

CWIC: Using Images to Passively Browse the Web

Quasedra Y. Brown
Computer Information Systems
Clark Atlanta University
Atlanta, GA 30314
qbrown@cis.cau.edu

D. Scott McCrickard
College of Computing and GVV Center
Georgia Institute of Technology
Atlanta, GA 30332
mccricks@cc.gatech.edu

Advisor: John Stasko (stasko@cc.gatech.edu)

Abstract

The World Wide Web has emerged as one of the most widely available and diverse information sources of all time, yet the options for accessing this information are limited. This paper introduces CWIC, the Continuous Web Image Collector, a system that automatically traverses selected Web sites collecting and analyzing images, then presents them to the user using one of a variety of available display mechanisms and layouts. The CWIC mechanisms were chosen because they present the images in a non-intrusive method: the goal is to allow users to stay abreast of Web information while continuing with more important tasks. The various display layouts allow the user to select one that will provide them with little interruptions as possible yet will be aesthetically pleasing.

1. Introduction

The World Wide Web has emerged as one of the most widely available and diverse information sources of all time. The Web is essentially a multimedia database containing text, graphics, and more on an endless variety of topics. Yet the options for accessing this information are somewhat limited. Browsers are fine for surfing, and search engines and Web starting points can guide users to interesting sites, but these are all *active* activities that demand the full attention of the user. This paper introduces CWIC, a *passive*-browsing tool that presents Web information in a non-intrusive and aesthetically pleasing manner.

CWIC provides the user with new directions to the World Wide Web using images as road signs. CWIC surfs selected Web sites, moving from page to page collecting images. The images are analyzed to identify the ones that graphically capture the meaning of the page. Then, CWIC uses display mechanisms like background wallpaper and screen savers to provide the user with alternative communication mechanisms for Web information. The information is in the form of artistic displays that use images instead of traditional text.

Why use images instead of text? Traditional text-based displays are quite common, but they typically ignore the rich graphical nature of the World Wide Web. The Web is not just text but hypertext, and a major part of that are the images. We do not plan to do standard searching, but rather we want to raise users' awareness in a non-intrusive, secondary manner using displays that do not require extensive cognitive processing. Since human cognition can deal simultaneously with verbal and non-verbal events [Pav86], a system based on images would allow the user to perform other reading, writing, and coding tasks while processing additional visual data from our system. In addition, people have a good memory for images and can distinguish quickly new images [SCH70]. Thus, a system based on image representations may help people

identify new information with little cognitive effort, allowing them to continue with other tasks while staying aware of changes to the Web.

We see a number of potential uses for our system. CWIC could be used to surf news and sports sites to help a user stay up-to-date on breaking stories. Pointing CWIC to local Web sites could enhance a sense of community in the workplace. One user used CWIC to monitor entries in his Netscape bookmark file to stay informed of updates and changes. Of course, there is also a significant entertainment factor to CWIC that could be enhanced by pointing it to comics, museums, and other fun sites.

The remainder of this document will introduce the various components of CWIC: the crawler, the image waiting room, the control panel, and the image display module. Section 3 will discuss in depth several aspects of CWIC, including the probabilistic algorithm for identifying images, the various display layouts, and the display mechanisms. Section 4 will explain how CWIC draws from and contributes to several research areas, including human-computer interaction, information interfaces, and software agents. Section 5 will summarize our contributions and explore future directions.

2. Architecture

CWIC consists of four modules. The modules are the control panel, image waiting room, Web crawler, and the Image Display Module (IDM) (see Figure 2.1). Each module has separate tasks to perform. The links between the modules illustrate the flow of data and how the separate modules communicate with each other. The control panel allows users to manipulate URLs to crawl and control nature of the display. The Web crawler collects images from the Web pages.

The image waiting room is a storage place for all the images that pass the image evaluation. The IDM allows the user to select a layout and a display mechanism for the images. There are four layouts: grid, fade, spiral, and random. There are four display mechanisms: screensaver, wallpaper, standalone application, and Web page.

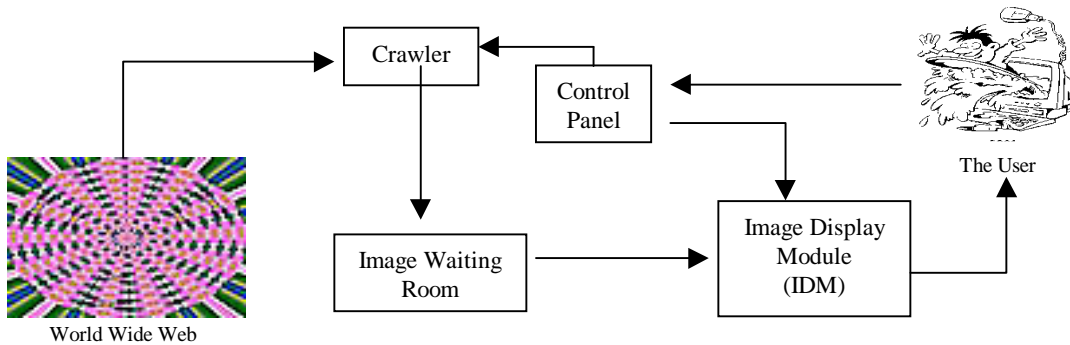


Figure 2.1.1: The crawler collects valuable information such as images and links from the World Wide Web. The images, if pass the evaluation process effectively, are stored into the image waiting room. Once the images are stored in the image waiting room, the IDM displays the images to the user utilizing one of the display mechanism and layout chosen by the user. The user has the ability to customize CWIC to their personal needs by providing important information to the control panel; which in turn communicates the information to the crawler and IDM.

What is a crawler? The crawler is responsible for starting at a given URL (e.g. www.cnn.com) and traversing all active links gathering information such as images and what page they refer to. The images are then sent through a probabilistic algorithm designed to limit the number of poor images like banners and bullets (see Section 3.1). If the image passes the evaluation, then the image is stored in the image waiting room. The crawler is also responsible for fixing incompletely specified links to make all active links to conform to the basic structure, (e.g. <http://www.cc.gatech.edu/index.html>).

Our Web crawler traverses the Web, moving from page to page collecting images. It starts at sites that the user specified in the control panel. Since we have to grab the images from various Web pages, we must have a procedure that verifies the active links as valid. The active links are corrected using a template URL. Once the links are downloaded, the link is compared

against the template URL. If the link does not conform, the link is evaluated for proper corrections by adding the server name, the hypertext tag or other cutting and pasting methods required.

The image waiting room manages which images are stored and processed as well as maintaining the size of the waiting room. The stored here after they correctly pass the evaluation process once the images are retrieved. This evaluation process is discussed in section 3.1, the probabilistic algorithm. When the IDM reaches its limit, the crawler executes a cleanup procedure to remove a percentage of the older images beginning at the front of the image waiting room.

The IDM displays images from the image waiting room utilizing the layout and display mechanism chosen by the user. The IDM controls the manner images are displayed to the user. When possible the links to the original Web page are provided so that the user can click on the image to visit Web pages that seem to be interesting. The layouts control how the images are positioned and displayed to the user. The various display mechanisms that control the manner in which the layouts are viewed is discussed in Section 3.2.

The major function of the control panel is to allow users to configure CWIC. The users can select starting points for the Web crawler from a list of URLs, or the user can manually enter starting points via an URL. The user can select a layout style and a display mechanism that will be used by the IDM.

3.2 Design and Implementation Details

The CWIC system is implemented using the Tcl/Tk language [Ous94], a platform-independent Web-aware scripting language. Since it is platform-independent, it works to various degrees on

multiple platforms. (Some aspects of CWIC require UNIX-based display mechanisms that have not been coded for other platforms.) The Web-aware capabilities of Tcl/Tk make it easy to download information from the Web, and Tcl/Tk contains powerful string-parsing routines that simplify the manipulation of the information.

The remainder of this section will discuss some of the interesting design and implementation problems we encountered in creating the probabilistic algorithm and display modules. We will also look at how the user input contributed to the design of CWIC.

3.1 Probabilistic Algorithm

The World Wide Web is full of images such as banners, buttons, and navigational bars that do not graphically capture the meaning of a page. In order to assure the user will not be displayed hundreds of banners or buttons; a probabilistic algorithm was constructed to filter out undesirable images. The algorithm has several constraints to reduce the chance that the user will be displayed poor images. The crawler utilizes this probabilistic algorithm to evaluate whether or not an image is suitable to display to the user.

While efficient filtering of text is a widely studied problem, there are fewer algorithms for effectively filtering images. While various vision algorithms may prove to be useful, they are often very computationally intense. Instead, we wanted to use readily available and easy to calculate characteristics. These characteristics and the range of appropriate values were identified by collecting images and dividing them into three separate categories: bad, medium, and good. Each image was evaluated for similar characteristics to determine why they received a good, bad, or medium rating. For example, we discovered that images that were of file type GIF were more likely to be bad images (button, bullets or little icons). Items like buttons are generally very small,

and banner ads generally have a very large or very small aspect ratio. Also, navigational tools for a particular Web page are considered a bad image again based on aspect ratio. If the images' aspect ratio was less than 1/3 or greater than 3, then the image was often a banner or navigation bar. All the information collected was combined and the various criteria for the probabilistic algorithm was born.

In order for the displays to have better images to display to the user, we developed an algorithm using readily available and easy to calculate characteristics: the files' type, the aspect ratio, and total area (see Figure 3.1.1). The images received a rating between zero and one based on various characteristics. Note that if an image failed one of the criteria and the rating was reduced, the image was not automatically disqualified for the IDM. For example, there are many good GIF images, and sometimes images with poor aspect ratios are still informative. In the end, images with a rating between 0.6 and 1 were accepted and sent to the image waiting room for storage.

3.2 Display Mechanisms

CWIC provides the user with four unique display mechanisms. The display mechanisms are standalone application, wallpaper, screensaver, and Web page. The mechanisms were chosen for their familiarity and ease of use. We wanted to provide a wide range of methods for communicating the images. The user achieves this by selecting and changing the display mechanisms in the control panel.

The standalone application mechanism works similar to any other application. A user starts and runs it in a portion of the monitor screen (see Figure 3.2.1). It is displayed by activating its own window and exhibiting the images using the chosen display layout. It is easy to

relocate and resize on demand, which the application will take up as little or as much screen space the user can allocate. The standalone application mechanism provides a significant amount of power and has the ability to be present at all times on the screen or iconicized until the user wants to see it. This application provides the user with the option to click on an image to view the corresponding Web page. The user can pull up the control panel and change settings on the application on the fly. The standalone application is practical for someone who wants CWIC to continuously browse the Web with the power to visit Web pages of interesting images.



Figure 3.2.1: CWIC running as a standalone application. The images are collected from the CNN Web site and displayed in a random layout. At the bottom is the URL of the image that the user is pointing to with the mouse. The user can visit the URL by clicking on the image.

The wallpaper mechanism presents the collected images in the background of the desktop (see Figure 3.2.2). It is always visible in portions of the screen not otherwise used by other applications. The mechanism is Unix-based because it uses the xv command to display the images on the background. Generally this technique has been used to put pictures of spouses or babies or similar fun images in the background. Just as PointCast (www.pointcast.com) brought screensavers from the realm of flying toasters to current events, we hope the CWIC wallpaper mechanism will bring current, relevant images to the user's screen background. The wallpaper is

continually visible unless the user covers the entire screen with windows. At certain times, more of the wallpaper mechanism is visible than other times, generally when the user is less busy or has fewer windows open.

Using the wallpaper, the user has the ability with the mechanism to view images from Web sites at any given time, either while performing daily duties or during free time, simply by looking at the wallpaper. This mechanism is very useful for someone who wants to be continually reminded of incoming information, but does not want to take up space on the desktop. This technique is great for someone with a larger monitor and who has space to spare.



Figure 3.2.2: CWIC is running as a wallpaper mechanism on the desktop. The images are collected from a link on the CNN Web site. The images are visible on the screen where other windows opened by user do not use.

The screensaver mechanism presents the collected images using a screensaver that updates periodically (see Figure 3.2.3). The screensaver mechanism will only activate at the computer's idle times and it is easy to remove when not needed. As such, it is unlikely to interrupt the user during busy times and can provide interesting information at times when the user is less busy. This mechanism leverages the xscreensaver program available for the Unix operating system. The screensaver uses an image map that is updated frequently using one of the display layouts chosen

by the user. This particular mechanism is useful for someone who has little time to spend browsing the Internet manually. The screensaver mechanism will inform the user about valuable information on the Internet at the computer idle times only; for example, when returning from a break or when relaxing at one's desk. This mechanism works by creating, displaying and updating an image map frequently and sending the images to the xscreensaver program. The user can remove the screensaver by pressing any key on the keyboard or mouse.

The Web page mechanism presents the collected images using a basic Web page design (see Figure 3.2.4). This mechanism generates a JPEG image and image map. An image map is a Web-based reference to an image that associates portions of the image with an URL. For example, in Figure 3.2.4 the Mark McGwire image is associated with the www.espn.go.com. The image map provides quick access to the corresponding Web page to view and positions it at a specific Web location. The Web page mechanism does not interrupt the user and is visible only when the user accesses the automatically generated Web page.

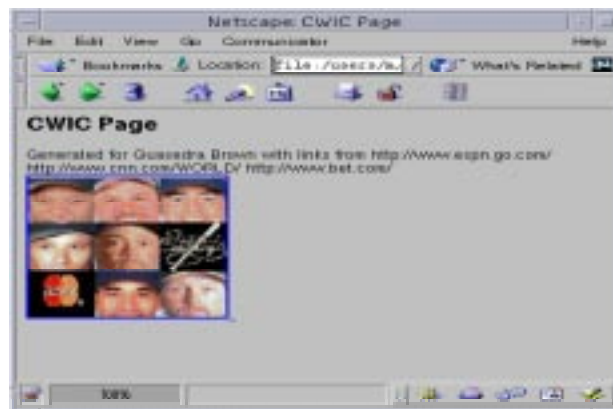


Figure 3.2.4: CWIC is running as a Web page display mechanism. The images and layout are visible on any basic Web page as an image map. This provides the user with quick access the associated Web page the image was downloaded from. The above images are collected from the www.espn.go.com, www.cnn.com/World/, and www.bet.com.

3.2 Display Layouts

CWIC provides the user with four possible display layouts. The display layouts are grid, spiral, random, and fade. The layouts were chosen based on style, familiarity, and simplicity. The user has the option to select whichever layout is suitable and desired. All the layouts have the ability to be resized by the user.

The grid layout presents images in a grid pattern (see Figure 3.3.1). Each image is cropped to fit in the user-selected size. The grid layout was chosen because its regular and repeating pattern makes good use of screen space. All images in the display are visible because there is no overlapping of images. The screen space is used efficiently by the grid layout because the images are placed from left to right and top to bottom. The number of columns and rows are calculated based on the size of the display space, and the size of the images all user configurable.

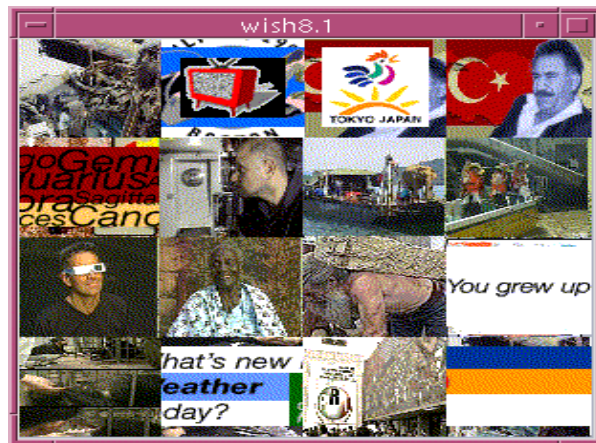


Figure 3.3.1: CWIC is running as a standalone application using the grid layout. The collected images are displayed to the user from top to bottom and left to right continually. The images are cropped to fit the size of the row-column lengths.

The spiral layout provides a unique view of the images utilizing a circular pattern (see Figure 3.3.2). The location of the most recent updates is easy to see because of the regular and recognizable pattern. The images are displayed starting from the center, spiraling counterclockwise to the perimeter, then back to the center. Calculating the perimeter of the canvas and radius forms the boundaries for spiral design. The drawback of the spiral is that images are partially overwritten.



Figure 3.3.2: CWIC is a standalone application using the spiral layout. The collected images are displayed to the user in a circular pattern starting from center to the perimeter to center counterclockwise and back from center to the perimeter to center clockwise. Each image is partially overwritten by another image.

The random layout presents the images by displaying the images randomly on the screen (see Figure 3.3.3). The overlapping and overwriting of images are often not as immediate as with the spiral, but the most recent updates are usually apparent due to the stacking order. After calculating the total area and perimeter of the canvas, the images are placed randomly throughout the area. Each random layout will be different due to the arbitrary positioning, so one person gets a different view than a neighbor might.

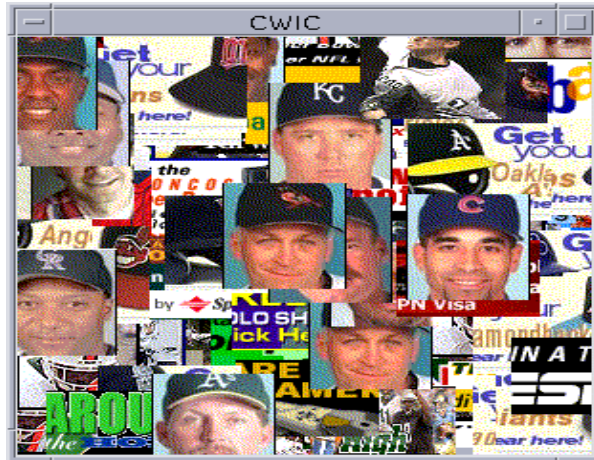


Figure 3.3.3: CWIC is running as a standalone application using the random layout. The collected images are displayed to the user randomly. The images are cut at the perimeter. All the images are partially overwritten as updates occur. The older displayed images are on the bottom while the newer displayed images are on the top. One person random display will differ from another person's random display.

The fade layout presents the images one at a time using a small display area (see Figure 3.3.4). The updates of images are gradual rather than sudden, which may prove to be less disruptive to the user [Gon96]. The fade technique was implemented using the Agentk animated toolkit extension to Tcl/Tk [MZ00]. Since the user will only view one image at a time, the fade layout will require little screen space. If necessary, the images are cropped to fill the canvas.

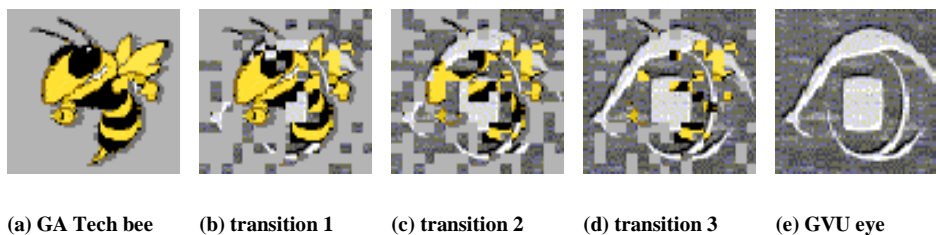


Figure 3.3.4: CWIC is utilizing its fade layout to display the images to the user. The first image, GA Tech bee, is displayed fully to the user (a). As the next image, Graphics, Visualization, Usability (GVU) eye, is being display; the previous image is being faded out slowly (b), (c), & (d). Finally, the previous image; GA Tech bee, is completely faded out and the next image; GVU eye, is fully displayed to the user (e).

3.3 Use/Usability

In developing CWIC, we obtained user input throughout the developmental process. There were a total of seven users that contributed to the components of CWIC. The users were chosen from the Georgia Tech College of Computing and represented a wide range of genders, races, ages, and educational levels.

The users enjoyed the four different display mechanisms and layout provided by CWIC. There was no distinct favorite mechanism or layout amongst the group. Each user appreciated the ability to add external links/URLs. The URLs included of www.obscurestore.com, www.unitedmedia.com, and www.doonesbury.com. One user pointed CWIC to his bookmarks file, so he could keep track of updates to pages he had bookmarked.

Though the user requested more complex displays that often included text, borders, and link information, simplicity was an important factor. This factor allowed any user either a novice or experienced to utilize, modify, and understand the behavior of CWIC. The user also wanted more control over the crawler depth and link selection within a page providing the user with complete control over the information being collected. Future versions may include this ability, but we want to be careful not to overburden the user with choices. Simplicity is important. CWIC should and will remain easy to start and easy to use.

4. Related Work

The main contributions of CWIC are to the fields of human-computer interaction (HCI), information visualization and interfaces, and software agents. The World Wide Web presents new challenges to HCI professionals. Unlike many information repositories, it changes frequently and

constantly. It is also different in that it is not just composed of text but of hypertext, text and images and more linked together.

Software agents and Web crawlers typically browse the Web and collect information. Tools like Yahoo (www.yahoo.com) and Alta Vista (www.altavista.com) traverse thousands of pages a day and provide several methods for accessing the information. Software agents perform more specific tasks; for example, CiteSeer retrieves and identifies publications from the Web [BLG98]. CWIC uses many of the same techniques as the crawlers and agents to identify and collect information from the Web, but we believe that alternative visual communication methods will broaden people's understanding of the Web. As such, we have tried to push the envelope and provide new ways to view the Web.

Much unique visualization to date only leverages the textual information at a site. For example, the ThemeScapes interface creates graphical mountains representing the textual content of an information space [WTPLPS95]. Rennison's Galaxy of News uses clustering and layout of text to provide a zoomable view of an information space [Ren94]. However, these systems ignore or de-emphasize the graphical component of a Web space, and when using them they consume our entire attention. CWIC leverages the graphics found at a site and presents it in less intrusive ways.

Most tools that try to keep us up-to-date on information on the Web do so using textual methods. For example, the AT&T Internet Difference Engine highlights textual differences to selected pages using a summary page [DB96]. Many other Web-based monitoring systems will send an email when a page changes. These types of solutions are text-based and as such ignore the rich graphical component of the Web. Also, they force users to seek out the solution, either by visiting a Web page or reading email, and as such are impractical for large numbers of pages or

for sites that change frequently. PointCast (www.pointcast.com) includes some graphics in its screen saver, though it is not the primary communication medium. CWIC brings the information to the user in a pleasant graphical manner that is easy for the user to process.

Fortunately, several other projects are considering the use of graphics in communicating information. Perhaps most notable are the Mandala and Montage systems [Hel99, Hel96].

Mandala uses images to represent Web information and stores them in imagemaps, which can then be placed on Web pages or exchanged with other users. Montage collects images from a shared Web proxy and randomly presents them in a screensaver or summary Web page. By utilizing a proxy, a user can see images from Web pages obtained by fellow users and leverage their surfing time. Another system, Netomat, responds to natural language questions not just with a textual list of sites but with a free flowing stream of text, images, and sounds [Jan99]. The Collage Machine is another system that creates collages of images and text as an alternative to traditional browsing [Ker97]. CWIC leverages the same principle that images can effectively communicate information about the Web, but it includes alternate layouts and delivery mechanisms, it allows easy access to the original Web page, and it has highly configurable starting points.

5. Conclusion/Future Work

This paper introduced CWIC, a passive browser that collects interesting images from Web pages. The several display mechanisms and layouts, and control panel provides the user with the ability to customize CWIC. CWIC provides the user with a graphical representation of interesting Web pages that reminds the user of updates and/or changes to Web pages.

CWIC uses image-based displays to provide the user with a different method for staying up-to-date on changes to the World Wide Web. A probabilistic algorithm filters out images that

are least likely to communicate information about the Web page. The display layouts and mechanisms give the users many choices for how to display the collected images.

In the future, the probabilistic algorithm will be enhanced so that it will better identify the images the user wants to see. We plan to include other layouts, and perhaps allow users to create their own layouts. However, we want to be careful to maintain the ease of use that attracts new users to CWIC.

Acknowledgements

The authors would like to thank College of Computing and Irfan Essa for their support via the Summer Internship Program. We would also like to thank all of our users for their helpful comments and patient use throughout the development of CWIC. We would particularly like to thank John Stasko and Alex Zhao for their technical and informational support. Finally, we would like to thank the Gvu Center and the USENIX Organization for their financial backing.

References

- [BLG98] K. D. Bollacker, S. Lawrence, and C. L. Giles. CiteSeer: An Autonomous Web Agent for Automatic Retrieval and Identification of Interesting Publications. In *Proceedings of the 1998 Autonomous Agents Conference (Agents 98)*, pages 116-123, Minneapolis, Minnesota, 1998.
- [DB96] F. Douglass and T. Ball. Tracking and Viewing Modifications on the Web. In *Proceedings of the 1996 USENIX Annual Technical Conference*. January 1996.
- [Gon96] C. Gonzalez. Does Animation in User Interfaces Improve Decision Making? In *Proceedings of the 1996 Conference on Human Factors in Computing Systems (CHI '96)*, pages 27-34, Vancouver, BC Canada, April 1996.
- [Hel96] J. Helfman. Montage as Cognitive Artifact: Passive Surfing in the Communal Cache. Human Computer Interaction Consortium, Frasier, CO, February 1996.

- [Hel99] J. Helfman. Mandala: An Architecture for Using Images to Access and Organize Web Information. In *Proceedings of the 1999 International Conference on Visual Information Systems (VISUAL 99)*, June 1999.
- [Jan99] R. Jana. Netomat: The Non-Linear Browser. *Wired News*, June 1999.
- [Ker97] A. Kerne. CollageMachine: Temporality and Indeterminacy in Media Browsing via Interface Ecology. In *Proceedings of the 1997 Conference on Human Factors in Computing Systems (CHI '97)*, Atlanta, Georgia, March 1997.
- [Ous94] J. Ousterhout. *Tcl and the Tk Toolkit*. Addison-Wesley, 1994.
- [Pav86] A. Pavio. *Imagery and Verbal Processes*. New York: Holt, Rinehart, and Winston, 1986.
- [MZ00] D. S. McCrickard and Q. A. Zhao. Supporting Information Awareness using Animated Widgets. In *Proceedings of the 2000 USENIX Conference on Tcl/Tk (Tcl2K)*, Austin TX, February 2000.
- [Ren94] E. Rennison. *Galaxy of News: An Approach to Visualizing and Understanding Expansive News Landscapes*. In *Proceedings of the 1994 User Interface Software and Technology Symposium (UIST 1994)*. Marina Del Ray, California.
- [SCH70] L. Standing, J. Conezio, and R. N. Haber. Perception and Memory for Pictures: Single-trial Learning of 2500 Visual Stimuli. *Psychonomic Science*, 19(10): 73-74, 1970.
- [WTPLPS95] J. Wise, J. J. Thomas, K. Pennock, D. Lantrip, M. Potter, and A. Schur. Visualizing the Non-Visual: Spatial Analysis and Interaction with Information from Text Documents. In *Proceedings of the 1995 IEEE Information Visualization Symposium (InfoVis 95)*, pp 51- 58, 1995.