Memory-equipped fisheye views for navigating large menu structures

Elaine G. Toms
School of Library and Information Studies
Dalhousie University
etoms@is.dal.ca

Mark T. Kinnucan
Graduate School of Library and Information Science
University of Western Ontario
kinnucan@julian.uwo.ca

Introduction

Typically, large hierarchical menus provide access to information systems such as the World Wide Web. Difficult to navigate, these menu structures obscure location within the hierarchy and marginalize the user's capability to accurately retrace steps through the hierarchy. These problems are due to menu presentation: as one moves upward or downward through the menu levels, the contents of lower and higher levels disappear from view. To date a mechanism that stores the history of menu levels (or information nodes) selected has been the primary tool to alleviate the problem.

In this study, we hypothesize that a variation of the fisheye view is a more effective approach to the presentation of history information in hierarchical menus. Current implementations of fisheye views identify location within a menu structure, but do not retain a history of a session within a menu. In essence, no memory of the path taken remains; analogically speaking, there are no breadcrumbs to collect. We suggest that by including past focal points in the calculation of node distances for a fisheye view, we can cause the fisheye view to retain some memory of the path taken. We wish to determine whether the resulting display provides an effective graphical overview of a menu structure.

History tools in interfaces

Several approaches have been explored to help users of computer systems retrace the history of their interaction with a system or reuse previous actions.These tools have been reviewed by Greenberg and Witten (1993) and by Lee (1992). Greenberg and Witten reviewed a variety of different approaches, of which only two are of interest in the present context: "history mechanisms" and "adaptive menu hierarchies."

History mechanisms

The history mechanisms reviewed by Greenberg and Witten consisted primarily of lists of objects. In command-based systems, the objects in the list are typically the previous commands issued, either in the current session or across several sessions. The "History" command and associated capability for editing and re-executing commands in Unix provide one example of such facilities. Similar mechanisms in the information retrieval realm include the "Display sets" command in Dialog and its counterparts in other retrieval systems. In yet another variation, Lemaire and Moore (1994) described a software tool that records and makes available the history of interactions between a learner and a human tutor.

It is becoming common in graphical user interfaces for these history lists to be placed in a separate window and to be manipulable (Greenberg and Witten 1993, 397–402). Kulander and Feiner (1990) referred to the capability to select items from history lists and manipulate them in GUI systems as "spatial browsing." They noted that spatial browsing is of limited usefulness in interfaces where much of the interaction is by pointing and clicking rather than by typing in the names of commands. They developed and presented an alternative called "editable graphical histories" for such interfaces.

In information environments, especially those that emphasize browsing, such as hypertext systems, the objects in the history list are more likely to be the previous locations visited rather than the previous commands issued. Greenberg and Witten referred to such history mechanisms as "navigational traces." The navigational trace may include all of the previous places visited, as in the "History" pull-down window in Netscape or Mosaic.On the other hand, the navigational trace may include only those places explicitly selected by the user, as in the bookmark feature available in Web browsers, hypertext systems, Gopher clients, etc. Engel, Andriessen, and Schmitz (1983) explored the possibility of combining a comprehensive navigational history with selectable bookmarks in the same display.

Research shows, however, that history lists and bookmarks in these browsers are seldom used. Tauscher (1996) found that only about one per cent of Web locations were chosen from history lists in Netscape or Mosaic. Only about three per cent were chosen from bookmarks. The low usage of these features could be due to several factors. Among the possibilities are the fact that the user has to activate the feature rather than having it continuously available, or that the information is presented in a simple list rather than a graphical display.

Adaptive menu hierarchies

Various researchers have explored ways to make a graphical menu display responsive to what choices the user (or a collection of users) has already made. Greenberg and Witten (1985) developed a method to alter the presentation of a menu based on frequency of past choices, so that more frequently chosen items were placed nearer to the top of the menu. Although it uses different techniques and requires more extensive record-keeping, this work is similar to our own. In a less computationally intensive approach, Nielsen (1990) illustrated the simple technique of adapting graphical overviews of "hyperspace" by placing checkmarks next to the labels of nodes that have already been visited.

More recently, there has been considerable interest in graphical display of the path a user has taken in World Wide Web session. Both Ayers and Stasko (1995) and Cockburn and Jones (1996) describe add-ons to Web browsers that provide this capability. The tool developed by Ayers and Stasko has one feature similar to fisheye views in that it allows the user to condense a part of the display into a more "zoomed out" version. However, this feature is entirely under user control, rather than being invoked automatically.In addition, both of these projects only display the actual nodes visited, rather than placing them in the context of the full range of possible nodes. With these considerations in mind, we decided to explore the possibility of incorporating history information in fisheye views.

Fisheye views

The fisheye view was devised by Furnas (1986) as a viewer that "can show places nearby in great detail while still showing the whole world simply by showing the more remote regions in successively less detail." He compared this naturally occurring representation to Steinberg's famous poster of New York, showing a street view of Manhattan, reducing the rest of the country to only its prominent landmarks and disappearing to the west. A fisheye view balances local detail with the global context. As Furnas elaborated, details at the local level are needed to understand local interactions, while details at the global level serve as orientation devices and enable a better interpretation of local detail.

A fisheye view as defined by Furnas (1986) is based on a focal point and an *a priori importance* (API). The focal point is the node currently being visited or attended to. API is the closeness of a node to the root of the tree (how high up it is on the "organizational chart"). Furnas calculated the distance (D) between any two points as the tree distance between those points (the number of nodes that would need to be traversed to get from one point to the other). In our usage, a point is one item on a menu that may contain multiple hierarchical levels. The fisheye view is based on a *degree of interest* (DOI) function that assigns to each point a number indicating how interested the user is in seeing that point. The DOI of a point is an increasing function of its API and a decreasing function of the distance of that point from the focal point. A fisheye view results from a display of only those elements with a DOI greater than a particular user-specified threshold.

Fisheye views have been grouped into three types: filtering, distorting and nesting (Noik 1993). Filtering fisheye views follow Furnas and constrain the display of information semantically, (e.g., Mitta 1990; Kinnucan 1992; Rivlin, Botafogo, and Shneiderman 1994). The presence of a node in a filtering fisheye display depends on its DOI. Distorting fisheye views, on the other hand, attempt to distort the display geometrically to obtain a more approximate visual fisheye effect. The distortion changes the position and size of the points (e.g., Sarkar and Brown 1994). In this case, the resulting display is more spherical in appearance and textual items on the periphery are so distorted as to be unreadable. Nesting fisheye views alter the size and shape of each element included in the display prior to positioning it in a layout, and hence remove the distortion factor (e.g., Noik 1993). A combination of zooming, panning, and magnification lets users examine details of the local context, detail of the target context and less detail of the context between the two. Of the approaches, the filtering and nesting views are the most appropriate for displaying textual information.

Originally, the API was calculated according to the distance from the root node where distance equalled the number of intervening nodes (Furnas 1986). Other alternatives include: a measure based on its accessibility to other nodes, the number of times a node was visited (Rivlin, Botafogo, and Shneiderman 1994), distance from a landmark (Valdez, Chignell, and Glenn 1988), and the occurrence of the search term in a node (Chen, Rada, and Zeb 1994). Rivlin and colleagues briefly discussed the use of a measure combining the structure, document style, and content, but did not elaborate. To enable a user to retrace steps—that is,give the fisheye view amemory—the focal point will be the previously accessed node.

With very few exceptions, only anecdotal evidence attests to the usability, efficiency, and effectiveness of fisheye views. Furnas (1986) compared navigating through a botanical taxonomic class using a fisheye view with a flat file. Although the experimental design was not elaborated on in the report, Furnas claimed that subjects who used the fisheye view were able to answer 75% of the task questions correctly. Schaffer and colleagues (1993) also compared fisheye views to traditional zooming on a hierarchically clustered network. They too found that users' performance improved with the fisheye view; users completed tasks faster and with fewer unnecessary transversals in the structure. Hollands, Carey, Matthews, and McCann (1989) compared a non-hierarchical graphical fisheye view of a subway map with scrolling a zoomed image; in this case fisheye view users performed only marginally better.

But fisheye views have problems. The information structure created may require zooming and panning to examine all nodes in a large structure; often the details are too small to be seen and difficult to interpret, as illustrated by Noik's (1993) nested fisheye view of 1,680 nodes. Ten to fifteen nodes is a rule of thumb (Foss 1989). Secondly, the use of two or more views requires ample screen real estate to see both the entire structure and its zoomed portion and forces the user to integrate views at two levels of the hierarchy. Foss (1989) noted that much semantic information can be lost by the reduction such that the resulting display may misrepresent the underlying information database. Thirdly, fisheye views suppress selected nodes on the basis of the API measure and may leave the user unable to determine if the node exists or not (Smith and Wilson 1993). Fourthly, constructing large displays of nodes is system intensive and may not be time efficient.

Finally, none of the variations of fisheye views provide any history information. In a fisheye view, the menu display is updated as each new point is accessed. If the current point displayed and the previous point examined by the user are spatially dispersed, then it is likely that the point previously accessed will no longer be displayed in a freshly drawn fisheye view. If the previously viewed node is a parent or a child to the current node, a standard fisheye view will likely show it. However, even in this case, the standard fisheye view may well not show a node that was visited as few as four or five steps back. Because points to be displayed are evaluated only according to their relationship with the point currently displayed and to the root node, a standard fisheye view loses all "memory" of points previously accessed. We will explore various ways of giving a fisheye view a memory.

Overview of the study

In this study, the capability of navigating a large menu structure devised for accessing the World Wide Web will be assessed by means of various versions of a fisheye view including the classic one proposed by Furnas. The questions guiding our study are:

- Can a fisheye view also function as a history list?
- Is a fisheye view with a "memory" a usable interface tool? That is, can the user determine both placement within the hierarchy as well as previous steps taken?

Methodology

Three versions of the fisheye view programmed in C will be compared. The three versions are:

1. The classic one (following Furnas)

   $DOI_{fisheye} (N_k|.=N_c) = API(N_k) - d(N_c,N_k)$

   where

   API = a priori importance

   d = distance

   N_k = root node, i.e., a node "above the main menu"

   N_c = the information node currently in focus, i.e., displayed

   $d(N_c,N_k)$ is the distance between the two nodes, i.e., the number of menu layers.

   $API(N_k) = -d(N_c,N_k)$ is the distance between the node and the root, expressed as a negative number. That is, the farther from the root, the less important a node is.

2. Memory fisheye view using last accessed menu choice as "root node":

   $DOI_{fisheye} (N_k|.=N_c) = API(N_k) - d(N_c,N_k)$

   where

   N_k = previous node accessed

3. Memory fisheye view with weighting:

   $DOI_{fisheye} (N_k|.=N_c) =(A * DOI(n-p)) + (B * API(N_k)) + (C*(-d(N_c,N_k)))$

   where

   A, B, C = adjustable weights (normalized so that A + B + C = 1)

   n = iteration (*n*th time a menu selection was made).

   p = "order" of the history (how many steps back).

The menu hierarchy used by *Sympatico*, a product of Maritime Tel & Tel's Advanced Communications that functions as a general user interface to the World Wide Web, will be used as the test vehicle. *Sympatico* has a four-layer hierarchical menu structure.

Tocompare the four resulting presentations, sample logs from user sessions will be used. For each user selection, the following questions will be asked of the resulting menu display:

- Can placement within the menu hierarchy be readily determined?
- Can the user readily retrace steps?

Each response will be scored and the scores averaged across transaction logs.

Finally an analysis of variance will be used to determine if there are any statistically significant differences among the three menu displays.

Conclusion

This ongoing study assesses ways of giving a fisheye view memory such that not only a perspective of local and global contexts may be presented, but an indication of nodes previously visited continues to be displayed.

Note

The first author is also a doctoral candidate, Graduate School of Library and Information Studies, University of Western Ontario.

Acknowledgements

This work was supported by grant from the Dalhousie University Research and Development Fund.

References

Ayers, E.Z., and J.T. Stasko. 1995. Using graphic history in browsing the World Wide Web. *Fourth International Wide Web Conference, Boston.* (http://www.w3.org/pub/Conferences/WWW4/Papers2/270)

Chen, C., R. Rada, and A. Zeb. 1994. An extended fisheye view browser for collaborative writing. *International Journal of Human Computer Studies* 40(5): 859–78.

Cockburn, A., and S. Jones. 1996. Which way now? Analysing and easing inadequacies in WWW navigation. http://kaka.cosc.canterbury.ac.nz/~andy/ijhcs.ps.gz

Engel, F.L., J.J. Andriessen, and H.J.R. Schmitz. 1983. What, where and whence: means for improving electronic data access*International Journal of Man-Machine Studies* 18(2): 145–60.

Foss, C.L. 1989. Tools for reading and browsing hypertext. *Information Processing and Management* 25(4): 407–18.

Furnas, G.W. 1986. Generalized fisheye views. In *Human factors in computing systems: CHI'86 conference proceedings, April 13-17, Boston,* 16-23. New York: Association for Computing Machinery.

Greenberg, S., and I.H. Witten. 1985. Adaptive personalized interfaces—a question of viability*Behaviour and Information Technology* 4: 31-45.

Greenberg, S., and I.H. Witten. 1993. Supporting command reuse: mechanisms for reuse*International Journal of Man-Machine Studies* 39, 391–425.

Hollands, J.G., T.T. Carey, M.L. Matthew, and C.A. McCann. 1989. Presenting a graphical network: a comparison of performance using fisheye and scrolling views. In *Designing and using human-computer interfaces and knowledge based sysstems: proceedings of the Third International Conference on Human-Computer Interaction, Boston, Massachuttes, September 18–22, 1989, volume 2*, ed. G. Salvendy and M.J. Smith, 313–20. Amsterdam: Elsevier.

Kinnucan, M.T. 1992. Fisheye views as an aid to subject access in online catalogs.*Canadian Journal of Information Science* 17(2): 25-40.

Kurlander, D., and S. Feiner. 1990. A visual language for browsing, undoing, and redoing graphical interface commands. In*Visual languages and visual programming*, ed. S.-K. Chang, 257–75. New York: Plenum.

Lemaire, B., and J. Moore. 1994. An improved interface for tutorial dialogues: browsing a visual dialogue history. In *Human factors in computing systems: CHI'94 conference proceedings, April 24-28, Boston*, 16-22. New York: Association for Computing Machinery.

Lee, A. 1992. Investigations into history tools for user support.Doctoral dissertation, University of Toronto. (http://zgdv.igd.fhg.de/papers/gui/contents.html#history-tools)

Mitta, D.A. 1990. A fisheye presentation strategy: aircraft maintenance. In*Human-computer interaction—INTERACT '90: proceedings of the IFIP TC 13 Third International Conference on Human-Computer Interaction, Cambridge, U.K., 27–31 August, 1990*, ed. D. Diaper, D. Gilmore, G. Cockton, and B. Shackel, 875–80. Amsterdam: North Holland.

Nielsen, J. 1990. The art of navigating through hypertext. *Communications of the ACM* 33: 296–310.

Noik, E. 1993. Layout-independent fisheye views of nested graphs. In *1993 IEEE Symposium on Visual Languages, August 24–27, 1993, Bergen, Norway: proceedings*, 336–41. Washington: IEEE Computer Society Press.

Rivlin, E., R. Botafogo, and B. Shneiderman. 1994. Navigating in hyperspace: designing a structure-based toolbox. *Communications of the ACM* 37(2): 87–96.

Sarkar, M., and M.H. Brown. 1994. Graphical fisheye views. *Communications of the ACM* 37(12): 73–84.

Schaffer, D., Z. Zhengping, L. Bartram,J. Dill, S. Dubs, S. Greenberg, and M. Roseman. 1993. Comparing fisheye and full zoom techniques for navigation of hierarchically clustered networks. In *Proceedings of the Graphics Interface, 1993*, 87–96. Toronto: Canadian Information Processing Society.

Smith, P.A., and J.R. Wilson. 1993. Navigation in hypertext through virtual environments.*Applied Ergonomics* 24(4): 271–78.

Tauscher, L. 1996.Supporting World-Wide Web navigation through history mechanisms. (http://www.cs.bgsu.edu/usable-web/chi96/tauscher.html)

Valdez, F., M. Chignell, and B. Glenn. 1988. Browsing models for hypermedia databases. In *Proceedings of the Human Factors Society 32nd Annual Meeting, Anaheim, California, October 24–28, 1988*, 318–22. Santa Monica: Human Factors Society.