DOI: 10.1111/j.1467-8659.2011.01921.x

Eurographics / IEEE Symposium on Visualization 2011 (EuroVis 2011)
H. Hauser, H. Pfister, and J. J. van Wijk
(Guest Editors)

Volume 30 (2011), Number 3

PaperVis: Literature Review Made Easy

J. -K. Chou[1] and C. -K. Yang[1]

[1]Department of Information Management, National Taiwan University of Science and Technology, Taiwan

Abstract

Reviewing literatures for a certain research field is always important for academics. One could use Google-like information seeking tools, but oftentimes he/she would end up obtaining too many possibly related papers, as well as the papers in the associated citation network. During such a process, a user may easily get lost after following a few links for searching or cross-referencing. It is also difficult for the user to identify relevant/important papers from the resulting huge collection of papers. Our work, called PaperVis*, endeavors to provide a user-friendly interface to help users quickly grasp the intrinsic complex citation-reference structures among a specific group of papers. We modify the existing* Radial Space Filling *(RSF) and* Bullseye View *techniques to arrange involved papers as a node-link graph that better depicts the relationships among them while saving the screen space at the same time.* PaperVis *applies visual cues to present node attributes and their transitions among interactions, and it categorizes papers into semantically meaningful hierarchies to facilitate ensuing literature exploration. We conduct experiments on the InfoVis 2004 Contest Dataset to demonstrate the effectiveness of* PaperVis.

Categories and Subject Descriptors (according to ACM CCS): I.3.3 [Computer Graphics]: Picture/Image Generation—Line and curve generation I.3.6 [Computer Graphics]: Methodology and Techniques—Graphics data structures and data types

1. Introduction

One critical problem that we modern people are facing today is not the lack of information but its inundation. It could be a serious issue for academics, who would like to conduct researches on relatively new areas. One would try to look up for keywords or concepts through google-like search engines, but finally loses focus after following numerous seemingly related hyper-links that normally bear complicated structures. To be more specific, sufficient amount of literature review is mandatory for entering a new research field; however, the intricate and complex citation and reference relationships among papers would always present non-trivial or even formidable difficulties, especially for newcomers. To address these issues, we propose to represent papers as a graph that facilitates the following functionalities. First, it is easier to find important papers for a certain research field and shows how important a paper is in its corresponding research field. Second, the relevance among papers become more intuitive and comprehensible. Third, the overall user-friendly visual structure provides better transition coherence while switching among views.

Our work, named *PaperVis*, endeavors to assist both experts and novices as follows. For expert users, they can find important papers in the specified categories by using keywords to filter out undesired information from the dataset. In addition, such a framework reduces the chances of missing the reference of some relevant or even important works from others. For novice users, a new research field becomes much more accessible as papers of higher importance can be easily discovered. Such a functionality can also be applied to a specific paper; that is, given a paper, say paper A, a frequently asked question is to identify the most relevant or important papers with respect to paper A. Most of us will agree that we should have read as many as possible papers while reviewing literatures, however, when there are too many related papers to read, it becomes more critical to prioritize the reading sequence. By starting with the most important/relevant papers, it significantly helps catching the overall picture of a specific research field in the early stage.

To sum up, *PaperVis* makes the following contributions. First, we represent interested papers as a graph that better depicts their complex relationships by modifying the existing *radial space filling* [AH98] and *bullseye view* [CK97] techniques. Second, visual cues, such as node colors, sizes and boundaries are used to indicate papers' importance or citation/reference relationships. In addition, the distances among papers are determined according to their relative similarities. Third, through the help of our visual representations, the visual coherence among viewing histories are preserved. Finally, we propose a novel *information clustering algorithm* that categorize papers into more meaningful groups to facilitate ensuing paper relationship exploration. We have conducted several case studies based on the InfoVis 2004 Contest Dataset to prove the effectiveness of *PaperVis*.

2. Related Work

In this section, we discuss the previous works in calculating the similarities/impacts/relationships among papers/documents and summarize the related works concerning the visualization of paper citation network.

2.1. Defining relationships among nodes

Several papers have defined different functions to assess the impact/similarity/relationship between two papers. Crnovrsanin et al. introduced a method to rank the relationships between two nodes in a graph, even if they were not connected at all [CCM09]. According to the ranking score, it helps digging out the implicit links which represent hidden connections in complex networks. Similarly, Song [Son98] computes the term *semantic similarities* in the document space, which are mainly used for clustering purpose. Furnas [Fur86] proposed the concept of computing the *degree-of-interest*, or *DOI* for short, which represents the importance level of one node to the current focus node, under a tree structure. Later, van Ham et al. [vHP09] extended DOI from trees to graphs while taking the user interests into account. The previous techniques are typically computing the node relationships over the entire dataset. Nevertheless, as mentioned in the Introduction section, the focus of this paper is to help prioritize or find relevant papers from a paper's bibliographic information. The calculation of *importance* and *relevance* in our work are inspired by Broder et al. [BGMZ97], who defined the term *resemblance* and *containment* to determine the syntactic similarity between two textual documents.

2.2. Visualizing paper citation network

There are numerous works involved in exploring the bibliographic meta data through various visualization techniques. We classify them into two main categories.

2.2.1. Network-based

The network-based approaches tend to present papers of a database altogether at once. This type of visualizations basically analyzes the structural features of a graph or adopt efficient clustering algorithms to find significant nodes or separate nodes into groups. In general, they provide meaningful global overview of the entire dataset, and allow users to catch initial insights before drilling down for details. Oftentimes, what people need is to begin with a node of interest and then navigate to learn new knowledge from whom it relates to. Another main issue for network-based visualization is its ability for interaction. The inefficiency of the computational cost over the entire graph usually makes them infeasible for user interaction. For these reasons, our work adopts the perspectives provided by Paper-centric methods.

2.2.2. Paper-centric

Similar to our approach, the main focus of paper-centric visualizations is to first allow users to select an interested paper from the database, and then to visualize the bibliographic network of the chosen paper. For example, Mackinlay et al. [MRC95] implemented *Butterfly* which enables users to navigate within the complex paper reference-citation cyberspace. Shen et al. proposed *BiblioViz* [SOTM06] for visualizing bibliographic information. It summarizes the pros and cons of the published papers in InfoVis Contest 2004, and integrates the desired functionalities into their design. They also argue that a node-link graph is better for conveying relationships among objects than other mechanisms. Zhang et al.'s work of *CiteSense* [ZQGS08], proposes an architecture to let users collect their own knowledge database and share their experiences. van Ham and Perer [vHP09] adapt the DOI function to obtain interested hidden information. *PaperCube* [BA09], a web-based application, proposed by Bergstrom et al., enables users to explore new and important research papers from online digital libraries. The idea of their work is to reduce the cognitive load by only showing relevant information. Carriere et al. first made use of the *bullseye view* to visualize the results from a web query [CK97]. By modifying Carriere et al.'s work, Yang et al. proposed to use *core trees* to present paper citation network data [YK99]. The core tree algorithm arranges paper nodes on different concentric circles accordingly. By far, the last three discussed papers are perhaps the closest ones to ours. They all generate the visualization results over a focus node's immediate neighbors. However, both Carriere et al.'s work and Yang et al.'s work place nodes through a hierarchical structure, where the distance between two nodes refers to their hierarchy, but not relationship. Moreover, the same node may appear more than once in one view. As we believe that distance should be intuitively associated with relationship, *PaperCube* [BA09] and the other papers seem to fail on placing nodes in appropriate positions, so that the exact distances among nodes become meaningless.

3. PaperVis

To perform as an effective exploration tool for literature review, *PaperVis* offers the following features:

Deploy screen space as efficient as possible We adopt the ideas of *radial spatial filling* [AH98] and *bullseye view* layout [CK97] to efficiently utilize the screen space.

Meaningful visual representation The color, size and boundary of a node and the distances among nodes visually represent meaningful characteristics of the results.

Paper-centric *PaperVis* places its focus on a specific paper or keyword, while other papers or keywords being arranged in a relative setup. And the main goal is to assist users to prioritize the reading order from a paper's citation network or to find important papers in a field of research via grouping papers by common keyword sets.

Detail on demand interaction Our friendly user interface allows users to intuitively switch among different views and easily drill down for details. Moreover, we provide animation and visual cues to help our users better comprehend the relative transitions during the interaction.

History review tree A history review tree mechanism is provided in *PaperVis*. To prevent the users from getting lost, we believe that it would be much more powerful and flexible to visually review the visualization history under a tree structure, instead of the undo/redo framework.

3.1. System Overview

Figure 1 provides a quick look of the user interface of *PaperVis*. There are totally five regions. The first one is the *system configuration region*, as marked in *A*. Within this region there are four further *controls* that can be selected. In *Mode*, a user can select one of the following three modes: *Citation-Reference Mode*, *Keyword Mode* and *Mixed Mode* to gain different perspectives of the involved paper network. In *scope*, a user can select to display the *reference* and/or *citation* of a selected paper, and restrict the exploration *level* of the network. *Color Coding* shows the current color settings for each level, as well as for the citation/reference relationships to the focus node. *Common Keyword* and its configurations are mainly used for paper clustering by counting common keyword sets. The second one, marked in *B*, is the *central view control region*. It records the user interaction history in a tree structure which offers a convenient interface for the reviewing purpose. As *PaperVis* is capable of performing animation during view transitions, users can also press the buttons of *play*, *pause* and *stop* to explicitly control the playback of the associated animations if any. The third region, marked in *C*, is the *data filtering and selection region* that allows a user to type a keyword to search for an interested keyword or paper, which is then used as the central focus node in region *E*. The fourth region, marked in *D*, is the *detailed information region*, which shows the detailed publication information of the central node. And finally, the *central view region*, marked in *E*, displays the visualization results according to the settings in region *A* or *B* and supports the following operations:

1. Move the mouse cursor over a node (either a paper or a keyword): the detailed information of the node will be popped up, and an example is shown in the light yellow rectangular box in region *E* of Figure 1.
2. Click on a paper node: the citation/reference relationships among the clicked paper and the other papers will be drawn. Straight lines represent the references, and dashed lines refer to the citations of the clicked paper.
3. Double click on a node: refocus by setting the selected node as the next central focus node.
4. Right click on a paper node: if the paper is stored in the local computer, then *PaperVis* opens the PDF file, or otherwise links to the webpage on the *ACM digital library*.

3.2. Citation-Reference Mode

We first describe the definition of *relevance, importance* and *level* used in this paper. To ease the description, we denote by p_{sel} the user-selected paper, by p_{other} any other paper in the dataset, by P_i the set of papers whose levels equal to i, with respect to p_{sel}, where $i = 1,2,3,... \infty$, by $Cite(X)$ the papers that have referenced the set of papers in X, by $Ref(X)$ the papers that the set of papers in X have referenced, and by $N(X)$ the number of papers in X. Given a user-selected paper p_{sel}, for each paper node p_{other}, we first define its *relevance*, denoted as $Node_r(p_{other}, p_{sel})$, with respect to p_{sel} as:

$$Node_r(p_{other}, p_{sel}) = \frac{N(Ref(p_{sel}) \cap Ref(p_{other}))}{N(Ref(p_{sel}))} \tag{1}$$

It basically measures the *co-reference percentage*. And it is evident that, the higher the relevance, the more papers they have co-referenced.

Second, its *level*, denoted as $Node_{level}(p_{other}, p_{sel})$, with respect to p_{sel}, is defined as:

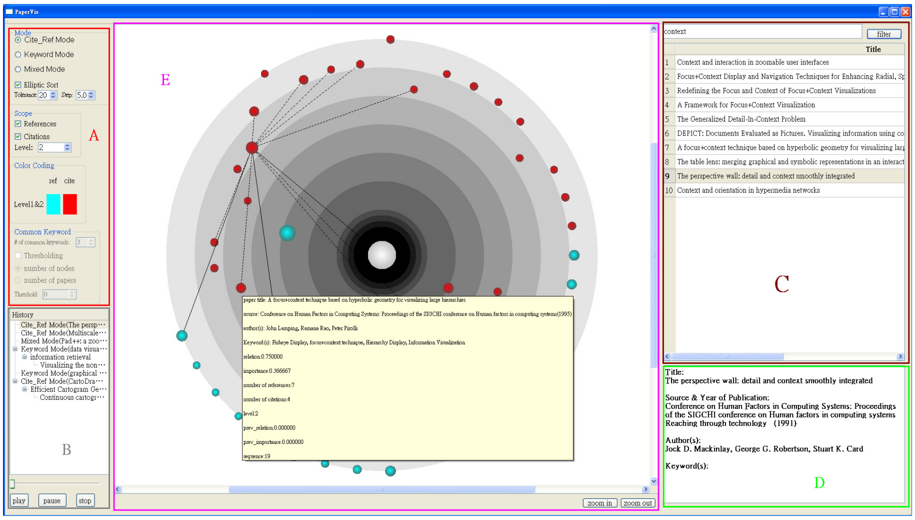
$$\begin{aligned} Node_{level}(p_{other}, p_{sel}) = & \\ & 1,\ if\ \ p_{other} \in Ref(p_{sel}) \\ & 2,\ if\ \ p_{other} \in Cite(p_{sel}) \\ & 3,\ if\ \ p_{other} \in Ref(P_1 \cup P_2) \\ & 4,\ if\ \ p_{other} \in Cite(P_1 \cup P_2) \\ & 5,\ if\ \ p_{other} \in Ref(P_3 \cup P_4) \\ & 6,\ if\ \ p_{other} \in Cite(P_3 \cup P_4) \end{aligned} \tag{2}$$

Notice that, a paper can only be set to a minimum level. For example, two papers (say paper *A* and *B*) are referenced by p_{sel}, so the levels of paper *A* and *B* are both 1. If paper *A* is also referenced by paper *B*, then the level of paper *A* can be 3 and the level of paper *B* can be 4. In this situation, the levels of paper *A* and paper *B* are assigned to be 1.

Finally, its *importance* value, which is denoted as $Node_{imp}(p_{other}, p_{sel})$, with respect to p_{sel}, is defined as:

$$Node_{imp}(p_{other}, p_{sel}) = \frac{N((P_1 \cup P_2) \cap Cite(p_{other}))}{N(P_1 \cup P_2)} \tag{3}$$

As this equation shows, the *importance* of a paper means the percentage of being cited by the papers in P_1 and P_2. We believe that a paper is considered important if it is cited by most of p_{sel}'s direct citations/references. To further ease the description, we abbreviate the notations of $Node_r(p_{other}, p_{sel})$



Mode
● Cite_Ref Mode
○ Keyword Mode
○ Mixed Mode
☑ Elliptic Sort
Tolerance: 20 Step: 5.0

Scope
☑ References
☑ Citations
Level: 2

Color Coding

	ref	cite
Level1&2	🟦	🟥

Common Keyword
# of common keywords: 3
□ Thresholding
● number of nodes
○ number of papers
Threshold: 0

History
- Cite_Ref Mode(The persp…
- Cite_Ref Mode(Multiscale…
- Mixed Mode(Pad++: a zoo…
- Keyword Mode(data visua…
  - information retrieval
    - Visualizing the non…
- Keyword Mode(graphical …
- Cite_Ref Mode(CartoDra…
  - Efficient Cartogram Ge…
    - Continuous cartogr…

paper title: A focus+context technique based on hyperbolic geometry for visualizing large hierarchies

source: Conference on Human Factors in Computing Systems: Proceedings of the SIGCHI conference on Human factors in computing systems(1995)

author(s): John Lamping, Ramana Rao, Peter Pirolli

Keyword(s): Fisheye Display, focus+context technique, Hierarchy Display, Information Visualization

relation:0.750000

importance:0.366667

number of references:7

number of citations:4

level:2

prev_relation:0.000000

prev_importance:0.000000

sequence:19

context

	Title
1	Context and interaction in zoomable user interfaces
2	Focus+Context Display and Navigation Techniques for Enhancing Radial, Sp
3	Redefining the Focus and Context of Focus+Context Visualizations
4	A Framework for Focus+Context Visualization
5	The Generalized Detail-In-Context Problem
6	DEPICT: Documents Evaluated as Pictures. Visualizing information using co
7	A focus+context technique based on hyperbolic geometry for visualizing larg
8	The table lens: merging graphical and symbolic representations in an interact
9	The perspective wall: detail and context smoothly integrated
10	Context and orientation in hypermedia networks

Title:
The perspective wall: detail and context smoothly integrated

Source & Year of Publication:
Conference on Human Factors in Computing Systems: Proceedings of the SIGCHI conference on Human factors in computing systems Reaching through technology (1991)

Author(s):
Jock D. Mackinlay, George G. Robertson, Stuart K. Card

Keyword(s):

Figure 1: *The user interface of* PaperVis. *The central area, region E, is for primary visualization. Region A contains the configuration options, while region B is used for displaying the viewing history. Data filtering and selection controls are placed in region C. Finally, details of the currently selected paper are shown in region D.*

to $Node_r$, and similarly for other notations, when there is no ambiguity. We then map the values of *relevance*, *importance* and *level* to the attributes of a node-link graph. The importance value is used to decide the size of a node, or more specifically, the width and height of a node. The value of relevance determines the distance between p_{sel} and p_{other}. Since the intuition is to position a node with higher relevance in a closer distance, we made $Node_r = 1 - Node_r$. And the hue channel in HSV color space is used to separate nodes into different colors according to its *level*. The central node p_{sel} will have the maximum size, denoted as $Node_{size}$, and is drawn in white. Note that the values of relevance and importance range from $0 \sim 1$. We will have to make a little adjustment on $Node_r$ and $Node_{imp}$ to avoid two paper nodes which have the same sets of references from sticking to each other and a paper node whose importance is zero from disappearing, respectively. Thus, we set $Node_r = \alpha + (1 - \alpha) \cdot Node_r$ and $Node_{imp} = \beta + (1 - \beta) \cdot Node_{imp}$ where α and β is currently set to be 0.1 and 0.3, respectively.

The next step concerns the layout design, since the node sizes and pairwise relevance between nodes have been determined. There are several existing works, including Kohonen's *Self-Organizing Maps* [Koh97], Fruchterman's *Force-Directed Placement* [FR91], and Frick et al.'s *GEM* algorithm [FLM94], etc. These approaches try to equally distribute the nodes and minimize edge crossings while still maintaining the relationships among nodes. However, they are relatively time-consuming, and therefore are not suitable for paper network visualization, where interactive exploration is almost unavoidable. In addition, the distance between 2 nodes may become less meaningful, which is the very drawback that *PaperVis* strives to improve. We finally choose *Radial Space Filling* approach [AH98] and the *Bullseye View* [CK97], as they can present a hierarchical structure of nodes and save the screen space at the same time. More works along this line can be seen from Yang et al.s approach on *InterRing* [YWR02], Stasko et al.'s *Focus+Context* approach [SZ00], Keim et al.'s *CircleView* [KSS04] and Collins et al.'s *DocuBurst* [CCP09]. Nevertheless, the citation network is not purely hierarchical. If we use this kind of algorithm to place the nodes, the problem of *duplicated nodes* could happen, that is, the same node may appear more than once. Our goal is to spatially arrange the paper nodes, relative to the central node, so that nodes would not collide with each other while the distances among them respecting their corresponding relevance values.

Such a *node placement* problem is shown to be *intractable*. To show that, we remove the constraints on node sizes by focusing only on node positions, as the consideration of both can only complicate the problem even further. Formally speaking, the node placement problem can be formulated into an *optimization problem* by minimizing the cost function C defined as the following:

$$C = XL_0X^T$$
$$subject\ \ to\ \ XL_iX^T = B_i,\ \ for\ \ i = 1, 2, \ldots, n \qquad (4)$$

where $X = x_0, x_1, \ldots, x_n$, and x_i is a 1 by 2 matrix, which means the position of node i in the 2D space. n is the number of other papers loaded. Note that node 0 represents p_{sel}, while nodes 1 to n the other papers. $L = D - W$, where W is a n by n matrix, and $W_{ij} = Node_r(p_i, p_j)$. D is a diagonal matrix, where $D_{ii} = \sum_j W_{ij}$. Equation 4 is a well-known problem of *Quadratically Constrained Quadratic Program* [BV09], and can be shown as a *NP-Hard Problem*.

As the primary goal of *PaperVis* is to serve as an interactive and efficient paper network exploration tool, we opt for modifying the aforementioned *RSF* and *Bullseye layout*. A simplified solution would be to only concern about the relevance (distances) between the selected paper and any other paper. The algorithm for node placement and node attribute assignment therefore works as follows:

1. A user first specifies the levels and the scope of the citation network to be explored and selects a paper of interest.
2. *PaperVis* loads the bibliographic information of the selected paper according to the user defined parameters. The relevance among the selected paper and the loaded papers, and the sizes and colors of the loaded papers are calculated. More precisely, the size of a node is decided by $Node_{size} \cdot Node_{imp}$. The hue in *HSV color space* of a node is decided by $(Node_{level}/max\_level) \cdot 360$.
3. The papers are distributed into 10 bins according to their relevance values, i.e., to the bin of ($Node_r \cdot$ 10). In each bin, we sort the papers first by relevance, second by importance, and last by level.
4. The 10 bins form 10 concentric circles, as the circle numbers are labelled in Figure 2(a), marked in yellow and black. Each circle grows outwardly from the previous circle. And each of them has a background color with different intensities for identification purpose. The radius of each circle is denoted as c_i, i = 1,2,3,...,10. For placing the paper nodes into the circles, we start from bin 1 and first check whether there is any paper in the bin or not. If not, then the growing size of that circle is set to be the minimum value. On the contrary, if the bin contains at least 1 paper, the radius of the circle grows with $Node_{size}$. And the node placement process begins with an initiated angle θ, a moving angle Δθ(360/number_of_papers_in_the_bin) and an angle accumulator $A(0, initial)$. The radius of a paper node is computed by $c_{i-1} + ((Node_r \cdot 10) - i) \cdot (c_i - c_{i-1})$. The possible position (x, y) of a node is then calculated as (cos(θ) · radius of the paper, sin(θ) · radius of the paper). And $\theta = \theta + \Delta\theta, A = A + \Delta\theta$. If a node is going to collide with other nodes at the current calculated position, then the possible position needs to be updated again until no collision would occur. If the accumulate angle A exceeds 360 degree, it means the circle should be expanded to accommodate more nodes. Circle 10 in Figure 2(a) shows such an example.
5. Repeat step 4, until all paper nodes have been assigned to a position on the central view.

By following the previous steps, we are able to place the nodes onto the central view, and the relevance (distances) between any other nodes and the central focus node are also preserved. However, maintaining the relationships among the papers other than the focus node may also be preferred when we would like to see the correlation among them. Therefore, a more complicated node placement scheme is considered. For that, we make modification to the fourth step of the proposed approach. First, we calculate the relevance among any pair of loaded papers and form a symmetric correlation matrix. Next, we leverage the power of elliptic sort [Che02] to acquire an ordering of all nodes concerning their pairwise relevance. The ordering sequence is denoted as $Node_{seq}$ hereafter. Instead of setting a random initiated angle for node placement, we further restrict the possible node position to be within an angular span. The span is computed according to $Node_{seq}$, *tolerance* and *step* in Figure 1, region *A*. The original angle θ of a node is decided as $(360 \cdot (Node_{seq}/number\_of\_loaded\_papers))$. If a node is going to collide with other nodes at the current position, then $\theta = \theta \pm$ *step*, A = A + *step*. If the accumulated angle *A* exceeds a *tolerance*, it means the concentric circle should be expanded to accommodate more nodes. Although we have successfully leveraged the elliptic sort so that relevant nodes would be placed closer to each other in an angular fashion, the involved computation effort is non-negligible. Therefore we leave the enabling of such an ordering as an option for users (shown in Figure 1, region *A*).

Another important feature of *Citation-Reference Mode* is the use of visual cues to assist users to get better comprehension during interaction. When a user double-clicks on a paper node, the paper will be set as the new central node, and the current view will be transformed into a new view based on the selected paper and the parameters. However, a user may not be able to find connection between these two views. That is, which nodes have appeared in the last view and what their previous settings are. Thus, *PaperVis* provides visual cues, which include the *alpha value* and *arc length* along a node boundary, to assist the connection. In particular, the *alpha value* refers to its previous importance, the *arc length* refers to its previous relevance, and the color of the arc is set to be the node color in its previous view to make clear the corresponding relationships.

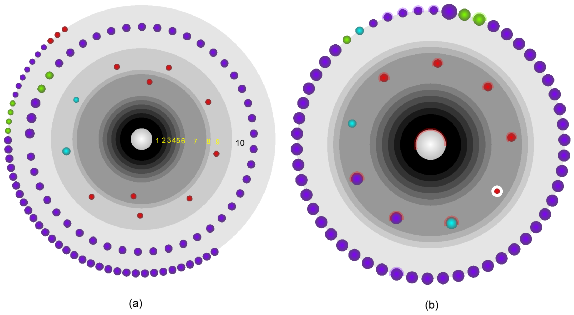


Figure 2: *Examples of* Citation-Reference Mode. *(a) The visualization result after a paper of interest being selected. The bin circles, marked in yellow and black, are expanded accordingly to accommodate the nodes in those bins. (b) In a refocused view, colored boundaries of nodes show their status in the previous view.*

3.3. Keyword Mode

The second viewing mode is *Keyword Mode*, and the main purpose of this mode is to find papers with common keyword sets or use keywords as a category to cluster papers into groups to find important papers in certain research areas. The closely-related algorithm, *CaseCluster* [AO06], limits a node to appear only once at each level, which may result in information loss or even discarding the optimal clustering at the early stage. Zhang et al.'s work [ZCL09] made use of the *FP-tree* algorithm to construct a hierarchical structure of a paper network. Our algorithm for demonstrating the hierarchical relationships of keywords works as follows:

1. A user selects a specific keyword, and the system loads all papers which contain the chosen keyword. The root node is then formed by the chosen keyword and the papers that contain this keyword. And it is now the current node.
2. Among all the papers in the current node, we collect all the keywords that have appeared in the papers, except the ones appeared in its ancestor nodes. A *histogram* is then constructed by counting the numbers of papers having the same keyword. After that, the keywords are sorted with their paper counts in a descending order.
3. The current node is split into child nodes according to the user specified threshold Γ (or the user can decide not to do so). We provide 2 ways to do the thresholding: "number of nodes" and "number of papers". If "number of nodes" is selected, then the top Γ keywords in the sorted list are created as the child nodes. On the other hand, if "number of papers" is selected, then the keywords with paper counts greater than Γ are created as the child nodes.
4. Set each child node created in step 3 as the current node and repeat step 2 and 3, until there is no keywords can be created as a child node or the number of the hierarchical tree structure reaches a certain number that the user has previously specified. Note that the *number of common keywords* corresponds to the levels in the tree structure and is shown in concentric circles in *RSF layout*.

Table 1 gives a more concrete example that helps to better illustrate the operations of this mode, and such an example is also used to compare with two existing approaches to further demonstrate the effectiveness of *PaperVis*. In this example, assuming there are four papers, and each of which contains a set of keywords. Figure 3 shows the resulting representations from *CaseCluster* [AO06], *FP-tree* [ZCL09], and *PaperVis*. It could be easily observed that paper *A* and *B* have 3 keywords in common. In *PaperVis*, we preserve the maximum number of common keywords between 2 papers, so that papers with similar set of keywords would be grouped into the same node. However, the algorithms in *CaseCluster* and *FP-tree* split paper *A* and *B* into 2 separated clusters in the node splitting process at first level.

We adopt the *RSF* technique for the hierarchical structure visualization in the *Keyword Mode*. The root node forms a complete circle. For other keyword nodes, the sibling nodes share the angle span of their parent's. The angle span and color of a fan-shaped keyword node is determined by computing the proportional number of papers in the node with respect to its siblings'. For example, the angle span of the root node is 360. Assume that the root node has 4 children, and their paper counts are 8, 7, 5 and 4, respectively. Therefore, the angle spans of the keyword nodes are 120,105, 75 and 60, respectively. The hue of each series of keyword node is determined to be 0, 120, 225 and 300, respectively. The lower a tree level, the smaller its alpha value is.

Table 1: *An example used to compare the* Keyword Mode *in* PaperVis *with other works.*

Paper	Keywords Included
A	visualization, data, mapping, 3D
B	visualization, WWW, mapping, 3D
C	visualization, data, tree, graph
D	visualization, data, WWW, design

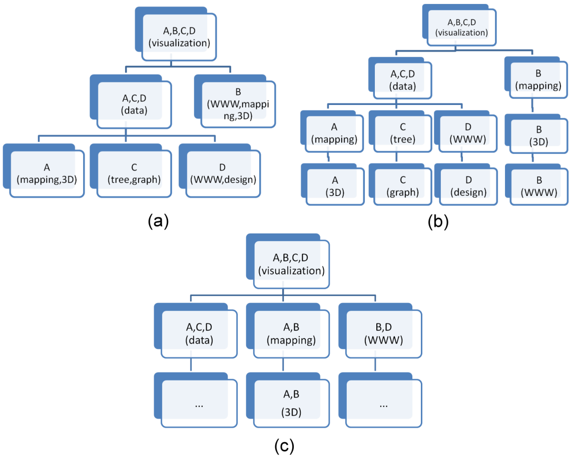


Figure 3: *Results from (a)* CaseCluster *,(b)* FP-tree *and (c)* PaperVis*. Compared with* CaseCluster *and* FP-tree*, our proposed algorithm keeps the largest common keyword sets between 2 papers. Paper A and paper B have 3 keywords in common in this example.*

A user can double-click on any of the keyword nodes to see more details. The node placement in the central view is now processed as that in the *Citation-Reference Mode*. The papers in the root node are loaded. And, the importance and the relevance of the paper nodes need to be recomputed, as the central node is now a keyword, not a paper. The importance of a paper is defined to be the number of papers that cite this paper, divided by the maximum citation count among all the loaded papers. The papers with more keywords in common are then placed closer to the central node. The hue value of the background is the same with the keyword node chosen in the last view. The hue of the nodes is defined as the opposite hue color of the background $((hue_{background}+180) mod 360)$. As the background and nodes move farther away from the central node, the alpha values decrease. Figure 4(b) shows such an example, assuming that the user double-clicks on the keyword node

Internet(5) around the upper left part in Figure 4(a). We can see that there are 22 papers loaded by the keyword *information retrieval*. 7 of them have at least 2 keywords (information retrieval and data visualization) in common, and 5 of them have at least 3 keywords (information retrieval, data visualization and Internet) in common.

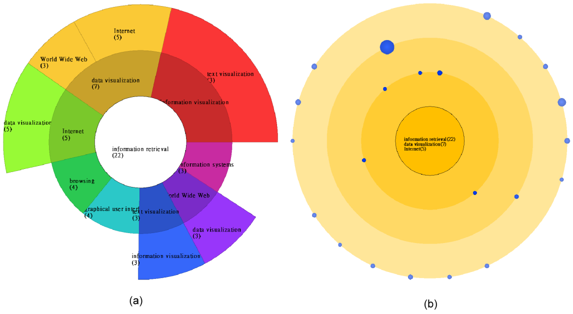


Figure 4: *Examples in the* Keyword Mode*. (a) Start with the papers containing the keyword* information retrieval*, and find their common keyword sets with 3 keywords in common. (b) The associate result after a node in (a) is double-clicked.*

3.4. Mixed Mode

Finally, in the *Mixed Mode*, the basic requirement is that the selected paper *should have at least 1 keyword*. In this mode, after a user selects a paper, *PaperVis* loads the papers just like in the *Citation-Reference Mode*, but arranges the layout as in the *Keyword Mode*, as shown in Figure 5(a). As the algorithm described in *Keyword Mode* step 1, the root node is formed by a main paper and other related papers. In step 2, the keywords are collected from the keyword list of the selected paper. Steps 3 to 5 are the same as in *Keyword Mode*.

When a user double-clicks a keyword node, *PaperVis* also transforms the central view into *Citation-Reference Mode*, as shown in Figure 5(b). But this time, the importance and level are calculated as in *Citation-Reference Mode*, and the relevance and background color are calculated as in *Keyword Mode*.

4. Case Studies

PaperVis is built on top of a machine with an Intel Core 2 Quad CPU Q9505 2.83GHz and 3GB memory. We demonstrate the effectiveness of *PaperVis* through some potential scenarios for literature review. Our case studies are based on the following dataset (http://www.cs.umd.edu/hcil/iv04contest/info.html). Note that the timings for the series of interactions in the following case studies are within 1 second, since the maximum number of loaded nodes is below 200. And the elliptic sort is used in all of the case studies.

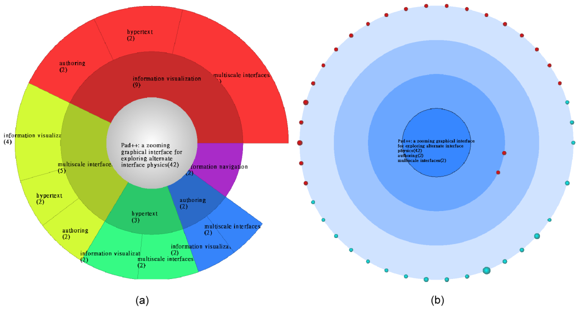


Figure 5: *Examples in the* Mixed Mode*. (a) Exploring papers with 2 common keywords within a paper's bibliographic data. (b) The associate result after a node in (a) is double-clicked. The paper nodes in the outer circle refer to the papers having no keywords in common with the center one.*

4.1. Case Study 1: Show me the relevant or important papers with respect to the central focus paper.

In this case study, we start with a paper of interest named *Managing multiple focal levels in Table Lens*. And we would like to see what papers are relevant or important with respect to the selected one in its citation network of four levels in depth. In Figure 6(a), we can see that the selected paper has only five references (marked in green) and one citation (marked in blue). But there are more than 50 papers which have referenced the papers in the first level of its citation network (marked in red). A possible explanation for this is that the references and the citations of the selected paper seem to be more important than the selected paper itself. In Figure 6(b), we click on the node with the largest size (the green node in bottom-left) to show its citation network in this view. As we can easily observe that almost every other red node has been brought to this view because of it. This kind of pattern tells us that the paper might be in a leading role in a certain area. Because it has been published earlier, it would be impossible for it to reference the papers published afterwards. Furthermore, because it is classical, people keep on referencing it. Thus, the node is being placed with a far distance, and with a relatively large size. Another interesting pattern could also be found in *Citation-Reference Mode*. As shown in Figure 6(a) and circled in red, it is evident that there is a node which is closer to the central node than any of the others. And there is a cluster of papers being placed right next to each other having pretty high relevance to the central node, even though they do not have direct links to the central node. This tells us that these nodes not only are relevant to the central node, but also are highly related to each other. If we click on the node closest to the central node, shown in Figure 6(c), then we can see that it has referenced 4 of 5 papers which are central node's references (solid lines to green nodes). And if we double-click on the node, the view will be refocused and the node will be treated as the new central node, as shown in Figure 6(d). In the newly refocused view, there are 3 references close to the central node. The boundary of each of them is almost a complete red circle, which means that their node colors in the previous view are red. And they are relevant to both the current and the last central nodes. Based on above observations, we can say that if one is reading *Managing multiple focal levels in Table Lens*, then he/she should also read the papers that we have mentioned.

4.2. Case Study 2: Finding important papers to read for a certain research field.

In *Keyword Mode*, one goal is to find papers with significant impact on specific categories. For this purpose, we start with the keyword *information visualization*, and search through the dataset to see that how many papers contain the keyword and have 3 keywords in common. The result is shown in Figure 7(a). Assume that a user is interested in seeing the relationships among the keywords *information visualization*, *World Wide Web* and *hyper text*. The node *hypertext(3)* is then double-clicked, and we expand all the edges coming out from the three nodes in the inner circle. As can be seen in Figure 7(b), the insights we can obtain from this view is that the three papers have certain relationships among each other: either they have direct links to each other or they have referenced the same paper. And we can conclude that if one is interested in doing a research involving *information visualization*, *World Wide Web* and *hypertext*, these three papers are definitely needed to be read.

4.3. Case Study 3: Show me which field the selected paper is mostly related to.

In the *Mixed Mode*, we select a paper, entitled *Stretching the rubber sheet: a metaphor for viewing large layouts on small screens*, and find that the paper itself and 15 out of its 30 references or citations share the same keyword *information visualization*, as shown in Figure 8(a). Then we double-click the keyword node *information visualization(15)*. There are 2 things might be interested in the refocused view as shown in Figure 8(b). First, the more important paper nodes (those with larger sizes) seem to have stronger correlation among each other than the others, as they are positioned next to each other at the upper arc of the first circle. Second, we can see that all of the 15 papers with the common keyword "information visualization" are its citations (red nodes). And we want to know why the other 15 papers do not have the keyword in common. So we move the mouse cursor over them to see their detailed information. We found that some of them have been published at earlier times, and were not tagged with any keyword. The others are tagged with keywords having similar concepts like *data visualization*, *visualizations*, etc. We can say this paper is closely related to *information visualization*. As we double-click on the biggest red node in Figure 8(b), the result is shown in Figure 8(c). We can see lots of nodes having red or blue boundaries. It means that those papers have direct relationship to both previous and current selected central nodes. The near complete boundaries mean that the papers have higher relevance with the first selected paper, and they also have been positioned close to the second selected paper. In Figure 8(c), the boundary of the central node is more than half circle, and the node with a white boundary (the first selected paper) is also close to the central node. We can say that these 2 papers are really relevant. The boundaries of the other nodes are worth observing as well. For example, the biggest blue node on the lower left of Figure 8(c) is also very important to both selected papers, as the alpha value of its boundary is high. Also, the two red nodes on the upper left are worth mentioning in this view, where they have high correlation to each other and their boundaries are nearly complete circles with high alpha values.

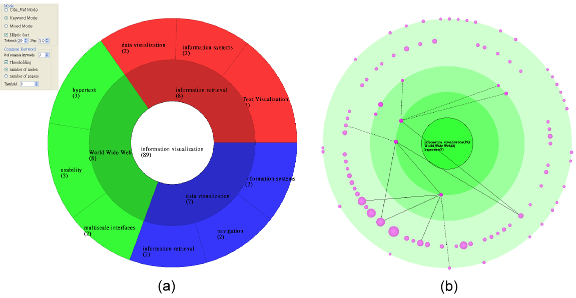


Figure 7: *Results of Case Study 2. (a) The numbers of papers contain the keyword* information visualization *and have 3 keywords in common are visualized. (b) The view is refocused with the node* hypertext(3) *in (a). More information could be dug out by expanding the citation/reference relationships of the 3 nodes in the inner circle.*

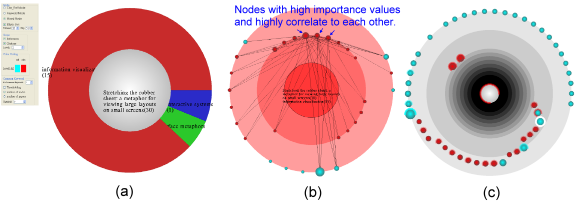

Mode
○ Cite_Ref Mode
○ Keyword Mode
● Mixed Mode
☑ Elliptic Sort
Scope
☑ References
☑ Citations
Level: 2
Color Coding
ref cite
Level1&2
Common Keyword
☐ Thresholding
information visualization (15)
Stretching the rubber sheet: a metaphor for viewing large layouts on small screens(30)
interactive systems (1)
face metaphors
Nodes with high importance values and highly correlate to each other.
Stretching the rubber sheet: a metaphor for viewing large layouts on small screens(30) information visualization(15)
(a)
(b)
(c)

Figure 8: *Results of Case Study 3. (a) There are 15 out of the selected paper's 30 references or citations share the same keyword. (b) After refocusing the view, we found that the papers with higher importance have bigger correlation among each other. (c) Refocusing with the largest red node in (b), and several patterns can be observed as well.*

5. Discussion and Limitation

This section includes some issues worthy of discussion in our current implementation. In *PaperVis*, we make use of the size of a node and the distance of 2 nodes for denoting importance and relevance, respectively. Both of the values are derived based on the bibliographic meta data. However, the formulation could also be application-specific. For example, one could take user interests into consideration [vHP09] or take more care on the *semantic similarities* of paper titles or

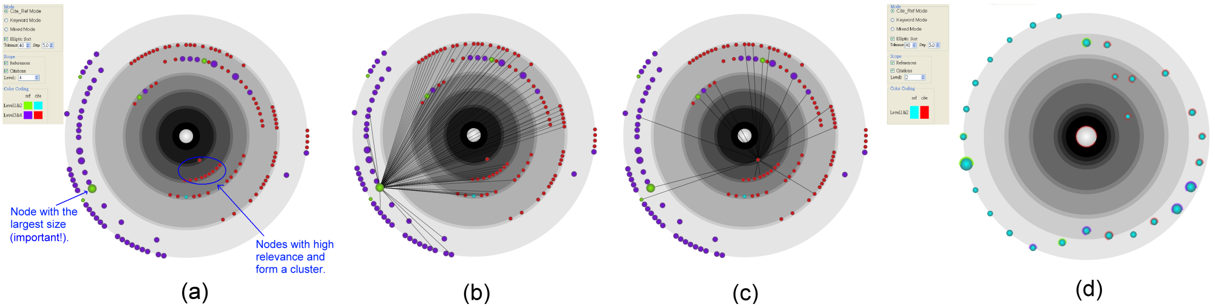

Node with the largest size (important!).
Nodes with high relevance and form a cluster.
(a)
(b)
(c)
(d)

Figure 6: *Results of Case Study 1. (a) The paper entitled* Managing multiple focal levels in Table Lens *is selected. And two evident patterns could be easily observed. (b) Expanding the citation/reference relationships of the node with the largest size. (c) Expanding the citation/reference relationships of the node which is the closest to the central node. (d) Refocusing on the node selected in (c). And some interesting things can also be found by inspecting the nodes with colored boundaries.*

keywords [Son98]. By modifying the definition of the relationship between two nodes, users are able to observe data from different perspectives.

The timing results and the scalability of *PaperVis* are summarized as follows. Loading the dataset into main memory and preprocessing time takes around 5 to 6 seconds. During the interactions, *PaperVis* wishes to provide users prompt responses. There are two factors hinder us from reaching this goal. The first factor is the time to calculate the elliptic ordering in *Citation-Reference Mode*. The elliptic sort algorithm iteratively generates the sequences of N samples from the symmetric correlation matrix until convergence. In each iteration, it takes O(N^2) to compute the correlation coefficients among N samples. The algorithm stops if the derived sequences remain the same in 2 consecutive iterations. Under the hardware specification used in this paper, the result of elliptic sort can be derived within 3 seconds when N is smaller than 500, which is normally sufficient for dealing with four levels of a paper's citation/reference network. The second influential factor is the clustering algorithm in *Keyword Mode*. The algorithm, exhaustively searches for all possible common keyword sets if no threshold is adopted, thus potentially yielding unnecessary and meaningless results. As a consequence, a proper parameter configuration is needed.

We compare *PaperVis* with recent researches, e.g. [vHP09] and [BA09], and point out the strengths and the potential weaknesses of our work. PaperVis keeps more relevant papers within shorter distances, more important papers with larger node sizes, and uses node colors to represent the paper levels. We believe that it is not only more intuitive, but also assists users to quickly identify interested papers. Another advantage of PaperVis is that we have maintained the coherence and histories among transitions, which makes our users keep on track while navigating in the citation network. Nevertheless, [vHP09] and [BA09] include more attributes, other than paper citations and keywords, in their analysis, which provides broader perspectives. [vHP09] takes user interests into account and [BA09] collects more detailed information, such as publication year and authors, etc.

6. Conclusion and Future Work

We have proposed a visualization framework, called *PaperVis*, which has made the task of literature review relatively easier. Three feature modes, namely *Citation-Reference Mode*, *Keyword Mode* and *Mixed Mode* are provided to explore an interested set of papers from different perspectives, while at the same time being user-friendly, intuitive, and interactive. We adopt *Radial Space Filling* and *Bullseye* to efficiently utilize screen space. Animations and visual cues are rendered during view transitions to vividly demonstrate the before-after correspondence. Several case studies are conducted to further demonstrate the usefulness and effectiveness of *PaperVis*. There is, however, still much room for further improvement. By now, we only allow a user to have one focus node in the center of the view. It would be challenging to provide the choice of multiple focus points. This might be useful for users, but the difficulties are the measurement of the relationships among nodes and how we place the nodes accordingly. Moreover, the coverage of analysis could be expanded by analyzing the publication years, authors, or even the content of papers, thus deriving a broader view of a paper's citation network. Another possible improvement could be to ameliorate the keyword clustering algorithm with a semantic analysis. For example, the terms "data visualization" and "information visualization" imply similar concepts in semantics. Papers with these keywords may be separated into different groups, but actually should have been classified in the same cluster. Finally, we believe *PaperVis*, given its current form, is already equipped with many merits or generalities that make it also applicable in the exploration or visualization of other domains as well.

Acknowledgements

We would like to thank Prof. Kwan-Liu Ma and Christopher Muelder from UC Davis for their valuable comments before the paper submission, and Prof. Chun-Houh Chen and Chiun-How Kao from Academia Sinica for providing the source code of elliptic sort.

References

[AH98] ANDREWS K., HEIDEGGER H.: Information Slices: Visualising and Exploring Large Hierarchies using Cascading, Semi-Circular Discs. In *Proceedings of IEEE Symposium on Information Visualization (InfoVis), Late Breaking Hot Topics* (1998), pp. 9–12. 1, 2, 4

[AO06] APITZ G., OGALE N.: Casecluster: Visualizing Case References between Supreme Court Cases, 2006. 6

[BA09] BERGSTROM P., ATKINSON D. C.: Augmenting the Exploration of Digital Libraries with Web-Based Visualizations. In *Proceedings of the 4th International Conference on Digital Information Management* (2009). 2, 9

[BGMZ97] BRODER A. Z., GLASSMAN S. C., MANASSE M. S., ZWEIG G.: Syntactic Clustering of the Web. *Computer Networks and ISDN Systems 29* (1997), 1157–1166. 2

[BV09] BOYD S., VANDENBERGHE L.: *Convex Optimization*. Cambridge University Press, 2009. 4

[CCM09] CRNOVRSANIN T., CORREA C. D., MA K.: Social Network Discovery Based on Sensitivity Analysis. In *Proceedings of the 2009 International Conference on Advances in Social Network Analysis and Mining* (2009), pp. 107–112. 2

[CCP09] COLLINS C., CARPENDALE S., PENN G.: Docuburst: Visualizing document content using language structure. In *Proceedings of Eurographics/IEEE-VGTC Symposium on Visualization (EuroVis '09)* (2009), pp. 1039–1046. 4

[Che02] CHEN C. H.: Generalized Association Plots for Information Visualization: The Applications of the Convergence of Iteratively Formed Correlation Matrices. *Statistica Sinica 12* (2002), 7–29. 5

[CK97] CARRIERE S. J., KAZMAN R.: WebQuery: Searching and Visualizing the Web through Connectivity. *Computer Networks and ISDN Systems 29* (1997), 1257–1267. 1, 2, 4

[FLM94] FRICK A., LUDWIG A., MEHLDAU H.: A Fast Adaptive Layout Algorithm for Undirected Graphs. In *Proceedings of the DIMACS International Workshop on Graph Drawing* (1994), pp. 388–403. 4

[FR91] FRUCHTERMAN T., REINGOLD E.: Graph Drawing by Force-directed Placement. *SOFTWARE – PRACTICE AND EXPERIENCE 21* (1991), 1129–1164. 4

[Fur86] FURNAS G. W.: Gerneralized Fisheye Views. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (1986), pp. 16–23. 2

[Koh97] KOHONEN T.: *Self-Organizing Maps*. Springer, 1997. 4

[KSS04] KEIM D. A., SCHNEIDEWIND J., SIPS M.: CircleView: A New Approach for Visualizing Time-related Multidimensional Data Sets. In *Proceedings of the working conference on Advanced visual interfaces* (2004), pp. 179–182. 4

[MRC95] MACKINLAY J. D., RAO R., CARD S. K.: An Organic User Interface for Searching Citatino Links. In *CHI '95* (1995). 2

[Son98] SONG Y.: BiblioMapper: A Cluster-Based Information Visualization Technique. In *Proceedings of the 1998 IEEE Symposium on Information Visualization* (1998), pp. 130–136. 2, 9

[SOTM06] SHEN Z., OGAWA M., TEOH S. T., MA K.: Biblioviz: A System for Visualizing Bibliography Information. In *Proceedings of the 2006 Asia-Pacific Symposium on Information Visualisation* (2006), pp. 93–102. 2

[SZ00] STASKO J., ZHANG E.: Focus+Context Display and Navigation Techniques for Enhancing Radial, Space-Filling Hierarchy Visualizations. In *Proceedings of the IEEE Symposium on Information Vizualization 2000* (2000), pp. 57–64. 4

[vHP09] VAN HAM F., PERER A.: "Search, Show Context, Expand on Demand": Supporting Large Graph Exploration with Degree-of-Interest. *IEEE Transactions on Visualization and Computer Graphcs 15* (2009), 953–960. 2, 8, 9

[YK99] YANG C., KAO C.: Visualizing Large Hierarchical Information Structures in Digital Libraries. In *Proceedings of the Second Asian Digital Library Conference. Taipei, Taiwan, ROC* (1999), pp. 217–225. 2

[YWR02] YANG J., WARD M. O., RUNDENSTEINER E. A.: InterRing: An Interactive Tool for Visually Navigating and Manipulating Hierarchical Structures. In *Proceedings of the IEEE Symposium on Information Visualization 2002* (2002). 4

[ZCL09] ZHANG J., CHEN C., LI J.: Visualizing the Intellectual Structure with Paper-Reference Matrices. *IEEE Transactions on Visualization and Computer Graphics 15*, 6 (2009), 1153–1160. 6

[ZQGS08] ZHANG X., QU Y., GILES C. L., SONG P.: CiteSense: Supporting Sensemaking of Research Literature. In *CHI '08* (2008), pp. 677–680. 2