A Simple 3D Visual Text Retrieval Interface

Ben Houston[†], Zack Jacobson[‡]

[†]Carleton University
Ottawa, Ontario, Canada
bhouston@chat.carleton.ca

[‡]Consulting and Audits Canada
Ottawa, Ontario, Canada
zack.jacobson@cac.gc.ca

Abstract

3D visual text retrieval interfaces are currently a fringe topic of interest. This paper suggests that 3D visual interfaces are fringe topics because of the complexity and abstract nature of many of the previous attempts in this area. In order for 3D visual interfaces to become mainstream this paper proposes that they must be concrete in the metaphor they employ, simple to use, and appear familiar to the average web user. In a set of two prototypes, AutoViz and NetViz, an attempt is made to fulfill these requirements. The prototypes represent the query terms one plane and the documents in a second parallel plane. A spring system is used to cluster the terms and documents into a meaningful structure. Document profiles are displayed to the user as they move the mouse on top of document representations.

Keywords: information visualization, visual query languages, search interfaces, information retrieval.

1   Introduction

Very little work has been done on 3D visual search interfaces in the past. The reasons that work in this area has been stifled is two fold. First, many of the projects took as a primary focus the exploration within the result space. Exploration is a secondary function of a search interface. Examining the results for the needed document is the primary goal, only after the needed document is believed not to be in the results is exploring and interacting with the result space an issue. Basically, exploration is not something that the average user will do on a regular basis. Second, many visual interfaces are too complex, feature overloaded and overwhelming for the average user. For the average user simplicity and familiarity are important and all too often ignored.

The goal of this 3D visual search interface is to be concrete in its representations and simple in its layout mechanisms and interactions. In addition the interface is meant to interact with existing search engines. Thus unlike some other visualizations this project does not require a specially processed document database.

The two prototypes, NetViz and AutoViz, discussed in this paper are both preliminary explorations into this area. More work is needed.

2   Related Work

The most interesting 3D visual search interface is the Document Explorer [Fowler *et al.* 1996, Fowler *et al.* 1997]. It is very effective in showing the semantic relationships between various documents in a set through spatial arrangements. Unfortunately, its 3D spatial layout of the documents leads to the appearance of complexity. The majority of the documents in the visualization are obstructed by other documents in a sea of overlapping text. Also the visual information in the interface is spread over various windows and views suggesting to the average user that the interface is very complex and difficult to learn.

Another interesting, though less flashy, information retrieval visualization is VQuery system [Jones 1998]. This system uses a direct manipulation interface based on Venn-like diagrams. In this system the user is able to create oval and associate then with particular terms. These ovals can then be placed in overlapping combination that imply specific Boolean search queries.

The interface described in this paper was developed independently of the two similar systems mentioned above.

3   Visualizing Search Results

The goals of this 3D visual search interface are concrete representations and simplicity. In a search there is two sets of data: the query and the results. The query in most text retrieval application is a set of terms. The results are usually a subset of total documents in the index.

3.1 Basic Layout

We chose to represent each of the terms in the query string by a sphere. A cylinder represents each of the documents in the results.

To make effective use of 3D space the terms and the documents are laid out in two parallel planes. The upper plane contains the terms while the lower plane contains the documents as shown in Fig. 1. This arrangement allows for an easy view of the whole topology of the results without excessive manipulation of the view.
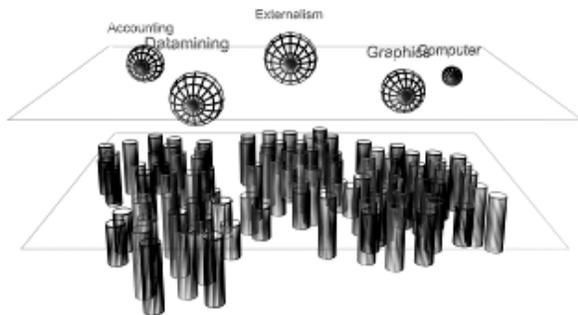

Figure 1. Schematic of the term and document planes used in visualizing search results.

Although not apparent in the schematic shown in Fig. 1 the documents will cluster underneath the terms that they contain. The clustering algorithm is an iterative energy minimizing spring system similar to Kamada and Kawai [Kamada and Kawai 1989]. The terms themselves are free to move as well and thus they will move into an arrangement such that terms that co-occur in the results will be located near each other. This results in a very Venn-like representation of the results as shown in Fig 2.
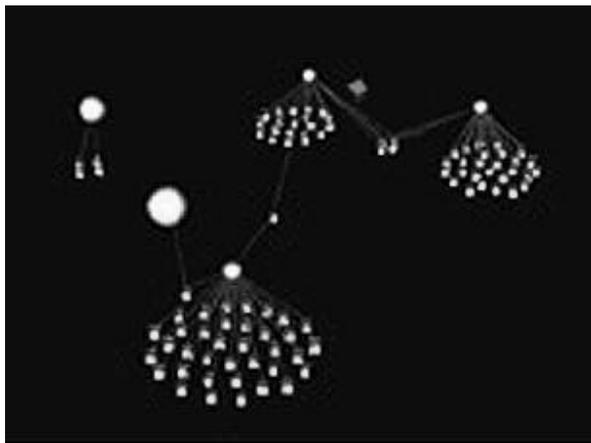

Figure 2. AutoViz screen. A set of documents is clustered around a series of terms in a Venn diagram like fashion.

3.2 Use of Color and Proportion

Each of the terms has a brightness/intensity that is relative to the information content of that particular term. The information content of a term is greater the more rare a term is. In other words, terms that are more obscure have a more specific meaning and are thus more important in narrowing down the search space. The size of a term is also relative to its information content.

In the current prototypes all the terms are colored yellow. In the future with the help of a semantic word network it could be useful to color terms according to how separated they are in the word network. Thus related terms such as 'keyboard' and 'computer' would be have similar hues while the unrelated term 'car' would have a very dissimilar hue.

Documents are varying in brightness/intensity. This dimension of intensity is used to identify the most relevant documents in the topology. It is interesting to note that usually the most highly relevant results will be clustered in two or more locations. Only about half of the time will you have a single grouping that contains the most relevant documents. This grouping or relevant results is one or more clusters allows for the user to only inspect a few documents in each cluster to determine the trends and decide whether to continue exploring in that cluster.

4   The Use of Text in the Visualization

Since the visualization is intended to be used for text retrieval it is necessary that text be displayed within the interface. Strangely, it was our findings from user feedback, that the less text in the interface the better. It seems that text can quickly clutter the visualization and this adds to its perceived complexity. As mentioned earlier it seems that an increase in perceived complexity leads to fear in users attempting to learn how to use the interface.

4.1 Labeling of the Terms

The terms as represented by spheres in the upper plane are always labeled. This is the only instance of persistent text within the visualization. The reason that the terms are labeled and nothing else is the result of the fact that the terms serve as landmarks or a road map for the underlying clustered document topology.

The terms are labeled by text that is fixed in a location relative to the representative spheres but

fixed in its orientation in terms of the viewer. Thus the term label as still being scaled based their distance from the viewer but their orientation will always be remain upright and facing the viewer. The fixed viewer orientation of the text ensures that it is always readable.

The term labels are all the same color, typeface and font size. It was found that varying the colors of the labels only served to decrease their readability.

4.2 Labeling the Documents

Determining an effective method for displaying the titles and profiles of the documents proved to be challenging. The term profile refers to a document's title, URL, summary, size and date of addition into index. The requirements for the labeling of documents were (1) since the number of documents is huge only one document's profile should be visible at any one time (2) the user should be familiar with the format the document information is provided in (3) the user should easily be able to view the information from any document in the visualization.

The three methods of displaying a document's profile or title discussed in the following paragraphs all rely on the same document selection technique. The selection technique works as follows: as the user move their mouse on to a cylinder representing a document a timer will be set. If the timer expires and the user still has their mouse cursor on top of that particular document its profile will be displayed. This method allows for the user to easily move their mouse around the scene without triggering the display of any document profiles while at the same time allowing the user to rest the mouse cursor on a document and almost instantly get the profile. This is is a method similar to how ScreenTips are displayed in Microsoft Office[1].

It should also be noted that in the visualizations in which a document is selected and its profile is displayed the associated terms in the visualization would also be highlighted. This is not very important when terms are connected by edges to documents as in the current prototypes (Fig. 2, Fig. 3) but in future it is planned that there will be no edges visible (see Fig. 1).

4.2.1 Method one: situated document titles.
The first method was simplistic and ineffective. The choice was made to have a documents label appear situated within the visualization. It was thought that this would be a nice way of spatially associating a label to a document. There are two major downsides to this usage of situated text. First, the document plane is quite dense with other documents and thus the situated document labels where easily obstructed. Second, there is not room for other text beside just a title within the visualization space.

4.2.2 Method two: relatively positioned overlays.
In this method a semi transparent rectangle was overlaid on top of the visualization and then filled with formatted text. This allowed for a large amount of text to be clearly readable. The association between the text and document was very clear. The downside of this method was that the text had to be removed from the visualization as soon as the user moved the mouse off of the document they were inspecting. The reasons for this was simply that the text overlay usually obstructed the view of a number of documents that were located underneath it.

4.2.3 Method three: fixed position overlays.
It was felt that it was advantageous to display a document's profile for as long as possible or at least until the user requested to view another document's profile. It was impossible to keep the profile on screen using the previous technique since many documents would be obstructed and thus the user would be prevented from inspecting them. In order get around the obstruction problem associated with the last technique it was suggested that the overlay be fixed to a particular non-obstructive location in the visualization. As visible in Fig. 3 the top left corner was chosen. This method worked quite well and it is the current method that is still in use.
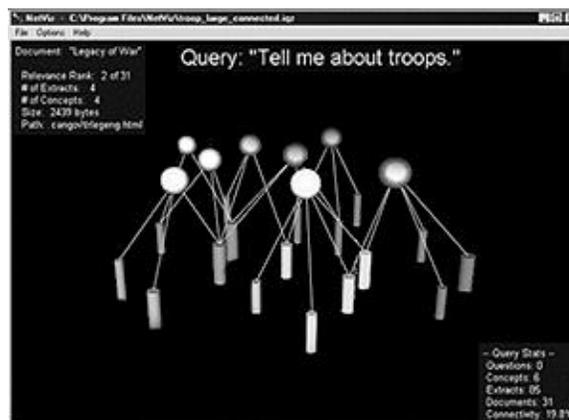

Figure 3. NetViz screen. A profile of a document is visible in the top left corner, the statistics on the search is visible in the bottom left corner.

---

1 http://www.microsoft.com/office

The current method employs the use of a semi-transparent rectangle with white text. In the next version the overlay will be designed in order to mimic the document profiles seen in most search engines (for an example of a Google profile [Brin and Page, 1998] see Fig. 4). Thus a white background will be chosen, the title will be a bold hyperlink, and the rest of the text information will be in a black font augmented by hyperlinks. Also along the side of the overlay a scroll bar will be present letting the user examine the documents in a serial fashion without selecting them from the 3D visualization. The usage of a profile that mimics a standard results from a 2D search engine should aid users in understanding the 3D interface.

Latent Semantic Analysis
... Latent Semantic Analysis Latent semantic...
...techniques. Although **latent** semantic analysis has shown...
www.cs.brown.edu/courses/cs295-3/latentsemanticanalysis.html - Cached - 4k - GoogleScout

Figure 4. A single document profile from the Google search engine.

4.3 Previewing a Documents Contents

Many commercial search engines provide short text summaries of a document to allow the user to make a judgment on weather a particular results could be fruitful. These summaries can be extracted from the META tags, they can be the first 256 characters of the document, or they can be a piece or two of the text that contains some query terms.

Unfortunately, it is hard to judge a document based solely on the equivalent of one sentences of text no matter how well the particular words are chosen. A complementary technology to text summarization would be the visualization of the regions of a document that are related to each of the individual terms of the query. The visual depiction would allow the user to judge whether the terms where consistently co-located or not.

In staying with the theme of simplicity the information about the intra-document term locations are hidden from the user until the user requests further information. Only when a user selected a particular document in order to view the profile would the extra information become apparent. The selected document (i.e. the cylinder representing the document) "opens-up" and reveals the location of term usage (See Fig 5). This idea of graphically displaying the relevant pieces of text within a document is based on a somewhat related 2D project by Eick [Eick 1994].
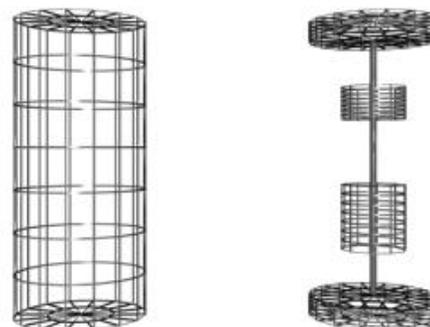

Figure 5. On the left is a "closed" document, on the right is an "open" document revealing the sections which are relevant to the query.

This usage of a visual summary is meant to serve as a complementary technique to existing text summary.

5 Interaction Methods

5.1 Adjusting the Relevance Threshold

When a search engine returns a set of results it will assign a uni-dimensional relevance factor to each element of the set. The relevance factor is generated based upon how well a document fulfills the query as a whole. Qualities such as the number of times a term is mentioned in a document, if a term appears in the title of a document, or a document has many incoming links influence the relevance factor.

AutoViz will display either the 1000 documents with the highest reputability or all the documents that fulfill at least partially the query – which ever is smaller.

The relevance threshold is a user-controlled scale that sets the minimum relevance that a document must meet in order to be displayed in the visualization. Once AutoViz has displayed all the documents, the user is then able to adjust the relevance threshold slider, in order to focus on only the most relevant documents (see Fig. 6 and Fig. 7).
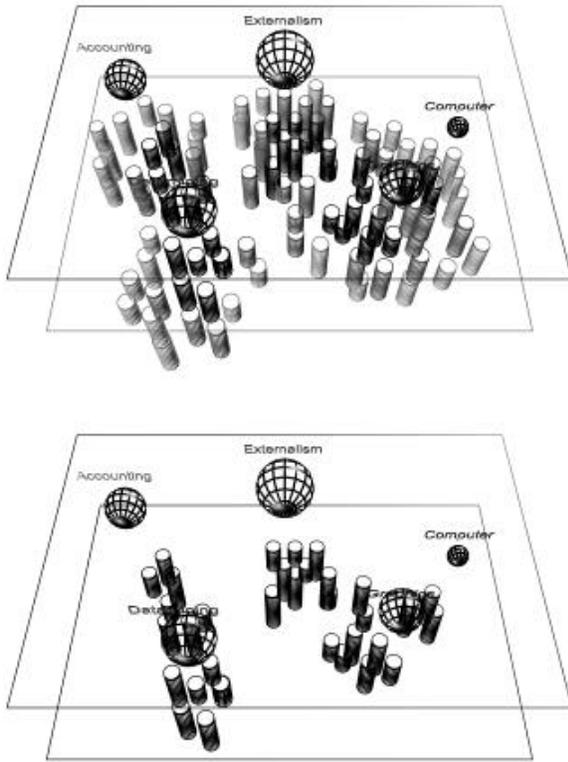
Figure 6. A depiction of showing the a low relevance threshold (top) and a high relevance threshold (bottom).
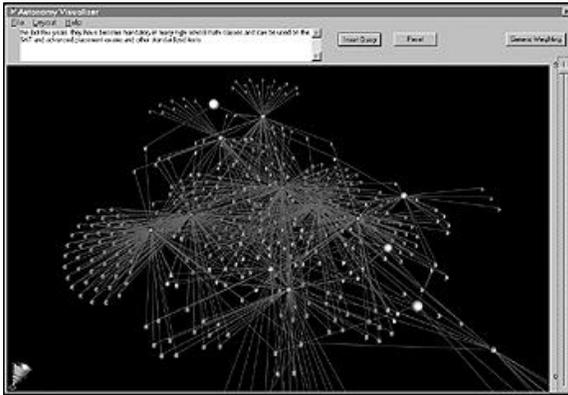

Figure 7. AutoViz screen. The relevance threshold scale runs vertically along the right side of the window.

5.2 Visualizing Multiple Queries

Assuming that the user does not get the result that they were looking for on the first query they will then have to somehow determine better more targeted queries to submit. This visual search interface allows for the submission of addition queries into the

visualization. It will allow the user to comprehend the cross-query trends in the results. Sometimes a document may be slightly relevant to a series of queries but not highly relevant to any particular query. This ability to view trends across queries will allow the user to notice persistent results and examine them to determine if it is what they are looking for.

6   Acknowledgements

7   Conclusions

The prototypes and ideas discussed in this paper are all on a path leading to simpler visualization tools for aiding users in their searches for information. No longer should users be scare of a large number of search hits when they have access to an interface that can organize that information in an obvious and meaningful way.

The next version of the interface may be implemented as a browser plug-in and could look like the depiction in figure 8.
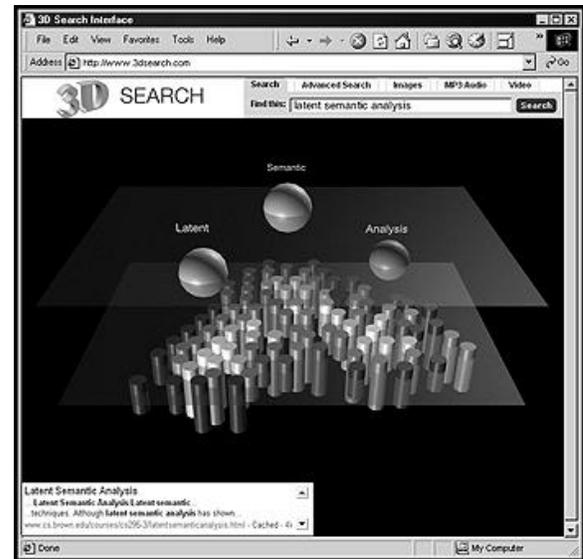

Figure 8. Artist's depiction of a future possible version of this visual search interface.

8   References

[Brin and Page, 1998] Sergey Brin, Lawrence Page. The anatomy of a large-scale hypertextual Web

search engine. In *Computer Networks and ISDN Systems*, 30: 107-117, 1998.

[Eick 1994] Stephen G Eick. Graphically displaying text. In *Journal of Computational and Graphical Statistics*, 3(2):127 – 142, June 1994.

[Fowler *et al.* 1997] Richard H Fowler, Wendy A L Fowler, Jorge L Williams. 3D visualization of WWW semantic content for browsing and query formulation. In *Journal of Network and Computer Applications*, 20: 135-149, 1997.

[Fowler *et al.* 1996] Richard H Fowler, Aruna Kumar, Jorge L Williams. In *Proceedings of The First Annual Conference on Emerging Technologies and Applications in Communication*, page 110-113, 1996.

[Jones 1998] Steve Jones. Graphical query specification and dynamic results previews for a digital library. In *Proceedings of UIST '98, ACM Symposium on User Interface Software and Technology*, November 1998.

[Kamada and Kawai 1989] T Kamada and S Kawai. An Algorithm for Drawing General Undirected Graphs. In *Information Processing Letters*, 31, pages 7-15, 1989.