Formative Evaluation of Flip Zooming: Towards Effective Integration of Detail and Context on Small Displays

Lars Erik Holmquist and Staffan Björk
The Internet Project
Viktoria Research Institute
Department of Informatics
S-411 80 Gothenburg
SWEDEN
{leh,bjork}@informatics.gu.se

ABSTRACT
We have developed and evaluated a prototype image browser based on *flip zooming*, a focus+context visualization technique. In a formative evaluation, we contrasted the technique with two other methods to present images. The evaluation showed that users appreciated the overview that the flip zooming prototype provided. However, we also learned that the current prototype lacked some important features, and that some users felt that the flip zooming layout was confusing. By using a formative evaluation method we gained valuable information which may be used when developing this or other focus+context techniques.

Keywords
Focus+context views, fisheye views, formative evaluation, image browsing, information visualization

INTRODUCTION
Small screens and the big picture
Despite the advances in human-computer interface research during recent years, the basic channel that is used for communicating information from the computer to the user has remained the same: a screen. Computer displays have not changed much since the days of the first workstations. But with the accelerating development in computer hardware and software, not to mention the fact that most computers nowadays are not isolated but part of a network, the users of today have access to hundreds or perhaps thousands of times more data than only a few years ago. How can we help making it possible to access all of this information through the same basic channel, a cathode-ray-tube screen 15 to 20 inches across? How can we present large amounts of information so that it makes sense and so that users can keep their overview, without getting lost in the details?

An obvious solution would seem to be to introduce bigger displays, but with current technology large and cost-effective screens with adequate resolution are still a long way off. In fact, it is entirely possible that the trend in the future will be in the opposite direction. The ongoing miniaturization of electronic components will most likely lead to smaller computing devices and hence smaller screens. It is obvious that practical strategies for displaying large amounts of information on small screens are becoming increasingly important as we approach the computing environment of the next century.

Focus+context visualization
In every-day life, people have developed strategies for working with large amounts of data. They spread out manuscripts on their desks to get an overview; they put up complicated diagrams on the wall and take a step back to "get the full picture"; they arrange notes and pictures on large surfaces to find meaningful patterns. An equivalent to these strategies applied to real-time computer graphics displays can be found in the various techniques collectively termed *focus+context visualization.* These are techniques that attempt to give users both the overview and the details that they need at the same time. Such techniques take their inspiration from the way that human vision works, in that they show the centre of attention – *the focus* – with a high amount of detail, while showing the surrounding information – *the context* – with much less detail. Thus, users can move their centre of attention to different areas, while maintaining the crucial overview.

In this paper, we describe the development and evaluation of a novel focus+context technique, which we hope will be a step towards solving the problem of displaying large amounts of information on small screens.

RELATED WORK
Focus+context techniques and applications
The *generalized fisheye view* provided the basis for much of the subsequent work on focus+context presentation [8]. An early graphical focus+context technique was the *bi-focal display*, which introduced horizontal distortion to the material outside the focus [22]. Techniques that introduce both horizontal and vertical distortion include the *rubber sheet view* [20] and the *graphical fisheye view* [19]. The *constrained zoom* [2] provided a fisheye view of graphs while maintaining the location of the nodes. Several techniques have used distortion inspired by geometry or optics, including the *perspective wall* [16], which gave a 3-dimensional view similar to the bi-focal display, and the *document lens*, which distorted a document in a way similar to a magnifying lens [18]. Several applications have aimed to provide focus+context views of information on the World Wide Web, including the *hyperbolic tree browser* [14], *zippers* [5], the *web book and web forager* [6], *imagetrees* [23], and the *zooming web-browser* based on the *PAD++ toolkit* [3, 4]. An overview of distortion-oriented techniques is given

in [15], and a relevant discussion of effective view navigation is given in [9].

Evaluations of focus+context applications

Evaluation of a zooming web-browser based on the PAD++-interface showed that experienced users found answers to questions significantly faster than when using a traditional browser [4]. When finding routes on a subway map, fisheye views showed a slight advantage over scrolling when all stations did not show up on the traditional display [10]. The performance of fisheye and full zoom views of a telephone network showed subjects completing a task more efficiently using the fisheye view [21]. An evaluation of *elastic windows*, a system for non-overlapping windows, showed it to allow faster performance than traditional window systems in some tasks [13]. Finally, the *CZWeb* browser was shown to aid users in navigating the World Wide Web and understanding its organization [7].

DEVELOPING A FLEXIBLE, LIGHT-WEIGHT FOCUS+CONTEXT TECHNIQUE

As is evident from the above, a large number of focus+context techniques have been developed. However, few seem to have found their way into real-world products. One reason may be that many of the techniques involve advanced calculations and hence require expensive hardware to be used effectively. Another may be that they are best suited for specialized data sets, such as graphs, and that there is no easy way to adapt existing data for use with these techniques. Finally, it is possible that some of these techniques are simply too complicated for every day use, and that the gains users can achieve from these methods are countered by the efforts needed to understand and learn to use them in the first place.

When we set out to develop a new focus+context technique, we found it very important that our solutions was practical and put low demands on the hardware that end users would need. It should also be easy to understand, and be flexible enough to adapt to many different data set. What we were looking for was a *flexible, light-weight focus+context technique.* The need for such a technique can only increase in view of the likely future proliferation of small, mobile computing devices, since such devices typically have very limited processing power compared to desktop workstations, and will commonly be placed in the hands of inexperienced users.

After studying the earlier focus+context techniques, we started experimenting with various ways to apply a focus view to a set of images or document pages laid out sequentially on a 2-dimensional surface. Our first attempt at a focus+context view (Figure 1, left picture) was inspired by the document lens, but with size reduction applied to each image individually, equally in both dimensions. A problem with this view is that the thumbnails close to the focus are smaller than those further away, which may be undesirable, since the information closest to the focus is often regarded as more important for the context. Worse, the distribution of the non-focus images is unclear – there is no obvious correlation between the un-focused view and the focus view.

A second attempt (Figure 1, right picture) was inspired by the rubber sheet view but without the deformation of the individual thumbnails that would occur in such a display. This view had the advantage of a clear correlation between the image distribution in the non-focus and focus views, but it made poor use of the display, with much space left empty.

We finally arrived at the simple method shown in Figure 2. A data set, such as a the pages of a document, a collection of images or a set of buttons in a graphical user interface, is laid out sequentially as a collection of thumbnail representations (Figure 2, left picture). When a thumbnail is brought into focus, it is zoomed to full size and placed approximately in the middle of the display. The remaining thumbnails are reduced in size and arranged around the focus (Figure 2, right picture).

To maintain the context in a sequentially ordered data set such as a document, it is important to give an indication of each image's place in the sequence. To accomplish this, in the un-zoomed view, the thumbnails are placed sequentially in a left-to-right, top-to-bottom fashion. In the zoomed view, the thumbnails which come before the focus image in the sequence are placed above and to the left of the focus page, and the following thumbnails are placed below and to the right. This solution may not communicate the exact place of the focus image in the sequence very well, but it should give quite a good idea of its approximate position in the material, as well as the relative size of the material.

THE FLIP ZOOM TECHNIQUE

We have called the technique described above *flip zooming* [12]. Users navigate through a data set by "flipping"

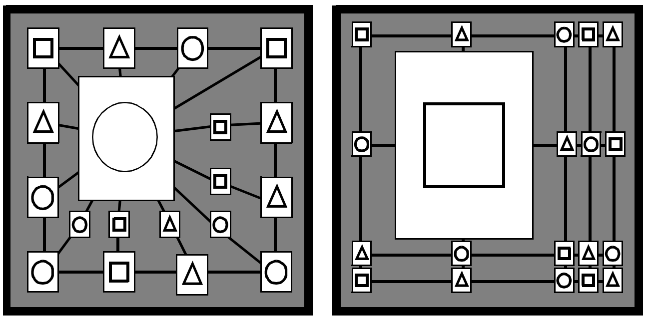

Figure 1. Early design ideas

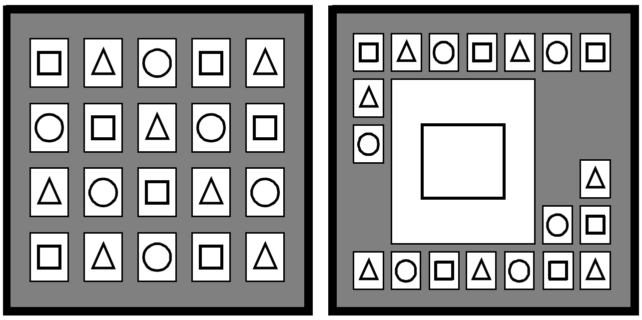

Figure 2. Flip zooming: Un-zoomed view (left); zoomed view (right)

through it, like the pages in a book. When users wants to examine an entry, such as an image, they select it by clicking on its thumbnail representation. The image is then zoomed to a readable size. The surrounding thumbnails are reduced in size and re-arranged to accommodate the expanded image. When users want to view a new entry, they can either “flip” to the next or previous in the sequence, using a GUI command or a keyboard short-cut, or they can select a new entry from another part of the document randomly, by clicking on its thumbnail representation.

Flip zooming offers a good real-time performance, since no advanced math is required for calculating the context display. The technique does not introduce spatial deformation of the context material, unlike many previous focus+context techniques. It offers the possibility to introduce other context information than thumbnails, such as keywords or icons. In comparison to currently used methods to present a set of thumbnails and a focus image, which typically involve two separate displays with thumbnails and focus image side by side, flip zooming constitutes a more direct connection between focus and context, since a thumbnail and the corresponding focus image never appear on the display simultaneously. Finally, flip zooming has the advantage that all thumbnails are always available, and no scrolling is needed to find the required image.

There are some problems with the technique. The most notable is that of scalability, an important issue with any focus+context technique. At some point there will simply be too many thumbnails on the display, and the method will break down. This might be overcome by introducing a hierarchical ordering of the information, to “hide” thumbnails that are not used. Another problem is that of the unpredictability of where the focus images appear, which might be confusing for users and break the context. Solutions might include smooth animation to ease the transition between different displays, and attempts to place the focus image as close to where users expect it to appear as possible. Extensive user studies will be needed to fine-tune such solutions.

FLIP ZOOMING PROTOTYPES

First prototype: A text-only web browser

The first implementation of flip zooming was the Zoom Browser [11]. This was a text-only browser for the World Wide Web, which presented web pages using the flip zoom technique. The implementation was too limited in functionality to be used as the basis of any extensive user studies, but it did serve as a proof-of-concept implementation that encouraged us to pursue the technique further.

Second prototype: An image browser

For our second prototype we implemented an image browser that presents a set of pictures using the flip zoom technique (Figure 3). The browser can present any type of images, such as photographs, drawings, lecture slides or scanned documents, which makes it more flexible than the first prototype. The application was written in Java, to allow easy integration with web-based material. The browser lets users click on any picture to bring it into focus, as described previously. There is also the option to supply a navigation bar, to let users flip to the next or previous picture by clicking a UI button.

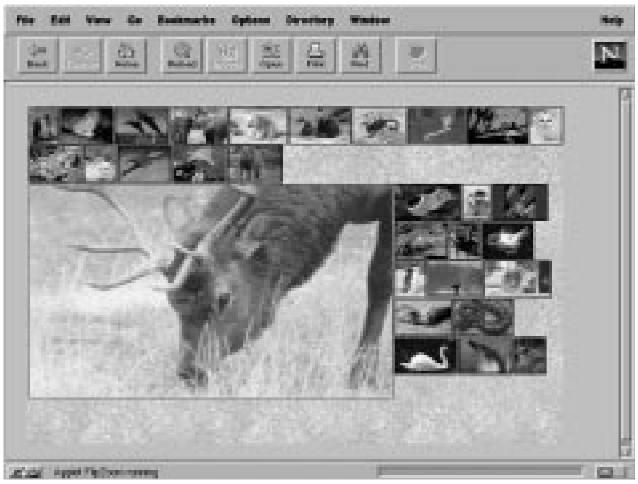

Figure 3. The flip zooming applet, running in a standard Java-enabled browser

The image browser can present image sets of any size, but with too many images on the display the thumbnails tend to become too small to be useful. It turned out that scaling pictures in real time was too inefficient in the then-current implementation of Java, so the applet used a set of pre-scaled thumbnails in different sizes, which were downloaded as they were needed. This limited the flexibility of the display somewhat but did not seem to pose any major problems.

With a stable prototype, we could now proceed to designing an evaluation which would help our further development work.

AIMS OF THE EVALUATION

The second flip zooming prototype had an application area – image browsing – which was general enough to formulate a variety of different tests. We decided to design an evaluation where we contrasted the flip zooming prototype with some other image browsing methods currently in use. When designing the evaluation we used an accepted standard work on human-computer interaction as our guide [17].

A goal of the evaluation would be to determine if the flip zoom technique offered the advantages that we hoped, those of a simultaneous good overview and access to the details of a large material. But perhaps more importantly, we wanted the evaluation to draw our attention to any flaws in the technique and/or the design of the prototype. The evaluation would be *formative* in the sense that evaluation process would affect an evolving design [17, pp. 603]. We wanted the evaluation to be very open-ended, to give us as much information as possible. We felt that completely free-form input from the users would serve as the most useful help in our continuing design process, since it would provide us with opinions and suggestions from people not previously familiar with the technique.

MATERIALS USED IN THE EVALUATION

Image sets

We decided to use two image sets as a basis for the evaluation: one which consisted of images that were designed to be read in order, and one which contained no inherent ordering. One of the image sets would consist mainly of text, and the other of pictures only. The image sets were:

- A set of lecture slides, a total of 34 images
- A set of photographs of various animals, a total of 30 images

Having determined which image sets to use, we then needed to find some presentation methods to contrast with flip zooming.

Presentation methods

We wanted to use popular and readily available applications, which attempted to provide some kind of overview over an image material. Furthermore, since we realized the importance of the Internet and the World Wide Web as a medium for distribution of images, we wanted all methods to work over the web.

We decided to use two methods in addition to the flip zooming image browser: a frame-based web-page, generated by Microsoft's PowerPoint application; and Adobe's Acrobat Reader for PDF files. The presentations were optimized for a standard 800 by 600 pixel display, and all images were cached on the local machine to avoid any differences due to network connection speed. In the following, we will describe the particulars of the three methods in more detail.

Method 1: The flip zooming image browser (Figure 3)

Here, the image sets were presented with the flip zoom technique described previously. No GUI or keyboard short-cuts for navigation were supplied, only the basic flip zoom functionality. Two sizes of pre-scaled images were used: a set of thumbnails, and a set of full-size images. The presentation was run in a Java-enabled browser.

Method 2: A web-page presentation with frames, as generated by PowerPoint (Figure 4)

Since one of our image sets consisted of slides prepared in PowerPoint, we used the presentation method provided by this application. PowerPoint includes functionality to automatically generate a web page with several frames. In the left-hand frame a list of slide titles is shown, and a full-size image of a slide is shown in the right-hand frame. For the second set of images, we improved the presentation by manually adding thumbnails (see below). VCR-style navigation buttons are provided in two other frames. The presentation method was run in a Java-Script-enabled browser.

Method 3: The Acrobat Reader (Figure 5)

The Portable Document Format (PDF) is a popular file format for distributing documents on the web. In the Acrobat Reader for PDF files, one or more columns of thumbnails are presented in a horizontal strip to the left of the main image. Typically, the thumbnails are presented as outlines only (see below). A set of VCR-style navigation buttons are provided. The presentation method was run as a browser plug-in.

A note on thumbnails and residue

In discussions of effective view navigation, information residue has been introduced as a measure of navigability [9]. By residue is meant the amount of information there is in one view of an information set about all other possible views of the same information set. Thumbnails would seem to be a way to add information residue while navigating a set of images, and as the subsequent evaluation showed, thumbnails were an important factor in users' appreciation of a a system. Surprisingly, however, neither PowerPoint nor the Acrobat Reader supplied a straightforward method to include thumbnails in the presentation.

PowerPoint generated a web-page with an index consisting of the header for each slide, but no thumbnails. There was, however, the opportunity to view the full text of each slide in the index. Since the second image set consisted of images with no headers, all index entries simply read "PPT Slide".

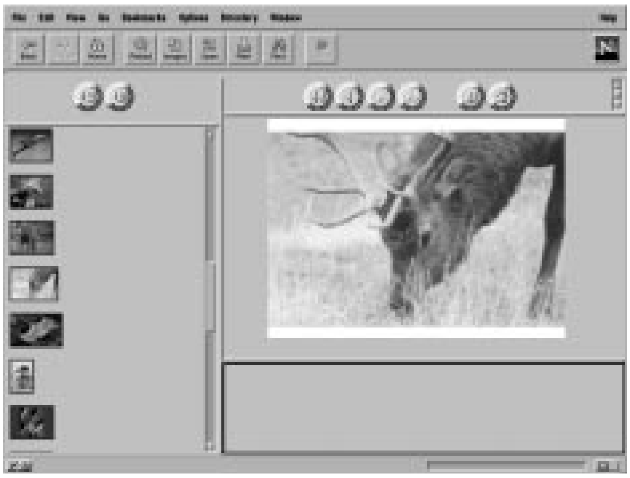

Figure 4. Web-page generated by PowerPoint, with thumbnails added manually

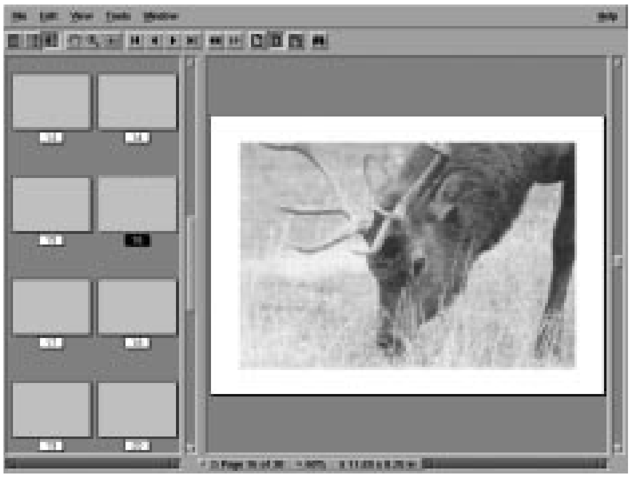

Figure 5. The Acrobat Reader; note that thumbnails are shown only as outlines

	Flip zooming	PowerPoint web Page	Acrobat Reader
Strengths			
Good overview	10	3	1
Alternative search methods	0	8	0
Clear structure	0	0	5
Weaknesses			
No / too small thumbnails	5	0	9
Unclear structure	5	1	0

Table 1: Strengths and weaknesses of the systems, as stated by 10 users

It might be argued that Power Point was not intended for presenting strictly picture-based information, and therefore we decided to enhance the index created for the second image set by manually adding thumbnails. (Figure 4)

A similar problem occurred with the Acrobat Reader. We were quite puzzled by the fact that the application did not show “real” thumbnail images, only outlines. Since the application is capable of scaling pictures in real-time, generating thumbnails should not be a problem. Eventually, we found that thumbnails have to be generated separately when a PDF file is created, but not even Adobe themselves seem to see a reason to provide them in the guide documents provided with the program [1]. We decided to use the thumbnail outlines only, partly because it seemed to be the norm for this presentation style and partly because we found it interesting to contrast a method not using “real” thumbnails with others that do. (Figure 5)

EVALUATION METHOD

The evaluation group consisted of ten subjects with computer experience ranging from moderate to very high. The group was chosen from co-workers and associates which had no previous experience of flip zooming. The group included four females and six males, with ages ranging from 20 to 60. The users were told that they would help evaluate three different systems for presenting images. The users knew that the flip zooming image browser had been developed by the research group. Think aloud protocols were used [17, pp. 622] and care was taken not to inhibit the participants.

The subjects were first asked to perform a number of tasks to bring their attention to different ways of using the systems. There were a total of nine tasks, which were rotated so that each task was performed with each system at some point. The tasks were varied so that they required the use of different search strategies, such as pattern recognition, image recall, and so on. The users were permitted to perform the tasks in any way they liked, using all available features of the respective applications. Examples of tasks were:

- ”Find the slide on which the terms *push* and *pull* are used”
- ”Find out if there are more cats than fishes in the pictures”

After the tasks were performed, each user had a debriefing session similar to the ones used in co-operative or participative evaluation [17, pp. 661-664], and answered a number of semi-structured interview questions [17, pp. 629]. The users were asked to point out the strengths and weaknesses of each of the presentation methods. At no point did the interviewer try to lead the subjects by suggesting any particular strengths or weaknesses of the different applications. The users were also asked to suggest ways in which the different applications could be improved.

After the interviews were finished, the material was processed. The stated strengths and weaknesses of each system were collected in groups, so that differently worded responses with the same meaning were grouped together.

EVALUATION RESULTS

The users were allowed to submit the same strength or weakness for more than one system. In the following, we will list those points which were mentioned regarding at least one of the presentation methods by at least 50% of the subjects (i.e. 5 or more out of 10 subjects). The results are summarized in Table 1.

When considering the results, it is important to remember that we wanted to evaluate the *flip zooming technique*, rather than the specific applications used. Our main goal was to find the strengths and weaknesses of the technique as currently implemented, and draw conclusions on possible areas of improvement for future prototypes. Because of this, we will not attempt to draw any conclusions on which of the applications was the “best”.

Strengths

Good overview (flip zooming)

All 10 users said that the flip zooming image browser provided a good overview of the material. We felt that this response was very encouraging in view of the fact that the responses were totally free-form, with no leading questions from the interviewers. Only 3 users felt that the PowerPoint-generated web page gave a good overview, and only one user felt that a good overview was a strength of the Adobe

Acrobat Reader. (It is quite possible that more users would have felt that the Acrobat Reader offered a good overview if we had included "real" thumbnails, but this should have little bearing on the result for the flip zooming application.)

Alternative search methods (PowerPoint web page)

8 users thought that the fact that the PowerPoint-generated web pages provided alternative search methods was a strength. The subjects used the built-in search function in the web browser to search for important keywords, probably because they were already familiar with this feature being offered by the browser. The flip zooming prototype provided no alternative to visual search, and although the Acrobat Reader does provide text search, this function was unknown to most users and was not used.

Clear presentation – "neat and tidy" display (Acrobat Reader)

5 users thought that a strength of the Acrobat Reader was that the material was very clearly presented – several users spoke of it as "neat and tidy". No user thought that any of the other methods had a clear presentation structure.

Weaknesses

No thumbnails / Too small thumbnails (Acrobat Reader / flip zooming)

9 out of the 10 users we interviewed thought that the lack of real thumbnails was a weakness in the Acrobat Reader (as mentioned above, only outlines of the images were shown). 5 users also complained that the thumbnails provided by the flip zooming browser were too small. If we had not manually added thumbnails to the PowerPoint-generated web page, it is likely that it would fallen under this point as well.

Confusing presentation – "messy" display (flip zooming)

5 users thought that the flip zooming image browser was confusing and gave no clear indication of the structure of the presented material. They thought that the display was too crowded and "messy". Only one user thought that the presentation on the PowerPoint-generated web page was confusing, whereas no users mentioned the Acrobat Reader in this regard.

Other observations

We also made a number of other observations while the users were performing the tasks, which were not easy to quantify. The most important were:

- *Overview allows approximative answers and "multi-tasking"*

  When asked questions such as "Are there more cats than fishes in the pictures?" many users would quickly give an approximate answer if they had an overview of the whole set. They then examined some pictures in detail to confirm the answer. The overview also allowed a form of "multi-tasking": When explaining a search strategy used in the flip zooming image browser, users sometimes continued to check the pictures, occasionally correcting the original answer while explaining how it was found!

- *"Brute force" searching is prone to errors*

  Using "brute force" searching, i.e. looking at each image in turn, often led to mistakes such as overlooking the right answer or confusing issues like "the number of cats" and "the number of pictures of cats". These errors were far less common when users took advantage of an overview display.

CONCLUSIONS AND DISCUSSION

Our main conclusion from the evaluations were:

- *Flip zooming provides a good overview*

  All 10 subjects in the evaluation pointed out good overview as a strength of flip zooming. In this respect, flip zooming performed significantly better than the other methods evaluated (although it is possible that if we had included "real" thumbnails, the Acrobat Reader might have performed better than it now did). The result serves as an indication that in situations when an overview is needed, the flip zoom technique provides a viable way to achieve such an overview.

- *A clear structure is important to users*

  The evaluation indicates that it can be a good idea to consider a trade-off between information density and clarity. Although some users complained that the thumbnails provided by flip zooming were too small, a cluttered display crammed with information might not be as useful as a well structured display with slightly less information.

- *Thumbnails are appreciated by users, but are not always easy to provide*

  Thumbnails provide important information residue, and this may be an important factor in the relative success of flip zooming. This was not particularly surprising, but what surprised us was that it was so difficult to provide thumbnails using the commercial applications in the study.

- *Users want alternative search methods*

  Again, not particularly surprising, but we had not thought to provide alternative search methods in our prototype, and most users never found the text search function provided by Acrobat Reader.

Our other observations seemed to confirm that the overview provided by flip zooming was a help to users when performing the different tasks. When using "brute force" searching, users seemed less efficient and often made errors, whereas a good overview allowed them to give approximate answers quickly and then refine them.

The evaluation served to encourage us to pursue the work with flip zooming further, while at the same time pointing out several areas for possible improvement. We feel confident in concluding that flip zooming provided a good overview, but the presentation will need to be tightened up to convey the structure of the material more clearly. We believe that focus+context methods can be useful for many situations when display space is scarce, and that flip zooming presents a realistic alternative to other methods.

FUTURE WORK

The evaluation described in this paper points to several possible improvements of the existing flip zooming prototype, including providing a more clearly structured display and alternative search methods. With an improved prototype, new evaluations could be performed, including quantitative tests to determine if the flip zooming image browser is more efficient than other applications for performing some well-defined task.

To date, we have implemented flip zooming for use with text documents and image sets. There is need to determine how to adapt flip zooming to other types of data. This might include a general presentation or windowing system which uses flip zooming instead of e.g. overlapping windows. Flip zooming might also be integrated in a graphical user interface, so that various buttons or menu items can be brought in and out of focus as they are needed.

The development of small but powerful computing devices should provide many interesting opportunities for focus+context visualization research. An important future work will be to adapt flip zooming and similar techniques to work on small devices such as PDAs and mobile phones.

ACKNOWLEDGEMENTS

We are grateful to our colleagues at the Viktoria Institute and the Internet Project for their assistance. In particular, Henrik Fagrell was instrumental in the process of putting this paper together. Roberto Busso developed the image browser used in the evaluation. The construction of the prototype was funded by NUTEK in cooperation with SSKKII. Christopher Ahlberg and Erik Wistrand were of great assistance during the initial development of flip zooming. Finally, we wish to thank all the test subjects for their cooperation.

REFERENCES

1. *Adobe Acrobat Reader Online Guide*, available with the Acrobat Reader software from http://www.adobe.com/prodindex/acrobat/main.html

2. Bartram, L., Ho, A., Dill, J., Henigman, F., The Continuous Zoom: A Constrained Fisheye Technique for Viewing and Navigating Large Information Spaces, in *Proceedings of ACM UIST '95*, pp. 207-215, ACM Press, New York, 1995.

3. Bederson, B., Hollan, J., Pad++: A Zooming Graphical Interface for Exploring Alternate Interface Physics, in *Proceedings of ACM UIST '94*, ACM Press, New York, 1994.

4. Bederson, B., Hollan, J.D., Stewart, J., Rogers, D., Druin, A., Vick, D., Ring, L., Grose, E., Forsythe, C., A Zooming Web Browser, to appear in Ratner, Grose, and Forsythe (eds.): *Human Factors in Web Development.* Also available as: http://www.cs.unm.edu/pad++/papers/web-browser/web-browser.html

5. Brown, M.H., Weihl, W.E., Zippers: A Focus+Context Display of Web Pages, in *CD-Rom Proceedings of WebNet '96*, Association for the Advancement of Computing in Education (AACE), 1996.

6. Card, S.K., Robertson, G., York, W., The WebBook and the Web Forager: An information workspace for the World Wide Web, in *Proceedings of ACM SIGCHI '96*, pp. 111-117, ACM Press, New York, 1996.

7. Fisher, B., Agelidis, M., Dill, J., Tan, P., Collaud, G., Jones, C., CZWeb: Fish-eye Views for Visualizing the World Wide Web, in *Design of Computing Systems: Social and Ergonomics Considerations* (Eds. Smith, M., Salvendy, G. and Koubek, R. J.), pp. 719-722, Elsevier, Amsterdam, 1997.

8. Furnas, G.W., Generalized Fisheye Views, *Proceedings of ACM SIGCHI '86*, pp. 16-23, ACM Press, New York, 1986.

9. Furnas, G.W., Effective View Navigation, in *Proceedings of ACM SIGCHI '97*, pp. 367-374, ACM Press, New York, 1997.

10. Hollands, J.G., Carey, T.T., Matthews, M.L. and McCann, C.A., Presenting a graphical network: a comparison of performance using fisheye and scrolling views, in *Designing and Using Human-Computer Interfaces and Knowledge Based Systems* (Eds. Salvendy, G. and Smith, M.), pp. 313-320, Elsevier, Amsterdam, 1989.

11. Holmquist, L.E., Focus+Context Visualization of Web Pages with Flip Zooming and the Zoom Browser, in *Extended Abstracts of ACM SIGCHI '97*, pp. 263-264, ACM Press, New York, 1997.

12. Holmquist, L.E. and Ahlberg, C., Flip Zooming: A Practical Focus+Context Approach to Visualizing Large Information Sets, in *Design of Computing Systems: Social and Ergonomics Considerations* (Eds. Smith, M., Salvendy, G. and Koubek, R. J.), pp. 763-766, Elsevier, Amsterdam, 1997.

13. Kandogan, E., Shneiderman, B., Elastic Windows: Evaluation of Multi-Window Operations, in *Proceedings of ACM SIGCHI '97*, pp. 250-257, ACM Press, New York, 1997.

14. Lamping, J., Rao, R., Pirolli, P., A focus+context technique based on hyperbolic geometry for viewing large hierarchies, in *Proceedings of ACM SIGCHI '95*, ACM Press, New York, 1995.

15. Leung, Y.K, Apperley, M.D, A Review and Taxonomy of Distortion-Oriented Presentation Techniques, in *ACM Transactions on Computer-Human Interaction*, vol. 1 no 2, pp. 126-160, 1994.

16. Mackinlay, J. D., Robertson, G. G., Card, S. K, The Perspective Wall: Detail and Context Smoothly Integrated, in *Proceedings of ACM SIGCHI '91*, pp. 173-179, ACM Press, New York, 1991.

17. Preece, J., Rogers, Y., Sharp, H., Benyon, D., Holland, S., Carey, T., *Human-Computer Interaction*, Addison-Wesley, 1994.

18. Robertson, G.G., Mackinlay, J.D., The Document Lens, in *Proceedings of ACM UIST '93,* pp. 101-108, ACM Press, New York, 1993.

19. Sarkar, M., Brown, M.H, Fisheye Views, *Communications of the ACM,* vol. 37 no. 12, pp. 73-84, 1994.

20. Sarkar M., Snibbe, S.S., Tversky, O.J., Reiss, S.P., Stretching the Rubber Sheet: A Metaphor for Viewing Large Layouts on Small Screens, in *Proceedings of ACM UIST '93,* pp. 81-91, ACM Press, New York, 1993.

21. Schaffer, D., Zuo, Z., Greenberg, S., Bartram, S., Dill, J., Dubs, S., and Roseman, M., Navigating hierarchically clustered networks through fisheye and full-zoom methods, *ACM Transactions on Computer-Human Interaction,* Vol. 3, No. 2, pp. 162-188, June 1996.

22. Spence, R., Apperley, M., Data base navigation: an office environment for the professional, in *Behavior and Information Technology,* vol. 1 no. 1, pp. 43-54, 1982.

23. Wistrand, E., Creating Image Context using Image-Trees, in *ACM SIGCHI '96 Conference Companion,* ACM Press, New York, 1996.