Focus + Context Screens: A Study and Evaluation

November 6, 2003

David Mitchell
Dr Andy Cockburn (supervisor)

Abstract

Display and manipulation of large documents on a standard display has long been a difficult problem. In this report we extend the work of Baudisch in Focus+Context screens. These screens combine a large low resolution display with a standard high resolution display to give a cost effective, large display. This report describes the physical and software implementation of our own focus+context screen. An ecological study is presented that finds focus+context screens have many potential areas of application. Two formal evaluations comparing the screen to other focus+context solutions and investigating the types of task the screen is best suited to. We find that the focus+context screen provides increased performance for tasks at a large scale, especially on documents the user is familiar with, and that users tend to enjoy using the focus+context screen.

Contents

Chapter 1

Introduction

Large documents cannot easily be effectively displayed on standard displays. A number of techniques exist to facilitate display of large documents, for example Zoom and Pan interfaces. However, if a user wishes to see a large area of a document using this they must zoom out, in which case they can no longer see the detail of a document. Once they zoom in again it is easy to become disoriented. An alternative to using different user interfaces on a standard display is to use a large, high-resolution display, which can display a large part of the document in high-resolution. Unfortunately, such displays are not cheaply available with today's technology and expensive, multiple screen, setups must be used.

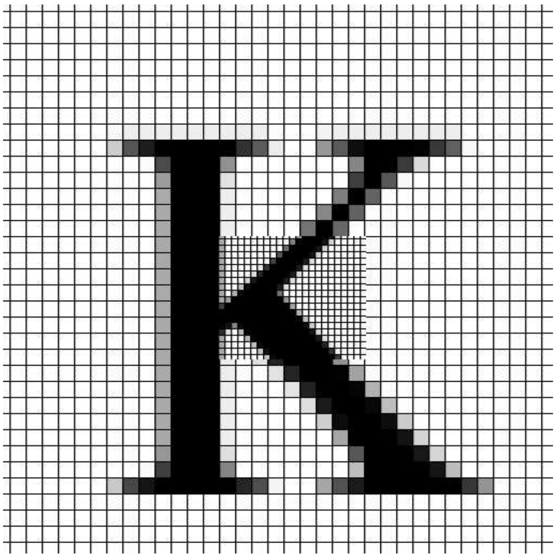


Figure 1.1: The high resolution focus area is nested inside the low resolution context

Baudisch et al (Baudisch, Good & Stewart 2001, Baudisch, Good, Bellotti & Schraedley 2002) introduced the notion of a focus plus context screen. This display projects a large, low-resolution display (the context) and uses a standard monitor to provide a high-resolution area in the centre of this (the focus). A single document is then displayed across both screens to provide a large mixed-resolution display. Figure 1.1 shows the arrangement of the focus+context screen. This allows the user to maintain context information and while being able to read detail in the centre of the document. Baudisch evaluated this against traditional setups (Baudisch et al. 2002) and found it provided improved performance for a range of tasks dealing with large documents such as a street map.

Baudisch's work was preliminary and there are a number of aspects of the focus+context screen that have yet to be investigated. This report documents three investigations into the focus+context screen. Firstly an ecological study is presented that informally investigates how focus+context screens could be applied to real-world tasks. Two formal evaluations of different aspects of the focus+context screen are also presented.

Chapter 2

Related Work

Baudisch et al. (2001) first proposed the focus+context screen as a way to "fit a larger piece of large visual objects into a display in order to let users save zooming interactions". This screen consists of a large board, onto which an image is projected, with a conventional monitor placed in a hole cut in the centre of the board. The monitor and projector are synchronised so that a single image can be displayed across the entire screen. This results in a high resolution area being present in the middle, allowing a detailed focus area to be surrounded by a large context area.

The focus+context screen spans a number of research areas. It is a new focus+context technology that may be suitable for augmentation by other visualisation techniques.

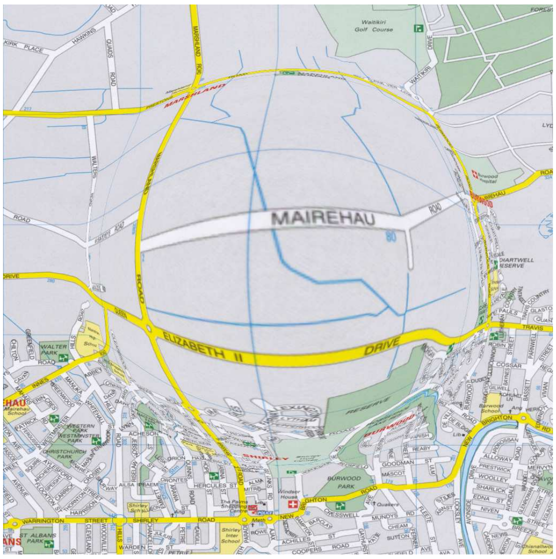


Figure 2.1: A fisheye map viewer

The focus+context screen is also related to research into large displays and extended desktops, since the context area of the screen is projected onto a large board. This report will concentrate on the focus+context screen as a focus+context solution rather than how it relates to large displays in general.

2.1 Related Work in Focus + Context Visualisation

Documents that are too large to be displayed at full size on a typical display are common. These so-called "multiscale" documents and the visualisation techniques used to effectively view them are an extensive area of research. The problem is obvious: how to view the focus of attention in sufficient detail while maintaining a feeling of the context of the focus in the whole or surrounding document.

2.1.1 Other Focus+Context Techniques

A number of other focus+context techniques have been developed. Zoom+pan allows the user to pan around their information space with a mouse and zoom in on areas of interest. This allows the user to gain an overview of the information space while at low magnification and to view detail after zooming in. If the user becomes disoriented, zoom+pan requires the user to zoom out to find their current location.

Overview+detail presents a permanently magnified view with a smaller overview that displays the entire information space. This is intended to assist users in maintaining their orientation within the document.

Fisheye views keep "a balance of local detail and global context" (Furnas 1986). A degree of interest function of items in an information space is used to allocate display space to those items. An "interesting" item will be allocated more display space than one that is less so. Furnas first used this technique to develop a code browser that allowed the user to "collapse" unimportant sections of code so that only the important parts of a file were visible. A number of other fisheye applications have been developed. Bartram, Ho, Dill & Henigman (1995) developed an application for viewing hierarchical graphs that enlarges graph nodes as they were selected. Bederson (2000) developed fisheye menus which enlarge menu items under the mouse cursor. Robertson & Mackinlay (1993) developed the "Document Lens" which lays a document across a 3D truncated pyramid. This is viewed from the top to provide a distorted view of the document with an undistorted focus region.

An example of a fisheye map browser is shown in Figure 2.1.

The hyperbolic tree (Lamping & Rao 1994) is a fisheye technique for the display of hierarchical trees that renders the tree on a hyperbolic plane and map this plane onto a circle. This has the effect of making nodes further from the centre appear smaller.

Speed-dependent Automatic Zooming (SDAZ) automatically zooms out a document when panning or scrolling quickly (Igarashi & Hinckley 2000). This allows users to more "efficiently and smoothly navigate through a large document without becoming disoriented" (Igarashi & Hinckley 2000). This technique gives the user a focus+context view of the document similar to zoom+pan , whilst performing zooming automatically.

2.1.2 Previous Focus+Context Evaluations

Baudisch et al. (2002) compared an overview+detail interface, a zoom+pan interface and a focus+context screen. They found the focus+context screen to be most efficient for a number of tasks on both static and dynamic data. Users reported a preference for the focus+context screen.

Overview+detail and zoom+pan interfaces have been compared with conflicting results. Baudisch et al.'s evaluation found the overview+detail interface produced faster task completion times than the zoom+pan interface. Hornbæk, Bederson & Plaisant (2002) compared overview+detail and zoom+pan interfaces and found the zoom+pan interface to be faster. Hornbæk et al. proposed some explanations for the improvement of subjects when using the zoom+pan interface. They propose that the overview is a distraction, requires mental and physical effort to use, or is too coarse when the detail is at a high level of magnification. Of these, the third provides a possible explanation for the contradiction with Baudisch et al.'s result. Baudisch et al. used a dual monitor interface that placed the overview on a second monitor, which resulted in increased clarity. It is possible that Baudisch et al.'s overview+detail interface was more conducive to fine-grained navigation.

Most of the fisheye techniques mentioned above have had little or no formal evaluation. Lamping, Rao & Pirolli performed an evaluation of the hyperbolic tree browser. The evaluation found no significant difference between the hyperbolic tree and a conventional 2D scrolling browser. This evaluation had only four participants.

2.2 Related Work in Large and Hi-Res Displays

Although not the focus of this report, large displays are an active area of research. Research has included how people use large displays, how the differing fields of view affect performance, new interaction techniques and use of large displays for collaborative tasks.

Grudin studied the use of multiple monitors in the workplace (Grudin 2001) and how the extra screen real estate was put to use. Grudin found the extra space was put to good use but that not many applications had good support for multiple monitors so users were forced to arrange windows themselves.

Tan, Gergle, Scupelli & Pausch (2003) investigated the effect a large display has on spatial performance with a constant field of view and found the large display produced increased performance for spatial tasks.

Czerwinski, Tan & Robertson (2002) studied how men and women responded to an increase in the field of view and found women experienced a greater performance increase than men.

Grossman, Balakrishnan, Kurtenbach, Fitzmaurice, Khan & Buxton (2001) designed a two-handed interaction technique for 3D modelling on large screens.

Tani, Horita, Yamaashi, Tanikoshi & Futakawa (1994) built a system that included a large screen to display overview of collaborative work, while each participant had an individual standard display for interacting with detail.

Although not investigated greatly in this report, all of these areas of research are relevant to the focus+context screen.

2.3 Other Related Work

Fitzmaurice (1993) investigated the use of spatially aware palm-top devices to display information about what they are held up to. For example a palm-top might be held up to a book in a library and it would display information about the book. We envisage a similar system in which a small, high-resolution map could be displayed on the palm-top to fill in the area which it was held up to.

Chapter 3

System Design

This chapter explains the physical construction and software implementation for the focus+context screen.

3.1 Physical Construction

Baudisch describes in detail how they built their focus+context screen, and their approach has been mostly followed here with a few modifications. Baudisch's focus+context screen leaned back away from the user against a wall. This allowed the projector to be mounted behind the user without the user's head casting a shadow. However, we feel a vertical display is more natural, especially as a standard display is often vertical. With this in mind a 5'4" by 4' section of 5mm plywood was cut and mounted on a custom built frame. A hole was cut in the centre large enough for a 17" CRT monitor. The screen was set on a desk and a 17" CRT flat screen placed in the hole. Paper was then used to mask the edges of the monitor and the gap between the board and the monitor. This allows the image projected on the screen to come all the way up to the edge of the monitor and produce a seamless image. Figure 3.1 shows the final screen from the rear. The final screen is capable of creating a display equivalent to approximately 5000x3750 pixels.

3.1.1 Shadow Casting

Baudisch leaned their screen back to avoid shadow casting, so to solve the shadow casting problem we mounted our projector to the side of the screen. This resulted in bad distortion of the image which was corrected for by the video driver. This had a significant performance hit so a new projector was sought that would allow the distortion to be corrected in hardware. The new projector cannot correct for the distortion from the side throw but can correct for a vertical misalignment. We decided to mount the projector above and behind the user similar to Baudisch's screen. It was found that some users cast a small shadow and the bottom of the image but it is not particularly intrusive so this compromise has been accepted.

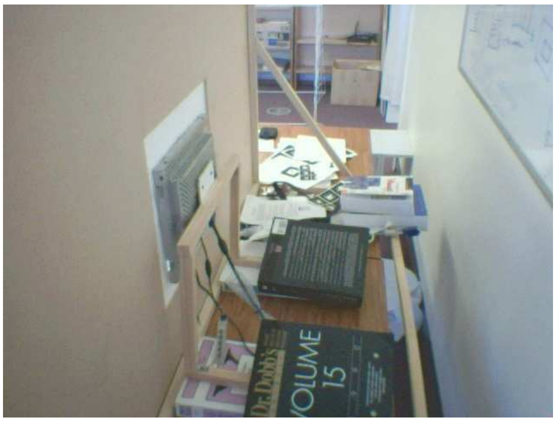


Figure 3.1: Rear view of the focus+context screen, illustrating the construction

3.1.2 Parallax Error

The CRT flatscreen monitor had the characteristic that the image seemed to sit about a centimetre behind the surface of the glass. This meant it was impossible to get the focus image flush with the projected context image. The solution was to use a LCD flatscreen monitor. The image on the LCD appears right on the surface of the screen. The plastic casing on the LCD was removed to allow it to be mounted flush with the projector screen.

3.2 System Implementation

This section describes the hardware and software used to implement applications on the focus+context screen.

3.2.1 Apparatus

The focus+context screen is run on a single 2.4Ghz Intel Pentium 4 with 512Mb RAM. The computer has a dual-headed nVidia GeForce4 Ti4200 video card. The LCD monitor used is a Dell 1503FP. The context is projected using a Panasonic PT-LC75E LCD projector. The computer runs Windows XP configured to spread the desktop across the two displays provided by the dual-head video card. One of these displays is the focus, the other is the context.

3.2.2 Software

Anything displayed on the context is blown up by the projector to approximately five times its original size. To compensate for this, the image displayed on the projector must be five times smaller than the one on the focus. This means the two will appear to be the same scale and the illusion of a single display is created. Because of this, the software for the focus+context screen had to be specially created, as there is no support in Windows XP for scaling down a desktop. Conceivably, this extension is possible for any window manager, but it is outside the scope of this report.

A number of applications were implemented on the focus+context screen. Firstly, a Java image viewer was written. This loads a single image file and creates two copies of the image: one full size and one scaled down for display on the context. These two images are then displayed in two separate windows. The window with the full size image is maximised on the focus display, while the other is moved onto the context display and maximised there. The software is written to keep the two images synchronised so that when one is panned, the other is too. This gives the desired single-image effect. This Java prototype has a number of limitations. It could only display images of severely limited size before performance became an issue. Also, calibration of the displays to align them properly was difficult. The prototype only supported variation of the position of the focus vertically, meaning that the projector must be repositioned by hand to get correct horizontal alignment. The prototype required the calibration information to be passed as command line arguments, which made calibration a trial and error exercise, requiring the software to be restarted for every calibration change.

The Java prototype was superseded by the current FcViewer. This is written in OpenGL, allowing it to take advantage of hardware acceleration. The FcViewer loads a processed image as a series of 1024x1024 pixel tiles. These are then converted to texture maps and placed on OpenGL quads. These quads are only rendered if they are at least partly visible, in the interests of performance. The FcViewer also has a much improved calibration system. It can be put into a calibration mode, which requires the user to drag four lines about the screen until they line up on both displays. A typical calibration now takes about 30 seconds. This FcViewer can display any appropriately processed image and is easily capable of working with images 8000x6000 pixels. Processing of an image consists of converting it to a series of 1024x1024 24bit TGA images and writing an appropriate configuration file to describe how the tiles fit together. Performance is only limited by physical RAM. Currently, images much larger than 250Mb will incur a performance hit every time a new tile comes into view. Increasing the amount of RAM will increase this limit.

The OpenGL FcViewer was also used as the base for the GIS Viewer described in Chapter 4.

The iD Software game “Quake 2” was adapted to run on the focus+context screen. Quake 2 is a 3D first person perspective shoot-em-up. Running on the focus+context screen was accomplished by running two copies of the game. The first copy was run full screen on the focus display. The field of view of the player was reduced to 65°. A second copy of the game was run on the context display and set to be a “spectator”. This spectator was set to track the player from the first game and positioned so that they saw the same as the focus player. Finally the field of view of the second game was set to 135° and the window positioned so that the two images lined up. When the player in the focus was moved or turned, the spectator in the context would move to match, giving an almost seemless display (see figure 3.2). This game was easily playable although there was a small lag between the two displays, which did not line up perfectly, although they were close. It was interesting to observe that people watching someone play this game were able to see enemies in the context area much more easily than the player themselves. It appears the player was forced to sit too close to the screen to take advantage of the added field of view that they were presented with.

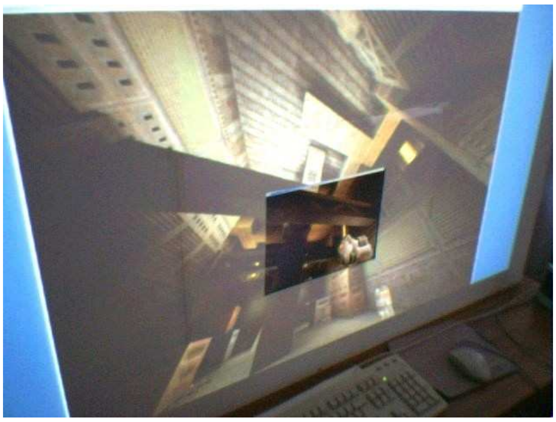


Figure 3.2: The Quake 2 game running on the focus+context screen

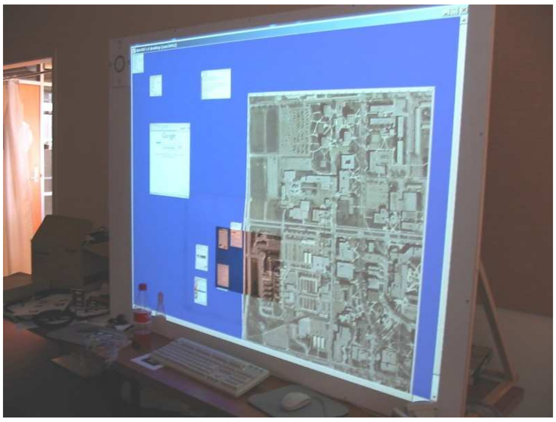


Figure 3.3: The VNC client running on the focus+context screen

The last application to be implemented was a full size desktop. This desktop effectively runs at 5000x3750 pixels. This is achieved by running a 5000x3750 desktop on a linux machine in a Virtual Network Computer (VNC) server. VNC allows a desktop to be connected to and interacted with remotely by receiving input events from the client across the network and sending images of the desktop back to the client. The focus+context computer then runs two copies of the VNC client. The first of these, the context, is configured to scale down the VNC desktop by five times. The second client is set to crop off most of the desktop except that that is visible in the focus. The first client is placed on the context isplay and the second on the focus display. This gives the effect of a single, very large desktop, as shown in figure 3.3 where a number of regular applications and a large image are open. Interacting with the context display allows this desktop to be used quite naturally. The main limitation is that it is impossible to pan around the desktop. Therefore, anything that must be read must first be dragged into the focus area. This is highly limiting when attempting to use a menu in the style of the Start menu, since that menu appears in the very bottom left of the desktop and the focus is just below the centre. Performance of this desktop is sufficient but degrades when moving very large windows as large images have to be transferred across the network in real-time. Alignment of the two screen images is also compromised as the image cannot be distorted to match as easily as it can using the custom viewer, so any mismatch in the aspect ratios of the projector and the LCD are propogated as alignment errors.

Chapter 4

Ecological Study

Implementation of focus+context screens for specialised applications is feasible with today's technology. To find if it is likely to be worthwhile, a number of small ecological studies were undertaken in which specialised prototype applications were developed and demonstrated to likely users. Researchers from university departments and local research companies were contacted, and interested parties were invited to demonstrate their applications. Focus+context screen versions of these applications were prototyped if they seemed viable and worthwhile. The relevant researchers then tried the prototype and offered any comments.

4.1 UML Diagrams

A UML class diagram of the Java Swing GUI classes was generated using TogetherSoft's Together IDE. This was then exported to a vector format which was converted to a format readable by the image viewer software. This diagram was too large to be loaded in its entireity so only 20% of the diagram was visible.

The diagram was shown to a computer science lecturer interested in software visualisation. The response was highly enthusiastic. The subject was immediately subjectively impressed by the additional real estate. UML was intended for use on whiteboards and for simple implementation on computer displays. Computer displays typically limit the scalability of UML but the subject found the focus+context screen significantly increased this limit. Users of UML diagram tools are often familiar with their diagrams and are aware of the location of important classes. Should a user need to view a class other than those in the current focus it is often necessary to zoom out to assist location of the target class. This zooming interaction is often distracting and reduces the continuity of using the interface. The focus+context screen diminished this requirement as many more classes were visible, although not all were legible.

This illegibility raised the problem that the subject would want to be able to read the information in the context area, were he using the focus+context screen frequently.

4.2 eXtreme Programming

One important phase of the eXtreme Programming (XP) process is the allocation of user stories to developers. This is ordinarily done using paper cards which are traded between developers until a suitable balance is found. A collaborative computerised version of this process has been developed in a separate project. Developers log on to this system and drag computerised user story cards into their pile.

We see the focus+context screen being useful in this environment. Each developer would have their own standard display and interact with the system as normal. The XP 'Coach' would use a focus+context screen and could see the entire workspace, and would be able to watch piles of cards form as developers selected and traded user stories. The coach could navigate this workspace and see the detail of a particular user. Figure 4.1 illustrates this idea.

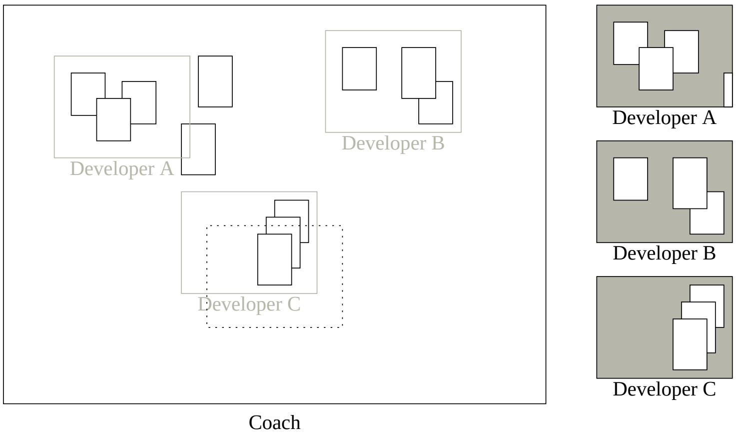


Figure 4.1: The proposed use of the focus+context screen in an XP environment

This idea was not implemented but the developer of the existing collaborative system was enthusiastic about the concept and noted that the screen real estate problems with the existing system would be relieved by the focus+context screen.

4.3 GIS Browsing

Geographic Information Systems (GIS) are databases containing information about geographic features. This information includes geometry such as coastlines, lakes and road centrelines, as well as feature information such as resource consents, groundwater quality etc. Display of maps from GIS systems is currently done largely on standard displays, even though the maps may be huge yet contain a fine level of detail. GIS is primarily used for the feature information rather than the maps that it can generate. Generation of maps is often done to demonstrate the capabilities of a GIS to customers or sponsors or to present information after it has been processed.

An employee of the National Institute of Weather and Atmospheric Research (NIWA) demonstrated the NIWA GIS system. This could display GIS maps and allowed the user to access feature information. The NIWA system supported basic browsing of the map but this was slow and difficult to use. The subject said a faster and easier map browser implemented on the focus+context screen would be useful for demonstration to customers.

The FcViewer software was adapted to load GIS data and display this instead of a loaded image. The sample data used consisted of approximately 90,000 rivers from the Canterbury region. When displayed in the context, this high level of detail became a meaningless blur. Therefore, feature information from the GIS data was used to filter out smaller rivers and only rivers above a certain size were displayed in the context. All rivers were displayed in the focus. Figure 4.2 shows a screenshot of the focus+context GIS viewer.

Figure 4.2: The focus+context GIS viewer renders only major rivers in the context

Display of this quantity of rivers presented a technical challenge to allow real-time interaction with the map. To achieve improved performance a quad-tree was used. This technique splits the map into quadrants, each of which is again split into quadrants. This happens several times down to a specified threshold. When the map is rendered the top level quadrants are checked for visibility. If a quadrant is entirely visible then its contents are rendered, if a quadrant is entirely offscreen then it is skipped, and if a quadrant is partially visible then this process is repeated on its sub-quadrants. This reduced the number of rivers that were being rendered each frame from about 180,000 (on the focus and the context) to approximately 75,000. This GIS prototype demonstrates that it is possible to greatly improve the performance of existing GIS viewers and that the focus+context screen may be effective for the display of GIS data. The developed prototype did not support multiple GIS layers or querying of features.

Chapter 5

Controlled Experiments

This chapter presents the results of two formal evaluation of the focus+context screen. The first investigates the effectiveness of the focus+context screen for map browsing tasks on an unfamiliar document and the types of task that the screen is better suited to. The second investigates the effect augmentation of the image with additional information has on performance for map browsing tasks.

5.1 Effectiveness for Browsing of Unfamiliar Documents

In order to put the time, money and effort that goes into the construction of a focus+context screen to good use, it is important to understand what the screen is and is not suited to. It seems likely that the focus+context screen should produce improved performance over a conventional display for many tasks relating to a large document such as a map. This evaluation compared the performance of 18 computer science students on each of three interfaces for a set of tasks on an unfamiliar map. Tasks were designed to be representative of actual tasks that a map would be used for.

Baudisch et al. (2002) compared the performance of two conventional interfaces to the focus+context screen for tasks performed on two large, unfamiliar documents. They found the focus+context screen to provide improved efficiency on both tasks, but also that there was a slightly larger increase in performance on one task. The difference between these two tasks was not formalised, so this evaluation seperated tasks into two well-defined categories in order to investigate which types of task the focus+context screen is best suited to.

5.1.1 Method

Design

Three interfaces were compared in this evaluation. The first was a small, panning only interface. This used a single 15" LCD monitor to display the map at a fixed zoom. The second was a small, zoom+pan interface. This used the same 15" LCD montor, but the display could be zoomed in and out by the user. Zooming was accomplished by holding the right mouse button and dragging the mouse along the NE/SW axis to control zoom. This approach was necessary as the supervisor had the keyboard and the mouse had no mouse wheel. The third interface was the focus+context screen.

Tasks were categorised into “following” tasks and “foraging” tasks. The former gave the subject precise directions to a target destination. For example, “Head north on state highway 19. Turn north-west onto state highway 16. Turn west onto Royal Road and find Royal Road school.”

The latter task type, “foraging”, asked the subject for a piece of information and required them to study an area of the map to determine the answer. For example, “Whenuapai Aerodrome is north-west of the harbour. What is the name of the school closest to the aerodrome?”

This evaluation was a 2-by-3 within-subjects ANOVA with factors of task type and interface type.

There were nine tasks for each task type, giving a total of eighteen tasks. Subjects were required to perform three tasks for each interface-task type combination. Tasks and interface type were randomised to minimise learning effects. The time the subject took to perform each task was recorded, times over a one minute threshold were counted as errors.

Subjects and Procedure

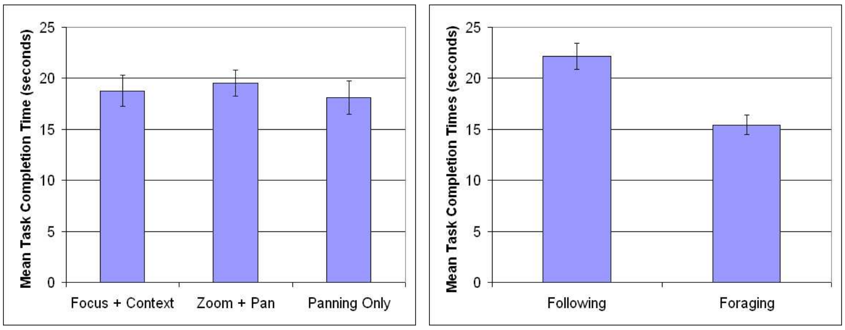
The subjects were 18 computer science students, who were each given $5 gift vouchers for their participation. Each had less than ten minutes prior experience with the focus+context screen.

Subjects were given a short training session on each of the interfaces. Before and after the timed tasks the subjects were asked a number of subjective questions.

For each task the subject was to perform they were handed a sheet of paper with the task written on it. They read this aloud and then were asked questions by the supervisor to ensure they understood the task. The subject was allowed to refer back to the task if they needed to while they were performing the task. Subjects were only shown the map while they were performing the task, at all other times the displays were blank.

Hypothesis

It was anticipated that the focus+context screen would produce the fastest task completion times, followed by the zoom+pan interface and finally the panning only. In terms of the tasks, foraging tasks seemed simpler and were expected to be easier to complete. Also, we expected that the focus+context screen would not be much faster, if at all, for following tasks, but would be significantly faster for foraging tasks as it eliminates the need for zooming interactions. We also believe that subjects will subjectively prefer the focus+context screen over the zoom+pan interface and the zoom+pan interface over the panning only interface, because of the large display presented by the focus+context screen.



(a) Mean task completion times for each interface type (b) Mean task completion times for each task type

Figure 5.1: Mean task completion times across factors

5.1.2 Results

Results were log transformed and analysed using a 2-by-3 ANOVA. There was no significant main effect for interface type ($F_{2,34} = 1.021, p = 0.371$) with mean task completion times for panning only, zoom+pan and focus+context were 18.1036 (9.929), 19.545 (7.5419), and 18.7754 (9.1912) seconds respectively as shown in figure 5.1(a). However, there was a signficant difference between the times taken for the two task types with the mean times for foraging and following tasks being 15.4365 (6.8940) and 22.1795 (9.4059) seconds respectively, with standard deviations in parentheses. $F_{1,17} = 44.580, p < 0.001$ (see figure 5.1(b). More interestingly, there was a significant interaction between interface type and task type with $F_{2,34} = 5.625, p < 0.01$. The mean times for each combination are displayed in table 5.1. The interaction becomes clear in figure 5.2. Error bars in all figures are $\pm$ one standard error.

The interfaces produced an average of 0.8333, 0.3889 and 0.5 errors out of the six tasks for the panning only, zoom+pan and focus+context interfaces respectively.

Subjects were asked to rate each interface on a seven point lickert scale, based on how efficient they perceived the interface to be. After the evaluation the focus+context screen, zoom+pan interface and panning only interface scored means 4.5, 5.0 and 3.28 respectively. A Friedman Chi-Square test showed a significant difference between the means ($\chi^2 = 19.44, df = 2, p < 0.001$).

Subjects were also asked to rank the interfaces by subjective preference. Subjects

	Foraging	Following
Panning Only	15.2422 (7.2487)	20.9650 (11.5364)
Zoom + Pan	18.1041 (6.7403)	20.9858 (8.2024)
Focus + Context	12.9633 (6.0122)	24.5876 (8.1389)

Table 5.1: Mean task completion times for each interface and task type

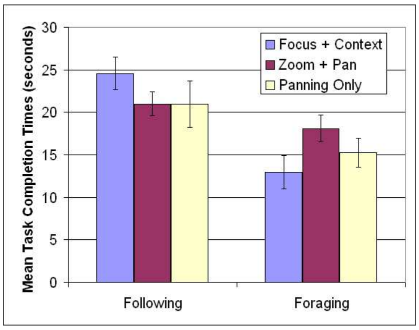


Figure 5.2: Mean task completion times for each interface across task types

preferences for interfaces differed significantly ($\chi^2 = 14.33, df = 2, p < 0.001$) with 13 prefering the focus+context screen, five prefering the zoom+pan interface and none prefering the panning only interface.

5.1.3 Discussion

The most interesting of these results is the interaction between interface type and task type. This effect demonstrates that there are some types of task that the focus+context screen is better suited to and other types of task that it is badly suited for. It is also interesting to see that the zoom+pan interface performed worse than the pan only interface for foraging tasks. We suspect the zoom+pan is more efficient once users get to the zoom level they want, but that this implementation makes zooming difficult.

The lack of a significant main effect surprised us. However, from our observations it appears that the extra information presented in the context area of the focus+context screen is distracting for "following" tasks. Subjects would seldom look away from the focus area during "following" tasks and one commented that "The focus+context was a little distracting".

The lack of clarity in the context region seemed to also impair subjects. Labels in the context area were not legible, rendering most of the information it presented useless. We expect augmenting the display in the context with larger, readable information would help the user.

Some subjects had difficulty with the zoom+pan interface, to the point where a number would very seldom use the zoom feature of this interface. Subjects were required to hold the right mouse button and drag to zoom. Many found this difficult or counterintuitive. We expect that if the zooming had been implemented to use the scroll wheel available on many mice then zooming may have been more popular.

Incredibly, although most subjects cast a small shadow on the context area just below the focus, not one of the subjects noticed this shadow. When asked if they were aware that they were casting a shadow all responded that they had not even noticed.

Many subjects found the panning only interface to be restrictive. One subject in particular summed it up the first time they used it with "This one's horrible".

When asked to compare the interfaces they used to a fold out paper map, all subjects said they would still prefer a paper map. It was pointed out that the paper map has an index which can be used to quickly locate target streets. This leads us to reassess the tasks presented to subjects, as it seems likely that the "foraging" tasks are fairly realistic, but that the "following" tasks do not accurately reflect how a map is used in reality. These tasks were intended to reflect the kind of task one my attempt when given directions to a friends house for instance. However, it seems more likely that map users would be told of a recognisable landmark near the target and then a simple set of directions from there.

These results do not appear at first to correlate with Baudisch's result. Baudisch found that the focus+context screen produced large, significant performance improvements. On close inspection of Baudisch's tasks it becomes apparent that both tasks were similar to "foraging" tasks. Both tasks required little reading of details but a lot of following of large lines.

The counterintuitive zooming method was a possible confounding factor. An improved zooming method would yield improved performance and provide a truer account of the impact of zooming interactions as users would less easily become confused by the zooming method. Comprehension of tasks was tested by the supervisor but it is possible and likely that subjects understood some tasks better than others. The supervisor was aware of the hypothesis being tested and may subconsciously have ensured subjects held a better understanding of focus+context tasks. Formalisation of the comprehension test would help to reduce these problems.

Tasks were randomised across conditions. It would have been better to counter balance tasks instead as it is possible tasks particularly suited to one interface were always or never tested on that interface.

5.1.4 Conclusions

We have presented a comparison of focus+context screens with zoom+pan interface and panning only interfaces for two different categories of task. We found no significant difference for the main effect of interface but a significant interaction between task and interface. The focus+context screen performs well on "foraging" style tasks that work with the document primarily at a larger scale. Detail in the context is useless if it is not legible. Users subjectively prefer using the focus+context screen over the zoom+pan interface.

5.2 Augmentation of Familiar Documents

For many real-world tasks, users will be familiar with the document they work with. For the second evaluation of the focus+context screen we presented subjects with a map of Christchurch and selected only subjects who were familiar with Christchurch.

As subjects were familiar with the map, tasks could be phrased to more accurately reflect how one would use a map and a large document in general, by making greater use of well known landmarks, rather than giving detailed directions.

One major outcome from the first evaluation was that users found the context area mostly useless because unfamiliarity with the map, combined with illegibility of all but major features meant users could not recognise any useful landmarks in the context. In an effort to combat this we propose augmenting the context map with additional labels on major landmarks.

5.2.1 Method

Design

This evaluation was a 2-by-2-by-2 within-subjects ANOVA, with factors of interface type, task part and augmentation.

Two interfaces were compared in this evaluation. The focus+context screen and zoom+pan interface were both tested. Each of these worked exactly as they did in the first evaluation.

Each task given to the user consisted of two parts. The first part was always an instruction for the subject to find a well known landmark such as a school. The second part required the subject to find another landmark and follow a short set of directions from there. An example of the first part is "Go to Riccarton High School". The second part of this task is "Find Rotherham Street. It is the last on the right before the Clarence/Riccarton Road intersection when heading towards town.". This was intended to reflect how many documents would be used.

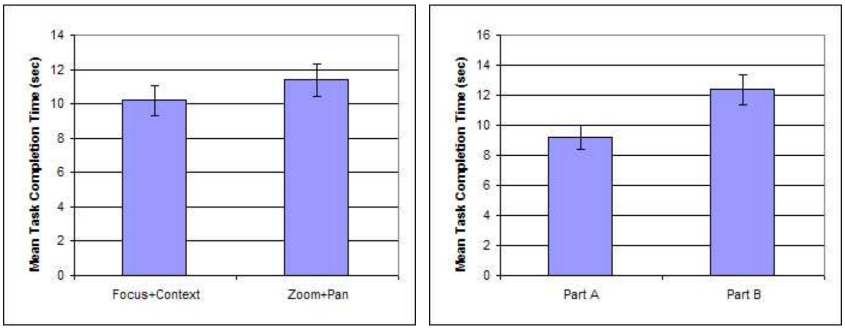
The third factor was augmentation. The map used has labels on each suburb, but these labels are not legible in the context or when zoomed out in the zoom+pan interface. The augmented condition puts larger versions of these labels onto the map. When the zoom+pan interface zooms out, these labels remain a constant size. The labels only appear on the context for the focus+context interface.

Subjects and Procedure

Subjects were 16 regular computer users. All subjects were screened to ensure a good knowledge of Christchurch. Subjects were required to perform four training tasks that introduced them to each interface, with and without augmentation. This training acted as the screening process; the experiment supervisor made a subjective judgement as to whether the subject was sufficiently familiar with the map to continue.

After training, subjects were asked to fill in a questionnaire aimed at getting subjective feedback about the interfaces and how efficient the subjects felt they were with each interface.

Subjects were handed each task as they were required. Subjects were asked if they recognised where they were to go. If they were unsure the supervisor would clarify. The supervisor only clarified the geographic location of targets, not the location on the map. For example, "Riccarton High School is between Main South Road and Blenheim Road". Once the subject was comfortable with the task they were shown the map and timed to find the target. Subjects were stopped between each part of the task.



(a) Mean task completion times for each interface type

(b) Mean task completion times for each part of the tasks

Figure 5.3: Mean task times for the two significant main effects

Hypothesis

From our observations in the first evaluation and our own experience using the focus+context screen, we expect the focus+context screen to increase the performance of subjects compared to the zoom+pan interface. It seems the context will be more useful when the subject can recognise features without needing to be able to read labels.

We predict that augmentation of the displays will produce an improvement in performance as subjects will be able to find the general area of the target more easily.

5.2.2 Results

The results showed a significant difference between the mean task completion times for each of the two interfaces, with the mean times for the focus+context screen and zoom+pan interface being 10.2021 (7.1203) and 11.3818 (7.6672) seconds respectively ($F_{1,15} = 5.418, p < 0.05$). Standard deviations are shown in parentheses. There was also a significant difference between the mean times for the two parts of each task. The means for parts A and B were 9.2116 (6.4159) and 12.3724 (7.9972) seconds respectively ($F_{1,15} = 6.441, p < 0.05$). Figure 5.3 shows each of these main effects. There was no significant difference between the mean times for with and without augmentation ($F_{1,15} = 0.115, p = 0.74$). These conditions produced means of 10.6436 (8.0534) and 10.9403 (6.7298) seconds respectively.

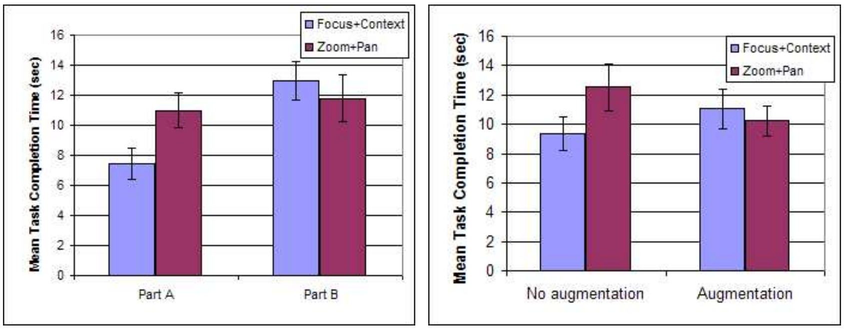
Breaking down the means shows some interesting results. There is a significant interaction between task part and interface type ($F_{1,15} = 4.668, p < 0.05$). The means for this are shown in table 5.2, standard deviations are shown in parentheses. There is also a significant interaction between augmentation and interface type ($F_{1,15} = 5.536, p < 0.05$), see table 5.3 for the means and standard deviations. Bar charts for these two significant interactions are shown in figure 5.4. There was no significant interaction between task part and augmentation. There was no significant interaction between all three factors.

	Part A	Part B
Focus+Context	7.4555 (5.8265)	12.9488 (7.3150)
Zoom+Pan	10.9677 (6.5833)	11.7959 (8.7049)

Table 5.2: Mean task completion times for the interaction between interface type and task part, in seconds

	No augmentation	Augmentation
Focus+Context	9.3560 (6.6115)	11.0483 (7.6055)
Zoom+Pan	12.5246 (9.1057)	10.2390 (5.8188)

Table 5.3: Mean task completion times for the interaction between augmentation and interface type, in seconds



(a) Interaction between interface type and task part

(b) Interaction between interface type and augmentation

Figure 5.4: Significant two factor interactions

5.2.3 Discussion

The significant difference between means for interface type is interesting. For these tasks the focus+context screen performs better. The previous evaluation did not find a significant difference for this factor. There are a number of possible explanations of this. The tasks in this evaluation tended to require less precise reading of the map than Following tasks in the previous evaluation. Tasks in this evaluation were more equitable with Foraging tasks in the first evaluation, and the first evaluation showed that the focus+context screen excelled for this type of task. Secondly, subjects were familiar with the document they were browsing, whereas in the first evaluation they were not. We believe a controlled evaluation comparing efficiency on familiar and unfamiliar documents would show the focus+context screen to be more useful for performing tasks on familiar documents.

The improved performance on Part A (Landmark oriented, low-precision) tasks by

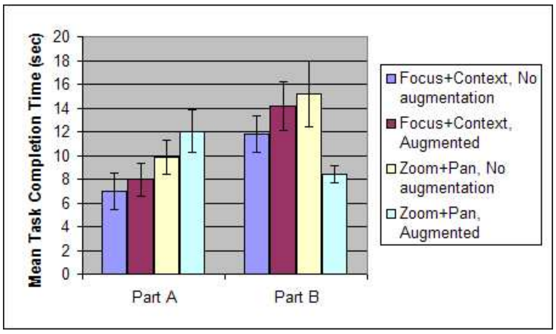


Figure 5.5: Interaction between all three factors

the focus+context screen (see figure 5.4(a)) is evidence, in addition to that provided by the first evaluation, that the focus+context screen is most beneficial for such tasks.

The lack of a significant difference between the means for augmented and regular interfaces was unexpected. The interaction between augmentation and interface type was anticipated, but the results did not go the way we expected. Figure 5.4(b) shows that the focus+context screen performed better without augmentation, whereas the zoom+pan interface benefitted from augmentation overall. Looking at 5.5 gives a clearer picture still. We can see that for all cases except zoom+pan for part B, augmentation has a detrimental effect on performance. For the zoom+pan interface, it appears performing Part B tasks benefits from augmentation.

We propose that the decreased performance on the augmented focus+context screen may be because the labels were too easy to read. We believe the boldness of the labels meant it was difficult to view the map without reading the labels, slowing the user down. Perhaps increasing the transparency of the labels would lessen this effect.

Subjects anticipated being more efficient on the focus+context screen. On a seven point lickert scale (1 = Very Inefficient, 7 = Very Efficient) the focus+context screen scored 5.375 for anticipated efficiency before the evaluation, and 5.5625 for perceived efficiency after the evaluation. This is a slight improvement over the zoom+pan interface, which scored 4.9375 and 4.500 for anticipated and perveived efficiency respectively. Subjects also felt the augmentation was some help. On a seven point lickert scale (1 = Very distracting, 7 = Very helpful), subjects rated anticipated helpfulness of augmentation as 5.75, and perceived helpfulness as 5.25. Overall this subjective feedback does not indicate a clear preference from subjects.

Subjects reacted in varied ways to the zoom+pan interface. Some found it immediately natural to use, while others struggled to cope with the zoom implementation. Switching between left and right buttons for different tasks presented problems to some users. It was observed that different users have different preferences for the direction that zoom should be. A number of users wanted to drag the mouse up and left to zoom in, rather than up and right. Other users were happy with the configuration present. If a zooming interface is implemented with the right mouse button, we would recommend the interface allow the user to select the direction for zoom.

A number of subjects mentioned that they often only looked at the focus on the focus+context screen, rather than the whole display. One mentioned that with experience, it became easier to view the display as a whole. Feedback for the focus+context screen was mixed, and less enthusiastic than the previous evaluation. Subjects felt they found the focus+context screen more useful when they were familiar with the target of the task. Subjects also found the focus+context screen became more useful as they practised with it. We believe that an expert user of the focus+context screen with a document they are familiar with will benefit from the use of the focus+context screen over other interfaces.

Unlike a paper map, none of these interfaces allowed the user to turn the map around. Some subjects found this challenging as they were used to turning the map when they read it.

As with the first evaluation, the zooming interaction is likely to have been a confounding factor. Choosing targets that many subjects are familiar is difficult and may have confounded the results of this evaluation, but we feel the subjects never became totally lost.

5.2.4 Conclusions

We presented a comparison of focus+context screens and a zoom+pan interface with and without augmentation. Focus+context screens perform better than zoom+pan interfaces for large scale tasks on documents with which the user is familiar. Augmentation of the display with additional labels has no significant effect on efficiency for documents with which the user is familiar. Users slightly prefer the focus+context screen for these tasks but reactions are mixed.

Chapter 6

Discussion and Further Work

When people first see the focus+context screen, they are almost always subjectively impressed. After running two controlled evaluations and finding subjects frequently prefer it over a standard interface it still appears to be subjectively satisfying to use, but perhaps less so than originally thought. One explanation for the highly enthusiastic initial reception given by many users is simply that the notion of a six foot screen appeals when compared to a regular 19”.

The results shown in the two controlled evaluations demonstrate that the focus+context screen is more useful when used correctly, but that it is not a panacea for large document browsing.

We have used the focus+context screen extensively. It has been used primarily for writing and testing the FcViewer application but the large VNC desktop described in Chapter 3 was used also. The screen was sometimes used after hours for recreation playing the Quake 2 game. All of these tasks did not seem to be hindered by the focus+context screen.

A limitation of our particular screen was the small LCD monitor used. The 15” monitor supports resolutions up to 1024x768. While this is plenty for the FcViewer map software or the Quake 2 game, it restricts the use of the VNC desktop. The desktop effectively removes the need for virtual workspaces and allows unused windows to be moved away from the focus. The focus area is limited on our screen to 1024x768, which is not very large. A larger monitor that supports 1280x1024 or higher would be useful. Unfortunately, large LCD monitors are often expensive.

6.1 Further Work

6.1.1 Large Desktop

The use of the VNC desktop in a real-world situation would be an interesting area of further study. Our own experience with it show it to be a more accurate analogy of a real desktop. The physical size makes the scale between the document and the desktop closer to that of a real desk. Moreover, the most important windows are in the focus and currently being used, and less frequently windows are moved further from the focus. This second point shows that it may allow users to arrange their computerised desktop more like they do their physical desktop. Malone (1983) studied the organisation of desks and it appears that the VNC desktop on the focus+context screen may be a useful way to interact with the computer. Further study of the VNC desktop should be pursued.

6.1.2 Document Familiarity

The two evaluations presented here look at documents of different familiarity to the user; the first uses unfamiliar documents, while the second uses familiar documents. The results of these two evaluations differ in the level of statistical significance of the result. This leaves room for further work to compare the performance of the focus+context screen on familiar and unfamiliar documents.

6.1.3 Hybrid Focus+Context Interfaces

Investigation of hybrid focus+context techniques could also be investigated. For instance, an SDAZ /Focus+Context or fisheye/Focus+Context interface. These techniques may be more effective with the extra screen space offered by the focus+context screen.

6.1.4 User's Seat Position

As demonstrated by the Quake 2 game, sitting directly in front of the screen, the same distance as from a regular monitor, may not always provide the best view of the data. Quake 2 showed that sitting further back from the screen allowed the seam between the two displays to be more easily overcome. The optimal distance to sit from a focus+context screen is not known and further study must be done into this and how the seam between resolutions can be overcome.

6.1.5 Variable Mouse Speed

Some subjects in the evaluations commented that the mouse speed was good in the context but too fast in the focus. The mouse rate was actually kept constant but subjects perceived it as faster because of the high resolution in the focus. A variable mouse speed may be worth investigating, although the tasks presented here are unlikely to benefit. We believe for tasks that require manipulation of documents a variable mouse speed may be beneficial.

6.1.6 Input Tasks

Neither Baudisch, nor the work presented here deal extensively with tasks that manipulate the document. Both Quake 2 and the large desktop take a lot of input from the user, but neither of these were investigated in depth. The only input for the map browsing tasks was panning operations. An architect using the focus+context screen must draw lines and curves, as well as viewing the document. These types of input task provide further topics for investigation.

6.1.7 Alternative Construction

Currently the focus area is fixed. We propose an alternative construction of the screen that would allow the focus to be moved and even to allow multiple foci. The context area would still be projected, but the image would be rear projected and the board onto which it is projected would be horizontal, like a table. This would allow anything to be put on top of the image or users to lean over the image without casting a shadow. The focus area would be implemented on a Tablet PC. The Tablet PC would be tracked using a magnetic tracker, inertial compass or some sort of optical tracker. The Tablet PC would determine its position on the context image and display the appropriate area in high resolution on its own display.

This is similar in concept to the spatially aware palm-top devices described in Fitzmaurice. The advantage of this is that each Tablet PC can implement its own rendering, allowing multiple foci. This reduces the need for panning the display using the mouse but allows the user to focus on their area of interest by physical interaction. Pontential applications for this include circuit and architecture diagrams, maps and collaborative applications such as the XP system discussed in section 4.2.

Chapter 7

Conclusions

We have presented a description of the construction of our focus+context screen and the implementation of software for that screen. A number of applications were implemented including an image browser, a desktop and a 3D game.

We have studied how the focus+context screen might be used for specific, real-world applications and implemented a GIS browser for this purpose.

Finally we have presented the results of two formal evaluations of the screen. These showed that for tasks requiring a knowledge of a large area of the graph and some fine detail the focus+context screen is beneficial. It also appears that the more familiar with the document the user is the greater the benefit they will gain from the focus+context screen. Augmentation

The focus+context screen holds promise as a visualisation technology for specialised applications. It is a young technology that has many aspects still to be investigated to find its full potential.

Bibliography

Bartram, L., Ho, A., Dill, J. & Henigman, F. (1995), The continuous zoom: A constrained fisheye technique for viewing and navigating large information spaces, *in* 'Proceedings of UIST'95 Symposium on User Interface Software and Technology', pp. 207–215.

Baudisch, P., Good, N., Bellotti, V. & Schraedley, P. (2002), Keeping things in context: A comparative evaluation of focus plus context screens, overviews, and zooming, *in* 'Proceedings of CHI'02 Conference on Human Factors in Computing Systems'.

Baudisch, P., Good, N. & Stewart, P. (2001), Focus plus context screens: Combining display technology with visualization techniques, *in* 'Proceedings of UIST'01 Symposium on User Interface Software and Technology'.

Bederson, B. B. (2000), Fisheye menus, *in* 'Proceedings of UIST'00 Symposium on User Interface Software and Technology', pp. 217–225.

Czerwinski, M., Tan, D. & Robertson, G. (2002), Women take a wider view, *in* 'Proceedings of CHI'02 Conference on Human Factors in Computing Systems'.

Fitzmaurice, G. W. (1993), 'Situated information spaces and spatially aware palmtop computers', **7**(36), 38–49.

Furnas, G. W. (1986), Generalized fisheye views, *in* 'Proceedings of CHI'86 Conference on Human Factors in Computing Systems', pp. 16–23.

Grossman, T., Balakrishnan, R., Kurtenbach, G., Fitzmaurice, G., Khan, A. & Buxton, B. (2001), Interaction techniques for 3d modeling on large displays, *in* 'Proceedings of the 2001 symposium on Interactive 3D graphics', ACM Press, pp. 17–23.

Grudin, J. (2001), Partitioning digital worlds: Focal and peripheral awareness in multiple monitor use, *in* 'Proceedings of CHI'01 Conference on Human Factors in Computing Systems'.

Hornbæk, K., Bederson, B. B. & Plaisant, C. (2002), *ACM Transactions on Computer-Human Interaction (TOCHI)* **9**(4), 362–389.

Igarashi, T. & Hinckley, K. (2000), Speed-dependent automatic zooming for browsing large documents, *in* ‘Proceedings of the 13th annual ACM symposium on User interface software and technology’, pp. 139–148.

Lamping, J. & Rao, R. (1994), Laying out and visualizing large trees using a hyperbolic space, *in* ‘Proceedings of UIST’94 Symposium on User Interface Software and Technology’, pp. 13–14.

Lamping, J., Rao, R. & Pirolli, P. (1995), A focus+context technique based on hyperbolic geometry for visualizing large hierarchies, *in* ‘Proceedings of CHI’95 Conference on Human Factors in Computing Systems’, pp. 401–408.

Malone, T. W. (1983), ‘How do people organize their desks?: Implications for the design of office information systems’, *ACM Transactions on Information Systems (TOIS)* **1**(1), 99–112.

Robertson, G. G. & Mackinlay, J. D. (1993), The document lens, *in* ‘Proceedings of UIST’93 Symposium on User Interface Software and Technology’, pp. 101–108.

Tan, D., Gergle, D., Scupelli, P. & Pausch, R. (2003), With similar visual angles, larger displays improve spatial performance, *in* ‘Proceedings of CHI’03 Conference on Human Factors in Computing Systems’.

Tani, M., Horita, M., Yamaashi, K., Tanikoshi, K. & Futakawa, M. (1994), Courtyard: integrating shared overview on a large screen and per-user detail on individual screens, *in* ‘Proceedings of the SIGCHI conference on Human factors in computing systems’, ACM Press, pp. 44–50.