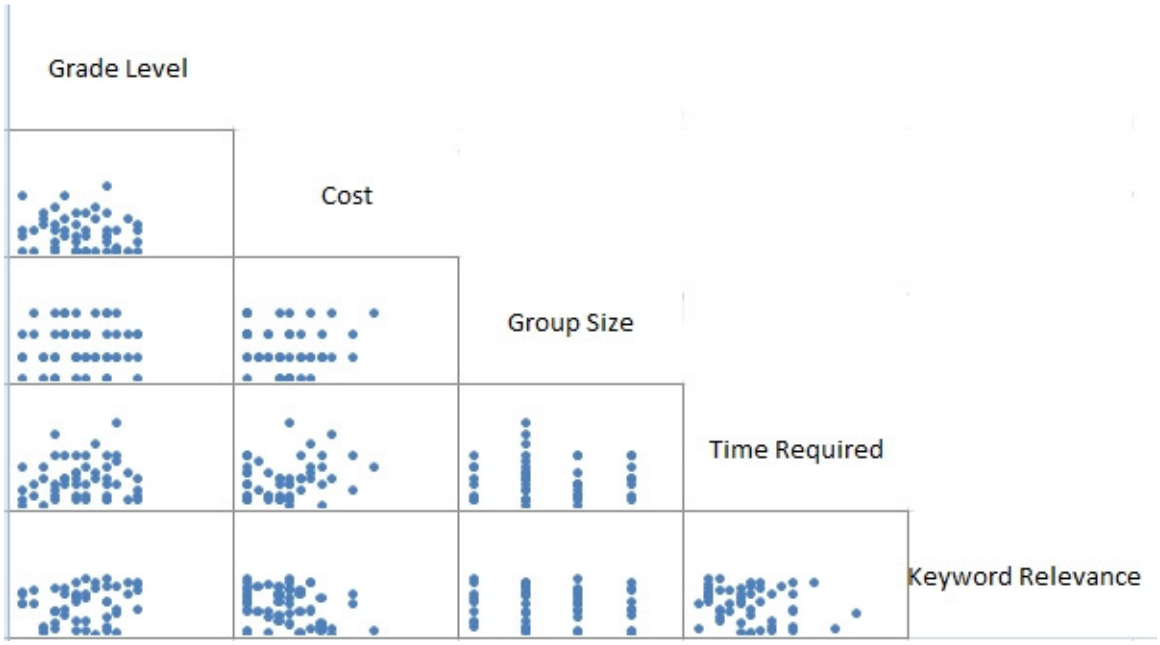


Figure 31: Triangular scatterplot matrix using TeachEngineering variables.

Though it is easy to compare any two variables, and there is a lot of information available for the user's interpretation, it is still a messy graphic that can take time to read. In fact, there is perhaps too much information to attempt to sort through. However, attempting to display five dimensions is a difficult task regardless; instead, let's take a look at a simpler scenario of three variables.

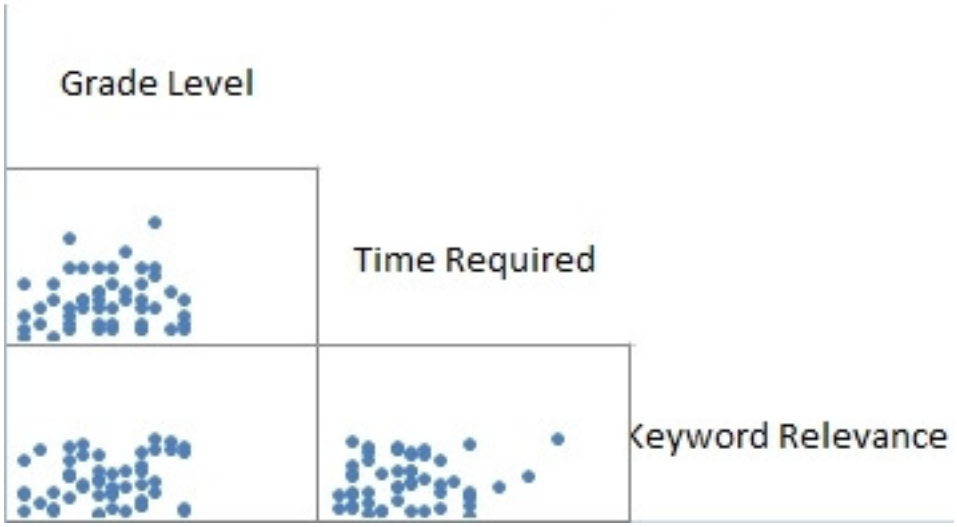


Figure 32: Three variables arranged in a scatterplot matrix.

Here we see that we are only comparing three variables. Even in this limited scenario, the scatterplot matrix provides for a more difficult graphic to read than encoding a third dimension, such as through the use of color, onto a regular scatterplot. Without the use of linking and brushing, the scatterplot matrix provides much less information as data items are difficult to compare. However, even when using these techniques, the scatterplot matrix does not seem to be an effective technique to visualize digital library information

Parallel Coordinates

At first glance, the parallel coordinates visualization technique appears very intriguing for the simultaneous display of multiple variables. However, when attempting to represent the multiple variables from TeachEngineering, a significant problem arises. As we have dealt with TeachEngineering variables so far, we have considered the use of both continuous (relevance) and discrete (grade level, cost etc.) data; even though *cost*

and *time required* are actually examples of continuous data, we are treating them as discrete by only considering them with integer values. One component of the parallel coordinates system is that all axes need to share a similar scale. Ignoring this limitation for now, we can draw what a parallel coordinates chart might look like, when implementing both continuous and discrete data.

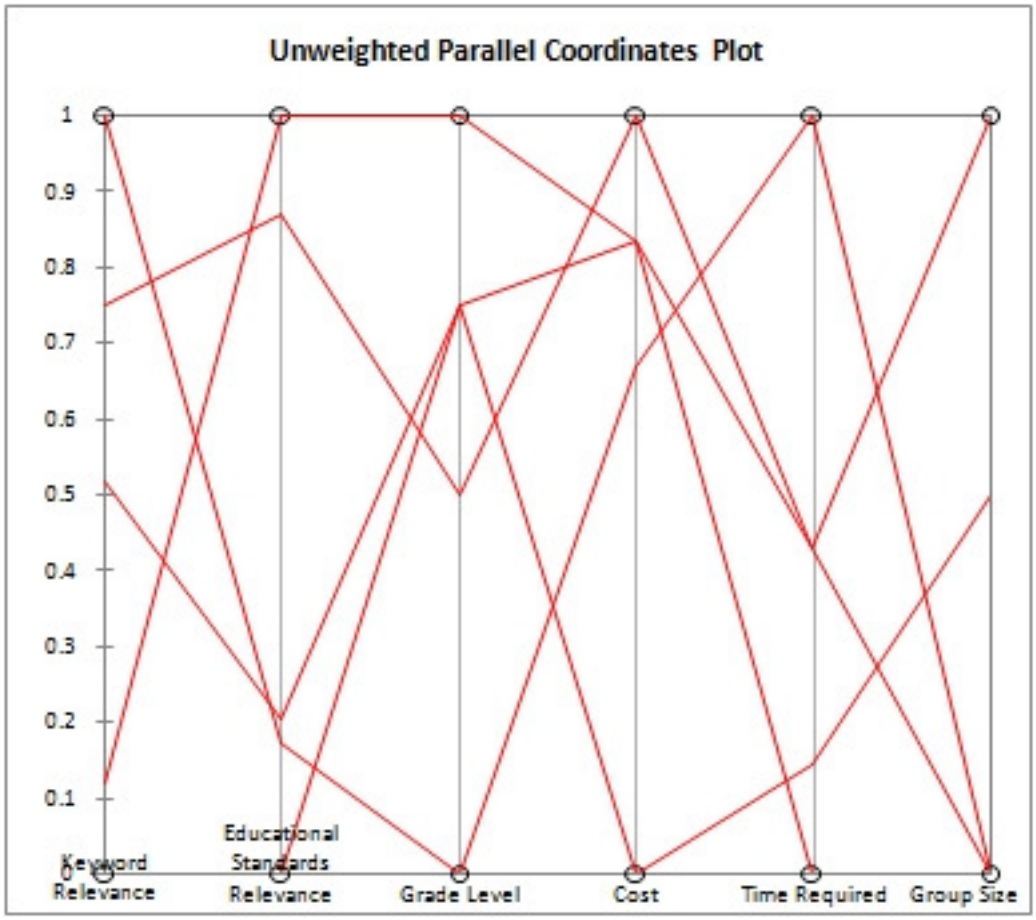


Figure 33: Data represented via improper parallel coordinates . Each vertical axis is specified by its own, unlabeled scale.

In Figure 33 different scales are associated with the various axes. For example, the *Grade Level* axis ranges from the low value of four to the upper limit of eight. The data used to generate this graphic can be found in Appendix A. Without proper labeling

of the axes, this chart is useless; despite containing a large amount of information that the user may be interested in, it is not easy to read. When looking for a sixth grade lesson plan, should the user really need to look for the midway point on that particular axis, and then near the top for the adjacent keyword relevance axis? The reason for parallel coordinates requiring similar scales on every axes is to prevent this sort of ambiguity. It becomes clear that in order to use the parallel coordinates system, all of our variables must be normalized through the use of a relevance score. For example, if the user is searching for sixth grade, then documents for sixth grade have perfect relevance, eighth grade is decent relevance, and kindergarten or twelfth grade would present low or no relevance at all. This relevance computation must then be applied to all of our variables, after which the information can be displayed on parallel coordinates; if this sort of conversion is not possible, then the parallel coordinates technique does not represent a good fit to visualize the data. Assuming that we have now converted all of our dimensions to a similar unit (relevance) and scale, we can see what this might look like.

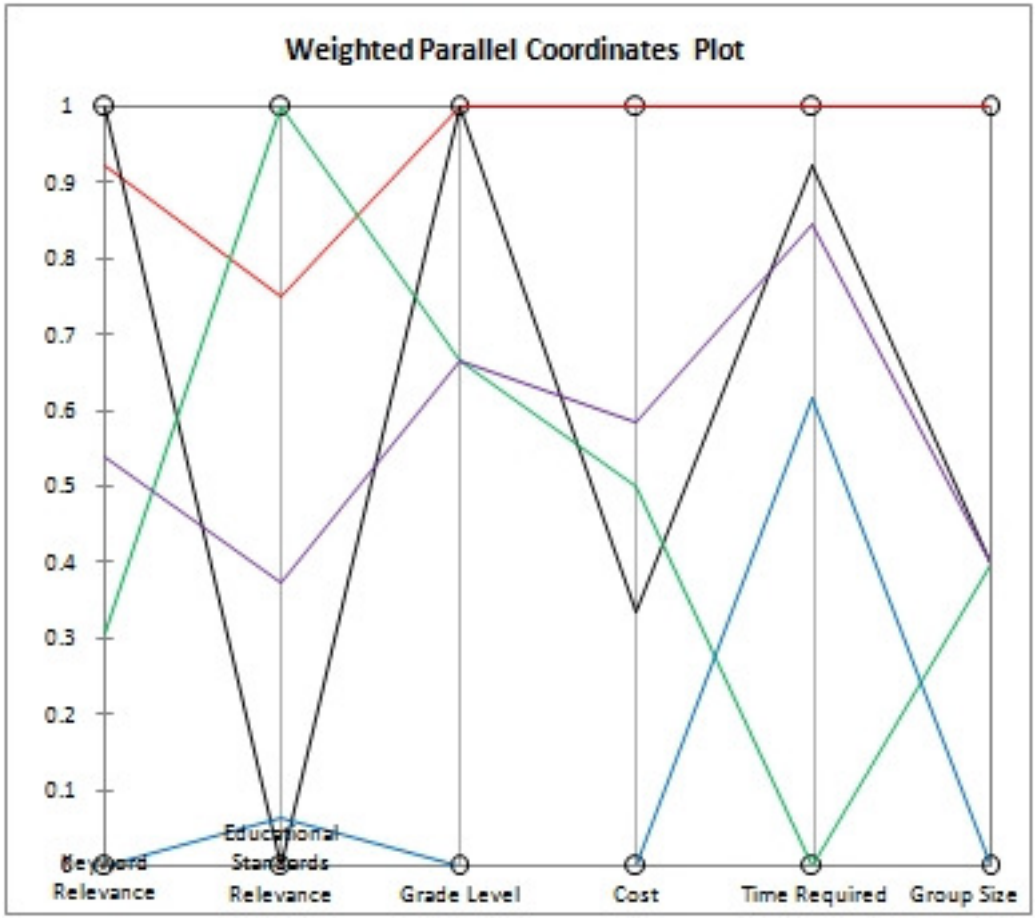


Figure 34: Parallel Coordinates representation with axes scales adjusted.

Based on this visual, we can immediately see that there is one document that represents nearly perfect relevance across every metric; other documents show strong relevance to several, but not all, of the search variables, and yet one line demonstrates weak relevance across all but one dimension. With this technique applied as in the example, it is easy to identify documents that may be interesting from the subset that is returned. However, there still remains the concern that too many data lines may be returned, which can turn the parallel coordinates into a confusing graphic of little value.

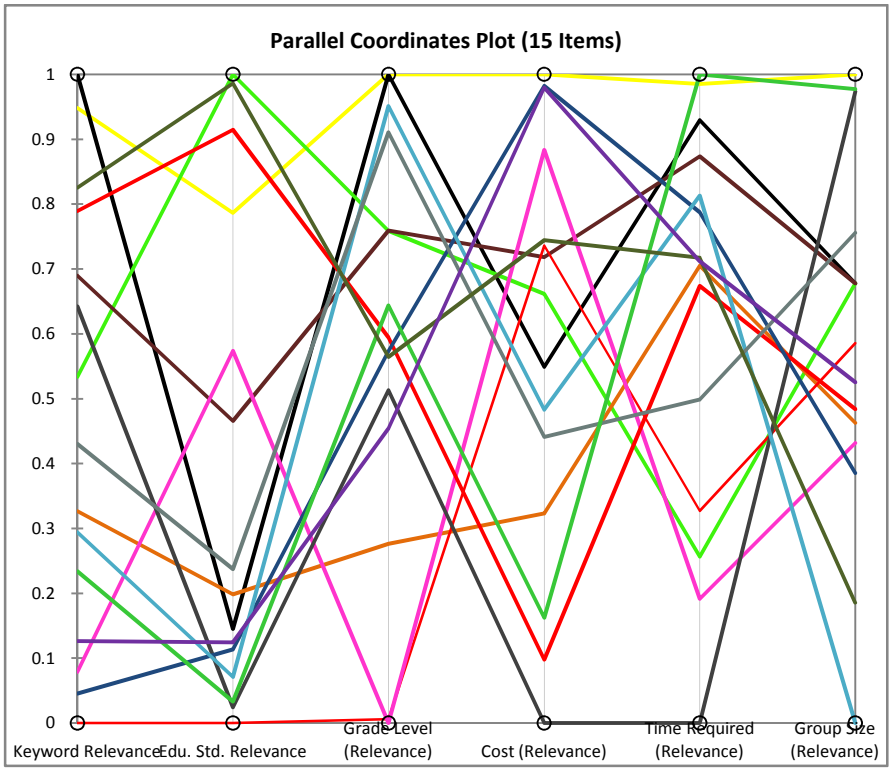


Figure 35: Fifteen data items plotted on a single parallel coordinates plot.

The use of color has been shown as an effective technique to aid against the problem of too many data lines. Representing each line with a different color helps to distinguish each line from one another and increases the readability of the graphic. User interaction can help to improve this interaction technique even further. For example, perhaps hovering over a specific data line could result in it being highlighted and cause a box to appear and display additional information about it.

As suggested by Becker and Cleveland (1987) user interaction may be used to select and delete data points from the chart. For example, if a particular parallel coordinates graph is overcrowded with lines, an individual data line could be selected and deleted, or removed from the graphic, thus allowing the user to focus on the results that are of best interest.

Although the use of the parallel coordinates technique requires all of the TeachEngineering variables to be scored via a relevance metric, it is an effective display technique to quickly visualize how a document rates according to all of the potential user variables. If the user only uses two or three variables, the parallel coordinates technique can be adjusted to only show those axes.

Starplot

The starplot, which is essentially the same thing as parallel coordinates but arranged with radial axes, contains the similar limitation of requiring each axis to share a similar scale. Again, this requires the conversion of previously discrete data (such as specifying a group size) into a continuous relevance score that indicates how well the parameters and document match; if this conversion is not possible, then this technique is not possible. Figure 36 shows the data line of a document that perfectly matches all of our search parameters. Figure 37 is the radial version of Figure 34 and contains several data lines that show strong, medium, and weak relevance across different dimensions.

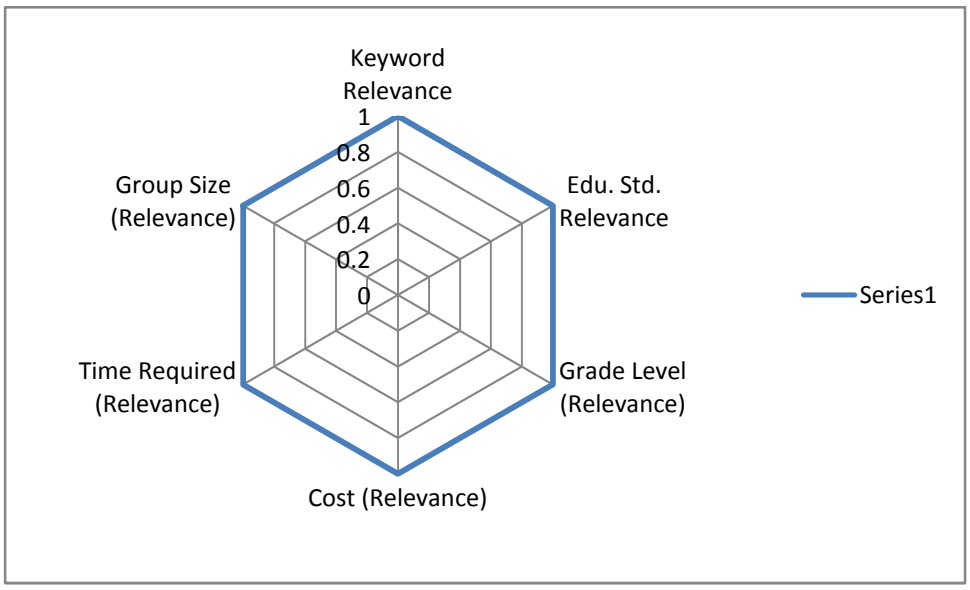


Figure 36: Star plot of a single document (series) representing perfect fit across six dimensions.

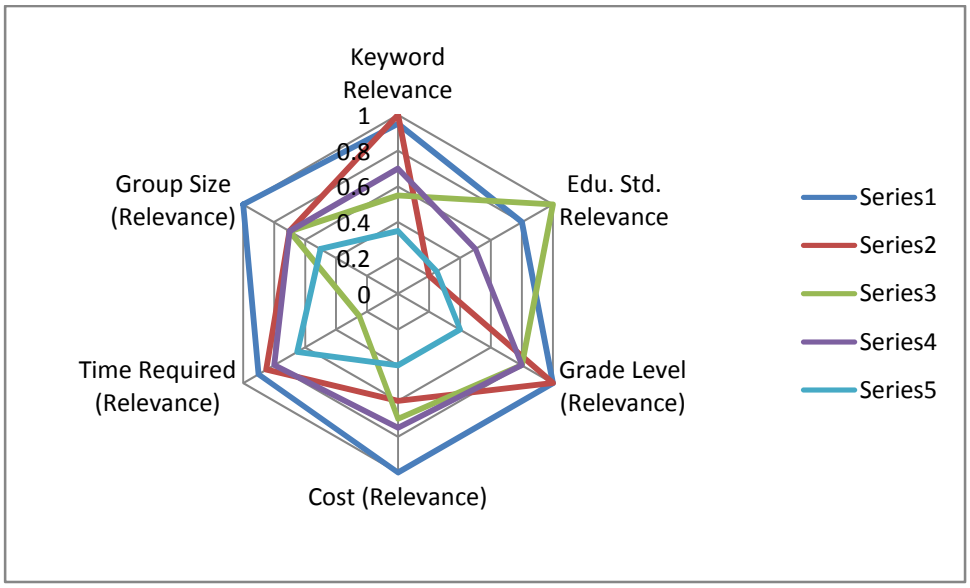


Figure 37: Star plot containing data of five documents (data series).

These starplots provide interesting graphics. Although the information provided is identical to parallel coordinates, the arrangement of axes around a radius makes this graphic seem more familiar, as it is similar looking to a normal set of orthogonal axes. For displaying a single data line, the starplot makes for a useful visualization. However,

the graphic quickly becomes crowded when additional data lines are entered on the same axes.

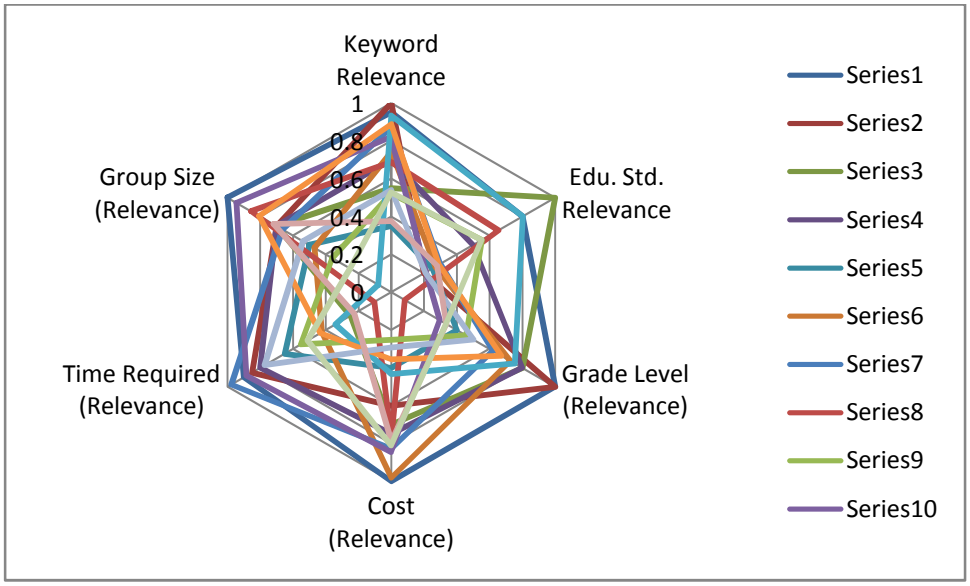


Figure 38: Overcrowded star plot containing fifteen document data lines (series).

Figure 38 contains the same data as Figure 35, but is represented in a radial form. Since both graphs demonstrate the same information, the choice of which approach to use could be considered user preference. However, I hypothesize that the parallel coordinates technique may be easier to read; since each of the axes are located directly horizontal to each other, it is easier to follow a data line across all dimensional axes and compare the attributes of different documents with each other. Clutter quickly becomes a problem when graphing multiple data lines onto a single star plot.

In identifying ways to apply the starplot, one idea that was generated consists of combining the starplot with the traditional list of search results, as seen in Figure 39.

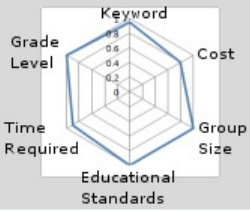
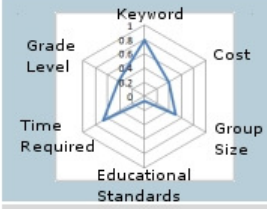
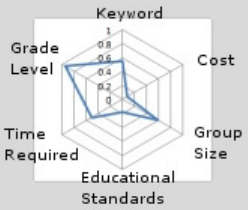
	Title ▲	Summary	Grade	Edu. Standards	Time	Associated Curriculum
	Work and Power: Waterwheel	Investigating a waterwheel illustrates to students the physical properties of energy. They learn that the concept of work, force acting over a distance, differs from power, which is defined as force a... ... more	7 (6-8)	**Colorado:** ◆ algebra 2 ◆ computations 6 ◆ interrelationships 5 ◆ number sense 1 ◆ physical science 2.3	50 minutes	**Subject Areas:** 2 **Curricular Units:** 1 **Activities:** 1 Details
	Many Paths	Students explore the composition and practical application of parallel circuitry, compared to series circuitry. Students design and build parallel circuits and investigate their characteristics, and apply Ohm's law.	4 (3-5)	**Colorado:** ◆ algebra 2 ◆ interrelationships 5 ◆ physical science 2.2, 2.3	50 minutes	**Subject Areas:** 2 **Curricular Units:** 1 **Activities:** 1 Details
	Swinging on a String	Students explore how pendulums work and why they are useful in everyday applications. In a hands-on activity, they experiment with string length, pendulum weight and angle of release. In an associated... ... more	6 (5-7)	**Colorado:** ◆ algebra 2 ◆ connections 6 ◆ data analysis, statistics, and probability 3 ◆ interrelationships 5 ◆ measurement 5 ◆ number sense 1 ◆ physical science 2.1	50 minutes	**Subject Areas:** 2 **Curricular Units:** 1 **Activities:** 2 Details

Figure 39: Search results incorporating star plots.

In this approach, the use of the starplot may be a quick heuristic to understand the attributes of a specific document. However, this still requires the use of the standard list of search results, and simply displaying the attributes in a text columns (rather than visually) may be just as much if not more effective.

Parallel Star Glyph

The parallel star glyph, likewise originating from the parallel coordinates concept, also is restricted by the limitation that all axes much share a similar scale. Assuming that all data variables are converted into a measure of relevance, we can essentially project multiple star plots into three-dimensional space, resulting in a very intriguing graphic. It

should be noted that color is only being used to help distinguish each individual three-dimensional star glyph from each other, and does not indicate any sort of value.

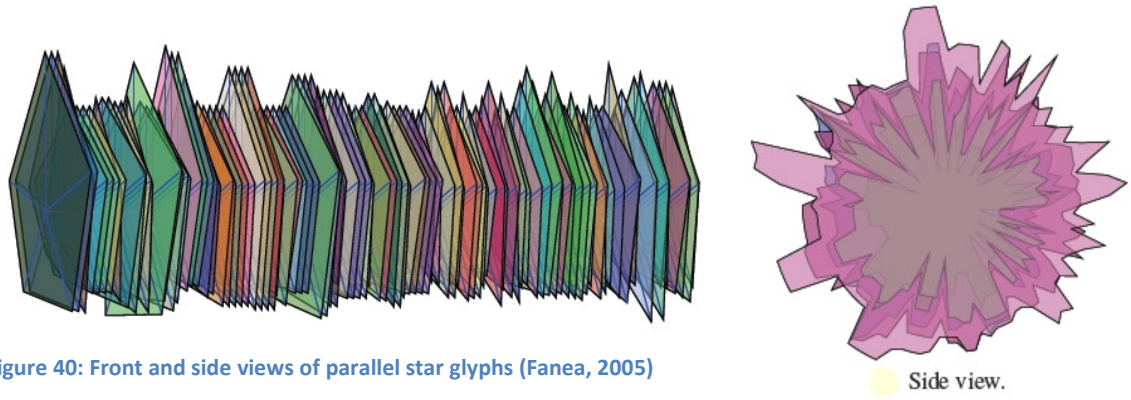
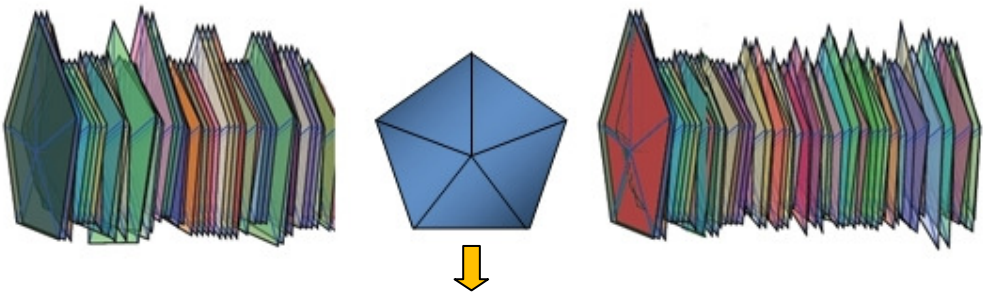


Figure 40: Front and side views of parallel star glyphs (Fanea, 2005)

Here we see that each starplot represents a distinct document in the search results. Looking at the starplots next to each other, I cannot help but draw parallels to the resemblance of a bookshelf, with the largest starplots representing the documents that have the highest relevance scores. From this angle, some of the axes are hidden or not easily viewable; allowing for user interaction to rotate the set of starplots around their radius could address this issue. Further, user interaction, as demonstrated in Figure 41, could be used to select and highlight a single document; a box could then appear to provide specific information about the selected document.



Title ▲	Summary	Grade	Edu. Standards	Time	Associated Curriculum
Work and Power: Waterwheel	Investigating a waterwheel illustrates to students the physical properties of energy. They learn that the concept of work, force acting over a distance, differs from power, which is defined as force a... ... more	7 (6-8)	**Colorado:** ◆ algebra 2 ◆ computations 6 ◆ interrelationships 5 ◆ number sense 1 ◆ physical science 2.3	50 minutes	**Subject Areas:** 2 **Curricular Units:** 1 **Activities:** 1 Details

Figure 41: Selecting a star glyph representing a document.

Although the visualization of the parallel star glyph is quite impressive, there is one glaring flaw that stands out – what happens if the documents are all pretty similar in size? Although the color differentiates one glyph from the next, if all of the documents are approximately the same size then that means they are all equally relevant (or irrelevant) and this visualization technique does not help much in finding the documents the user is likely to be looking for. One potential way to address this is to combine the relevance scores from each axis and arrange the glyphs in order from most to least relevant (based on the composite relevance score). Documents being nearly the same size may continue to be a concern, but arranging them by some measurement of total relevance helps to identify the results that best fit the search parameters.

A secondary concern is that there may be too many elements returned to make much sense of the glyphs – that they might be packed so close together to fit them all on

the screen, that the visualization becomes less useful. If the glyphs were arranged, as previously suggested, from greatest to smallest overall relevance, then perhaps the visualization could be limited to showing the first twenty documents as three-dimensional glyphs; a button could be used to generate the next set of glyphs, which would be slightly less relevant.

A last concern regards the technical feasibility of implementing this visualization technique, as it would need to be generated in real-time and viewable in a web browser to be a viable option. Regardless, the parallel star glyph represents a unique approach to representing documents that have relevance to search criteria.

Hyperbolic Browser

Of all the variations on the hierarchical tree structure, the hyperbolic tree is the visualization method that demonstrates the most promise for use with the TeachEngineering system. This technique allows for an essentially limitless amount of items to be displayed on the screen simultaneously. By interacting with the canvas, items of interest can be focused on; other items, while still visible on the screen, shrink in size. Overall, this allows for data to be focused on without losing perspective of other parts of the tree.

In looking to adapt the hyperbolic browser visualization, the primary issue to address is how the different variables will be represented within this tree structure. The hierarchical nature of the tree actually allows for a convenient way to map multiple dimensions – each level of the hierarchy represents a new dimension. For all of the discrete and categorical data types, this use of a hierarchical tree works extremely well.

For example, if our root node was *Science*, it might have child nodes of *Physics, Chemistry, Biology*, all of which represent subject types. Moving downward through the hierarchy, we can display other variables. Below a specific subject area a set of nodes such as *seventh grade* can be created in the tree; moving through the tree, we can find the branch that represents the topic, grade, cost, time and group size that we are looking for. Ideally there will be documents at the end of this branch that match all of our specifications; if not, then similar documents to what we are looking for will be located in nearby branches. The spatial layout of the documents in the hyperbolic tree helps to identify how they are similar or unlike each other.

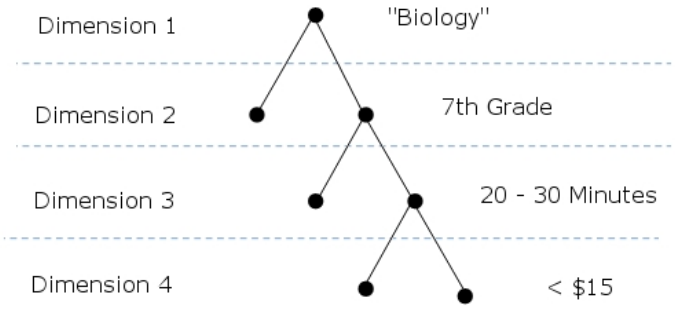


Figure 42: Mapping TeachEngineering variables into a tree hierarchy

However, when attempting to display continuous data, such as the attribute of *relevance* (to a keyword or educational standard), some issues arise. The notion of relevance cannot easily be mapped into a hierarchy level on the tree, and requires breaking it down into categories such as *strong* or *weak* relevance. Due to the nature of the hyperbolic tree, space between nodes dynamically changes as the user focuses on one area or another; thus, space cannot in itself be used as a factor in indicating relevance between a specific document and a topic, keyword or educational standard.

Other approaches to representing relevance on the hyperbolic tree can be found through the application of size, shape or color. The use of size, however, is discouraged,

as typically size indicates difference between levels of a hierarchy. It is common for the size of a parent node to be larger than that of a child, with the root node being the largest; however in other cases, the size two sibling nodes (at the same level on the hierarchy) might have different sizes, representing how many overall nodes descend from them. To an even further extent, the size of the nodes may change as they come into or out of focus. Overall, using node size to represent the level of relevance is not a good solution to this issue. Using icon shape to indicate a document's level of relevance, although a better method than using size, is still not ideal; again, using shapes requires categorizing the notion of relevance into groups such as *strong relevance, fair relevance,* and *weak relevance*. Color faces the similar limitation that in order to easily represent continuous data such as relevance, this data much be grouped into categorical units. In using color, one approach would be to again categorize the level of relevance (to a keyword or educational standard) by grouping documents into colors; for example, yellow may represent an excellent fit, whereas blue would be a poor fit. It should be noted that any number of colors could be used to categorize different levels of relevance. This leads to thoughts of an alternative approach that could use some sort of color gradient scheme to indicate relevance; although this

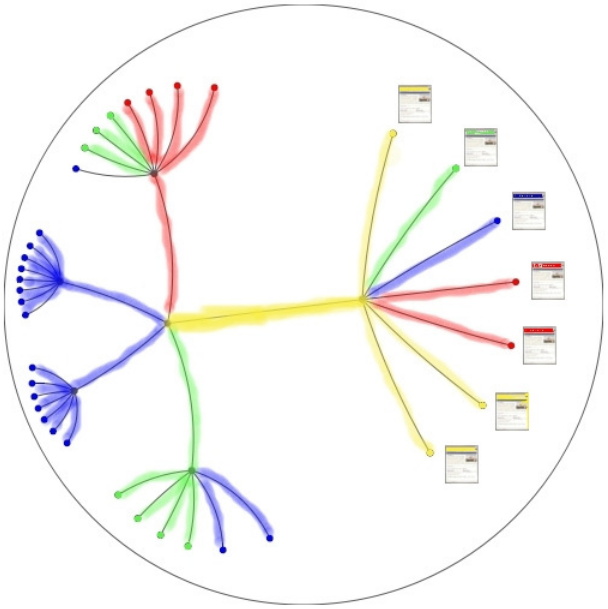


Figure 43: Using color brushing on tree branches to demonstrate document relevance.

might more accurately represent levels of relevance, this would undoubtedly make user interpretation more difficult.

One advantage of using color with a hyperbolic tree is that it can be easily seen, even when a document is out of focus and represented by a tiny icon. An additional proposed technique would be to paint the tree branch lines the color of their *most relevant* descendent; this would make sorting through the hyperbolic tree even easier, as clusters where lots of relevant documents exist could easily be spotted. It should be noted that the use of color and icon shape together may combine to provide even more information to the reader, or to make the graphic easier to interpret.

Because the hyperbolic tree is hierarchical and different dimensions are mapped to individual levels of the hierarchy, the question of how to order these dimensions becomes critical. Some users may consider keyword relevance the most important part of their search, and thus may want that to be represented by one of the top nodes in the hierarchy; another user may have a different opinion on how their data is arranged. The solution then, is to let the user decide how to map these dimensions to the hyperbolic tree through a simple ranking system. As seen in Figure 44, a situation could be established where the user orders the way the search variables are mapped to the hierarchy; if one dimension is not particularly of concern, it can be removed from the search dynamically. Overall, this approach to interacting with the data via the hyperbolic tree provides a great level of user interactivity and allows them to truly *search* through the virtual digital library space.

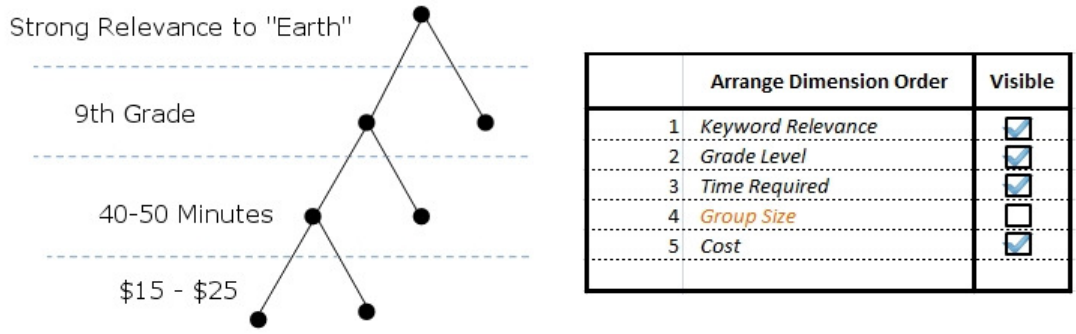


Figure 44: User-defined hierarchy structure of a hyperbolic tree using TeachEngineering variables.

In addition to the ability to display information well, the hyperbolic tree also renders extremely fast and can be generated within a Java applet in a web browser. Based on these observations, it is clear that the hyperbolic tree browser visualization technique has considerable merit in the discussion of how to display search results from TeachEngineering. Supporting this idea, it should be noted that this technique has been used before to map the hierarchies of digital libraries, including the Library of Congress as seen in Figure 45.

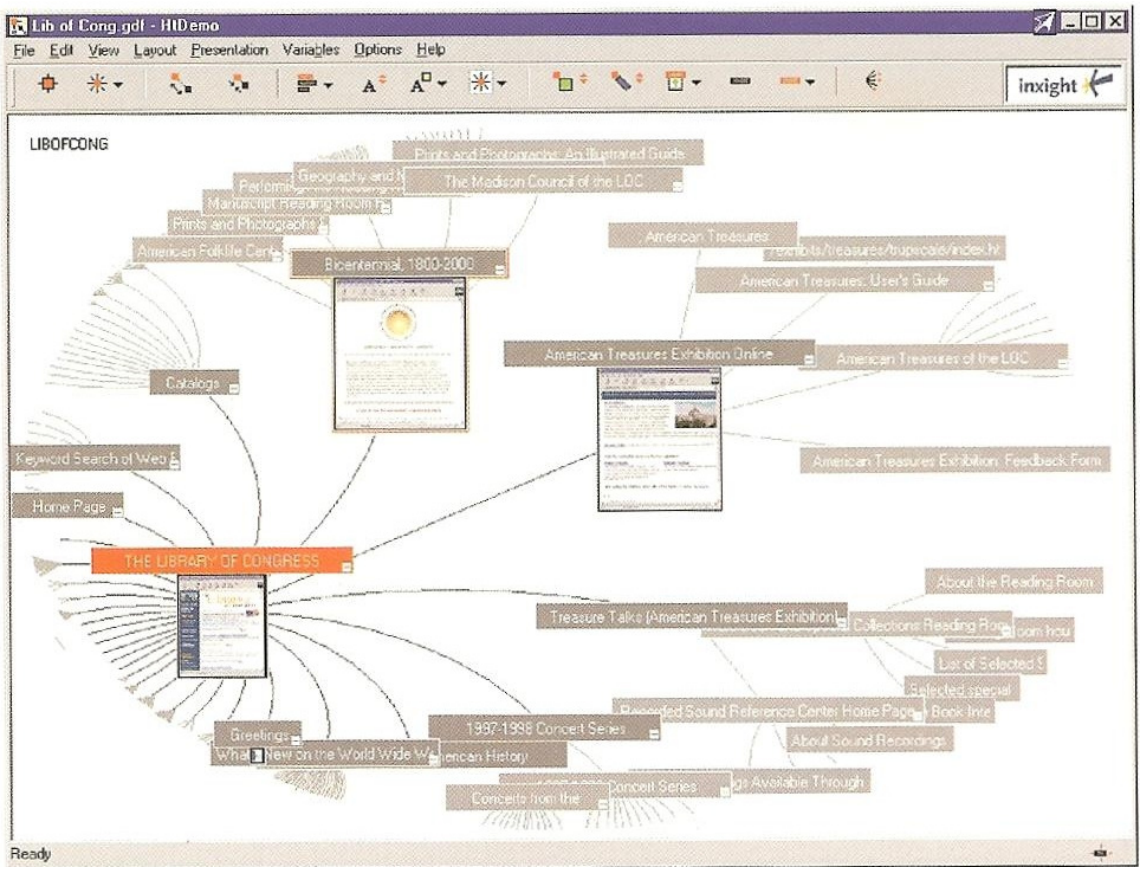


Figure 45: Library of Congress structure and items represented in the Hyperbolic Browser (Spence, 2001)

Additional Approaches

In addition to the previous techniques that have been analyzed with specific regards to TeachEngineering, there are other visualization methods mentioned throughout the paper that have not received this level of attention. The rationale for this is that they clearly do not represent ideal methods of visualizing a digital library and are better suited, perhaps even designed, for different tasks. For example, analysis of the Hyperslice technique was omitted because it was designed to display multivariable scalar functions, and not a collection of data items; additionally, it is very similar in appearance to scatterplot matrices. Andrews plots were left out for similar rationale, and the concept of using box-plots on parallel coordinates was only an identification of how these two

concepts could be combined. The TileBar technique, while extremely insightful, is not particularly applicable for the TeachEngineering system, as typically users are looking for subjects as opposed to specific information regarding multiple keywords; this visualization would be greatly suited towards search results for digital libraries of longer documents, such as academic journals. Although cone trees and treemaps did present ways to visualize multiple variables, the hyperbolic browser was identified as similar but superior method of visualizing the TeachEngineering data via a hierarchy.

In addition to these methods, there were a few other visualization techniques that were also removed from consideration early on. Pixel-Based approaches for visualizing data are typically designed for situations of high multidimensionality; in them, each attribute value is represented by the color of a single pixel on the screen. The worlds-within-worlds technique explores embedding three-dimensional graphs inside of other three-dimensional graphs, and doing this as many times as needed to represent all of the dimensions. Typically this technique holds some variables constant while manipulating others, and requires significant data interaction via a virtual hand; being able to generate this in an online, real-time manner would likely prove difficult, and users would need training to understand how to manipulate the system. Themescapes, which look like a map of terrain landscape, work best in clustering data and allowing the user to drill down for details; however, many of the advantages posed by this technique are limited by the difficulty of interacting with the system; ironically, queries occur after a themescape has already been built, rather than the other way around (Spence 183). Lastly, Kohonen maps are another method to cluster data, and bare a strong resemblance to tree maps.

These and even more visualization techniques were discovered and considered during the course of research, but it was determined that it was best to focus on the methods that demonstrated the greatest ability to help users identify useful documents in a digital library system.

DISCUSSION & RECOMMENDATIONS

Even after an extended analysis attempting to understand various visualization techniques and how they might be applied to the TeachEngineering digital library, no single method has emerged as a clear "solution". Each of the techniques that were specifically analyzed for TeachEngineering have their merits, advantages and limitations.; they all displayed pretty much the same information, albeit differently. In reality, the challenge of visualizing information may be partially due to differences between users. It is a bit ironic that everyone who interacts with a given system is grouped together as part of the collective *users*, when individually these people may have very little in common. One user may know exactly what they are looking for and be familiar with a lesser-known visualization technique such as the parallel coordinates plot; another user may have an idea, but be unsure of what they are looking for and not feel comfortable with using anything outside of a standard list. This is probably why there is no definitive solution to the problem.

Although it has been essentially ignored for all of the previous discussion and analysis on how to visualize the TeachEngineering digital library, it is important to note the idea of visualizing data with a simple list. For situations where data is univariate this method is ideal; after all, what exactly is a list other than a one-dimensional, vertical axis along which search results are sorted by the measure of relevance. In most information systems, this type of search response is both the default and only visualization method available. Google would be such an example, where regardless of the query, a list of results is returned, ranked via relevance. However, in contrast to Google, with an

essentially unlimited number of pages to sort through, a structured system such as TeachEngineering has limited and specific information to return to the user. Because the type of information is fairly specific (lesson plans), users interacting and exploring the data through a more advanced visualization technique may find answers to questions they did not know that they had. Regardless, for some situations, including the TeachEngineering *simple search*, the best method of displaying search results can be found in the standard relevance-sorted list.

By no means, however, is this conceding the notion that the standard list is the best way to visualize all data, as that is certainly not the case. Indeed, for multivariate search situations, it makes sense to display results in a manner that is intuitive and fitting to the number of parameters being searched upon. For two or three parameters, the use of a scatterplot to visualize document retrieval appears to be an effective option. However, when attempting to visualize four or more variables, this technique appears to lose its effectiveness and other approaches such as the parallel coordinates plot or hyperbolic tree may make more sense.

When it comes to having trouble identifying exemplary methods of visualizing information repositories, I am by no means alone. About data repositories, Chen and Wang (2001) describe “This information is completely abstract, so the data must be mapped into a physical space representing the relationships contained in the information as accurately and efficiently as possible. This lets observers understand, through spatial relationships, the correlations in the library. However, finding a good spatial representation of the information remains a challenge”. Although not particularly associated with digital libraries, popular visualization tools such as InfoVis and Spotfire,

neither of which are web-based, avoid addressing this issue and instead visualize multiple techniques (scatterplots, parallel coordinates, etc.) simultaneously.

My exploration of information visualization techniques, and how they might be applied to the TeachEngineering digital library, has resulted in finding multiple potential solutions – specifically scatterplots, parallel coordinates, and hyperbolic trees. Each technique has advantages and weaknesses, and seems to have a situation where it would be the ideal visualization method; however, all of these examples were created with hypothetical data, and may have been unintentionally designed to allow a visualization technique to appear more promising than it actually is.

Regardless, scatterplots remain a very strong technique for displaying the results of search queries; documents deemed relevant can all be found within a short spatial distance from each other. As evidenced in our examples, the use of color and icon shape can be extremely useful in embedding additional dimensions into the technique, although attempting to use the *icon size* had a less positive outcome. Even limitations of the method, such as concerns about the plot becoming overcrowded, can be reduced via simple techniques. Despite seeming like a very ordinary, perhaps even boring, visualization technique, the scatterplot, as we have applied it, shows that it can be used to display search results where there are two or more parameters.

The use of parallel coordinates in its many forms (including star plots), although intriguing, appear to be less effective than some of these other techniques. Parallel coordinates are excellent at displaying all of the information on a single plane, in a readable format. However, attempting to use traditional parallel coordinates plots, it becomes clear that even a small number of data lines can make the graphic difficult to

read and interpret. Star plots fare even worse in the problem of over-plotting, but do have a few potential uses, such as displaying the attributes of single documents as was shown in the mock search results page of Figure 39. Although parallel coordinates represent an excellent way to directly compare different variables to each other, they are probably best utilized to compare data trends of an entire dataset and not to pick out a few single, relevant search results.

If forced to choose one visualization technique to recommend for further exploration regarding this manner, it would certainly be the hyperbolic tree. Not only does the hyperbolic tree represent hierarchical data, which library information is typically represented as, but it has a dynamic and intuitive interface. Further, the hyperbolic tree allows for part of the data to be focused on, while still maintaining a view of all the rest of the data, which makes it less likely that the user will become lost in the data. As Figure 45 demonstrated, I am clearly not the first person to realize the potential of using the hyperbolic tree to visualize the structure of digital libraries. However, it should be noted that my approach, to visualize search results as opposed to the entire library, is unique; whether this visualization technique holds up under advanced queries, utilizing multiple variables, remains a key question.

Although this study attempted to qualitatively compare these different techniques, the use of a quantitative test would help in determining the effectiveness of each visualization technique. Future steps would include designing an experiment that uses human testers to gauge the effectiveness of each method. This could be accomplished in several ways. In its simplest form, this could involve showing the user examples of the visualization outputs and seeing what they are able to identify. Although there are several

steps in between, a more complex test could involve helping the user build a query, and then having them answer questions; their answers could be used to determine if they actually found the *best* search results, and the time it takes the user to complete the task could be recorded. Upon completion, users could be given a qualitative test to see how confident they were in their responses, and if exposed to multiple visualization techniques their preferences can be recorded as well. Depending on how the study is designed, it may be necessary to have some interactive element that mimics the functionality of a real implementation, particularly for the hyperbolic tree to be tested. Further experiments could help to answer which of these various visualization techniques is really the most effective in helping the user to find the search results that they are seeking.

As has been demonstrated, the challenge of visualizing digital libraries is not a trivial task. Many potential solutions exist, but none have been refined to the point where they are considered the overwhelmingly superior method. Much of this may have to do with the users, who have various needs and expectations in searching for lesson plans. Regardless, surveying the available visualization techniques has led to the identification of several candidate solutions that demonstrate promise in visualizing TeachEngineering search queries. Although additional empirical research is needed, the further development of these methods should result in increasingly interactive and dynamic information visualization techniques that allow the users to find the documents that they want and are looking for.

BIBLIOGRAPHY

Andrienko, G., & Andrienko, N. (2004). Parallel Coordinates for Exploring Properties of Subsets. *Second International Conference on Coordinated and Multiple Views in Exploratory Visualization, 2004. Proceedings.*. 93-104.

Becker, R.A., & Cleveland, W.S. (1987). Brushing scatterplots. *Technometrics*. *29*, 127-142.

Calitz, A., & Munro, D. (2001). Representation of Hierarchical Structures in 3D Space. *Proceedings of the 1st international conference on Computer graphics, virtual reality and visualisation*. 59-64.

Card, S., Mackinlay, J., & Shneiderman, B. (1999). *Readings In Information Visualization: Using Vision To Think*. Morgan Kaufmann Publishers.

Chen, J.X., & Wang, S. (2001). Data Visualization: Parallel Coordinates and Dimension Reduction. *Computing in Science & Engineering*. *3*, 110-113.

Chiricota, Y., Jourdan, F., & Melancon, G. (2004). Metric-Based Network Exploration and Multiscale Scatterplot. *IEEE Symposium on Information Visualization, 2004. INFOVIS 2004.*. 135-142.

Elmqvist, N., Stasko, J., & Tsigas, P. (2007). DataMeadow: A Visual Canvas for Analysis of Large-Scale Multivariate Data. *IEEE Symposium on Visual Analytics Science and Technology, 2007.*. 187-194.

Elmqvist, N., Stasko, J., & Tsigas, P. (2008). DataMeadow: A Visual Canvas for Analysis of Large-Scale Multivariate Data. *Information Visualization*. *7*, 18-33.

Engineering Statistics Handbook, (2006). Star Plot. Retrieved August 11, 2008, from NIST/SEMATECH e-Handbook of Statistical Methods Web site: http://www.itl.nist.gov/div898/handbook/

Fanea, E., Carpendale, S., & Isenberg, T. (2005). An Interactive 3D Integration of Parallel Coordinates and Star Glyphs. *IEEE Symposium on Information Visualization, 2005. INFOVIS 2005*. 149-156.

Friendly, M. (2008). "Milestones In The History of Thematic Cartography, Statistical Graphics, and Data Visualization". Retrieved Aug 9, 2008, from http://www.math.yorku.ca/SCS/Gallery/milestone/milestone.pdf

Hearst, M. (1995). TileBars: Visualization of Term Distribution Information in Full Text Information Access. *Proceedings of the SIGCHI conference on Human factors in computing systems*. 59-66.

Hearst, M., & Pederson, J. (1996). Visualizing Information Retrieval Results: A Demonstration of the TileBar Interface. *Conference on Human Factors in Computing Systems*. 394-395.

Inselberg, A. (1997). Multidimensional Detective. In S. Card, J. Mackinlay & B. Shneiderman (Eds.) *Readings In Information Visualization: Using Vision to Think* (pp.107-114). San Francisco, CA: Morgan Kaufmann Publishers

Jianxin, C., Wenxue, H., & Haibo, G. (2007). Research on Optimization of Multivariate Information Feature Extraction Based on Graphical Presentation. *The Eighth International Conference on Electronic Measurement and Instruments, 2007*. 321-324.

Johnson, B. & Shneiderman B. (1991). Tree-Maps: A Space-Filling Approach to the Visualization of Hierarchical Information Strructures. In S. Card, J. Mackinlay & B. Shneiderman (Eds.) *Readings In Information Visualization: Using Vision to Think* (pp.152-159). San Francisco, CA: Morgan Kaufmann Publishers

Kromesch, S., & Juhasz, S. (2005). High Dimensional Data Visualization. *6th International Symposium of Hungarian Researchers on Computational Intelligence*. 1-12.

Lamping, J. & Rao R. (1995). The Hyperbolic Browser: A Focus+Context Technique for Visualizing Large Hierarchies. In S. Card, J. Mackinlay & B. Shneiderman (Eds.) *Readings In Information Visualization: Using Vision to Think* (pp.382-408). San Francisco, CA: Morgan Kaufmann Publishers

Lamping, J., & Rao, R. (1996). Viusalizing Large Trees Using The Hyperbolic Browser. *CHI '96: Conference Companion on Human Factors in Computing Systems*. 1-2.

Lanzenberger, M., Miksch, S., & Pohl, M. (2005). Exploring Highly Structured Data: A Comparative Study of Stardinates and Parallel Coordinates. *Ninth International Conference on Information Visualisation, 2005*. 312-320.

Lee, M., Reilly, R., & Butavicius, M. (2003). An Empirical Evaluation of Chernoff Faces, Star Glyphs, and Spatial Visualizations For Binary Data. *Proceedings of the 2003 Asia-Pacific symposium on Information visualisation* . 1-10.

Liu, W., Meng, H., Hong, W., Wang, L., & Song, J. (2007). A New Method for Dimensionality Reduction based on Multivariate Feature Fusion. *Proceedings of the 2007 IEEE International Conference on Integration Technology*. 108-111.

Mijksenaar, P. (1997). *Visual Function: An Introduction to Information Design*. New York, NY: Princeton Architectural Press.

Moustafa R., & Wegman E. (2006). "Multivariate Continuous Data – Generalizations of Parallel Coordinates". *In: Unwin, A., Theus M., Hofmann, H.* (Eds.), *Graphics of Large Datasets: Visualizing a Million,* Springer*:* 143–156*.*

Pillat, R., Valiati, E., & Freitas, C. (2005). Experimental Study on Evaluation of Multidimensional Information Visualization Techniques. *Proceedings of the 2005 Latin American conference on Human-computer interaction*. 20-30.

Robertson, G., Mackinlay, J. & Card, S. (1991). Cone Trees: Animated 3D Visualizations of Hierarchical Information. *Proceedings of the SIGCHI conference on Human factors in computing systems: Reaching through technology*. 189-194

Spence, R. (2001). *Information Visualization*. Harlow, Essex, UK: ACM Press.

Statistics Toolbox - Visualizing Multivariate Data Demo. Retrieved August 11, 2008, from MathWorks Web site: http://www.mathworks.com/products/statistics/demos.html?file=/products/demos/shipping/stats/mvplotdemo.html#4

TeachEngineering: Resources for K-12. (n.d.). Retrieved August 9, 2008, Web site: http://teachengineering.org/about.php

Thai, V., Handschuh, S., & Decker, S. (2008). Tight Coupling of Personal Interests with Multi-dimensional Visualization for Exploration and Analysis of Text Collections. *2008 12th International Conference Information Visualisation*. 221-226.

Tominski, C., Abello, J., & Schumann, H. (2004). Axes-Based Visualizations with Radial Layouts. *Proceedings of the 2004 ACM Symposium on Applied Computing*. 1242-1247.

Tufte, E. (1983). *The Visual Display of Quantitative Information* . Cheshire, CT: Graphics Press.

Van Wijk, J., & Van Liere, R. (1993). Hyperslice: Visualization of Scalar Functions of Many Variables. *In Proceedings of IEEE Visualization '93*. 119-125.

Voigt, R. (2002). *An Extended Scatterplot Matrix and Case Studies in Information Visualization*. Unpublished Masters thesis. Virtual Reality and Visualization Research Center, Vienna, AU. Retrieved August 10, 2008, Web site: http://www.vrvis.at/via/resources/DA-RVoigt/index.html

Wang, H., Wang, C., Liu, K., Meng, B., & Zhou, D. (2004). VISDM-PC: A Visual Data Mining Tool Based On Parallel Coordinate. *Proceedings of the Third International Conference on Machine Learning and Cybernetics*. 1244-1248.

Wegman, E. (1990). Hyperdimensional Data Analysis Using Parallel Coordinates. *Journal of the American Statistical Association*. *85*, 664-675.

Weiss-Lijn, M., McDonnell, J.T., & James, L. (2002). An Emperical Evaluation of the Interactive Visualization of Metadata to Support Document Use. In K. Borner & C. Chen (Eds.) *Visual Interfaces to Digital Libraries* (pp. 50-64). New York, NY: Springer-Verlag Berlin Heidelberg

Wong, P., Crabb, A., & Bergeron, R.D. (1996). Dual Multiresolution HyperSlice For Multivariate Data Visualization. *IEEE Symposium on Information Visualization '96*. 74-75.

Yang, J., Patro, A., Huang, S., Nehta, N., Ward, M., & Rudensteiner, E. (2004). Value and Relation Display for Interactive Exploration of High Dimensional Datasets. *IEEE Symposium on Information Visualization, 2004. INFOVIS 2004.* 73-80.

APPENDIX: DATA TABLES

The following table contains the data used to generate the Figure 31 scatterplot matrix. These values are hypothetical and were created to reflect what a subset of TeachEngineering data might look like. The relevance score was randomly generated:

Grade Level	Cost	Group Size	Time Required	Keyword Relevance
1	0	1	50	0.741518489
1	15	2	20	0.218590777
1	20	3	15	0.989423185
1	50	2	30	0.596481513
2	0	3	35	0.098463388
2	20	4	25	0.123085389
3	25	2	40	0.514497823
3	30	1	50	0.930152717
3	35	2	15	0.952526493
4	0	4	20	0.173621345
4	10	3	25	0.264203105
4	5	2	30	0.255666049
4	20	1	35	0.133698167
4	25	1	60	0.793096778
4	40	2	80	0.552510531
5	50	3	60	0.32232985
5	30	4	40	0.843858567
5	20	3	35	0.002628399
5	15	4	40	0.157601154
6	0	3	45	0.508517219
6	0	2	20	0.183492678
6	15	1	25	0.344683952
6	25	2	40	0.926929343
6	35	3	35	0.305757933
6	0	4	60	0.172489312
6	10	2	45	0.407134811
7	20	1	20	0.438163116
7	0	3	25	0.524326535
7	15	2	35	0.021371607
7	0	3	60	0.472956037
7	35	2	50	0.50130894

8	30	2	70	0.988033519
8	40	4	40	0.565980066
8	0	2	45	0.225806559
9	15	1	20	0.702619652
9	10	3	25	0.284120362
9	0	4	35	0.886371851
9	15	2	40	0.180613813
9	20	1	20	0.706449304
9	35	2	60	0.320638742
9	60	4	50	0.708862311
10	5	2	55	0.007783443
10	0	3	35	0.129968684
10	20	2	90	0.520882337
10	0	4	60	0.014530689
11	0	3	20	0.470722413
11	30	2	45	0.111771054
12	20	1	40	0.366725025
12	25	3	30	0.308228477
12	0	3	25	0.730446305
12	10	2	20	0.243307189

The following table consists of the data used to populate the Figure 33 parallel coordinates graphic:

Keyword Relevance	Edu. Std. Relevance	Grade Level	Cost	Time Required	Group Size
1	0.405315011	4	20	50	2
0.32647179	0.801609583	6	30	30	4
0.582854002	0.264214023	7	0	20	3
0.179369964	0.087845014	7	25	15	2
0.327490057	1	8	25	30	2

After modifying the values of the non-continuous data types (grade level, cost, time required and group size) to fit the notion of relevance and a scale of zero to one, the following data was used to populate parallel coordinate Figures 34 and starplot Figure 37:

Keyword Relevance	Edu. Std. Relevance	Grade Level	Cost	Time Required	Group Size
0.95	0.8	1	1	0.9	1
1	0.2	1	0.6	0.85	0.7
0.55	1	0.8	0.7	0.25	0.7
0.7	0.5	0.8	0.75	0.8	0.7
0.35	0.25	0.4	0.4	0.65	0.5

The following table contains the values used to populate parallel coordinate Figure 35 and starplot Figure 38:

Keyword Relevance	Edu. Std. Relevance	Grade Level (Relevance)	Cost (Relevance)	Time Required (Relevance)	Group Size (Relevance)
0.95	0.8	1	1	0.9	1
1	0.2	1	0.6	0.85	0.7
0.55	1	0.8	0.7	0.25	0.7
0.7	0.5	0.8	0.75	0.8	0.7
0.35	0.25	0.4	0.4	0.65	0.5
0.740344833	0.268651711	0.726616337	0.98102	0.422386732	0.4678575
0.87207299	0.297815263	0.620527627	0.82667	0.976543196	0.6636074
0.686443593	0.653730264	0.082029843	0.77238	0.104323596	0.8554118
0.521527017	0.553419295	0.456907286	0.25296	0.551682665	0.3606238
0.819806747	0.21117753	0.296114984	0.84571	0.884463241	0.9446311
0.933457313	0.802737405	0.753602438	0.43219	0.340923723	0.0801841
0.888755659	0.290871431	0.672195877	0.35502	0.43363296	0.8066599
0.540011546	0.207658041	0.502407755	0.29264	0.772521359	0.5382718
0.374712866	0.279989181	0.335784682	0.78751	0.227079119	0.7189201
0.527633688	0.55194799	0.335839372	0.80989	0.510407102	0.2511333