EG UK Theory and Practice of Computer Graphics (2007)
Ik Soo Lim, David Duce (Editors)

Bifocal Radial Visualization of Intranet Search Results using Image Caching

Andreas. J. Zschimmer and Jonathan C. Roberts

Computing Laboratory, University of Kent, UK

Abstract

This paper presents a novel image caching strategy implemented for our bifocal radial visualization for Intranet search visualization. This visualization displays search results by combining hierarchical information with focus+context. It is designed specifically for domain or Intranet searches, where specific search results are overlaid (highlighted) on top of a hierarchical circular visualization, which enables these results to be viewed in context with the whole dataset. Drawing large hierarchies is time-consuming, hence the main visualization is cached in a unique way to provide interactive visualization and manipulation. The caching is complicated by the use of the bifocal distortion where a portion of the visualization is magnified, thus two sets of images are cached. This paper describes the caching technique in detail.

Categories and Subject Descriptors (according to ACM CCS): H.5.2 [Information Interfaces and Presentation]: User Interfaces; I.7.4 [Computing Methodologies]: Document and Text ProcessingElectronic Publishing

Keywords: search result visualization, image caching, focus+context, bifocal display

1. Introduction

Search engines can retrieve a very high volume of documents for a user's query. This is true for both Internet search engines such as Google, Yahoo and AltaVista, as well as for those integrated into websites and Intranets (such as through ht:Dig or Google's domain search using "site:http"). Typically, both kinds of search engine present the results in rank-ordered lists on multiple pages, and results can number many thousands if not millions. The presentation of typically 10-50 results per page in a rank-ordered list (as used by the majority of search engines) does not provide a concise overview of all the results nor an indication of how one result relates to another (i.e. topical result groups). In addition, navigation of the large number of result-pages is time-consuming, rendering this form of presentation inadequate for the user to quickly focus on relevant results or groups of results.

When a user does a search on an Intranet (on a local server) they may be interested in locating specific files or information related to the domain. For example, users may wish to search a Company database for a specific finance report, or a user may wish to find a specific file on their own disc. These local databases and Intranets contain a lot of implicit meta-data, that could be used to create better search results, topical result groupings, and more importantly could be visualized. Thus, it would be possible and useful to generate a *map* of the Intranet and annotate the search results on top. This would provide context for the displayed results and allow users to more easily find what they were looking for.

In this work we present the bifocal radial visualization (Figure 1). The radial visualization realizes the hierarchical content taken from the website structure, this is *static* information, i.e. this information would only change when a developer adds another webpage to the site, hence we assume that the information of the web-structure does not change during a web-search session. Thus, this visualization is stored in an image cache. But, because we include a magnified area of the display the image caching is not straightforward, hence we cache two sets of images (one for the high magnification and the other for the lower magnification) and merge them together to form the whole visualization. This paper details this novel caching and merging strategy.

It is certainly much quicker to rotate and manipulate the cached image rather than re-drawing the whole hierarchy. Even with recent advances in computing power, large hierarchies are still time consuming to render and hence image-

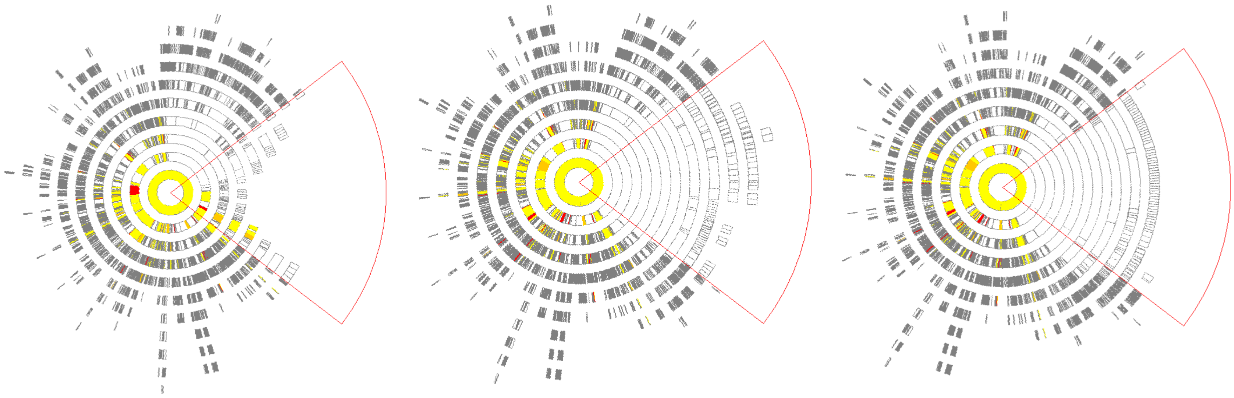

Figure 1: *The images show the bifocal radial visualization. The user can rotate the whole display to show different parts of the visualization at a larger magnification.*

caching techniques are still required. Furthermore, such an image could be quickly transmitted over the web and viewed on a client computer. This would have the added advantage that the same image could be used by multiple users and it would only need to be drawn once. In addition, it would permit an additional level of security if the host organization did not wish to publish the raw data of the hierarchy.

2. Related work

There are two main areas of related research that are relevant to this work: (1) image caching techniques and (2) visualization, including references to web search visualization, focus+context visualizations and circular visualization techniques.

2.1. Caching & its use in Visualization

Caching techniques abound in general computer science, and while image-caching is used in web interaction, computer graphics and rendering it does not feature much in visualization. For instance, CPU and GPU's utilize caches for temporary and quick access of frequently used datum, web browsers may locally cache recently viewed webpages and images, while proxy servers may cache web pages and images for a whole organization.

In computer graphics various image caching methods have been developed. Principally, the approach exploits the fact that as users navigate through a complex scene some parts of the display do not appear to change. Therefore, the rendering speed can be increased in comparison to naively re-rendering the complete scene. One example of this idea is *billboards*, where a two-dimensional bitmap is used in place of the three-dimensional object (such as trees) and are always oriented to face the user. Taking this further, Maciel and Shirley [MS95] introduced the idea of an *imposter*. In this case the object (or scene) is replaced by a rendered version of the scene placed as a texture on a polygon. Later this work developed into hierarchical image caches to accelerate walkthroughs of complex environments [SLS*96, SS96]. Image-based rendering (IBR) utilizes these images to first create a model of the scene that is used to generate some new views of the scene.

Most of the research with visualization on image-caching has focused on scientific visualization and particularly to optimize aspects of the rendering. For example, Yoon et al. [YDKN97] use an IBR image-cache technique to optimize volume visualization, where the cached image is used to create the next frame, which takes advantage of temporal coherence between viewpoint changes. While LaMar and Pascucci [LP03] use imposters to speed up the rendering of scientific visualizations. In information visualization the use of image caching is not well documented, utilized or researched. For instance, Stasko and Zhang [SZ00] do mention that they cache an image but exact details are vague.

2.2. Visualization

Previous work in the field of search result visualization and visualization of hierarchies has seen a large variety of methods and focus. Types of representation include visualizations based on text [RS03], abstracted two-dimensional graphs [SFRG00a, SVM*99], three-dimensional graphs [CLS00, MH97] and enhanced thumbnails [WFR*01].

Text visualizations are obviously the most common: the user is familiar to this interface and it is implicitly clear what the information represents. However, it is difficult to realise large quantities of data, and a complete overview of the information is often hard to achieve (e.g. users of search tools such as Google and Yahoo have to scroll down, or click through next pages, to see more information). Techniques of distortion have been developed in visualization to enable focus+context. In fact Suvanaphen and Roberts [RS03] integrated text distortion techniques with search result queries, while Munzner et al. [MGT*03] in their TreeJuxtaposer provide a technique to visualize and compare large tree hierarchies.

Two dimensional graph visualizations are the closest techniques to our circular visualization method. For example, techniques like DiskTree represent hierarchical structures in a circular two-dimensional form [CPM*98], Sunburst [SZ00] displays a radial visualization of information hierarchies and includes a detail method that enlarges a region of the hierarchy outwards. Lamping et al. [LRP95] use hyperbolic trees in combination with a fish-eye technique to present circular focus+context for a hierarchy. DiskTree is the most similar to our method; however, apart from the visual differences in the layout of the structure and our use of a bifocal distortion, DiskTree annotates usage and change on the general structure while our method annotates search results. In particular, each of these techniques have been used to show directory structures and they have not been integrated with web searching.

On the other hand, techniques such as SQWID [SC97] and Sparkler [HHP*01] do plot search result data onto a 2-dimensional axis but do not show the results in context. For example, SQWID positions the results depending on their similarity with given terms; while Sparkler displays results from multiple search engines on different quadrants of a circle. Shneiderman et al. [SFRG00b] present *hieraxes* which display search results in a two-dimensional table. Each result is represented by dot (or a bar chart); the user can move up and down hierarchies and specific items can be search for and dynamically displayed. Furthermore, Spoerri [Spo06] in MetaCrystal compares the results from multiple search engines, visualizing them in a circular way.

Three-dimensional plots have also been used to display search result data. For example, Cugini at al. [CLS00] plotted individual search results into three dimensions using various methodologies. Many of their methods were circular in nature, where orders of the items were positioned due to either rank or various metrics based on keywords. However, their visualizations merely plotted the search results rather than displaying the results in context with some known data. Sebrechts et al. [SVM*99] present a comparative evaluation of many types of representation, they show that 3D plots are often hard to navigate and interpret. They take longer to locate pertinent information because they comprise of highly abstracted multi-dimensional information. Finally, although not originally designed for web search result data, Cone-Tree [RMC91] is a seminal work on visualizing graph data in a 3D form. This visualization uses animation to move nodes in the tree to the front, such to allow the user to view specific nodes.

It is often difficult to display any visualization in full detail and this is especially the case for web-search result data. There is simply not enough real estate on the screen. Hence distortion techniques are used to display detail in context. Stasko and Zhang [SZ00] also use a bifocal display to display detail, but their method grows a region of the hierarchy outside or inside the main hierarchy circle. Lamping et al. [LRP95] use hyperbolic trees in combination with a fish-eye technique to present detail in context for a hierarchy. The fish-eye magnifies objects in its centre, as a focus of interest, and surrounding objects appear gradually smaller as they approach the outer bounds of the visualization space. In our implementation we use a bifocal display, where a segment of the circle is set at higher magnification. For further details on distortion techniques see the taxonomy by Leung and Apperley [LA94].

3. Design

There are two parts to the radial visualization; first the data is gathered and stored, then the data is visualized.

3.1. Gathering the data

First the structure of the website is gathered and stored in a hierarchical data structure. This is then visualized in the radial visualization.

Second, the user enters a query into the search engine. The search results are obtained by running a search over the whole website html content, retaining only data relevant to the search. The search engine used in this application is ht://Dig [htD], and the test dataset is the University's website. This search engine was used such that it could be closely integrated with the application.

Third, individual search results can then be matched to particular parts of this website-hierarchy. This means that the program needs to find both the point in the hierarchy and where it is plotted on the display; an associated list is used to construct this mapping. It is feasible to extract groupings of similar results and only plot the results where they are located in website structure; in much the same way as the *directories* projects visualize similar information, as described in section 1. However, in those directory visualizations the user would not benefit from viewing the overall context of how the search results fit in with the complete website structure.

3.2. Bifocal Radial Visualization

There are four main components to the radial visualization: arcs akin to the rings of a tree represent the hierarchy, the bifocal display, the results are annotated on top of the static web-structure, and finally the user can interact with the visualization.

First, the centre of the radial visualization symbolizes the top most directory of the website, with each subsequent arc

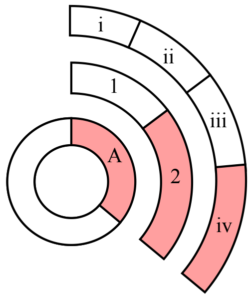

Figure 2: *This figure shows that levels in the hierarchy are represented by subsequent rings, with each leaf node being plotted within the angle of its parent arc.*

of the ring corresponding to a level in the directory hierarchy. Each leaf node is plotted within the angle of its parent arc. This is sketched in Figure 2.

Second, part of the circle is distorted. This bifocal display enables part of the plot to be realized in higher detail. This is especially useful if the hierarchy contains a very large number of directories, in which every "leaf" directory has only a small graphical representation. We use a discrete boundary on the distortion to provide a clear distinction between the high and lower quality. It is possible to imagine other types of distortion, such as smooth distortion or multiple distorted arcs. However, for our initial realization we wanted to keep a simple and uncomplicated design. The user can rotate the visualization to display different parts of the hierarchy at this higher detail. In fact 75° of the circle is distorted and the rings are plotted in higher detail. Essentially, this can be thought of as 5° of the *original* circle being plotted within 75°, and the remaining 355° of the *original* circle being plotted in 285°.

Third, the user's search results are plotted on top of the static web-structure radial visualization. This is achieved by matching the result-URLs directory structure to segments representing the same directory structure within the radial plot. Segments are filled with varying colours to indicate the number of results a directory contains (we use red for high numbers, through orange to yellow for low numbers). This immediately directs the user's focus to areas of interest, where the most hits are located.

Fourth, search queries are entered into the search field and executed by clicking the 'Search' button. After all results have been retrieved and drawn, rotation of the visualization is simply a matter of clicking and dragging with the mouse. If an area of interest is rotated into the overview view, hovering with the mouse over a segment displays its URL and the number of results contained. Clicking on the segment shows all contained result pages' URLs in a list, which can then be selected and displayed in an external browser according to the user's preferences.

4. Technical Implementation

Because the radial visualization acts as a context onto which the results are displayed, it does not change its form. It is only going to change its form when the website is changed. Hence, it is much quicker to pre-process this radial visualization and cache it as an image, and subsequently apply image manipulation operations onto it to provide both the bifocal distortion and the interaction capabilities. Hence the full creation and image manipulation operations are described below.

4.1. Directory structure, creation & storage

The website's directory structure is first extracted into a text file through a recursive directory listing program run on the web-server(s). This file is then converted into a hierarchical XML structure using JAXB (Java Architecture for XML Binding). All classes for interaction with the xml data are automatically generated by the JAXB Binding Compiler (xjc) from an XSD.

On creation, the following attributes in the XML hierarchy are set: the maximum directory depth for the whole structure, the number of subdirectories contained in a directory, each directory's starting degree, extent of degrees covered by the directory and the URL.

4.2. Bifocal Display & Cached Images

The first stage of visualization is the creation of ring segments (arcs) for all directories. This is relatively a simple operation, of calculating the directory hierarchy and working out the sizes of each subsequent arc section, but it is further complicated by the inclusion of the bifocal plot, and the use of cached images. The overall process is sketched in Figure 3. The process is as follows:

First, assuming that 5° of the original plot represents 75° in the bifocal display, the whole directory structure is rendered at normal resolution into a segment of 290°. (i.e. $285^\circ + 5^\circ$). This means that the data either side of the arc (as shown in Figure 3A) abuts each other: points o and o'. This image is cached. In fact, for ease of clipping we cache two versions of this disc: one with the bifocal segment missing on the right, and another with it missing on the left.

Then, the directory data for the detailed-resolution of the bifocal display is rendered. In this case we render a full circle of the data at this higher resolution. Thus, 24° of the data (at high resolution) is plotted into one whole circle (i.e. 360° of detail-resolution correlates to 24° of the normal resolution). This image is then cached. In fact, we pre-render and cache 15 detail-resolution discs. These represent the whole directory data at high-resolution.

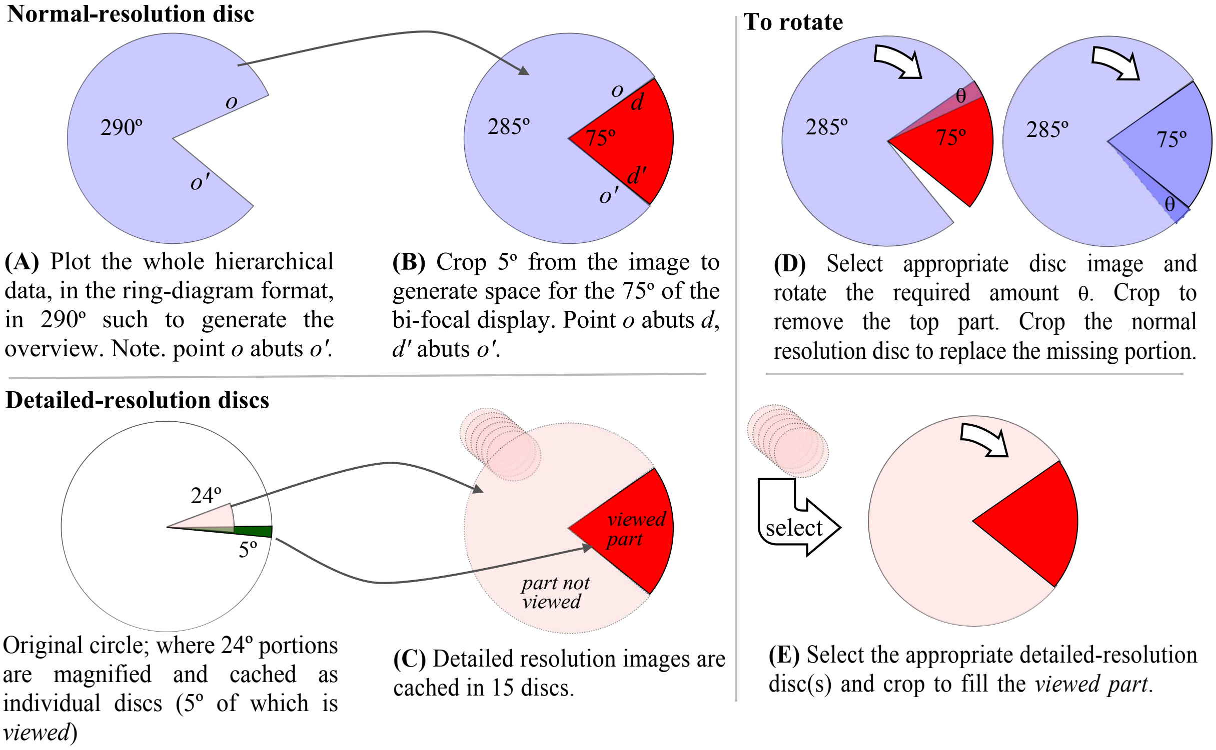

Figure 3: *The diagram shows the caching and cropping procedure for the bifocal display. (A) plot the whole data into* 290° *of a circle. (B) Crop* 5° *from the plotted data (to leave the* 75° *gap for the focus view), (C) plot the data for the focus view, i.e., plot the whole data into 15 separate discs (each* 24° *of the original), hence* 75° *of these secondary discs are used for each focus display. (D and E) Now to do a rotation operation; rotate the main disc, crop to remove the top of the main disc, crop the normal-resolution disc to add the missing portion; select an appropriate detail-resolution disc and crop the required portion to fill the focus part of the display.*

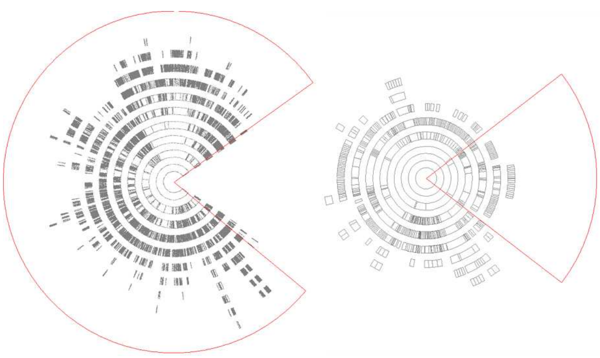

Figure 4: *The bifocal radial visualization; showing a normal resolution disc on the left, and a detail-resolution disc on the right, with cropping arcs annotated in red.*

In order to display the bifocal display, a composite of one normal-resolution image and one of the 15 detail-resolution discs is made, i.e. 5° is clipped from the normal-resolution image, and merged with 75° taken from a detailed-resolution disc. This composite image is then displayed. Figure 4 shows the normal resolution image on the left, with one of the detailed discs on the right.

To achieve a rotation, both resolution discs are rotated. For small angles (i.e. those less than 24°) the appropriate angle θ from the detailed-resolution disc needs only to be rotated and cropped in place. However, by rotating the normal-resolution disc a gap θ would appear, as shown in Figure 3 part D. Hence, a proportion θ of the complementary normal-resolution disc is cropped into place to fill the gap. The representation of all rotations becomes a composite and therefore requires two clipping and drawing operations.

Obviously, when the rotation angle θ is greater than 24° then a new detail-resolution disc would be used instead. Again, there may be a situation when parts from two detail-resolution discs are required; in which case the detail-resolution bifocal segment becomes a composite of two detail-resolution image discs, Figure 5.

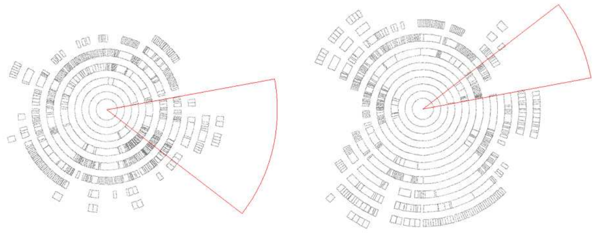

Figure 5: *The angle may overlap two detail-resolution discs, hence two of these discs need to be cropped.*

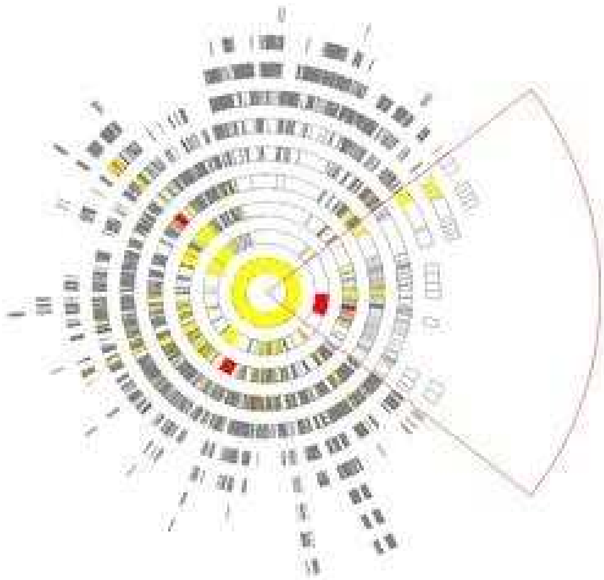

Figure 6: *The search result data is mapped on top of the directory hierarchy; again by storing them into cached images, this time with transparent background.*

4.3. Search-Result Visualization

Now the cached images have been created the results can be annotated, see Figure 6. In fact we use a similar technique for the search result visualization as the cached images above. We create cached 'twin' images that only contain the visualization of the search result data. The search-result information is plotted with the appropriate colour: according to the number of search results contained in the arc segments with the appropriate angles, as above. The only difference is that these result-images are created with a transparent background (RGBA). Transparency is used in order to allow these images to be overlaid (and cropped) onto its twin. Rotation and clipping arcs are only calculated once and applied to both twins, thus reducing the amount of calculations necessary during the drawing process. This overlay method is useful because, the old images are merely exchanged with the new cached-images for a subsequent search.

4.4. Additional enhancements

In addition to the aforementioned caching of discs, various values are cached and stored to increase performance. The 'original' (non-scaled) degree measurements (starting angle and angular extent) of segments are stored within the XML hierarchy to enable fast creation of ring segments at the visualization stage. At the generation stage separate lists, of what data each disc contains, is stored for each detail-resolution image. If the mouse pointer is within the overview area, the segment it points to can be found by searching only the list(s) of the detail-resolution disc(s) currently displayed in the bifocal area. This shortens the search time considerably. Furthermore, this idea is used in the process of finding which result-segments are to be cropped and displayed. During the initial drawing of each detailed-resolution disc of the result-segments the index of the image is recorded in a HashMap along with its URL acting as the key.

5. Discussion

The speed of rotation achieved with the above methods of clipping is considerably faster than the real-time re-drawing of all segments in both detail and enlarged form. However, there are some drawbacks.

Currently we utilize a fixed angular extent of 75° for the bifocal display. This provides a neat way to cache the images, and it is easy to calculate the quantity of detail-resolution discs. But, challenges arise if the user is permitted to change this angle. The developer would need to adapt the cached images when the angle is altered. It would be useful to let the user manipulate this angle to focus on a larger area. Moreover, the bifocal display uses a binary focus+context area, however it would be useful to change this to a smooth focus area.

Another challenge is with the circular format itself. If the number of directories is very large or the resolution at which the visualization is displayed is low then a sub-directory may be realized by a miniscule pixel or subpixel point. This problem is more apparent with leaf nodes that become rendered in the normal-resolution area. In which case the indication of search results may not be visible and the user would not be able to establish which area in the hierarchy contains the most search results. A second drawback is that the neatness of the circular display relies heavily on the well-formed and well-ordered hierarchical organization of the website directories. It is certain that some website hierarchies would not form such a neat visualization. We are currently extending the idea to generate a different *static* visualization that can be used for 'in context' web search result visualization.

6. Future work & Conclusions

This work is part of a larger project looking at search result visualization. In this paper we have presented full technical details of the caching strategy for our bifocal radial visualization, especially designed for Intranet 'in context' visualization. This work is currently being examined and evaluated in various ways. We wish to investigate the usefulness of 'in context' visualization. We predict that such techniques visualization are useful but we need to evaluate how it is used and whether it is more useful than other techniques.

Furthermore, we propose to extend and adapt the bifocal display such to allow multiple bifocal regions, and investigate other non-binary mappings of the focus area. By exchanging the bifocal style of magnification to a multi-focal transition area the user should would perceive a smoother magnification process [LA94], which may be more useful when very high rated of magnification are used. But, both these techniques would complicate the caching and cropping of discs.

The combination of the 'focus+context' method with hierarchical information in a radial format utilizing a bifocal display, with the image-caching method presents a novel visualization. It is feasible to imagine how the image-caching technique could be used for other visualizations. Our bifocal radial visualization provides a full contextual overview of the hierarchy and detailed information about areas of interest at the same time. Navigation of the visualization is simple and its interpretation is intuitive. This allows the user to focus and explore the results themselves.

7. Acknowledgements

We would like to acknowledge the webmasters of Kent University for providing us with the complete directory data.

References

[CLS00] Cugini J., Laskowski S., Sebrechts M.: Design of 3d visualization of search results: Evolution and evaluation. In *Proceedings of IST/SPIE's: Visual Data Exploration and Analysis (VDA 2000)* (San Jose, CA, 2000), IST/SPIE, pp. 23–28.

[CPM*98] Chi E. H., Pitkow J., Mackinlay J., Pirolli P., Gossweiler R., Card S. K.: Visualizing the evolution of web ecologies. In *CHI'98* (Los Angeles, April 1998), ACM, pp. 400–407.

[HHP*01] Havre S., Hetzler E., Perrine K., Jurrus E., Miller N.: Interactive visualization of multiple query results. In *Symposium on Information Visualization (INFOVIS '01)* (October 2001), IEEE Computer Society Press, pp. 105–112.

[htD] http://www.htdig.org.

[LA94] Leung Y., Apperley M.: A review and taxonomy of distortion-oriented presentation techniques. *ACM Transactions on Computer-Human Interaction (TOCHI) 1*, 2 (1994), 126–160.

[LP03] LaMar E., Pascucci V.: A multi-layered image cache for scientific visualization. In *PVG '03: Proceedings of the 2003 IEEE Symposium on Parallel and Large-Data Visualization and Graphics* (Washington, DC, USA, 2003), IEEE Computer Society, p. 9.

[LRP95] Lamping J., Rao R., Pirolli P.: A focus+context technique based on hyperbolic geometry for visualizing large hierarchies. In *Proceedings SIGCHI* (1995), ACM Press, New York, pp. 101–108.

[MGT*03] Munzner T., Guimbretière F., Tasiran S., Zhang L., Zhou Y.: Treejuxtaposer: scalable tree comparison using focus+context with guaranteed visibility. *ACM Trans. Graph. 22*, 3 (2003), 453–462.

[MH97] Mukherjea S., Hara Y.: Focus+context views of world-wide web nodes. In *Proceedings of the eighth ACM conference on Hypertext* (1997), ACM Press, New York, pp. 187–196.

[MS95] Maciel P. W. C., Shirley P.: Visual navigation of large environments using textured clusters. In *SI3D '95: Proceedings of the 1995 symposium on Interactive 3D graphics* (New York, NY, USA, 1995), ACM Press, pp. 95–101.

[RMC91] Robertson G., Mackinley J., Card S.: Cone trees: Animated 3d visualizations of hierarchical information. In *Proc. ACMSIGCHI '91 Conf. on Human Factors in Computing Systems* (New Orleans, Louisiana, 1991), pp. 189–194.

[RS03] Roberts J., Suvanaphen E.: Visual bracketing for web search result visualization. In *7th International Conference on Information Visualization (IV'03)* (2003), IEEE Computer Society, pp. 264–269.

[SC97] S. M., C. K.: Visualizing search results using SQWID. In *Proceedings 6th International World Wide Web Conference* (1997).

[SFRG00a] Shneiderman B., Feldman D., Rose A., Grau X.: Visualizing digital library search results with categorical and hierarchical axes. In *Proceedings 5th ACM conference on Digital libraries* (2000), ACM Press, New York, pp. 57–66.

[SFRG00b] Shneiderman B., Feldman D., Rose A., Grau X. F.: Visualizing digital library search results with categorical and hierarchical axes. In *Proceedings of DL-00, 5th ACM Digital Library Conference* (San Antonio, US, 2000), pp. 57–66.

[SLS*96] Shade J., Lischinski D., Salesin D. H., DeRose T., Snyder J.: Hierarchical image caching for accelerated walkthroughs of complex environments. In *SIGGRAPH '96: Proceedings of the 23rd annual conference on Computer graphics and interactive techniques* (New York, NY, USA, 1996), ACM Press, pp. 75–82.

[Spo06] SPOERRI A.: Visualizing meta search results: Evaluating the metacrystal toolset. In *Proceedings 69th Annual Meeting of the American Society for Information Science and Technology (ASIST)* (Austin (US), 2006).

[SS96] SCHAUFLER G., STÜRZLINGER W.: A three dimensional image cache for virtual reality. *Computer Graphics Forum 15*, 3 (1996), 227–236.

[SVM*99] SEBRECHTS M., VASILAKIS J., MILLER M. S., CUGINI J. V., LASKOWSKI S. J.: Visualization of search results: A comparative evaluation of text, 2d, and 3d interfaces. In *Proceedings 22nd SIGIR* (1999), ACM Press, New York, pp. 3–10.

[SZ00] STASKO J., ZHANG E.: Focus+context display and navigation techniques for enhancing radial, space-filling hierarchy visualizations. In *INFOVIS '00: Proceedings of the IEEE Symposium on Information Vizualization 2000* (Washington, DC, USA, 2000), IEEE Computer Society, p. 57.

[WFR*01] WOODRUFF A., FAULRING A., ROSENHOLTZ R., MORRSION J., PIROLLI P.: Using thumbnails to search the web. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (2001), ACM Press, New York, pp. 198–205.

[YDKN97] YOON I., DEMERS J., KIM T., NEUMANN U.: Accelerating volume visualization by exploiting temporal coherence. In *Proc. IEEE Visualization 97* (1997), pp. 21–24.