CircleView: Scalable Visualization and Navigation of Citation Networks

Peter Bergström, E. James Whitehead, Jr.
Dept. of Computer Science
Univ. of California, Santa Cruz
Santa Cruz, CA 95064
{pbergstr, ejw}@cs.ucsc.edu

ABSTRACT

CircleView is a citation network browser that uses circles around circles as its visualization method to show a focus paper and two levels of its citation network. This method scales to varying numbers of papers and references, and has a structured layout that makes the visualization more readable. Bibliographic metadata is available via mouseover for all displayed papers. General requirements are presented for citation network visualization and navigation tools. An important requirement is the ability to integrate with existing institutional digital libraries, satisfied by CircleView's use of web browser facilities for its user interface. CircleView is shown to have good visualization and performance scalability characteristics.

Categories and Subject Descriptors

H.3.7 [**Information Storage and Retrieval**]: Digital Libraries – *user issues*; H.3.7 [**Information Storage and Retrieval**]: Digital Libraries – *systems issues*.

General Terms

Design, Documentation

Keywords

Visualization of citation networks, information visualization, reference networks, bibliographic metadata, digital library

1. INTRODUCTION

A hallmark of scientific writing is the use of references to related work. Readers mine these references pursuing multiple goals. Newcomers to a discipline can follow chains of references through papers in the field, thereby orienting themselves to its problems and approaches. Frequently recurring papers in the citation network are usually authoritative papers in a field, worthy of deep study [2]. Disciplinary experts use references to stay abreast of current work, and to understand who is using published results, and how. Conference reviewers use references to determine the originality of a submission, and assess whether it has sufficiently referenced related work. Personnel review committees use citations as a rough measure of the influence of research [1]. Finally, historians and patent attorneys use references to establish the chronology and pathways of contribution for inventions.

A common thread running through these different uses of references is the need to efficiently navigate through the citation network. Current digital libraries, such as those run by ACM or IEEE, provide some support for this navigation. Someone browsing through these libraries can get to a web page for a specific paper, thereby displaying its abstract, bibliographic metadata, and a link to the paper's full text. Each page also lists the paper's references, hyperlinked to the web page for each referenced paper. Library patrons can navigate through the citation network by searching or browsing to a paper's page, then clicking on an individual reference link, bringing up the reference's page, and so on. While the reference mining tasks listed above can be performed using this style of navigation, the approach is far from ideal. The heart of the problem is the single paper orientation of the digital library web interface, and the need in all of the reference mining tasks to derive meaning across multiple reference-linked papers. The digital library interfaces focus on a single node in a multi-node citation network. A researcher sees only one paper at a time, and that is not enough to place a paper in context.

To provide better support for information seeking tasks involving citation networks, digital libraries must support the visualization and navigation of citation networks. Visualization is necessary to show multiple papers at the same time, more easily supporting the derivation of meaning across a set of papers. Furthermore, this capability must be scalable, handling large numbers of papers, and widely varying numbers of references per paper. Navigation is necessary to support the cognitive model of following references from paper to paper in the citation network. Navigation also helps address issues of visual scale: only a portion of a large paper set need be shown on screen at any one time, with navigation causing a shift in the papers currently being viewed.

In the remainder of this paper we explore issues surrounding the visualization and navigation of citation networks. In Section 2 we describe general requirements for citation network visualization and navigation systems. Following, we present our realization of these requirements in the CircleView system by giving a system overview, scenario of use, implementation details, limitations, and a performance characterization. We conclude by examining the related work, and describing future directions.

Before continuing, a brief note on terminology. When one paper references another, it creates a relationship between those papers. It is useful to distinguish between the outbound and inbound ends of that relationship. If a paper *A* has in its text a reference to a second paper *B*, we say that *A references B*. This is the outbound end of the relationship. The inverse (inbound) form is called a citation, *B cites A*. This choice of terminology is confusing, since conventional usage conflates the meaning of reference and citation, and once combined, it is difficult to later cleave a distinction between them. We have not been able to develop improved terms, but this choice of terminology is consistent with that used by the ACM Digital Library.

2. GENERAL REQUIREMENTS FOR CITATION NETWORK NAVIGATION AND VISUALIZATION

Over the past ten years, a handful of systems have explored the design space of visualizing and navigating through a citation network (in chronological order, Butterfly [3], BIVTECI [4], RefViz [5], and CiteWiz [1]). We reflect on these systems and our own work to develop a general list of requirements for systems that support navigation and visualization of citation networks.

A theme running through these requirements is the recognition that this is more than just a standard graph visualization problem. To support people's citation mining tasks a system must, of course, provide a scalable graph visualization, but it must additionally provide support for understanding the contents of the papers that are the graph nodes, and the ability to navigate from paper to paper through the graph structure. Similarly, it is also insufficient to provide only navigation support without a supporting graph visualization; this is the current state of web based digital libraries. The key to adequate task support is the combination of graph visualization, navigation, and the ability to easily reveal substantial details about each paper in the visual field.

2.1 Scalable Network Visualization

Current digital libraries contain hundreds of thousands of papers, and even disciplinary subsets have paper counts in the hundreds to low thousands. Clearly any visualization of the citation graph among a set of papers must scale to the expected dataset. A corollary is that one way to address scalability concerns is to keep the size of the dataset down, by choosing some policy to limit the number of papers visualized on screen at any one time.

Another issue for graph visualization scalability is the number of line crossings. Citation networks involve a wide variation in the number of edges leaving and entering each node; we observed one survey paper with 280 references. Beyond a small number of papers, a graph visualization that shows all edges will make the edges hard to see or count, causing difficulty in understanding the significance of a given paper.

In order to provide an effective visualization, systems will rarely, if ever, have recourse to visualizing an entire large dataset at once. Once the dataset is larger than low 100's of papers, it is hard to imagine a visualization that would not suffer from problems in understanding the relationships among papers, and difficulty in using pointing devices to select individual papers to see their bibliographic metadata and abstract. As a result, only parts of a large dataset will ever be visualized, leading to the need to be able to move around throughout the dataset to bring new paper subsets into view, and our next requirement.

2.2 Support Navigation

A user needs to be able to navigate through the citation network, bringing different portions of the network into the visualization with each navigation step. Navigation transitions should occur within a reasonable period of time to avoid user frustration, and ensure users do not lose their task context waiting for a refresh.

There are two models for navigating through a citation network, paging and panning. In the paged approach, the visualization focuses on a specific paper, and shows some portion of the citation network centered around it. Such is the case with Butterfly [3] and CiteWiz [1]. BIVTECI's "specific view" is a slight twist, allowing a small number of focus papers simultaneously [4]. Navigation then involves shifting to a new focus paper, thereby bringing up a different portion of the citation network. One can think of the visualization for each paper as forming a separate page, and hence navigation involves jumping from page to page (really, from view to view).

In the panning approach, the entire citation network for a dataset is visualized in a large virtual space, with only a view into this large space being visible at any one point in time. Current 3D graphic toolkits easily support this approach. Navigation then involves moving the view window across the existing visualization, possibly also supporting zooming and tilting. RefViz exemplifies this approach. It provides a visualization where each paper is a square, with squares clustered depending on keyword similarity. The interface allows panning and zooming across this space composed of potentially large numbers of clustered papers. RefViz does not visualize the citation network, and while this avoids the reference line crossings issue, it also makes the system not directly comparable to the others [5]. BIVTECI's "relevance view" is similar to that provided by RefViz. BIVTECI also has a "general view," a view of the citation network of a paper dataset that is not centered on a given paper. While BIVTECI does not explicitly address scalability issues, a panning navigation scheme could work in this case.

For paged navigation of the citation network, there are two additional requirements.

2.2.1 Focus on a Given Paper

The visualization needed to be centered on a given paper. This supports the user model of jumping from paper to paper in the citation network. It also makes it clear which paper is the current focus, thereby allowing some of this paper's bibliographic metadata to be prominently displayed. For example, our CircleView implementation displays the focus paper's title at the top of the screen, and displays its bibliographic metadata by default when a new screen is displayed.

By focusing on a given paper, it is also possible to construct a local tree out of the citation network, thereby allowing for a hierarchical visualization. In general, tree layout is much easier than graph layout.

2.2.2 Navigation History

The page orientation provides the benefit that it makes it possible to record the browsing history. A simple browsing history tracks papers that have been viewed. The history should be viewable, and support backtracking. This history is similar to a web browser's history feature, and similarly acts to reduce "lost in hyperspace" problems by remembering a linear navigation path over the network structure.

2.3 Integrate into existing digital libraries

Citation network visualization and navigation systems should be designed to integrate into existing web based digital libraries. Early systems like Butterfly and BIVTECI were designed before the advent of web digital libraries, and hence are standalone desktop applications. In the case of Butterfly, the UI was assumed to be the primary interface into the digital library, so the system provided search support.

The current stability of the web's architecture makes it likely that existing web-based digital libraries will exist in more-or-less their current form for the foreseeable future. As a result, it makes sense

to design citation visualization systems such that they could be deployed as part of these systems, and do not duplicate existing functionality. Citation visualization systems need not provide search services, since these are better provided by the libraries themselves, or by web search engines like Google Scholar.

There are two broad approaches for achieving this goal, depending on where the user interface is generated. The easiest to field and deploy is a web application architecture, where the visualization is generated on the server side of a web based digital library. The visualization is then displayed to the user in the web browser. This has the benefit of not requiring any special software installation, permitting quick rollout to a broad audience of scholars. The drawback is a loss of responsiveness in the user interface during navigation operations. More responsive user interfaces are possible by creating standalone desktop applications that then interacts with the digital library via some form of remote procedure call interface. However, supporting a wide range of users and platforms is problematic at best.

2.4 Display Bibliographic Metadata

Many citation mining activities involve determining the relevance of a paper's contents to a person's task. For example, a novice exploring a citation network is interested in finding papers relevant to their chosen research topic. Making determinations of relevance requires some information about the paper's contents. As a result, there must be a way to display bibliographic metadata and the paper abstract for user-selected papers in the visualization. Ideally, this metadata will include standard bibliography items like the title, authors, publication venue (conference/journal), pages, volume, issue, etc. Additionally, the visualization should show the number of referenced works, and number of citing works.

Since this is the most commonly accessed information, it should be as easy as possible for a user to cause its display. The information should be displayed on the screen without needing to reload the entire page from a server, and should be available for all papers on the screen. In the CircleView system, this information is available via mouseover on all papers, thereby needing no user action beyond moving the mouse.

2.5 Ready Access to Paper Fulltext

Since the bibliographic metadata and paper abstract are often not sufficient to determine the relevance of a paper to a user's information seeking task, the system needs to provide a mechanism for easily accessing the paper's fulltext. Additionally, once a person identifies articles that are relevant to their task, they often want to download them onto their local machine, for later viewing or printing.

2.6 Paper Signficance Cues

The visualization should be able to give the user visual cues to what papers are more important than others. It should also be able to show if a paper occurs more than once in the immediately visible citation network. While there are many ways to establish the significance of a paper, one simple way is to note the number of times it has been cited, and provide a secondary visualization of this citation count.

2.7 Scale to Large Data Sets

The system's implementation needs to scale to datasets the size of existing institutional digital libraries. There are two primary issues here. First, the database schema and technology needs to be able to accommodate large numbers of papers, their reference lists, and inbound citations. While challenging, this issue has already been addressed by the large digital libraries. A second issue is efficiently supporting queries that return portions of the citation graph. This appears to be an open issue for large datasets, and high transaction volumes.

3. THE CIRCLEVIEW APPLICATION

Our method of visualizing academic citation networks is realized through an application we call CircleView. CircleView represents papers as circles, and uses circles around circles to show a paper and two levels of its references.

Figure 1 shows a typical screenshot of CircleView using our sample dataset from the ACM Digital Library. The circle in the middle is the *focus paper* and the surrounding circles are the papers it references. The smaller circles on the reference circles are those papers' references. References of the main paper that have more than 10 references of their own are represented with a bolded line instead of showing individual references (For example, the two papers in the upper left corner of Figure 1). The references of the main paper are sorted by the number of citations each has, with the number of references signified by the color of the line connecting it to the main paper. Colors were chosen in a manner analogous to a colored temperature scale where papers with more than 50 citations have a red line and papers with 0 citations have a black line (we recognize that this color scale will be only partially legible for people who are color blind). Assuming that a paper that has been cited more often is more important than ones that have been cited fewer times, the most important paper is shown as the detailed reference by default.

The *detailed reference* is the larger circle in the top right. When a bold-line paper becomes the detailed reference, all of its references will be shown, no matter how many references it may have. If the user clicks on any of the main paper's other references, that paper will become the detailed reference. If the user then clicks on the detailed reference, the view will shift, making it the focus paper, with all of its references around it. Hovering the mouse over any of the circles will cause the bibliographic metadata to be shown in the bottom right corner and will cause any other occurrence of that paper in the visible network to be highlighted. This enables a user to quickly count the number of times a paper is the referenced by the papers in the immediate network. The metadata shows the title, all authors, publication information, publisher, as much as possible of the abstract, and a link to the ACM Digital Library. If a paper is a reference but its metadata is not in the database, i.e. it is at the edge of the dataset, the user has the option to retrieve its metadata from the ACM Digital Library.

The circles around circles layout technique, combined with a focus paper, are central to achieving scalability of the visualization. Since CircleView only displays papers one or two reference hops away from the focus paper, the total number of papers that must be visualized is limited. We do not attempt to visualize the entire dataset at once. Additionally, our visualization employs graceful degradation when faced with situations that yield overly cluttered or unreadable visualizations. The use of a bolded circle to represent a reference with too many references is one such approach. Since graceful degradation results in some information being hidden, the detailed reference paper provides a mechanism to re-expose this hidden information for a single paper.

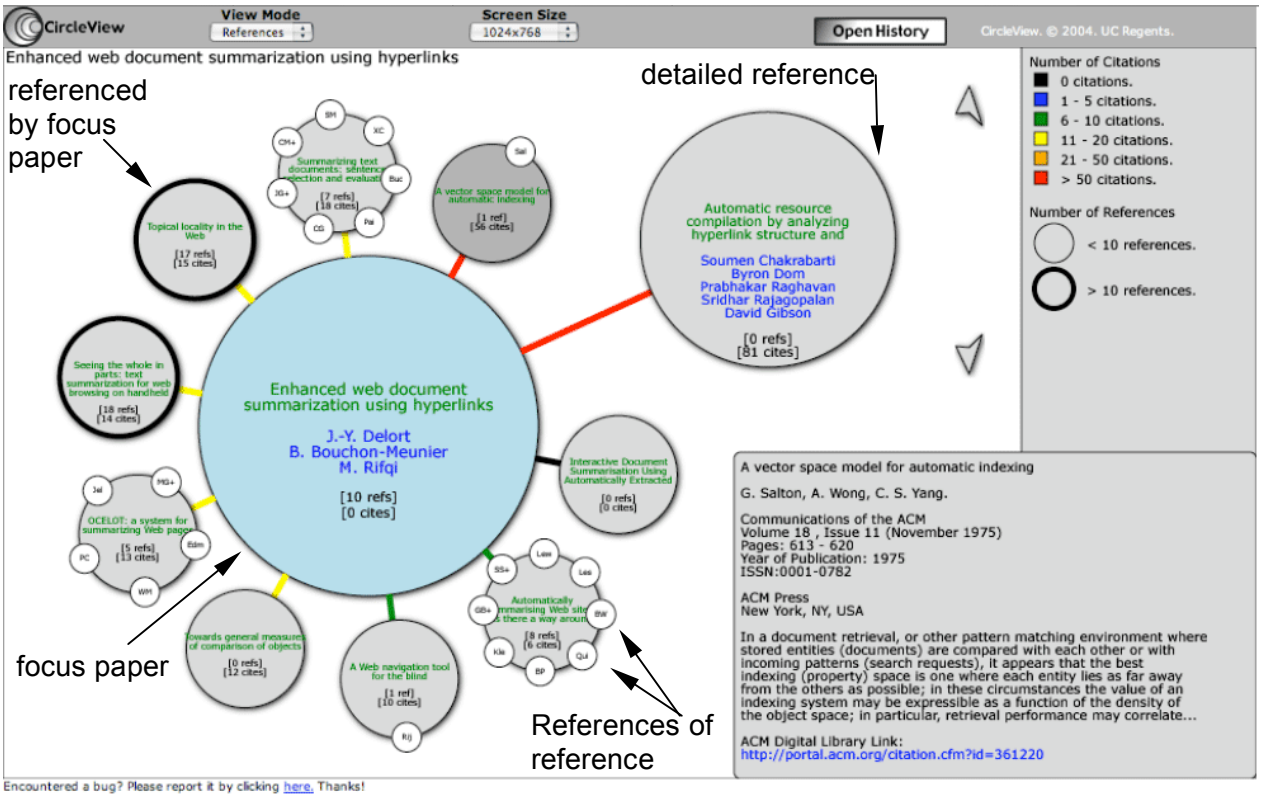


Figure 1. CircleView user interface.

A user can change from reference view to citation view. With citation view, the visualization uses the same circles around circles approach to show all the papers that cited the main paper. This is useful if one wants to see related papers on a certain topic, or to follow a thread of research forward in time from a given paper.

Users can change the size of the view area depending on the resolution of the user's screen. The visualization is ideally viewed at a resolution of 1600x1200 pixels, however it can go down as far as 800x600. No visual information is lost at lower resolutions, due to CircleView's use of the vector-oriented SVG, but since text becomes smaller, it can be difficult to read.

CircleView keeps a browsing history, shown in Figure 2, so that a user can keep track of where in the network he is currently located. It also allows for quick access to any of the views in the history by clicking on the title and loading it in the CircleView window. There is also a back button that makes CircleView go back one view to the previously viewed paper in a manner similar to a web browser's back function.

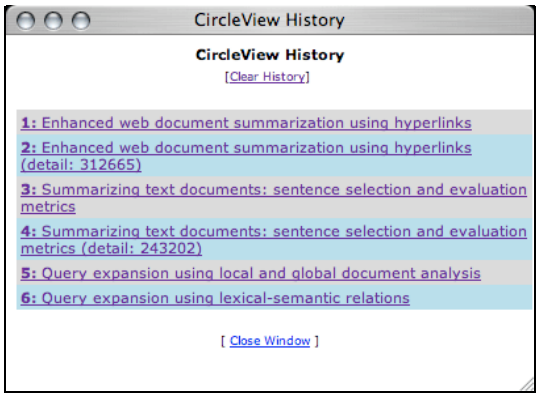


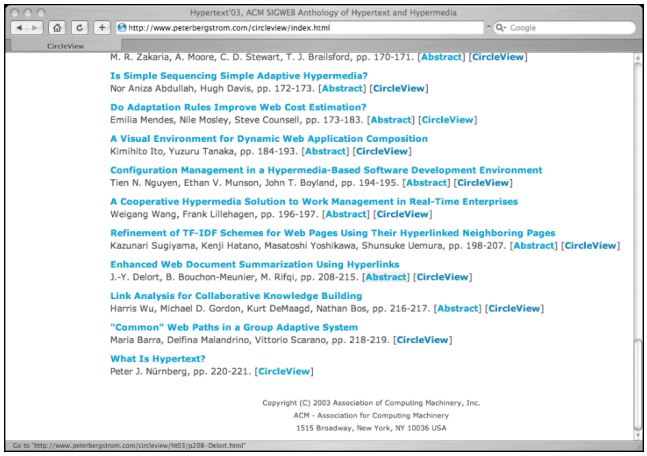
Fig. 2: CircleView History

4. A SCENARIO USING CIRCLEVIEW

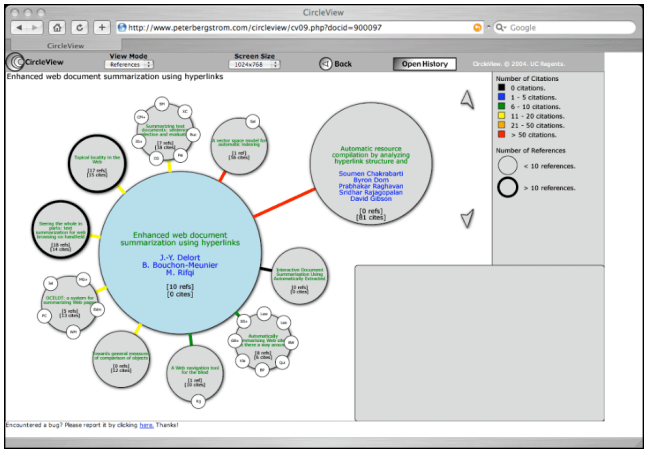
In order to illustrate the way that CircleView might be used, we present a scenario that covers the main functionality of the application. It covers a typical information-gathering task that a researcher may perform using CircleView.

Our researcher, Melissa, begins by looking at the online version of the proceedings of the Hypertext 2003 conference (Screen 1, below). Screen 1 shows the Hypertext 2003 table of contents from the ACM's SIGWEB Anthology of Hypertext and Hypermedia CD, modified to include links to CircleView next to each paper. Clicking on the CircleView link brings up the visualization, centered on that paper. This screen shows one possible way to integrate CircleView into the user interface of existing institutional digital libraries.

Melissa is somewhat familiar with the hypertext field and finds the paper titled, "Enhanced web document summarization using



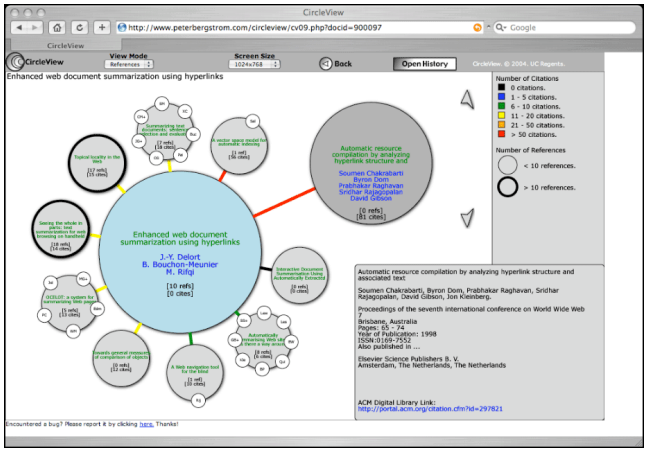
Screen 1: The Hypertext 2003 proceedings page.



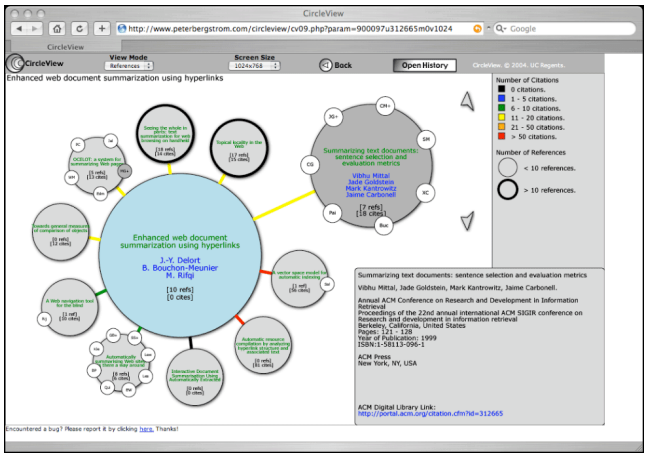
Screen 2: Default view in CircleView of "Enhanced Web Document Summarization Using Hyperlinks."

hyperlinks," pertinent to her current line of research. Melissa decides to use CircleView to see supporting literature the paper references, in hopes of finding other work related to her research. To do this, Melissa clicks on the "CircleView" link next to the paper, bringing up Screen 2 with "Enhanced web document summarization…" as the focus paper of a CircleView page.

Screen 2 shows that the focus paper has 10 references, ordered counter-clockwise from the detailed reference by the number of citations. The detailed reference paper has 81 citations. Melissa thinks that the paper title seems interesting. Melissa moves her cursor over the circle for the detailed reference paper, yielding Screen 3, which now shows the bibliographic metadata for the paper in the lower right hand corner. Since the paper has been in print for only five years by the Hypertext 2003, yet has 81 citations, she decides that it is probably an influential paper, and would be good to read. She clicks on the "ACM Digital Library Link," causing a new window to pop up, generated by the ACM Digital Library (in this case the link is to the actual ACM Digital Library, and not to the contents of the SIGWEB Anthology CD). From there, Melissa has access to the paper's fulltext, which she downloads. Closing the window returns her to CircleView.



Screen 3: Melissa puts her cursor over the detailed reference of "Enhanced Web Document Summarization Using Hyperlinks" to see its metadata.



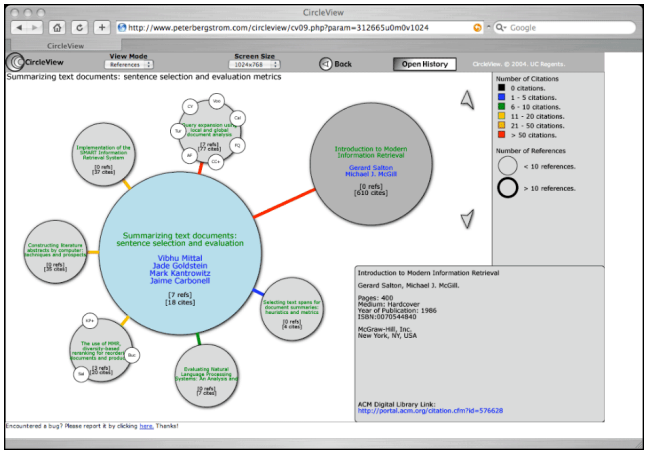
Screen 4: Melissa views the metadata for the 3rd most cited reference of "Enhanced Web Document Summarization Using Hyperlinks."

Melissa now wants to look at the third most cited paper in more detail. To do this, she can either click on the reference circle or click the down arrow twice to rotate the references clockwise by two. After rotating the view, Melissa sees Screen 4.

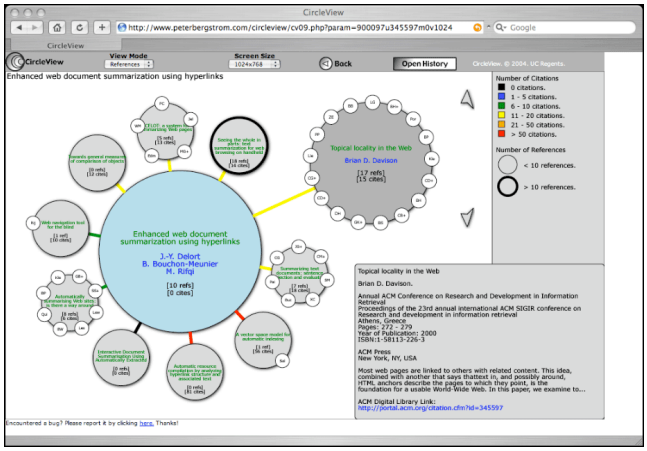
Melissa knows this paper already but has never thoroughly examined its references. She notices that the detailed paper was cited by another reference. Melissa sees this because there are two highlighted circles in the visualization that signify the same paper (the detailed reference is highlighted, as is one of the references of the paper at appx. 10 o'clock). She clicks on the detailed reference circle to refocus the visualization on it, yielding Screen 5.

Melissa moves her cursor over the highest ranked reference and sees that it has a whopping 610 citations. Clearly this must be an important work, and worthwhile to read. Examining its metadata in the lower right window, she notices that this work is a book, which she'll need to check out from her local library.

Melissa decides to return to the original paper of interest by using the back button, bringing her to Screen 4. She thinks the fourth most cited paper looks interesting, and decides to examine it more closely. This paper is represented as a bold circle because it is has more than 10 references. To make the references visible, Melissa



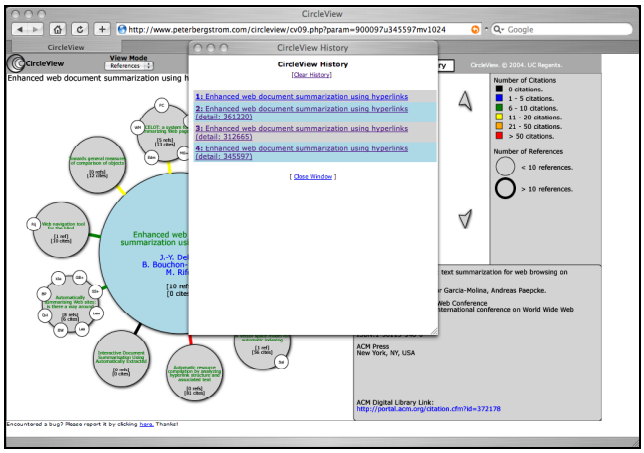
Screen 5: Melissa views the metadata of the most cited reference of "Summarizing Text Documents: Sentence Selection and Evaluation."



Screen 6: Melissa views the 4th most cited reference of "Enhanced Web Document Summarization Using Hyperlinks."

rotates the visualization clockwise to make this paper the detailed reference, yielding Screen 6 (below). Melissa moves her cursor over the detailed circle to see the metadata of the paper. This paper also looks relevant to her research, so she downloads it by following the library link in the lower right corner. Melissa wants to see what papers she has already viewed, so she clicks on the "Open History" button in the top toolbar. In Screen 7, a window pops up with links to all the papers she has viewed, listed in visitation order. She wants to return to the first CircleView screen, so she clicks the first link in the list, bringing her back to Screen 2.

Melissa thinks she has found enough reading for today, so she finishes her CircleView session by closing the browser and continuing on with her work.



Screen 7: Melissa viewing CircleView's browsing history.

5. CIRCLEVIEW IMPLEMENTATION

5.1 Architecture

CircleView is a three tier web application. Application logic is written in PHP, using the MySQL relational database for persistence of the paper dataset and for citation network queries. We chose Structured Vector Graphics (SVG) [6] because of its vector representation of lines and text, thereby allowing the visualization to adapt to different screen sizes while still producing attractive, proportionally scaled output. Figure 3 (below) details the overall architecture of CircleView and how it interacts with the user and the ACM Digital Library.

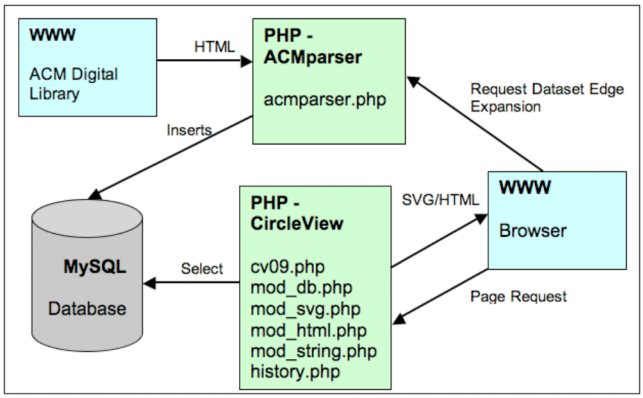


Fig. 3 CircleView Architecture.

At the heart of CircleView there are two PHP components that interact with the client. The CircleView component retrieves the necessary data to compose the current CircleView page, then outputs the appropriate HTML and SVG to the client. The ACMparser component handles user requests to expand the dataset by one paper. The need for this functionality derives from our decision to replicate only a portion of the entire ACM digital library in the CircleView database, thereby guaranteeing that users will eventually reach the edge of the dataset. To ensure that this condition does not bring their activity to an immediate halt, the user may request the insertion of an edge node. If CircleView were fully integrated into an institutional digital library, this functionality would not be necessary.

The ACMparser component takes the request, consisting of a reference identifier based on the ACM Digital Library's format, and retrieves the page from the digital library. Then, the parser parses out all the information and inserts it into the database. The client refreshes and the added paper is now in the network.

5.1.1 Database Interaction

Within the CircleView component, the database module handles the database connection as well as the depth first recursive function that retrieves all the necessary data for the visualization. The main module passes the numerical identifier of the main paper to the database module and recurses down two levels of references[1]. The algorithm determines which papers in the two levels are actually visible in the page being generated. For example, a reference of the focus paper with more than 10 references will not display these references, instead substituting a bolded circle perimeter.

The SVG code output to the client is optimized by eliminating metadata for papers not visible in the current view. This optimization is important because SVG rendering and object selection times are related to the size of the SVG document. In the case of object selection, reduced SVG sized resulted in noticeably more responsive reference mouseovers. The ECMAScript function that manages the highlighting and showing of the metadata needs to iterate through two arrays, which is time consuming, and performance depends heavily on the speed of a user's computer. Typically, it takes less than half a second to highlight a circle and show the metadata, but the lag time can be as much as 2 seconds when there are more than 50 unique papers in a view.

The database module generates a multidimensional array containing the paper ID, title, abstract, source, publisher, authors, and references or citations of all the visible papers. This array is passed along to the SVG module.

5.1.2 SVG Generation

Within the CircleView component, the SVG module generates the SVG code. The SVG module consists of a very large recursive function that produces the SVG markup. The function behaves differently depending on its current level within the recursion. First, it draws the main circle and the information boxes, and generates all the metadata that is visible when a user puts the cursor over a circle. Second, the function draws the first level references one by one. After each first level reference, it recurses and draws the second level for that reference. The circles and their positioning are calculated using simple geometry using sine and cosine. After all the SVG markup is generated, the program returns to the main module where it makes the SVG HTML safe, and then outputs the SVG to the client browser.

5.2 Paper Dataset

To test CircleView we needed a dataset large enough for testing yet not large enough to become unmanageable. Initially, we decided to use the CiteSeer database from the University of Pittsburgh [7], because CiteSeer has made its metadata publicly available online in XML format. However, after importing the data into CircleView and inspecting the data, we noticed that the dataset was incomplete and the number of papers was too large for our implementation. Unfortunately, the quality of CiteSeer's citation data was not as good as expected because CiteSeer relies on users to input papers, the result being that many paper references are omitted. In areas where we knew the literature well, the incompleteness of the CiteSeer data led us to lack confidence in the utility of CircleView. The visualization and navigation features were useful for finding papers in the CiteSeer dataset, but we knew that many other relevant papers were missing.

The CiteSeer database is also very large, with over 500,000 papers. Since our primary intended contribution is in the visualization, and not in the database representation, it was no surprise that our relatively straightforward use of MySQL became slow when loading a view. Our expectation is that any real-world deployment of CircleView will involve integration with the repository schema of an existing institutional digital library, where this data scaling problem has already been addressed.

To reduce the size of our dataset, and to improve the reference and citation data quality, we turned to the ACM Digital Library. Since this digital library is also very large, we did not undertake to replicate the entire library, instead focusing on a subset of its collection. Unlike CiteSeer, the ACM Digital Library does not make its data publicly available in an easily parsable format. To import data from the ACM Digital Library, we wrote a crawling HTML parser in PHP to screen-scrape the data. Given a starting paper, the parser recursively retrieves all the papers listed in the reference section of a paper. For us, a nice aspect of the ACM Digital Library is its hyperlinking of most reference papers, allowing easy extraction of reference relationships. This is significant because it allowed us to avoid the challenging problems of parsing the text of paper references, and mapping references to unique paper identities. It does, however, mean that our reference data quality is only as good as that of the ACM Digital Library. While very good, not all references of a paper are hyperlinked in the ACM Digital Library, and we ignore all non-hyperlinked references due to parsing difficulty.

The parser crawls down three levels from the starting paper. This allowed us reach four levels of papers because the crawler retrieves references for the third level (thereby requiring the creation of nodes representing papers at the fourth level) but does not retrieve the metadata for these fourth level of papers. A key issue in the implementation of the reference crawler was subdividing the work to avoid out-of-memory errors due to holding too great a portion of the citation graph in-memory at any one time.

For our dataset, we chose the Proceedings of the 2003 ACM Conference on Hypertext and Hypermedia [8], since we are familiar with this literature. The proceedings consisted of 38 long and short papers. Retrieving the 38 papers down to 3 levels yielded 5,305 unique papers, 12,694 total (non-unique) authors (a given author will appear in our dataset as many times as they have written papers in our dataset; we did not perform any processing to create unique author entries, since this was not necessary for the visualization), 173 editors, 28,674 references, and 127,374 citations.

The retrieval process took 10 days to complete. One reason the retrieval took so long was that we intentionally slowed the rate at which we accessed the ACM Digital Library in order to be

[1] We have only recently realized that a more efficient query approach appears possible, involving a self-join of a reference endpoint table with itself. The table directly expresses the first level of references, and the self-join returns the second level. We have not yet tried this implementation approach.

courteous and avoid being blacklisted. Therefore, we limited page requests to once every 3 seconds. There turned out to be almost an order of magnitude more citations than references, due to the fact that the number of papers retrieved is small subset of all the available papers in the ACM Digital Library. The large number of papers outside of CircleView's relatively small dataset makes it statistically likely that more papers would reference the papers in the dataset than the papers in the dataset would reference papers outside of the dataset.

5.3 Performance

Though many aspects of the implementation of CircleView would be changed in a full scale deployment, it is still useful to have a broad characterization of the performance of the existing implementation.

We instrumented the server-side CircleView component to measure SVG generation time for the default view of each paper in the database, repeating the test 5 times per paper. Then, we averaged the SVG generation time for papers with the same number of references. Results are shown in Figure 3. The x-axis represents the number of papers (circles) displayed in a given page. The y-axis represents the number of milliseconds required to generate the given page. Tests were performed on a Mac Powerbook G4 with 512MB of RAM, running Apache 2 and mod_php.

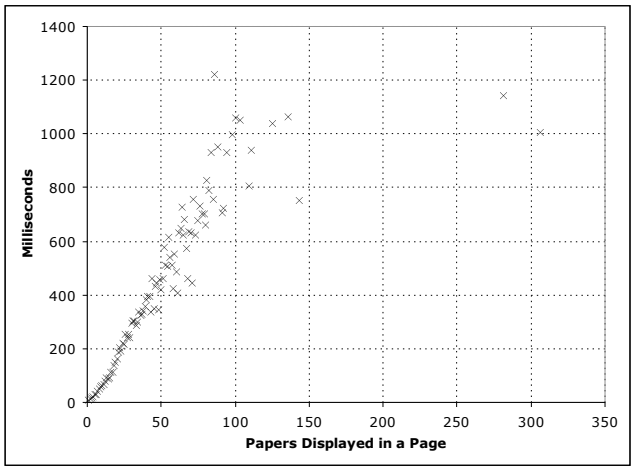


Fig. 3: Plot of number of papers displayed per page vs. SVG rendering time (in milliseconds)

The results of this test showed a generally linear trend up to 111 papers, at which point the dataset becomes sparse, with available data points indicating a leveling out between 1 and 1.2 seconds. The longest page generation took 1.22 seconds for 86 papers, yet the largest number of papers in a view, 306, was generated in an average of 1.00 second. This discrepancy may be due to the fact that the 86 papers might have contained more metadata than other papers. Since more metadata leads to a larger amount of SVG to be generated, this would result in a longer page generation time.

5.4 Scalability of the Visualization

When the number of references of a paper becomes very large, CircleView's method of visualization becomes progressively more difficult to interpret. The best method to deal with scale is to alternate the distance each reference circle is from the main circle, thereby increasing the effective usable area. This allows for more references to be displayed in such a way that the circles are larger than they would be if they are all directly next to each other. This measure is fairly good at maximizing the possible size of the reference circles, but it only works up to a point.

When the number of references exceeds 60, the visualization becomes too crowded. We note that the related systems BIVTECI, Butterfly, and CiteWiz do not substantively address visual scalability at all, and hit scalability limits around 20-40

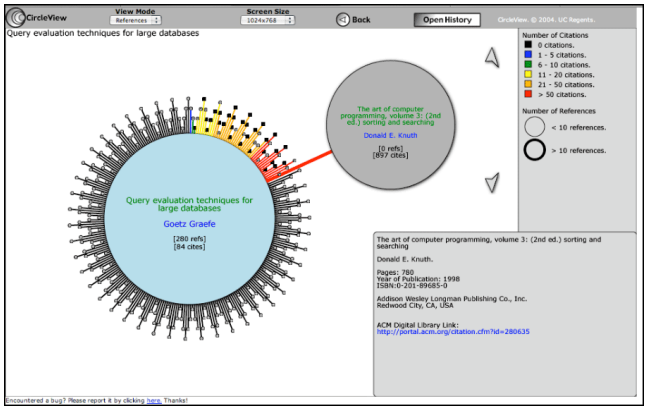


Fig. 4: An extreme case of a paper having 280 references.

references. Figure 4 below is an extreme example of a paper that has 280 references. The user can still access all the metadata and use all the other functions of CircleView but it is very hard to read or navigate. Therefore, CircleView is limited in its effectiveness when the number of references is large.

How serious is this problem? The answer depends on the distribution of reference counts. If it is very infrequent to encounter papers with 60 or more references, then the lack of visual scalability of CircleView is not as critical. Figure 5 shows the count of papers that possess a given number of references in our dataset. It clearly shows that the majority of papers in our dataset have a low number of references and less than 0.4% of papers will have scalability issues.

We note that 2,650 out of 5,305 papers have 0 references. This represents several things. Papers at the edge of our dataset do not have their references loaded into our dataset, and hence will appear to have zero references. Another factor is that some referenced works aren't hyperlinked by the ACM Digital Library, either due to parsing problems with the reference, or the fact that it doesn't maintain a record for the work. Finally, some of the items in the digital library just don't have references, common examples being a conference chair's introduction, or an overview of a panel session. Generally the count of papers with zero references should predominantly be interpreted as an artifact of our data collection technique, and the limits of ACM's reference linking technology.

The average number of references per paper in our dataset is 9.5 when excluding papers with 0 references (5.4 when including all the papers with 0 references). Both averages are well within the capabilities of CircleView. Therefore, one is much more likely to see a paper with less than 20 references than a paper with over 60.

Still, a user of CircleView will occasionally encounter a paper with high reference counts, and it would be nice if CircleView behaved better in these circumstances. While we have not substantively addressed this issue, one possibility is to create a set of stacked focus circles, with each page in the stack holding 30 or

fewer references. User interface features would then be added to permit flipping through the stack. Stacks would be organized in order of citation count, with the topmost stack, shown by default, holding the papers with highest citation counts. We intend to explore this in future work.

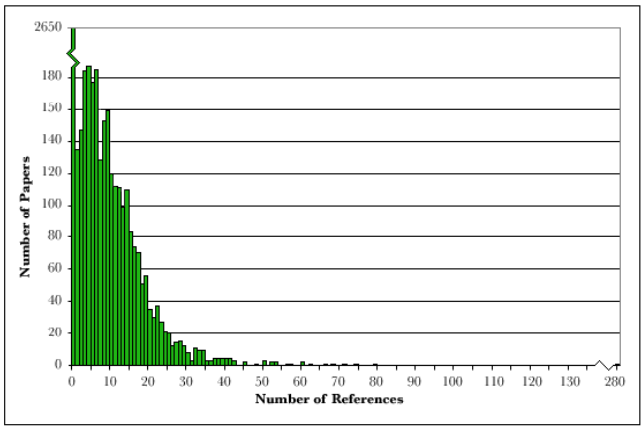


Fig. 5: Bar chart showing the count of papers (y-axis) having a given number of references (x-axis).

6. RELATED WORK

6.1 Hierarchical Information Visualizations

Since CircleView transforms the citation network into a tree for on-screen display, it can be viewed as a form of hierarchical information visualization. Hierarchical information visualizations are a good way to show a large amount of information on a single screen. Fisheye views visualize large amounts of information by showing the root of a hierarchy in high detail and subsequent levels of the hierarchy in less detail [9]. This allows the current root to take up more of the screen yet also show the context where it fits in with the rest of the hierarchy.

One can modify the fisheye view by having the tree hierarchy be uniformly arranged along a circle as shown by Lamping [10]. This modification allows a smooth blending of focus and context because the whole hierarchy is laid out uniformly on a hyperbolic plane then mapped onto a circular display. CircleView similarly uses focus plus context, but takes advantage of the fact that it only displays two hops from the focus paper to use a more straightforward, non-hyperbolic layout.

Another method of graphically showing hierarchical information is through the use of Tree-Maps [11]. Tree-Maps provide a dense information display, using the size and placement of a particular component in the map to show its significance. CircleView also shows which papers are more significant than others, using color and ordering.

There are several ways to represent hierarchies in three dimensions. Cone Trees allow for the display of hierarchies in 3D space in such a manner that it retains the vertical structure of a tree but allows the nodes in each level to be arranged in two (rather than one) dimensions [12]. The Perspective Wall [13] and the Document Lens [14] are visualization methods that allow a user to focus on a specific area yet see the rest in perspective. The Butterfly [3] browser developed by MacKinlay allows for searching and navigation of citations located in an online database. It uses a "butterfly" layout, similar to the two sides of an open book, where a papers metadata is shown at the top of the butterfly and the wings of the butterfly contains the paper's references on the left and the paper's citers on the right.

CircleView opted for a simpler 2D visualization of the citation network, instead of a 3D approach. This was due to the added complexity of providing a 3D visualization within a Web browser, as well as a design sense that it is better to use a 2D visualization if that sufficiently meets the visualization requirements.

6.2 Bibliography and Citation Network Visualization

Thomson ResearchSoft produces the commonly used Endnote [15], a tool that allows scholars to organize their collected bibliographic metadata, and automatically generate paper reference sections. However, Endnote does not represent reference relationships among papers in an Endnote database, and this is the key relationship CircleView uses for its visualization. As a result, Thomson's EndNote companion product for bibliographic visualization, *RefViz* [5], does not visualize citation networks. Instead, RefViz uses keywords available for each paper to cluster together papers with related keywords. The BIVTECI system's relevance view is similar to the clustering performed by RefViz [4]. However, the goal of BIVTECI and RefViz is similar to CircleView in that these applications aim to give users a visual way to find related papers.

The 2004 InfoViz conference held a visualization contest, involving the visualization of a dataset of InfoViz conference papers from 1995 to 2002 (8 years) [16]. Visualizations were required to address the four contest tasks: create a static (non-interactive) overview of the 8 years of the InfoViz conference, characterize research areas and their evolution, show where a particular researcher fits into the research areas, and show the relationships between two or more researchers. Due to the focus on static visualization, these visualizations did not support navigation. As well, due to the chosen tasks, all contest entries performed some kind of clustering of papers, similar in spirit to RefViz.

The winning student entry by Ahmed et al. provides a large-scale visualization of the entire citation network of the dataset in a single figure [17]. Unfortunately, it is impossible to pick out individual papers in the resulting visualization, severely limiting its utility. The entry by Ke et al. provides a more interesting citation visualization, in which papers are circles, and the size of the circle indicates the number of citations it received [18]. Lines between circles represent reference relationships. Coloring is used to indicate the age of papers. To reduce the number of papers that need to be shown in the visualization, they show only papers that were cited 20 times, as well as papers that cited one of these and were cited at least 7 times themselves. This reduced the 614 paper dataset down to a more manageable 47, which was then visualized. Since their approach is a graph visualization, there are still a large number of line crossings in their visualization. Overall, this citation filtering approach appears to be quite valuable.

Related to citation network visualization is visualization of co-citation networks. A co-citation occurs when separate documents by two or more authors cite each others' work. Analysis of co-citation is useful for finding clusters of related papers, since authors that cite one another frequently tend to be working in similar areas of research. Chen and Carr [19] provide one example of computing and visualizing co-citation networks within the

Hypertext literature. Their visualizations present co-citation networks as 3D graphs, realized using VRML technology.

7. CONCLUSION

CircleView is a new method for visualizing and navigating academic citation networks using a web-based graphical interface. By limiting its display to two reference hops from the focus paper, it permits scalable visualization of citation network subgraphs. Compared to using a web browser and accessing an online paper database such as the ACM Digital Library, where one can only view a single paper and its metadata, CircleView is much more powerful. It allows a user to see many relationships between papers. Instead of clicking through several web pages to get to a given paper, the paper can be visible on the screen from the beginning.

We also provide a coherent set of requirements that citation network visualization and navigation systems must support. A key requirement that differentiates our work is the desire to integrate this feature into existing, web-based institutional digital libraries. The CircleView implementation has demonstrated the feasibility of delivering reference navigation capability to a standard web browser with reasonable client-side and server-side performance. Additionally, the CircleView visualization method has reasonable visual scalability for 99.6% of all papers encountered in our dataset. Our hope is that CircleView is sufficiently compelling, that librarians of existing institutional digital libraries are willing to take the next step, and deploy this, or similar functionality to a broad audience of library patrons.

ACKNOWLEDGEMENTS

We would like to thank Melissa Chan and Mark Slater for their contributions to this work.

REFERENCES

[1] N. Elmqvist, P. Tsigas, "CiteWiz: A Tool for the Visualization of Scientific Citation Networks," Tech. Report no. 2004-05, Dept. of Computing Science, Chalmers Univ. of Technology and Göteborg University, 2004.

[2] S. Kim, E. J. Whitehead, Jr., "Properties of Academic Paper References," *Proc. Fifteenth ACM Conference on Hypertext and Hypermedia (Hypertext'04)*, Santa Cruz, CA, Aug. 9-13, 2004, pp. 44-45.

[3] J. Mackinlay, R. Rao, S. Card, "An Organic User Interface for Searching Citation Links," *Proc. ACM CHI'95 Conference on Human Factors in Computing Systems*, 1995, pp. 67-73.

[4] D. Modjeska, V. Tzerpos, P. Faloutsos, M. Faloutsos, "BIVTECI: A Bibliographic Visualization Tool," *Proc. 1996 Confernce of the Centre of Advanced Studies on Collaborative Research*, 1996, p. 28.

[5] Thomson ISI Web of Science, RefViz, http://www.refviz.com/.

[6] J. Ferraiolo, J. Fujisawa, D. Jackson, "Scalable Vector Graphics (SVG) 1.1 Specification," W3C Recommendation, Jan. 14, 2003, http://www.w3.org/TR/SVG/

[7] CiteSeer.IST Scientific Literature Digital Library, http://citeseer.ist.psu.edu/

[8] L. Carr, L. Hardman, *Proceedings of the Fourteenth ACM Conference on Hypertext and Hypermedia (Hypertext'03)*, ACM Press, Nottingham, UK, Aug. 26-30, 2003.

[9] G. Furnas, "Generalized Fisheye Views," *Proc. ACM SIGCHI Conference on Human Factors in Computing Systems*, April, 1986, pp. 16-23.

[10] J. Lamping, R. Rao, P. Pirolli. "A Focus+Context Technique Based on Hyperbolic Geometry for Visualizing Large Hierarchies," *Proc. of the SIGCHI Conference on Human Factors in Computing Systems*, May 1995, pp. 401-408.

[11] B. Johnson, B. Shneiderman. "Tree-Maps: A Space-Filling Approach to the Visualization of Hierarchical Information Structures," *Proc. 2nd Conference on Visualization '91,* pp. 284-291.

[12] G. Robertson, J. Mackinlay, S. Card. "Cone trees: Animated 3d visualization of hierarchical information," *Proc. ACM SIGCHI Conference on Human Factors in Computing Systems,* April 1991, pp. 189-194.

[13] J. Mackinlay, G. Robertson, S. Card. "The Perspective Wall: Detail and Context Smoothly Integrated", *Proc. SIGCHI Conference on Human Factors in Computing Systems*. March 1991, pp. 173–176.

[14] J. Mackinlay, G. Robertson. "The Document Lens," *Proc. 6th Annual ACM Symposium on User Interface Software and Technology,* December 1993, pp. 101-108.

[15] Thomson ISI Web of Science, Endnote, http://www.endnote.com/.

[16] InfoViz 2004 Contest, "The History of InfoViz." http://www.cs.umd.edu/hcil/iv04contest/

[17] A. Ahmed, T. Dwyer, C. Murray, L. Song, Y. Wu, "InfoViz 2004 Contest: WilmaScope Graph Visualization," Information Visualization Benchmarks Repository, http://www.cs.umd.edu/hcil/InfovisRepository/contest-2004/

[18] W. Ke, K. Börner, L. Vizwanath, "Major Information Visualization Authors, Papers, and Topics in the ACM Library," Information Visualization Benchmarks Repository, http://www.cs.umd.edu/hcil/InfovisRepository/contest-2004/

[19] C. Chen, L. Carr, "Trailblazing the Literature of Hypertext: Author Co-Citation Analysis (1989-1998)", *Proc. of the Tenth ACM Conference on Hypertext and Hypermedia (Hypertext'99)*, ACM Press, Darmstadt, Germany, February 21-25, 1999, pp. 51-60.