The Shape of Shakespeare: Visualizing Text using Implicit Surfaces

Randall M. Rohrer[1] David S. Ebert[2] John L. Sibert[1]

[1] The George Washington University
Department of EECS

[2] University of Maryland Baltimore County
CSEE Department

"Words. Words. Words."
- Hamlet, William Shakespeare [23]

Abstract

Information visualization focuses on the use of visual means for exploring non-visual information. While free-form text is a rich, common source of information, visualization of text is a challenging problem since text is inherently non-spatial. This paper explores the use of implicit surface models for visualizing text. We describe several techniques for text visualization that aid in understanding document content and document relationships. A simple method is defined for mapping document content to shape. By comparing the shapes of multiple documents, global content similarities and differences may be noted. In addition, we describe a visual clustering method in which documents are arranged in 3D based upon similarity scoring. Documents deemed closely related blend together as a single connected shape. Hence, a document corpus becomes a collection of shapes that reflect inter-document relationships. These techniques provide methods to visualize individual documents as well as corpus meta-data. We then combine the two techniques to produce transparent clusters enclosing individual document shapes. This provides a way to visualize both local and global contextual information. Finally, we elaborate on several potential applications of these methods.

Keywords: information visualization, text visualization, procedural visualization, implicit surface modeling, blobby models, document clustering, information retrieval, graphics, user interfaces

[1] The George Washington University, Department of EECS, Washington, DC, 20052. {rohrer, sibert}@seas.gwu.edu

[2] University of Maryland Baltimore County, CSEE Department, 1000 Hilltop Circle, Baltimore, MD 21250. ebert@cs.umbc.edu

1 Introduction

Information visualization provides a tool for exploring and understanding information. While some information is inherently spatial (e.g. atmospheric temperature samples), much is not. Visualizing the *non-visual* requires mapping the abstract to the physical. Perhaps, the most common and widely-used form of abstract information is free-form text. From the earliest days of human writing, through the popular plays of Shakespeare, to the current Information Age, free-form text is integral to human communication. Text is among the richest and largest sources of information consumed, produced, and analyzed by humans. Hence, text is a prime potential source for applying visualization techniques. Text visualization uses visual metaphors for analyzing and understanding content and relationships in text documents. For example, document or corpus content might be mapped to shape, and inter-document relationships might be represented as a network graph. By specifying and controlling a useful visual mapping, text visualization can be used to better understand document content and relationships.

In this paper, we describe several new techniques for text visualization. First, we define a simple method for mapping document content to shape. By comparing the shapes of multiple documents, global content similarities and differences may be noted. In addition, we describe a visual clustering method in which documents are arranged in 3D based upon similarity scoring. Documents deemed closely related blend together as a single *concept shape*. Hence, a document corpus becomes a collection of concept shapes. These techniques provide methods to visualize individual documents as well as corpus meta-data. We then combine the two techniques to produce transparent clusters enclosing individual document shapes. This provides a way to visualize both local and global contextual information ("focus+context"). In addition to describing these visualization techniques, we briefly discuss several applications that seem useful for these methods.

Our visualization methods use underlying implicit surface models to generate flexible, procedural shapes. We believe that procedurally-generated shapes pose an interesting and useful alternative for representing abstract spaces such as text. By *procedural modeling*, we mean techniques that use algorithms and code segments to dynamically generate, abstract, and encode model detail [10, 12]. Such approaches provide a flexible, dynamic framework in which a small set of parameters may be used to control the generation of detailed models. The domain of text visualization entails a very large multi-dimensional space. Appropriate selection of a small set of representative parameters of this large space can be used to procedurally control the dynamic generation of very complex, detailed models and shapes without an unwieldy, many-to-many parameter mapping. In addition, since visualizing text requires mapping the abstract to the physical, procedural modeling provides a nice interface for providing this layer of abstraction.

2 Related Work

2.1 Text Visualization

Even though text-based information is ubiquitous, a limited number of text visualization approaches have proven useful. Problems stem from mapping a non-spatial entity into an artificial physical space. Meaningful mappings often are domain and application dependent. To be useful, text visualization must provide superior ways for humans to analyze and understand large document sets. As corpus size grows, visual clutter produced by visualization often confuse and obscure meaning. Such general issues make effective text visualization difficult but intriguing. Given the large, rich abundance of text-based information, finding effective visualization methods is important.

Several text visualization systems are particularly interesting. Themescapes presents corpus themes in the form of geographic terrain or mountains (literally, a landscape of themes) [24]. Text documents are analyzed to produce a theme summary that is mapped to peaks and valleys of a 3D terrain. The DEPICT system visualizes text documents using underlying context vectors and Kohonen Self Organizing Maps [16, 22]. It uses neural networks for learning context of word usage and how regions of high-dimensional space map into projected 2D regions. Visually, the clustered document space is represented as a 2D grid of circular nodes with color and size used to show document properties. Acquaintance/Parentage uses n-gram analysis for language-independent categorization of text documents and arranges them as a 2D proximity graph to show document relationships [7, 8]. Stereo Field Analyzer (SFA) has been used to show a clustered document space as a 3D scatterplot using stereo [11]. The VIBE system also visualizes document similarities by arranging nodes in measured proximity to fixed spatial reference points [17, 19]. MIT's Galaxies of News provides a rich 3D information space in which to navigate, explore, and find news items within a thematically clustered space [21].

These examples of text visualization use shape and spatial arrangement for depicting document content and relationships. Corpus analysis yields pertinent information that is organized and presented as visual and spatial representations. Interaction with the resulting visualizations allow further exploration and understanding.

2.2 Implicit Surface Modeling

Implicit surface modeling is a flexible, procedural approach to shape generation. It uses an underlying density field to represent volumetric information that can be modeled as a collection of procedurally-generated density field sources. Using procedural control, this density field can be evaluated to produce amorphous, generalized shapes that exhibit interesting animation and blending effects. These "organic" shapes may be viewed volumetrically (volume rendering) or by finding an iso-surface of interest. Implicit surface models are also known as blobby models, metaballs, or soft objects. See [2, 3, 18, 25, 26] for further details. Figure 1 shows several examples of shapes produced by implicit surface models. Any information that can be mapped to a density field may be modeled using these techniques. A nice property of these models is that the underlying source fields may be simply summed and blended without detailed knowledge of the processes that generated them. In this way, very desperate entities can be combined using a common density field model. (If desired, complex warps, blends, and constructive solid modeling operations can also be used when combining field sources.)

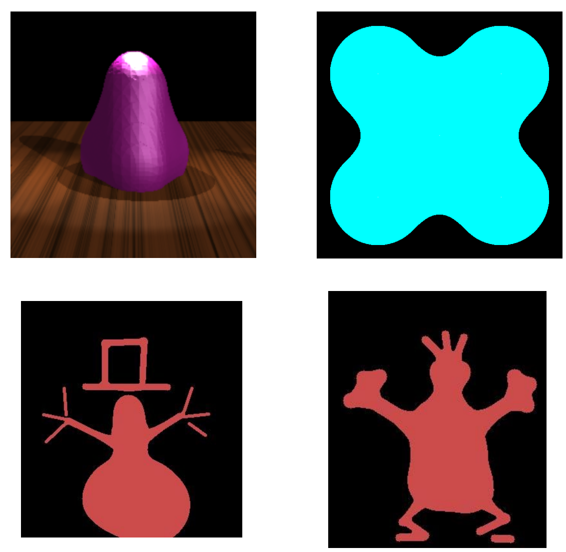

Figure 1: Shapes produced by implicit surface models

3 Blobby Text Visualization

Using ideas from text visualization and implicit surface modeling, we have developed several technique for visualizing text by mapping document content and relationships to blobby models. We sometimes refer to the resulting visualizations as *blobby text*. Implicit surface models are attractive since they use a generalized, procedurally-generated 3D density field to represent shape. Shape modeling is achieved by controlling the generation and evaluation of these density fields. To visualize text, we map text document and corpus characteristics to a density field representation. From this density field, we can generate generalized shapes.

Humans are quite adept at perceptual visual cues and recognizing subtle shape differences. In fact, it has been shown that humans can distinguish shape during the pre-attentive psychophysical process [20]. (Humans are pre-wired for understanding and visualizing shape. Visualization is based upon harnessing these skills of shape perception.) We hope to take full advantage of these perceptual skills by generating shapes in a dynamic manner. Hence, we chose to use implicit surface modeling for shape generation. While mapping selected characteristics of text documents to simple density field representations may seem arbitrary and simple, its simplicity provides for flexible shape generation. Since text is abstract and inherently non-spatial, a simple arbitrary spatial mapping results in no inherent loss of meaning. However, it is incumbent upon the visualization designer to provide, reveal, or reinforce some meaning.

In the following sections, we describe two visualization methods that use implicit surfaces for representing text. These blobby text methods are used to visualize document content and document relationships. We then combine the two techniques to visualize both local and global contextual information.

3.1 Document Visualization

Since text documents have no inherent spatial properties, visualizing the content of documents requires a spatial mapping. We have defined a simple mapping in which significant word occurrences in a document correspond to a representative 3D shape. By comparing the shapes of multiple documents, a high-level overview of document similarities and differences may be noted. Resulting document shapes may be further explored and compared by direct manipulation and "spread-sheet" interaction [6]. In [9], researchers report using superquadric ellipsoid surfaces for mapping multivariate data to representative glyphs and shapes. They control shape generation by interpolation. Our approach is related in concept. However, we use implicit surfaces for shape generation which provides a more flexible, procedural approach using volumetric density fields.

Like others, we use an underlying vector-space model to represent document content. Our document vectors summarize word occurrence in documents and how these words relate to other documents in the corpus. While n-gram analysis has the advantage of language-independence, we feel there are many advantages of using linguistic-based analysis (i.e. semantic meaning). Since we wish to apply our document model to applications in information retrieval, word "chunking", rather than pattern "chunking", seems more useful and meaningful. In analyzing document content, we ignore "stop-words" that provide little contextual meaning and do not currently use word stemming. In fact, we view our document analysis process as minimal and choose to concentrate on the visualization aspects. (A better document analysis engine could easily be substituted.) Given a corpus, we generate representative document vectors for each document as well as an overall corpus vector. The dimensionality of this vector space corresponds to the total number of unique words occurring in the entire text corpus. (Hence, it is a very high dimensional space that is commonly greater than 1000 dimensions.) Each document vector is specified in terms and dimensionality of the overall corpus vector. The document vectors are normalized with respect to the corpus vector to provide a correlation with the overall corpus content. Figure 2 shows an example document vector of N dimensions. (This document has 202 occurrences of the word "hamlet" and 0 occurrences of "java".)

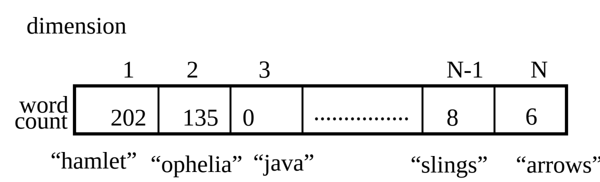

Figure 2: Document vector

After generating document vectors for the entire corpus, we perform a shape mapping for each document. One of the common problems in visualization is how to project meaningful information from high-dimensional space into a manageable, meaningful display space (i.e. 2, 3, or perhaps as many as 7 dimensions). Techniques like multi-dimensional scaling, metric scaling, and factor analysis can be used to help guide this projection. In our current system, we simply sort our document vectors in descending order of word occurrence. We select the top N occurring words in the corpus and use this as the basis for sorting document vectors. (Here, for example, statistical analysis could be applied to determine the N most influential or "best" occurring words as opposed to the N "most" occurring words.) At this point, the sorted document vectors reflect the relationship and weighting of how each dimension contributes to the overall corpus content. The chosen N dimensions are then mapped to N equally spaced directions in 3-space emanating from the origin of a sphere (Figure 3a). (The top N document dimensions correspond to these N directions in 3D.) In practice, we typically choose 14 principal directions in 3D: the 6 main orthogonal 3D directions (+X, -X, +Y, -Y, +Z, -Z axis) and also the 8 directions that bisect each of the main axis quadrants. While any value of N may be chosen, as N grows, less spacing occurs between these principal directions. We define the magnitude of the N spherical direction vectors to be linearly proportional to the document's normalized word count for the corresponding dimension. At the end of each one of these N direction vectors, we place a spherical source field function which partially defines an implicit surface model (Figure 3b). The N source fields populated about the sphere define a 3D density field. We treat this procedurally defined density field as an implicit surface model which can visualized using standard methods for implicit surface rendering. Figure 4 shows a possible shape generated by evaluating the implicit surface model of Figure 3b. (Note that we show our examples in 2D for clarity but in practice use 3D.)

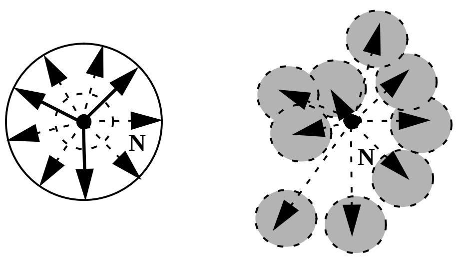

Figure 3: a) N principal directions b) implicit surface model with N source fields

Figure 4: Implicit surface generation

Experience and intuition dictate that 14 directions may be nearing the upper limit of practicality for arrangement of spherical source fields (depending on the extent of the arrangement space). However, using greater than 14 principal directions may be quite possible using alternatively shaped density functions (i.e. skeletal bones). Currently, we use identically sized spherical source functions at the end of every vector. We could vary the radius of these source functions as well. In ordering the principal directions, we currently use a fixed layout algorithm that attempts to spread high frequency words evenly about the sphere. However, one could use an alternative layout algorithms that attempt to use words semantics to guide this arrangement.

This process of mapping document vectors to spherically arranged density field sources is repeated for each document of interest in the corpus. Evaluating each document's 3D density field results in 3D blobby shapes that are based upon the content of the underlying document. Each shape uses the same scale and modeling space and is based on the same N-dimensional vector space model. For example, an upward bulge in the "hamlet" direction on one shape corresponds to the same upward "hamlet" direction on other shapes.

In our implementation, we choose to use a blobby potential field function derived by Blanc and Schlick that is flexible, efficient, and has fixed decay distance [1]. (This function generate densities ranging from 1.0 at its center to 0.0 at its boundary of influence.) To visualize, we evaluate the overall density field and compute an iso-surface (0.5 iso-surface) using Jules Bloomenthal's polygonization method [4]. We currently render the iso-surface polygons using VRML and display within VRML-enabled web browsers. Navigation and interaction with the resulting visualization is primarily performed through browser-supplied functionality. (This typically includes direct manipulation as well as common metaphors such as "fly-through" and "go-to".) In addition, VRML provides a broad set of sensor capabilities that allow one to define and add more interesting navigation and interaction aids.

Figure 5 shows a single text document visualized as blobby text. The underlying document is a paper titled "Web-based Information Visualization". Consistent with the title, the most frequently occurring words are "web", "visualization", and "information". These terms correspond to the three main, extruding bulges that appear on the top, left, and right portions of the shape. We may optionally choose to display the corresponding terms at each end of the direction vectors.

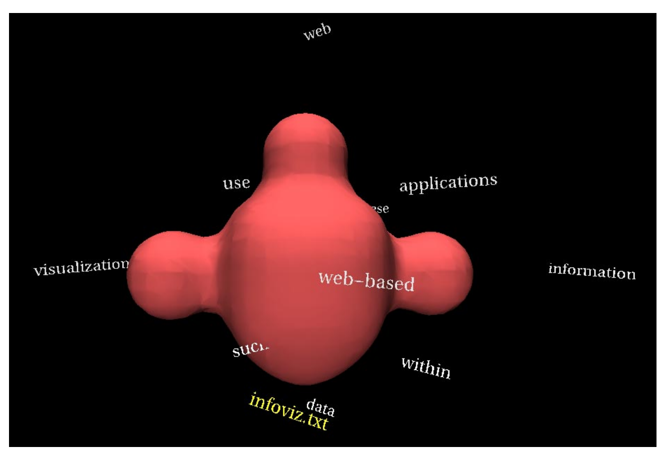

Figure 5: Blobby text document ("Web-based Information Visualization")

Extending this approach to a set of documents, we can compare document content by comparing a resulting set of shapes. To better inspect and compare multiple shapes, we use a spread-sheet style interaction method in which manipulating one shape (document) triggers corresponding changes (i.e. transformations) in all other shapes [6]. That is, if one selects a single document shape and rotates it using standard browser interaction tools, all other documents rotate in a corresponding, synchronized manner. This insures that all shapes maintain a consistent information space. (Spreadsheet interaction is currently implemented using VRML sensors and routes.) Figures 6 and 7 show examples of multiple text documents represented as a collection of blobby text shapes. Documents with similar content will tend to have generally similar shape. Documents with bulges or troughs in the same direction exhibit common word usage. (For example, in figure 6, the shapes in the upper left corner and upper right corner, each have a protruding bulge to the right which corresponds to the word "information".) While precise, detailed analysis may not be possible with this method, we see this visualization approach as a broad overview for comparing document content. We envision families of shapes that correspond to families of document themes. Such families may have overall similar shape but small local differences. In addition, these initial views might be used to initiate visual queries or to link to underlying or related information.

.

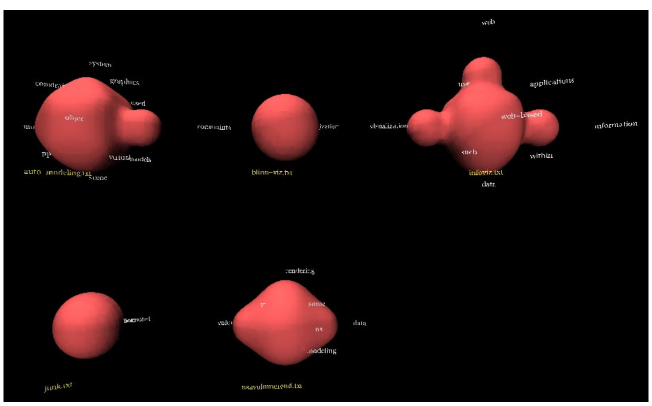

Figure 6: Multiple documents shown as blobby text

3.2 Corpus Visualization

A second text visualization method has been developed for understanding and visualizing an entire corpus and associated meta-data such as inter-document relationships. This technique employs a common document clustering algorithm and uses implicit surfaces to represent clusters of documents deemed to be similar. Our approach is similar in concept to the blobby clustering method reported by Gross and colleagues [13]. We use blobby clustering as a visual technique for representing clusters (relationships and hierarchies) of related documents. Spatial proximity of documents is used to show similarities. Documents deemed to be very closely related are shown as a cluster or connected shape that we call a *concept shape*. Underlying implicit surface models automatically blend related documents as connected components.

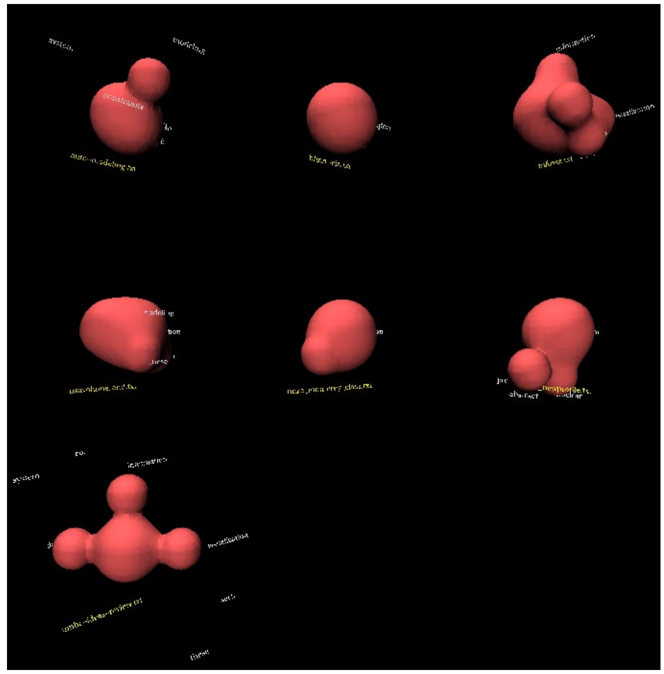

Figure 7: "Spreadsheet" of blobby text documents

We use the same document vectors described in section 3.1 to represent a collection of text documents. A document similarity matrix is computed by finding the dot-product between each pair of corresponding document vectors. This normalized similarity score provides a scalar value that represents how closely related each document is to the others. This completely connected relationship graph of documents is mapped to a corresponding mass-spring model in which particle locations correspond to document locations. Initially, the particles (documents) are randomly distributed throughout a specified arrangement region. A spring, representing a document relationship, is placed between each document pair. We use the pair's similarity score as a guide to establish the physical properties of the spring and make the spring rest length proportional to the similarity score. Documents deemed to be very similar will have spring properties that attempt to attract each other while springs of dissimilar documents will attempt to repel each other. Using this physically-based model with random initial mass locations, we allow the physical simulation to run to completion or to a sufficiently stable configuration. During this process, documents are pushed and pulled into a relative configuration that satisfies the physical properties of the mass-spring model. The final particle locations of the simulation correspond to document locations with distance being inversely proportional to similarity score. This physically-based clustering method is a common approach used by others. An incremental method that is suitable for interactive drawing is described in [7].

Once the physical simulation moves documents into a meaningful arrangement, each document's position is treated as a location of a point source field of an implicit surface model. A corpus of N documents will have N source fields representing its corresponding blobby model. Using standard implicit surface rendering, an iso-surface is found within the global density field defined by the collection of N field sources. A natural visual clustering of closely related documents is automatically created by the blobby model. Documents that are closely related, and hence closely located, are blended together as a single connected component of the implicit surface model. We often think of these connected components as concept shapes, concepts or themes shared by related documents. An entire corpus becomes a collection of concepts represented with these connected components. Figures 8 and 9 show examples of using this blobby clustering technique to visualize document relationships in two different corpuses. In each example, the concept clusters are easily visible. However, 3D interaction may be needed for full examination. Figure 10 shows a close-up view of Figure 9. The three documents clustered together in the foreground of Figure 10 all discuss issues of nuclear energy. This automatic blobby clustering approach highlights documents with similar concepts and reduces the visual clutter of individual document glyphs. Another advantage is that humans are much more adept at seeing continuous connected components as opposed to evaluating distance relationships between disconnected objects. This method provides a global view of a corpus and associated meta-data.

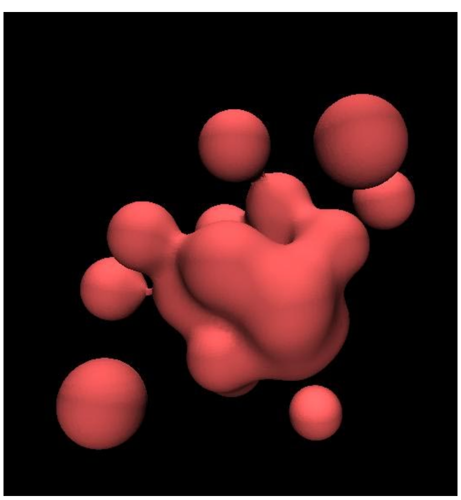

Figure 8: Clustered documents (blobby clusters)

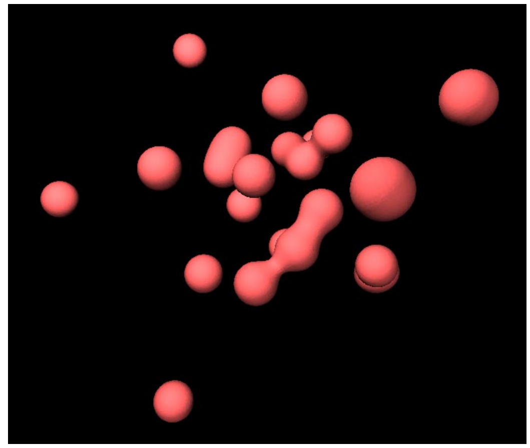

Figure 9: Clustered documents (blobby clusters or concept shapes)

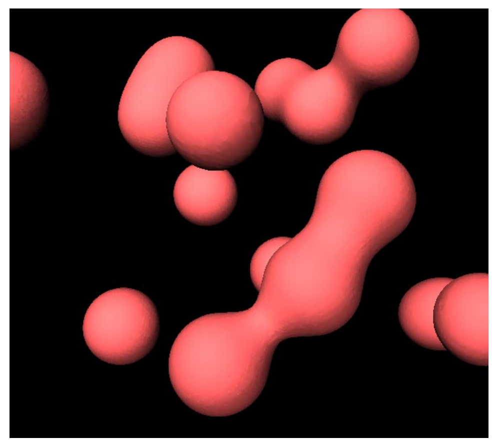

Figure 10: Close-up view of clustered documents in Figure 9

3.3 Focus + Context

Local and global context is very important in producing effective visualizations. We need to see and understand local detail (focus) in context of global information. This desired effect is known as "focus+context". Distortion viewing (i.e. fish-eye lens) is a popular approach to addressing this issue [5]. We addresses it by combining our reported techniques of document and corpus visualization (section 3.1 and 3.2) to produce semi-transparent clusters enclosing individual document shapes. The clustered concept shapes produced in corpus visualization are rendered as semi-transparent shells surrounding individual documents. The shapes of the individual documents within each cluster are based on their relative content using the shape-generating method described in section 3.1. Figure 11 shows a clustered collection of documents shown with semi-transparent cluster shells and individual document shapes. (Figure 12 is a zoomed view of Figure 11.) Figure 13 shows a detailed example of 3 documents. The two that are clustered together are Shakespeare's plays *Richard II* and *Richard III* while the solitary document within its own cluster is a document on information visualization (two vastly different concepts from vastly different ages). By providing both semi-transparent cluster shells as well as individual document shapes based on content, we achieve both local and global context within the entire information space.

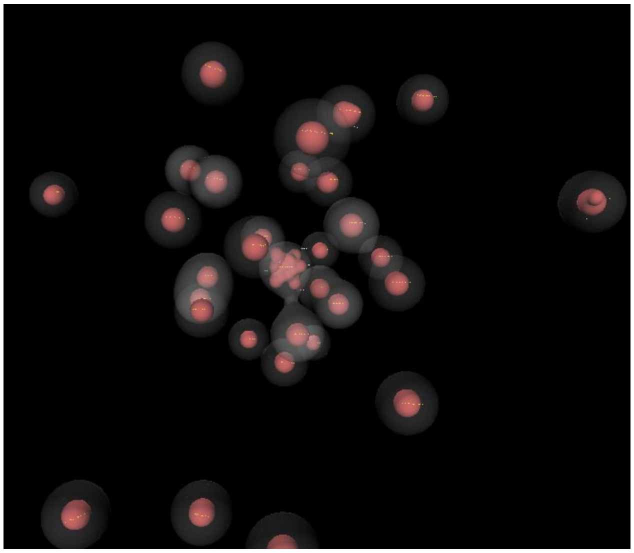

Figure 11: Semi-transparent clusters enclosing document shapes

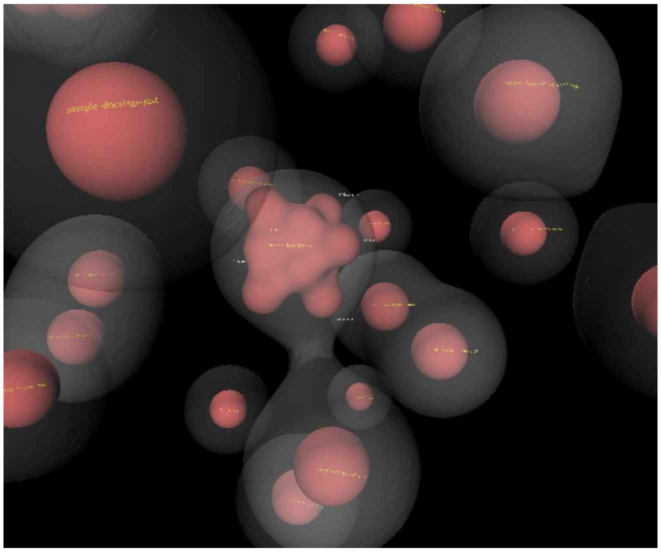

Figure 12: Zoomed view of figure 11 clusters

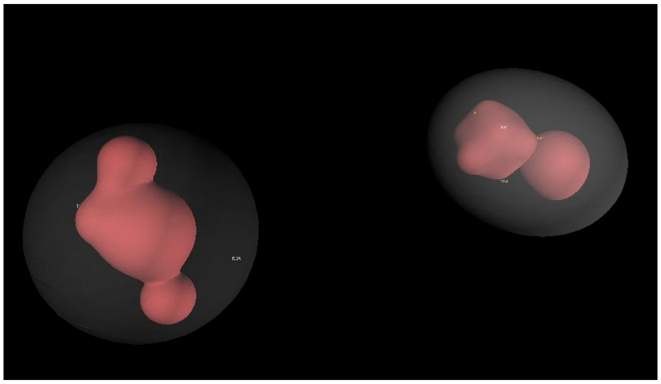

Figure 13: Information visualization cluster and Shakespeare cluster (Richard II, Richard III)

4 Applications and Future Work

We view these early experiments in blobby text visualization as very promising and are exploring numerous applications for these methods. The visualization components that we describe can be used as a visual interface for underlying information retrieval engines as well as for visualizing the results of such systems. Queries might be constructed and executed simply by marking, selecting, or brushing parts of a blobby document surface. Results of such queries might be visually highlighted across the corpus by using color, shape, or texture changes in the surface. As information volumes grow, data mining tools will likely need to rely on visualization for supplying global context and will require tighter coupling between query engines, knowledge bases, and visualization. We think that visualization through procedural, implicit surfaces may be suitable and attractive for such applications. While clustering and multi-dimensional scaling are often used to help visualize very large information spaces, visualization techniques themselves may be used to help explore and understand new clustering and scaling algorithms.

Because implicit surface models exhibit interesting and useful animation effects such as blending and deformation, they seem conducive for visualizing time-varying information. In fact, it would be interesting to visualize how a text corpus and its relationships change over time. Since we are currently using VRML for rendering, we can also embed appropriate web hyperlinks throughout our blobby surfaces which can serve as a very simple but useful information retrieval mechanism. (In fact, we currently use a limited number of hyperlinks in our current implementation to link to underlying documents.)

While blobby text visualization poses some interesting possibilities, there are some important questions left to be explored. To date, we have not performed a formal evaluation or user study to determine how meaningful and useful these techniques are. Blobby text needs to be compared with competing analysis and visualization techniques to determine its applicability, utility, and scalability. As we refine and improve our visualization and document analysis components, we will perform a much more rigorous evaluation of these techniques.

Perhaps, the most interesting direction for future work is to compute a hierarchy of document clusters (concept shapes) that map to multiple semantic layers of the information space. In this approach, multi-resolution visual components correspond to multi-resolution semantic concepts. By changing the granularity of clustering or changing the iso-surfaces (concepts) computed, a nested, multi-level, hierarchy of related concepts could be created. From the highest-level view, the overall corpus would be seen with its main concept shapes. As one dives in closer to specific clusters, more detailed views of sub-clusters (sub-concepts) would be revealed. An appropriate viewing scheme could be devised such that one always sees two levels deep into the nested hierarchy while simultaneously seeing the current level (Figure 14). If each of these nested shape layers correspond to nested semantic information layers, information query tools could adapt to appropriate levels of information detail at each level. A hierarchical, multi-resolution clustering approach such as this would be very useful for providing "focus + context".

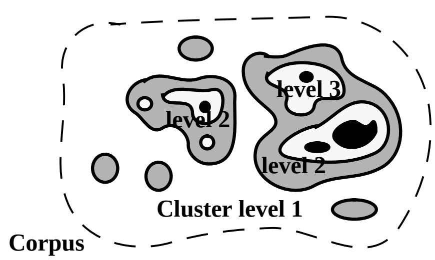

Figure 14: Nested cluster hierarchy

While iso-surface evaluation may still be appropriate with these methods, volumetric visualization of hierarchical concept layers may be a better approach. In visualizing, hierarchical layered structures, combinations of transparency, texture, and multiple iso-surfaces may be appropriate. Techniques for effective display of such layered volumetric information is addressed by Interrante [14, 15]. We plan to further expand our current work using these ideas and exploring these applications.

5 Conclusions

We have explored the use of implicit surface models for visualizing text. The power and flexibility of these underlying procedural models provides strong support for visualizing abstract, non-spatial information such as text. Using these models, we have developed techniques for visualizing the content of individual documents and for visualizing corpus meta-data. To visualize document content, we devised a simple, but representative, mapping from content to shape. By comparing the shapes of multiple documents, overall content similarities and differences may be noted. In addition, we developed a visual clustering method in which documents are arranged in 3D based upon similarity scoring. Documents deemed closely related blend together as single concept shapes that are easily observed. This provides a way of visualizing an entire corpus and its meta-data relationships. By combining these methods, a third visualization technique is produced that provides both local and global context of documents within a corpus. We are continuing to explore these techniques for text visualization, will expand their application to information retrieval and data mining, and will perform a rigorous evaluation of these techniques. While visualizing text (visualizing the non-visual) remains a challenging problem, new approaches like blobby text appear promising for providing insight and better understanding of large text corpuses. To this end, the "shape" of Shakespeare may be significant.

Acknowledgements

Thanks to Jules Bloomenthal for making his implicit surface polygonizer available for all.

References

[1] Carole Blanc and Christopher Schlick, "Extended Field Functions for Soft Objects", *Proceedings of Implicit Surfaces `95*, (Ed.) B.Wyvill and M.P. Gascuel, pp. 21-32, 1995.

[2] James F. Blinn, "A Generalization of Algebraic Surface Drawing", *ACM Transactions on Graphics*, Vol. 1, No. 3, pp. 235-256, 1982.

[3] Jules Bloomenthal (Ed.), Chandrajit Bajaj, Jim Blinn, Marie-Paule Cani-Gascuel, Alyn Rockwood, Brian Wyvill, and Geoff Wyvill, *Introduction to Implicit Surfaces*, Morgan Kaufmann Publishers, Inc., San Francisco, CA, 1997. ISBN 1-55860-233-X.

[4] Jules Bloomenthal. "An Implicit Surface Polygonizer", *Graphics Gems IV*, (Ed.) Alan Paeth, Academic Press, Inc., pp. 324-349, Chestnut Hill, MA, 1994.

[5] M. Sheelagh T. Carpendale, David J. Cowperthwaite, and F. David Fracchia, "Extending Distortion Viewing from 2D to 3D", *IEEE Computer Graphics and Applications*, Vol. 17, No. 4, July/August 1997, pp. 42-51.

[6] Ed Huai-hsin Chi, Phillip Barry, John Riedl, and Joseph Konstan, "A Spreadsheet Approach to Information Visualization", *Proceedings of Information Visualization Symposium '96*, IEEE Computer Society Press, Los Alamitos, CA, 1996, pp.

17-24.

[7] Jonathan D. Cohen, “Drawing Graphs to Convey Proximity: An Incremental Arrangement Method,” *ACM Transactions on Computer-Human Interaction*, Vol. 4, No. 3, September 1997, pp. 197-229.

[8] Marc Damashek, “Gauging Similarity with n-Grams: Language-Independent Categorization of Text”, *Science*, 10 February 1995, Volume 267, pp. 843-848.

[9] David S. Ebert, James M. Kukla, Christopher D. Shaw, Amen Zwa, Ian Soboroff, and D. Aaron Roberts, “Automatic Shape Interpolation for Glyph-based Information Visualization”, *Late Breaking Hot Topics Proceedings*, IEEE Visualization Conference, Oct. 1997, pp. 57-59.

[10] David S. Ebert, “Advanced Geometric Modeling”. In Jr. Allen Tucker, editor, *The Computer Science and Engineering Handbook*, chapter 56, CRC Press, 1997.

[11] David S. Ebert, Chris Shaw, Amen Zwa, Ethan L. Miller, and D. Aaron Roberts, “Minimally-immersive Interactive Volumetric Information Visualization”, *Proceedings of Information Visualization Symposium ‘96*, IEEE Computer Society Press, Los Alamitos, CA, 1996., pp. 66-67.

[12] David S. Ebert (Ed.), F.K. Musgrave, D. Peachey, K. Perlin, and S. Worley, *Texturing and Modeling: A Procedural Approach*, AP Professional, Boston, MA, 1994. ISBN 0-12-228760-6.

[13] M.H. Gross, T.C. Sprenger, and J. Finger, “Visualizing Information on a Sphere”, *Proceedings of Information Visualization Symposium ‘97*, IEEE Computer Society Press, Los Alamitos, CA, 1997, pp. 11-16.

[14] Victoria Interrante, “Illustrating Surface Shape in Volume Data via Principle Direction-Driven 3D Line Integral Convolution”, *Proceedings of SIGGRAPH 97*, (Los Angeles, CA, August 3-8 1997), In *Computer Graphics* Proceedings, Annual Conference Series 1997, ACM SIGGRAPH, pp. 109-116.

[15] Victoria Interrante, Henry Fuchs, and Stephen Pizer, “Illustrating Transparent Surfaces with Curvature-Directed Strokes”, *Proceedings of IEEE Visualization 96*, IEEE Computer Society Press, Los Alamitos, CA, 1996, pp. 211-218.

[16] T. Kohonen, *Self-Organizing Maps*, Springer-Verlag, Berlin, 1995.

[17] R.R. Korfhage, “To See or Not to See -- Is That the Query?”, *Proceedings of 14th Annual ACM SIGIR Conference*, 1991, pp. 134-141.

[18] H. Nishimura, M. Hirai, T. Kawai, T. Kawata, I. Shirakara, and K. Omura, “Object Modelling by Distribution Functions”, *Electronics Communications*, Vol. 68D, No. 4, pp. 718-725 (in japanese), 1985.

[19] K.A. Olsen, R.R. Korfhage, K.M. Sochats, M.B. Spring, and J.G. Williams, “Visualization of Document Collection: the VIBE System”, *Information Processing & Management*, Vol. 29, 1993, pp. 69-81.

[20] Andrew J. Parker, Chris Christou, Bruce G. Cumming, Elizabeth B. Johnson, Michael Hawken, and Andrew Zisserman, “The Analysis of 3D Shape: Psychophysical principles and neural mechanisms”, In Glyn W. Humphreys (ed), *Understanding Vision*, chapeter 8, Blackwell, 1992.

[21] Earl Rennison, “Galaxies of News: An Approach to Visualizing and Understanding Expansive News Landscapes”, *Proceedings of UIST 94*, Marina Del Ray, CA. November 4-6 1994.

[22] David A. Rushall and Marc R. Ilgen, “DEPICT: Documents Evaluated as Pictures, Visualizing Information Using Context Vectors and Self Organizing Maps”, *Proceedings of Information Visualization Symposium ‘96*, IEEE Computer Society Press, Los Alamitos, CA, 1996, pp. 100-107.

[23] William Shakespeare, *Hamlet*, Act II, Scene ii, 1600.

[24] J.A. Wise, J.J. Thomas, K. Pennock, D. Lantrip, M. Pottier, A. Schur, and V. Crow, “Visualizing the Non-Visual: Spatial analysis and interaction with information from text documents”, *Proceedings of Information Visualization ‘95*, IEEE Computer Society Press, Los Alamitos, CA, 1995, pp. 51-58.

[25] B. Wyvill, C. McPheeters, and G. Wyvill., “Animating Soft Objects”, *Visual Computer*, 2: 235-242. 1986.

[26] G. Wyvill, C. McPheeters, and B. Wyvill, “Data Structures for Soft Objects”, *Visual Computer*, 2: 227-234. 1986.