

# New Human-computer Interaction Techniques for the Digital Library

Robert J.K. Jacob  
MIT Media Laboratory  
Cambridge, Mass. U.S.A.  
*and*  
Department of EECS  
Tufts University  
Medford, Mass. U.S.A.  
jacob@eecs.tufts.edu

Horn-Yeu Shiao  
Department of EECS  
Tufts University  
Medford, Mass. U.S.A.  
hshiao@eecs.tufts.edu

Gregory R. Crane  
Department of Classics  
Tufts University  
Medford, Mass. U.S.A.  
gcrane@perseus.tufts.edu

## ABSTRACT

We present our approach to developing new interaction techniques that users could someday use for reading the information in a digital library. Specifically, we focus on techniques for communicating spatial information as an ancillary aspect of text material in the library, without distracting from the main reading task. We use a "lightweight" or "non-command" user interface that follows the user and responds appropriately. This enables us to present information to the user in a non-distracting, background or peripheral mode, in a situation where reading the text ought to remain the primary focus of the user's attention. We describe our approach, design rationale, and prototype implementations of several of these new interaction techniques.

## Categories and Subject Descriptors

H.1.2 [Models and Principles]: User/Machine Systems—*human factors*; H.5.2 [Information Interfaces and Presentation]: User Interfaces; I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—*virtual reality*.

## General Terms

Documentation, Design, Human Factors.

## Keywords

Digital library, human-computer interaction, interaction technique, lightweight interaction, non-command interfaces, virtual reality, geographical information.

## 1. INTRODUCTION

We seek to invent new human-computer interaction techniques for interacting with digital libraries in the future. Our project, A Digital Library for the Humanities, sponsored by the National Science Foundation and National Endowment for the Humanities, under the DLI-2 program, seeks to expand the range and kind of humanities material in the digital library and builds on the work of

the highly successful Perseus Digital Library at Tufts University[3, 4]. One part of this project is a collaboration between Electrical Engineering and Computer Science, Psychology, and Classics, in which we are attempting to devise new interaction techniques that users may someday use for reading the information in the library. We are focusing on ways of using and communicating spatial information as an ancillary aspect of the text material in the library. This paper describes our initial approach to this problem.

Our goal is to communicate some spatial information to the reader without distracting from the main reading task. We want to provide a "lightweight" or "non-command" user interface that follows the user and responds appropriately, without requiring much explicit action or attention from him or her. We are developing new interaction techniques for this purpose. An interaction technique is a way of using a physical input device to perform a generic task in a human-computer dialogue, such as entering a particular type of value into the computer (e.g., pull-down menu, slider, or text area)[5]. Our interaction techniques are designed to provide some ancillary spatial information almost "for free," while the user just reads the text normally.

### 1.1 Platform

The Perseus Digital Library began life in the days before the World-Wide Web, as a CD-ROM containing a set of hypertext pages[2]. It is now on the web, as hypertext, and is now accessed through conventional web browsers. However, our research looks toward future ways of interacting with the digital library, and we try not to be constrained by the capabilities of current web browsers. We are building prototypes to examine alternative interaction techniques using various non-standard platforms and mockups as needed.

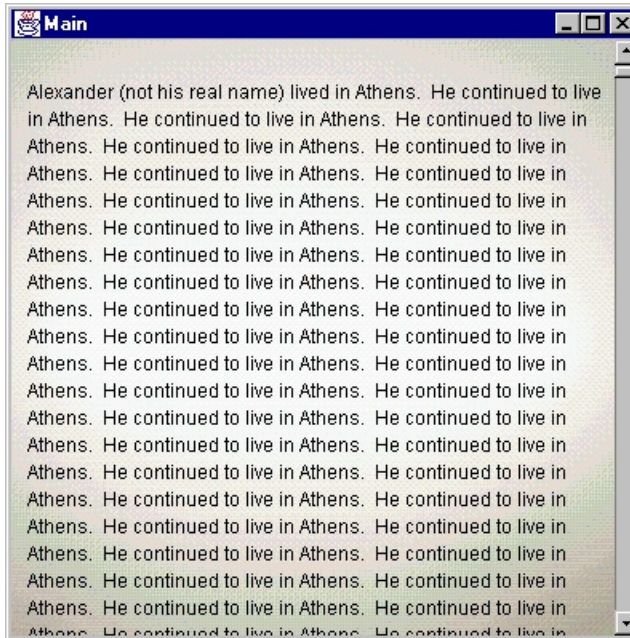
### 1.2 Goal

We begin with a desire to communicate spatial information related to a text that the user is reading, with minimal distraction to the reader. Imagine a text that involves locations, for example, it might contain, "Leonidas walked toward the Agora." Geographic or spatial information about places in the text would enhance the reader's understanding of such a text, particularly for a modern reader who might be less familiar with such places than the author's contemporaries would have been.

A conventional solution might be to provide a traditional hypertext link in the text ("Leonidas walked toward the Agora [click here for picture of Agora]") In a conventional browser,

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

Conference '00, Month 1-2, 2000, City, State.  
Copyright 2000 ACM 1-58113-000-0/00/0000...\$5.00.



**Figure 1. The background display prototype, showing sample text in the foreground and the blurred image behind it.**

selecting the link displays the picture in the main window, replacing the text the user was reading; a "Back" button or other link would then take him or her back to the text to continue reading. We envision a situation where the text is primary, and the spatial information is a relatively minor, ancillary piece of information. Receiving that spatial information ought not to distract the reader from the text. Clicking the hypertext link to another page is thus precisely the wrong approach, because it interrupts the flow of the reader's primary task. We want to let the reader continue reading and to make no particular effort on his or her part, while we simply provide him or her with some small amount of the spatial information somehow "for free," without distracting from the reading, without requiring any explicit action on the user's part, and without using up screen space that would normally be devoted to the text.

### 1.3 Lightweight Interaction

The idea of lightweight interaction, also sometimes called passive, tacit, non-command, low-intentional, or perceptual interfaces, is to try to obtain useful information from a user, without much effort from him or her. The user thus need not always give explicit commands, but the system might observe, guess, infer, and take hints from him or her [8, 10]. Input for this purpose can include eavesdropping on user commands intended for other purposes (as will be seen in our examples), physiological sensors (skin resistance, facial expression), or behavioral measures (typing or mousing speed).

We use this lightweight input approach to design our interaction techniques. That is, the user simply reads the main text and takes no other particular action to use our techniques. We also want the output to take little or no attention away from the text reading task and use up little or no screen real estate. We picture the user simply reading the text on the screen as he or she would otherwise, while extra information is flowing into his or her eyes

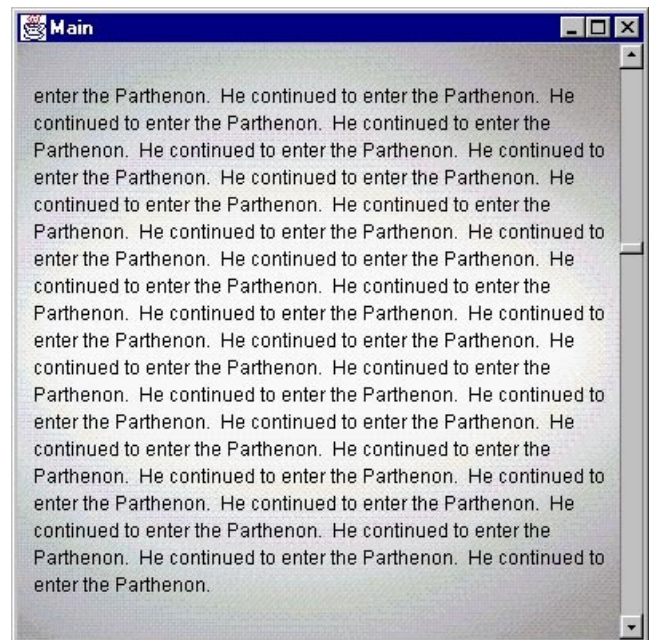
to convey some spatial information related to the text. This paper presents several experimental interaction techniques we are developing for this purpose. For each, we describe the real-world metaphor upon which it is based, the design rationale for our design, and our implementation of it.

## 2. INTERACTION TECHNIQUES

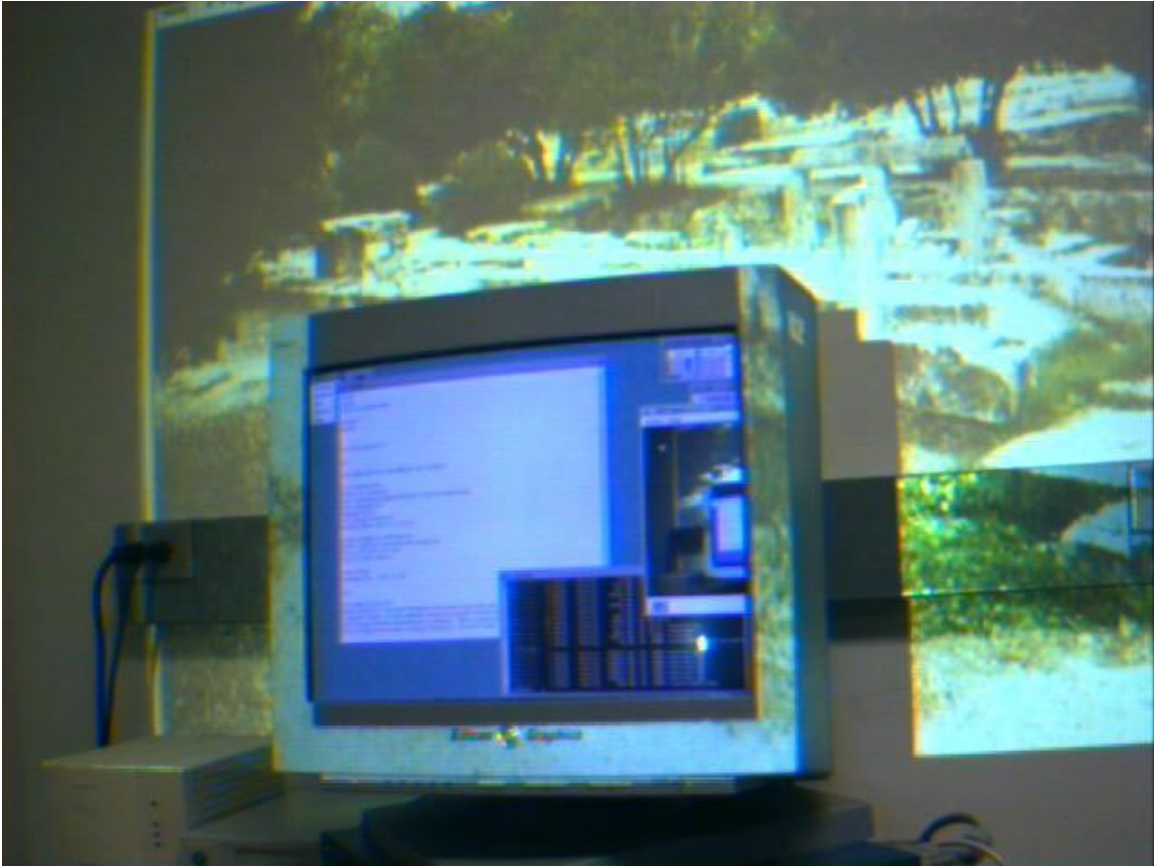
### 2.1 Background Display

Imagine you are reading text printed on a piece of clear plastic, and you can see, behind the plastic, a picture of the locale for that text. As you read the story, the background picture changes to reflect the places mentioned at each point in the text. The picture is toned down so as not to detract from the text, somewhat like a poster with text printed over faint background graphics. The picture is also greatly blurred, removing fine detail that would conflict with the text. Even with considerable blurring, this background behind the text subtly provides an approximate sense of a whether the scene is indoors or outdoors, in a street or a forest, daytime or nighttime. The extra information presented by the background display is very subtle, in the background, but somehow aids understanding--and the reader is still focussed on the text, not the pictures.

To implement an interaction technique along these lines, we collected the appropriate background images for a piece of text and marked up the text to add a tag at each point where a new image should be displayed. We then processed each image to make a suitable blurred, toned-down background. First, we filtered each background image down to just 3x3 pixels. This captured the overall color of the scene and some difference between foreground and background areas, but removed all fine detail. We then enlarged the 9-pixel image back to normal size and blurred it by linearly interpolating between the 9 regions.



**Figure 2. The background display prototype again, after the user has scrolled the text, causing a new background to appear (although this change is hardly visible in the black and white reproduction).**



**Figure 3. Photograph of the setup for the Peripheral Border Display interaction technique, showing the main CRT screen used for reading text and an image related to the current text projected onto the surrounding wall. The portion of the projected image that would otherwise cover the glass area of the CRT is blacked out by the projector.**

Finally, we increased the image brightness, to make a more subdued background against black text. These operations are performed on the fly, as the user scrolls.

As the reader proceeds through the text, the background changes to the appropriate image for each portion of the text. The change can be made with a gradual dissolve, again, so as not to distract from the reading. Figure 1 shows a screen from an initial prototype of this approach, with some sample text in the foreground and the blurred image behind it. Figure 2 shows what happens after the user scrolls forward and a different background appears.

While virtual reality systems strive for higher resolution for walkthrough displays, this application requires a very low resolution display so as not to distract from the foreground text. This can be thought of as an extreme in low resolution walkthrough, the 9-pixel walkthrough. For even simpler distinctions like indoors vs. outdoors, it could even usefully be reduced to a single pixel, that is a solid background color that changes as the reader proceeds through the text.

We track the user's reading through his or her scrolling commands. To facilitate this, we fade the background image more strongly (making the overlaid text clearer) in an elliptical bullseye pattern around the center of the page, and gradually fade it less,

outward from there (as seen in the figures), in order to encourage the reader gently to read always from near the center (the type there will stand out a little better). We then synchronize the background picture to the text at that central area, so we have a better measure of where in the text the reader is reading.

We also use an eye tracker in our laboratory for eye movement-based interaction techniques. We could use eye tracking to determine more precisely where the user is reading[7, 11], but our initial prototype just follows the user's scrolling.

An extension of this approach would be try to measure or estimate exactly when the user wants to read the text and when he or she wants to see through it to the background, and then to fade each of them up or down appropriately. We could look for indicators to help decide when to fade it up or down, perhaps reading speed (as determined by user scrolling). Suppose we assume roughly constant reading speed or at least a reasonable minimum speed. We then posit that any pauses slower than this minimum speed must be to see the picture (or to do some other task), but not to read the text any further, because the reader would not read any more slowly than this minimum speed for the text. We could measure this speed for an individual user by watching him or her read the first page or two.

A further extension could add background sound that changes as you go from indoors to outdoors, near or far from walls. Like the highly blurred image, the sound could be just a little background white noise that changes slightly from one locale to another, or some other greatly toned down general background sound.

## 2.2 Peripheral Border Display

Along the lines of the Background Display interaction technique, here we use peripheral vision instead of the background behind the text. For this technique, suppose you were reading a text while riding on a bus, which was driving through the same places mentioned in the text at the same time they were mentioned. Even without looking up from your book, you would be able to get a rough sense of where you were--indoors, outdoors, urban, rural, daytime, nighttime--through your peripheral vision.

We also want to continue our philosophy that the reading task is primary and the spatial information ought not to distract from it. We thus prefer not to use any of the main screen real estate that would be devoted to reading, just as the reader in the bus can see the peripheral views without obscuring the book. Our approach is thus to display the peripheral information outside of our main computer screen, using an auxiliary video projector. The user reads on the main screen without modification, while the peripheral information is projected onto a wall just behind the screen, as a very large, somewhat blurry image. (We do not blur this image explicitly, but the fact that it is a large video image located fairly close to the user makes its pixels fairly large and effectively blurs it.) In our configuration, the background image is simply front-projected from behind the user's head onto a wall just behind the monitor. The projected image has a black cutout (and, optionally, a physical mask in front of the projector lens) to cover the portion of it that would otherwise project over the real computer monitor.

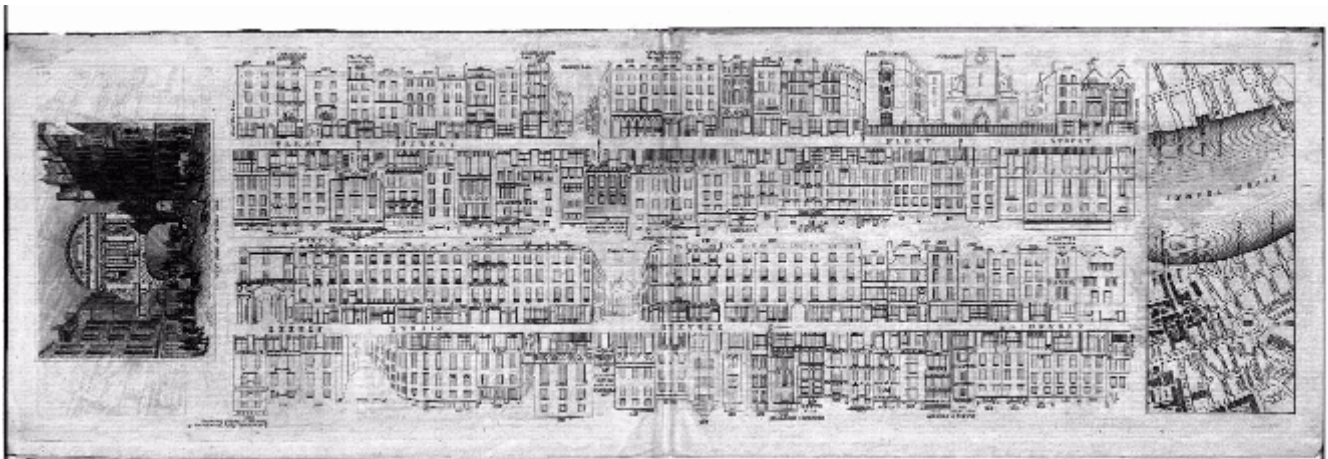
The reader can of course look up at the outer screen at any time. But our intention is that he or she need not ever look at it, but can simply receive information through peripheral vision, just like the imaginary reader riding in the bus. The reader thus uses his or her foveal vision for the reading task (foveal vision is always required for reading). Peripheral vision is little used in reading so we view it as an extra available channel in this setting, and we exploit it for communicating an approximate sense of place. This combination



**Figure 4.** A schematic picture of the Peripheral Border Display prototype. The screen image that is sent to the video projector is shown in the outer window, and the screen image of the regular CRT monitor is shown (here greatly reduced) in the cartoon picture of the CRT in the center.

makes a fuller use of the reader's available perceptual resources.

While the Background Display prototype runs in a single Java application window on a single machine, this one requires additional equipment. However, recall that our goal is to develop future interaction techniques, not necessarily intended to work on current browsers. This prototype runs on two Silicon Graphics workstations, one providing the main display, and one driving the video projector. It is written as two Java application programs, running on the two different displays, connected by a Unix pipe (or optionally a regular network socket) to communicate when the background image should change. Figure 3 shows a photograph of the overall setup, with the entire main CRT screen used for reading text and the appropriate image related to the current text projected onto the surrounding wall. Figure 4 shows a schematic screen picture of the prototype. The screen image that is sent to the video projector is shown in the outer window, and the screen image of the regular CRT monitor is shown (here greatly reduced) in the cartoon picture of the CRT in the center. In practice this center item is an entire CRT, and the outer "window" is a wall-



**Figure 5.** Example of one of the original Tallis maps of London, from which we created a virtual reality walkthrough.

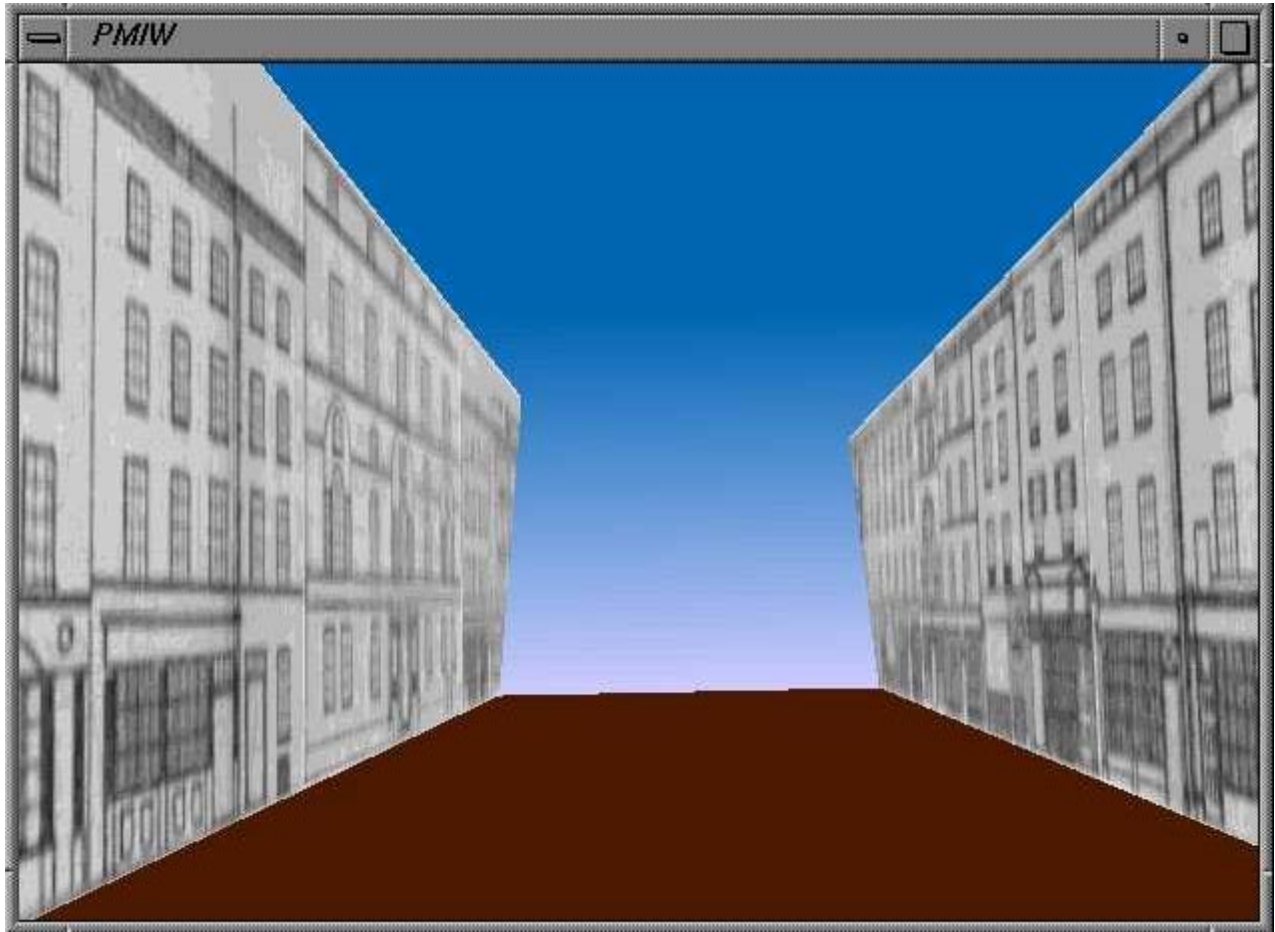


Figure 6. Screenshot of an image during London Tallis map walkthrough.

sized video projector image.

### 2.3 3-D Peripheral Display

An extension of the previous approach generates the peripheral view from a 3-D model rather than from a set of 2-D images. The 3-D model allows us to display a trajectory through the space as an animated walkthrough, showing a "movie" made by walking through a 3-D model in a particular trajectory. Another part of our current project is developing an historical 3-D model of portions of London a set of historical, block-by-block maps, the Tallis maps, in the Bolles Collection of the Tufts University Archives[1]. Figure 5 shows one of the original maps. We are scanning the maps, assembling a 3-D model of the corresponding blocks of London, and then applying the scanned maps as texture maps on top of them, to provide a simple historical 3-D walkthrough. We then mark up the text, as in the previous technique, but here we place tags containing 6-D world coordinate values at particular spots in the text, rather than background image file names. Figures 6 and 7 show the views seen by the user in the head-mounted display as he or she walks through the system. The same technique here might thus apply while reading, for example, a Sherlock Holmes story. As the reader reads "He followed the suspect down Baker St...", the peripheral display shows a continuous walkthrough of the

London model following the same path. And, again, the reader can view it just in peripheral vision, for a general approximate sense of place, while still reading with foveal vision.

Yet another approach would be to produce the border image as a fishtank virtual reality display, where the image is shown on a regular display (or projector), but its 3-D viewpoint changes slightly as the user move his or her head, so the user can see some



Figure 7. Another view of the London Tallis map walkthrough.

parallax. The head motion is tracked by putting a small magnetic (Polhemus) tracker on the user's head. The image on the screen changes depending on the user's head position, reproducing the head-motion parallax that one sees in the real world. The effect is somewhat like looking into a fishtank[12].

We provide an additional window on the main display, with a "you-are-here" map, showing the user's current location and recent trajectory superimposed on a map of the area. The user can also click on a location on the map, which causes the walkthrough to proceed to the selected location and the text window to scroll to the material corresponding to that location.

## 2.4 Peripheral Workspace

This interaction technique takes the peripheral display technique of the previous sections and turns it into an additional workspace for the reader, beyond the normal CRT screen. For a metaphor for this, imagine sitting in a library reading a book at a table, consulting cross-references, dictionaries, and footnotes. You wish to view a reference mentioned in the main book or perhaps to look up additional sources about other aspects of the main topic you are reading, or to compare several sources on a particular point. In a physical reading table, you might move your main book aside, still open to the page you were reading, and serving as a reminder of the context of your reading, even as you examine the other materials. Even if you move the first book to a distant corner of the table, too far to read the words clearly, it still serves as a reminder of your previous context; and it allows you to return to that context easily, by dragging the open book back to the center of the reading table. You might similarly spread out the cross-references and other sources on the table around you, each serving as a reminder of its topic, even if it is moved to a far corner of the table. You might visit some of the cross-referenced items briefly and then put them away. But, for others, you may want to keep them around for future study or to remind you of their content. You might place the materials in a meaningful spatial arrangement on the table around the periphery of your workspace, to help organize them and to let you use spatial memory to keep track of them. Then, when you go back and study them, you can see where they are in your arrangement. Or if you follow up on one of them, to go deeper, you can remember the previous context in the previous layout, and then drill down.

In a real library, you might sit at a large table and spread many books and documents in front of you. They would help remind you of context or related materials while you read the material located at the center of the table. Windows on a computer screen allow you to do this, but the available area is very small. Current computer displays do not have nearly the resolution needed to make a desk-sized display (2-3 feet on a side) with approximately the resolution of a regular CRT or LCD monitor (80-100 dpi). A current display can hold only a little more than a single sheet of letter-size paper at full resolution; its capacity is more like that of an airplane tray table than a real desktop[6]. Our peripheral workspace technique is aimed at simulating the large reading table by providing a large workspace consisting of the regular CRT plus the video projector surround. The central area has the full resolution of a current CRT display, while the surround is displayed with lower resolution. While a large, high-resolution display (not yet available) would be preferable, even the physical

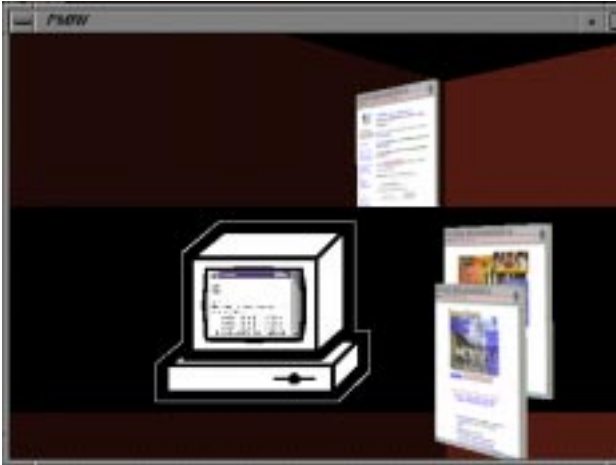


**Figure 8. Schematic of peripheral workspace prototype.** As in Figure 4, the screen image that is sent to the video projector is shown in the outer window, and the screen image of the regular CRT monitor is shown (reduced) in the CRT icon in the center. The CRT is running an unmodified Netscape browser; our prototype, running on the video projector, communicates with it to send and retrieve information about the pages to view.

reading table does not allow the user (unless he or she has extraordinary eyesight) to read materials far from the center of the table. They are visible as a reminder, but they must be dragged back to the center to be read. The physical table is, in effect, a small high-resolution display surrounded by a larger, lower-resolution display. Our approach approximates this within the limits of current display technology. While you cannot read the material in the low-resolution surround, you can use the additional area to exploit spatial memory to organize your references and other materials.

Additional capabilities can be provided because this is a computer display rather than a physical desk. The user can sort or rearrange the objects on the surround automatically, upon command, in non-physical ways based on criteria from the user. For example, the items could instantly be sorted by date or language or primary vs. secondary sources. They could also automatically be restored to the user's original spatial arrangement upon command. We thus combine some of the benefits of the large physical reading table with benefits only obtainable from its computerized analogue. We provide a manual or "natural" mode of interaction (dragging), augmented with some additional non-natural or "magical" commands (instant sorting).

Figure 8 shows a schematic of our initial prototype of such a system. One workstation, connected to the main CRT display, is running an unmodified Netscape browser under Unix. The second workstation, connected to the video projector, is running our prototype system, written as a Java program. It accepts mouse commands from the user to drag the miniature icons around the peripheral workspace and to move them in or out of the area corresponding to the browser running on the main CRT. Our implementation exploits the fact that the Unix version of Netscape



**Figure 9. Schematic of 3-D drag prototype, with central CRT and surrounding video projector. Here, the surround displays a 3-D image.**

allows limited command line access to a running browser from a separate process (and, in our case, from a separate workstation remotely connecting to the same X display object). When the user drags from the browser area onto the peripheral workspace, our Java program can thus retrieve the current URL from Netscape, make a miniature icon for it (by taking a screendump), and associate the two. When the user drags one of the miniature icons back onto the main browser area, we can send its saved URL back to the browser, via the Netscape command line interface, which causes it to return to the page that had been represented by the miniature icon. A keyboard command rearranges the miniatures into a timeline based on when they were created; and another command restores them to their original arrangement.

## 2.5 3-D Drag

We can also extend the Peripheral Workspace interaction technique to 3-D, by making the peripheral window show a 3-D world, containing little 3-D "objects" with the text or images painted on them. The underlying idea is to make the peripheral workspace a little more realistic, more like the real three-dimensional world. Figure 9 shows a schematic of this system. The prototype is written in C++, using SGI Performer for the 3D graphics and our PMIW user interface management system for input handling[9]. As with the previous prototype, the main workstation runs a conventional Netscape browser, and the second workstation runs our C++ program, driving the video projector, and communicating with the browser as needed. The user can navigate through the 3-D view in real time if desired. The 3-D view could also be presented using fishtank virtual reality, as in the previous section. This would be particularly useful here because the user could also move his or her head inward to zoom in on one of the miniature icons, read a little of the headline of it, to remember what it was without actually having to open it up back in center screen.

## 3. CONCLUSIONS AND FUTURE WORK

We have described our approach to developing new interaction techniques for the digital library. Specifically, we focused on

techniques for communicating ancillary spatial information while reading. We based our designs on the idea of lightweight or non-command user interaction. We believe this approach will allow us to present information to the user in a non-distracting, background or peripheral mode, in a situation where reading the text ought to remain the primary focus of the user's attention. We have described our prototype implementations of several of these interaction techniques, along with our design rationale for each. They show the technical feasibility of these methods, though in some cases they are not yet mature enough to evaluate usefulness.

The next step in this work is to narrow it down to the most promising new interaction techniques and refine our prototype implementations to make them suitable for use in experiments. We will then use these to test the usability and usefulness to users of these techniques, particularly how well they can communicate the spatial information while the user is engaged in a different primary task.

## 4. ACKNOWLEDGMENTS

We thank Leonidas Deligiannidis, Stephen Morrison, and Vildan Tanriverdi for their help with this work, particularly in developing the software infrastructure we are using; they are graduate students and recent alumni of the Electrical Engineering and Computer Science Department at Tufts. We also thank Holly Taylor of the Psychology Department, Greg Colati, the University Archivist, and the members of the Perseus Project in the Classics Department for their collaboration.

This research is principally supported by the National Science Foundation and National Endowment for the Humanities under the Digital Libraries Initiative Phase 2 Program. It was also supported in part by the NSF Interactive Systems Program, Office of Naval Research, Berger Family Fund, and the Tufts Selective Excellence Fund.

## 5. REFERENCES

- [1] Bolles Collection, Tufts University Archives, Tufts University, Medford, Mass., 2000. <http://www.library.tufts.edu/archives/london/tallis.htm>
- [2] G.R. Crane (Ed.), *Perseus 1.0: Sources and Studies on Ancient Greek Culture*, Yale University Press, New Haven, 1992. CD-ROM
- [3] G.R. Crane, "Building a Digital Library: the Perseus Project as a Case Study in the Humanities," *Proc. First ACM International Conference on Digital Libraries*, pp. 3-10, 1996.
- [4] G.R. Crane (Ed.), *The Perseus Project*, <http://www.perseus.tufts.edu>, Tufts University, Medford, Mass., 2000.
- [5] J.D. Foley, A. van Dam, S.K. Feiner, and J.F. Hughes, *Computer Graphics: Principles and Practice*, Addison-Wesley, Reading, Mass., 1990.
- [6] D.A. Henderson and S. Card, "Rooms: the use of multiple virtual workspaces to reduce space contention in a window-based graphical user interface," *ACM*

- Transactions on Graphics*, vol. 5, no. 3, pp. 211-243, 1986.
- [7] R.J.K. Jacob, "The Use of Eye Movements in Human-Computer Interaction Techniques: What You Look At is What You Get," *ACM Transactions on Information Systems*, vol. 9, no. 3, pp. 152-169, April 1991.
- [8] R.J.K. Jacob, "Eye Movement-Based Human-Computer Interaction Techniques: Toward Non-Command Interfaces," in *Advances in Human-Computer Interaction, Vol. 4*, ed. by H.R. Hartson and D. Hix, pp. 151-190, Ablex Publishing Co., Norwood, N.J., 1993.  
<http://www.eecs.tufts.edu/~jacob/papers/hartson.txt> [ASCII];  
<http://www.eecs.tufts.edu/~jacob/papers/hartson.ps> [Postscript].
- [9] R.J.K. Jacob, L. Deligiannidis, and S. Morrison, "A Software Model and Specification Language for Non-WIMP User Interfaces," *ACM Transactions on Computer-Human Interaction*, vol. 6, no. 1, pp. 1-46, March 1999.  
<http://www.eecs.tufts.edu/~jacob/papers/tochi.pmiw.txt> [ASCII];  
<http://www.eecs.tufts.edu/~jacob/papers/tochi.pmiw.ps> [Postscript].
- [10] J. Nielsen, "Noncommand User Interfaces," *Comm. ACM*, vol. 36, no. 4, pp. 83-99, April 1993.
- [11] V. Tanriverdi and R.J.K. Jacob, "Interacting with Eye Movements in Virtual Environments," *Proc. ACM CHI 2000 Human Factors in Computing Systems Conference*, pp. 265-272, Addison-Wesley/ACM Press, 2000.  
<http://www.eecs.tufts.edu/~jacob/papers/chi00.tanriverdi.pdf> [PDF];  
<http://www.eecs.tufts.edu/~jacob/papers/chi00.tanriverdi.ps> [Postscript].
- [12] C. Ware, K. Arthur, and K.S. Booth, "Fish Tank Virtual Reality," *Proc. ACM INTERCHI'93 Human Factors in Computing Systems Conference*, pp. 37-41, Addison-Wesley/ACM Press, 1993.