

SCREEN MAGNIFIERS: ARCHITECTURE, EVOLUTION AND EVALUATION

**P. BLENKHORN, D. G. EVANS, A. KING,
S. H. KURNIAWAN & A. SUTCLIFFE**

**Dept. of Computation, UMIST
P.O. Box 88, Manchester M60 1QD, United Kingdom**

1 INTRODUCTION

Designing multimodal interfaces for disabled users presents difficult challenges in amplifying the bandwidth of communication. Perceptual disability inevitably restricts communication, so the designer's motivation is to empower users with enhanced displays and substituting modalities, for instance, audio screen readers for blind users. This paper reports the evolution of screen magnifier technology that provides users with a magnified area for more detailed displays; however, this presents a dilemma. Magnified displays restrict the viewable area of a document, so users lose track of where they are. The design problem is how to give the user context for location awareness with a magnified display within the available screen real estate. Perceptual modalities are difficult to separate from user interface controls. Conventional user interfaces assume we can see cursors and achieve fine grain eye-hand coordination, but this is not possible for users with poor vision. The second design problem we investigate is how to coordinate perceptual modalities with control, in our case of the zone of magnification.

Visually impaired people (VIP) have problems in interacting with computer systems that present the majority of output in a visual form. VIP who have no functional sight (blind people) use screen readers, which are assistive programs that present the visual output of a computer system as speech or which drive refreshable Braille displays (for further information on the architecture of screen readers see [1]). The most widespread example of such software is Narrator¹, a limited functionality screen reader that is shipped with Microsoft Windows 2000 and XP. VIP who have some functional vision (partially sighted people) do not require such a radical change to the presentation of the visual output; enlargement of the visual display is generally sufficient and this is most commonly achieved by using an assistive program called a screen magnifier. The most widespread example of a screen magnifier is Microsoft Magnifier² which is shipped with Windows 2000 and XP, but it is atypical of modern screen magnifiers in that it does not provide enlargement over the full screen (see Section 4).

In contrast to a screen reader, a screen magnifier is a perceptual interface because a screen magnifier takes the graphical output of the computer system and presents an enlarged portion of the output on the visual display. However, it should be noted that some screen magnifiers are multimodal interfaces, i.e., they produce visual as well as audio outputs. The basic operational characteristics of a magnifier are explained in Section 2. The mechanism through which the screen magnifier captures and redisplay the computer's graphical output is critical and in Section 3 of this paper we examine a number of historical and contemporary approaches. There are a number of usability issues faced by the designer of a screen magnifier. These concern the quality and style of the image presented to the user and the operational modes provided by the magnifier; these are issues examined in Section 4. Even though the features presented in Section 4 are found in most screen magnifiers, they have rarely been tested with their prospective users. This can pose problems for users because, depending on the operational modes, there are large variations in the demands on users' cognitive processing as well as the over the control users have over the content (magnified) information and contextual (unmagnified) information areas. Section 5 goes some small way towards addressing this issue by describing the evaluation of some of the common features of screen magnifiers with visually impaired users. In addition, Section 5 also suggests the use of an absolute positioning device to help users orientating themselves, a common problem that visually impaired people experience when using screen magnifiers, especially at a high magnification level.

¹ Microsoft Narrator can be run by selecting Start, Programs, Accessories, Accessibility, Narrator

² Microsoft Magnifier can be run by selecting Start, Programs, Accessories, Accessibility, Magnifier

2 A BRIEF INTRODUCTION TO THE OPERATION OF SCREEN MAGNIFIERS

This section gives a brief overview of the operation of screen magnifiers. The purpose of this section is to give the reader sufficient background to understand the points raised in Section 3 and to introduce a common set of terms.

In this paper we will refer to the image generated by the computer system prior to enlargement as the ‘original image’ and the enlarged portion presented to the user as the ‘magnified image’; we will use this term irrespective of whether the magnified image occupies the full area of the screen or not. There are some considerations to be made with regard to the cursor or mouse pointer and we will use the term ‘cursor’ in the sense of the current text entry position in a character-based operating system (such as DOS) and the ‘mouse pointer’ or caret position of a GUI (such as Windows).

The basic purpose of a screen magnifier is to present an enlarged portion of the original image on the computer’s monitor. As noted above, the magnified image may cover the full screen or just a part. For the moment we will assume that it takes over the full screen and postpone the discussion of other operating modes until Section 4. The magnified image should be that portion of the original image that is of interest to the user. This is often, but not always, the area that is around the cursor. The cursor itself should be displayed at the correct position in the magnified image and suitably enlarged. As the user moves the cursor (typically by using the mouse or the keyboard), the screen magnifier must track the movement of the cursor so that it is presented roughly in the centre of the screen; of course, special considerations are required when the cursor moves toward the edge of the original image. As the cursor is tracked, the magnified image that is presented needs to scroll smoothly; flickering can be very disturbing. Occasionally the area of interest is not the operating system’s cursor. For instance, when a drop-down menu is selected by a keyboard shortcut in a word processing application, the magnified image should jump to the highlighted menu item. There are other occasions, for example, pop-up warnings, information from application assistants and changes in status, when the ‘focus’ (the centre of attention of the magnified image) should be changed.

3 SCREEN MAGNIFIERS

In this section we consider how the visual information presented by the computer system is captured, enlarged and presented to the user.

3.1 THE ARCHITECTURE OF SCREEN MAGNIFIERS

As noted earlier, a screen magnifier must capture the visual information presented by the computer system and display an enlarged portion of it to the user. In addition, it must display a suitably enlarged representation of the cursor or mouse pointer at the appropriate position on the screen. In order to examine the approaches to screen magnification in a systematic way, the approaches adopted can generally be classified into four groups:

1. Custom hardware solutions
2. Systems that exploit the underlying hardware
3. Device driver-based magnification
4. Operating system-based magnification

Approaches 1 and 2 are best considered in the context of a simple operating system (for example DOS). A very simplified view of the system’s architecture is presented in Fig. 1.

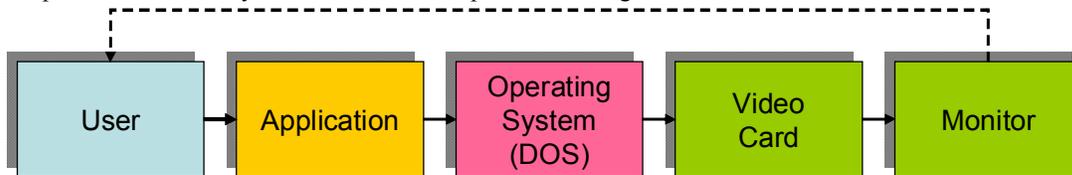


Fig. 1: Simplified architecture of the graphics display part of a DOS-based system

Approaches 3 and 4 are best considered in the context of a modern multiple-windowed GUI. We confine our discussion to the Windows operating system. A simplified view of the system’s architecture is given in Fig. 2.

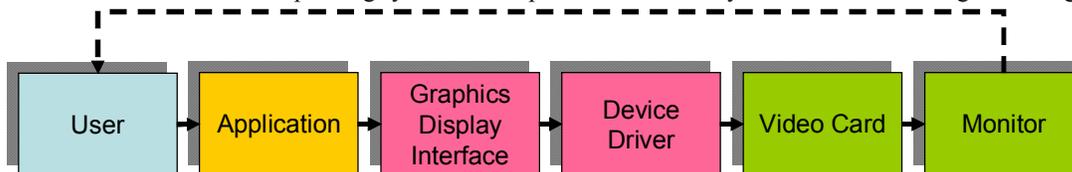


Fig. 2: Simplified architecture of the graphic display part of a Windows-based system

3.2 DOS-BASED SYSTEMS

On the whole DOS-based systems required the enlargement of the screen on a character-by-character basis. The earliest approaches carried out magnification in custom hardware. Two solutions were adopted, either the ‘standard’ video card was replaced by a custom magnifying card or a magnification system was introduced between the output of the video card and the monitor.

Later approaches used software that exploited features of the underlying hardware. Generally, the operating system built the image to be displayed at one screen resolution but the magnifier set up a different resolution on the video card. The earliest approaches (e.g. [2]) presented the magnified image as a window at the bottom of the screen, with the remainder of the screen taken over with the original image. Later systems provided full-screen magnification generally by accessing the original image in a page of video memory, processing this image to create a magnified image in another page of video memory and then displaying the magnified image from the second page.

3.3 GUI-BASED OPERATING SYSTEMS (WINDOWS)

With the advent of GUI-based operating systems, such as Microsoft Windows³, two major changes were required to the design of magnifiers.

1. There was a requirement to magnify both graphics and text.
2. The standard video modes used changed significantly from predominantly text-based ones to those that were primarily graphical.

The first point requires that the magnified image be constructed on a pixel by pixel rather than a character by character basis. The second point is best considered with reference to Fig. 2. A Windows application writes to the screen by making calls on the Graphics Display Interface (GDI). This provides a set of functions that allow an application to write, for example, text and bitmaps to specified areas of a window. In order to render the set of windows that form the screen image the GDI calls a device driver (DD) for the video card. The DD is specific to the video card and is generally supplied by the manufacturer of the card⁴. The GDI can interrogate the DD to determine the set of features provided by the video card.

3.3.1 Device Driver-based Magnification

The video card’s DD is replaced by a custom DD that is responsible for carrying out magnification. It controls the operational characteristics of the video card by manipulating its registers (see Fig. 3). Sensory Software’s Magnus (a Windows 3.1 magnifier) used this approach, replacing the standard VGA DD with a custom VGA DD. It changed the video registers to point the memory-mapped display to a ‘vacant’ area of memory-mapped screen memory on the video card, i.e. the displayed image was routed from this ‘vacant’ area. The calls made by the GDI constructed the image in ‘normal’ display memory. The magnified image was constructed by copying the ‘appropriate’ portion of the ‘normal’ display memory to the ‘vacant’ area. Magnification in the vertical direction was achieved by manipulating the VGA video registers so that lines were replicated. Magnification in the horizontal direction was achieved by copying pixels.

This approach was successful and used commercially, but it is subject to severe limitations. Because it uses a custom VGA driver, it works only in VGA mode, a very significant limitation these days. Moreover, the more advanced features of graphics cards cannot be used.

3.3.2 Operating-system based magnification

We believe that the approach that is most widely used today is to intercept (to ‘filter’ or ‘hook’) the calls made by Windows on the GDI or DD (see Fig. 2) and to construct the original ‘full screen’ image by using these calls. The portion of the original image corresponding to the desired magnified image can then be selected and processed. A number of strategies can then be used to display the image including using the GDI/DD path [3] or Microsoft’s DirectX. The latter can be used in a variety of ways; one interesting technique, which we describe later, is to present the magnified image as an overlay ‘on top’ of the original image [4].

One variation of this approach, referred to as ‘hooking high’, is to intercept the calls made to the GDI and to use these calls to construct the original image – from which the magnified image can be derived (see Fig. 4). The magnifier can then construct the magnified image to be presented to the user – probably by making calls on the DD. Alternatively the calls made on the DD can be intercepted – referred to as ‘hooking low’. Again the

³ We will confine our discussions to Microsoft Windows.

⁴ Generic video card device drivers exist, but they may not take advantage of the full set of features provided by the card.

magnifier constructs the original image and creates a suitably magnified portion that is presented to the user (see Fig. 5). Care must be taken if the DD is used to display the magnified image otherwise the calls made by the magnifier to the DD will also be intercepted by the magnifier [5].

There are a number of techniques that can be used to hook at either the GDI or DD level, is supported by Microsoft is the DDI (Device Driver Interception) hook that is provided by Microsoft Active Accessibility (MSAA)⁵ in Windows 95/98/Me which allows calls made on the DD by the GDI to be intercepted.

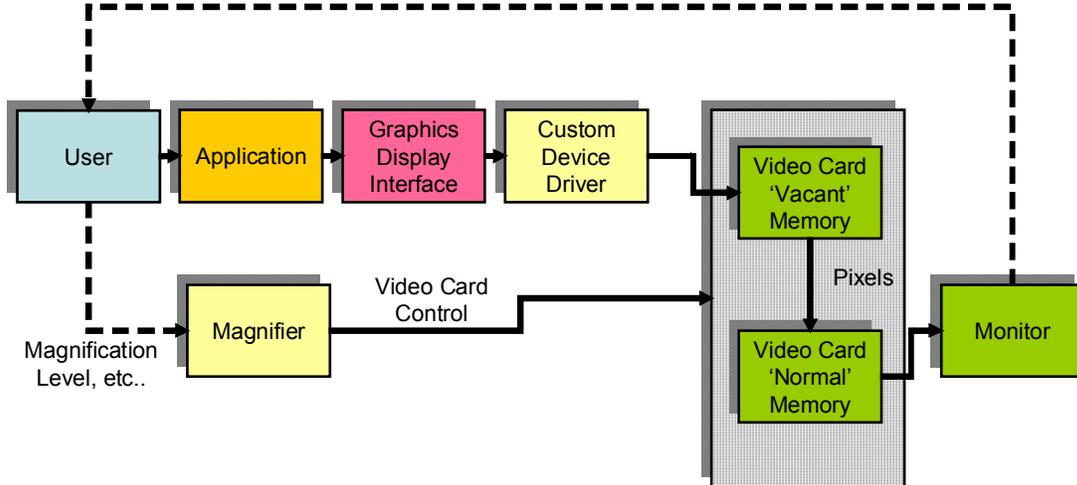


Fig. 3: Schematic of the Magnus Screen Magnifier

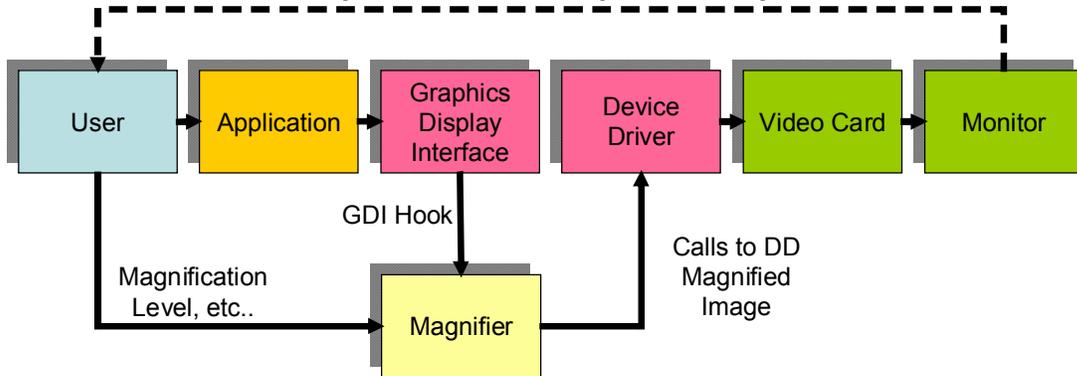


Fig. 4: Schematic of a Screen Magnifier using GDI hooking. The figure assumes that the magnified image is created by using the DD – this may not be the case.

The approaches described here can work quite successfully and we believe they are the basis of a number of commercial systems. However, there are a number of potential problems. Firstly, different hooking strategies may have to be developed for different versions of Windows – for example DDI hooking is not supported by Windows NT-based operating systems, most importantly Windows 2000 and XP. Secondly, because the magnifier has to interpret calls to construct the original image it incurs a significant processing overhead. Finally, some applications bypass the GDI-DD chain when writing to the screen, when using, for example, Microsoft DirectX. When GDI or DD hooking is used, the output of such applications cannot be magnified.

One approach that can be used to overcome these problems is to allow the original image to be constructed by the computer system using the standard GDI-DD chain and other methods such as Direct X. The problem with this approach is that the original image now appears on the screen unless it has been redirected. With such a wide variety of video cards it is no longer possible to redirect the original image in a similar manner that carried out by the Magnus system, described earlier. However, DirectX provides overlays that allow graphics to be overlaid on the video image. This feature seems to be provided mainly to support games programmers to easily move ‘sprites’, but since overlays can obscure the full screen, a magnified image written to an overlay can be

⁵ MSAA is a COM-based technology that improves the way accessibility aids work with applications running on Microsoft Windows. It provides dynamic link libraries that are incorporated into the OS and application programming elements that provide reliable methods for exposing information about user interface elements. [5]

used to obscure the original image, presenting only the magnified image to the user. The magnifier uses the bitmap held on the video card as the original image (see Fig. 6) and thus no additional computation time is used in the construction of the image. This approach is described in detail in [6]. This approach is not a universal solution because it is not supported by NT4 and not all video cards support overlays.

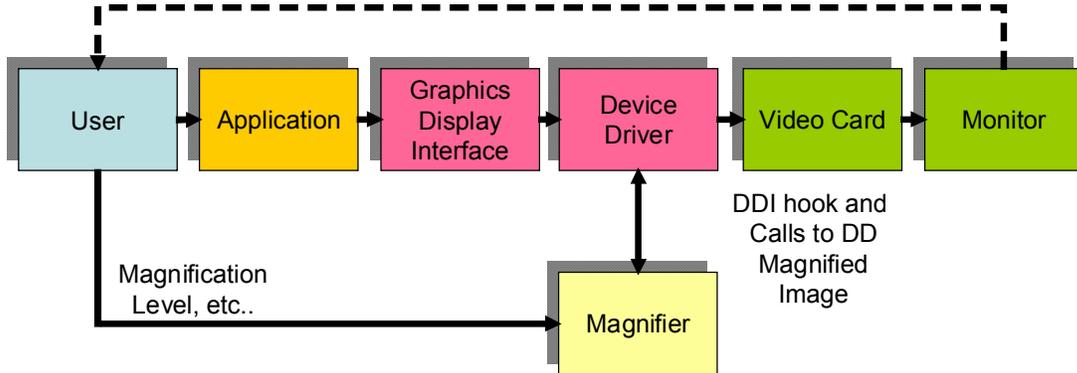


Fig. 5: Schematic of a Screen Magnifier using DD hooking. The figure assumes that the magnified image is created by using the DD – this may not be the case.

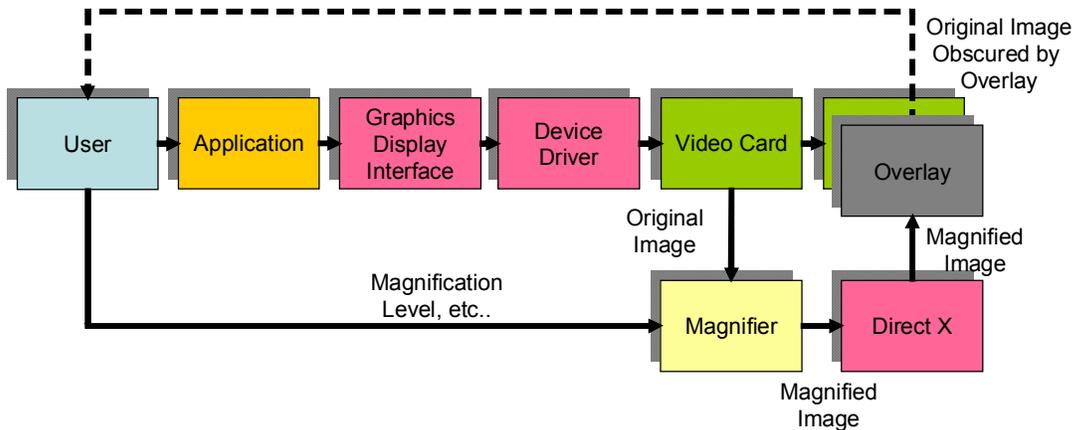


Fig. 6: A magnifier using overlays.

4 USER INTERFACES FOR SCREEN MAGNIFIERS

In this section we describe the common features provided by modern screen magnifiers. The first two of these (Colour Inversion and Enlargement) address the visual requirements of partially sighted users; the third, Smoothing, is required to reduce some undesirable artefacts of the enlargement process. The final three features are found in modern screen magnifiers and seem to attempt to address problems associated with users' understanding of where they are on the screen. Knowledge from the field of Human-Computer Interaction (HCI) is used to speculate on the usefulness of and the cognitive demands associated with these features.

4.1 COLOUR INVERSION

Many people with visual impairments prefer to work with alternative colour schemes. This typically involves inverting the screen so that light text is presented on a dark background (e.g. white text on a black background or yellow text on a blue background, commonly called 'negative contrast'). Colour inversion is the simplest form of colour mapping that can be provided. Some commercial magnifiers (e.g. ZoomText) provide customisable colour mapping where individual colours can be mapped. Customisable colour mapping can be useful for several purposes: ageing theory suggests that presenting text in negative contrast reduces screen glare, which is particularly useful for elderly people and people with Age-related Macular Degeneration (AMD); while presenting text in high contrast can be useful for people with colour blindness and elderly people with cataracts.

4.2 ENLARGEMENT

The main purpose of a screen magnifier is, of course, to enlarge the original image so that it can be read by a visually impaired person. Most screen magnifiers provide enlargement in integer steps between 2x and 16x (although some magnifiers provide up to 32x magnification). There is a trade-off to be made between the

magnification level and the proportion of the original image that can be viewed. Assuming uniform magnification, i.e. equal magnification levels in the horizontal and vertical directions, the proportion of the original image that is viewable is, of course, $1/n^2$ where n is the magnification level. This means that at 32x magnification the user can see $1/1024^{\text{th}}$ of the screen⁶ – which may correspond to a single character or less. In general, a user will try to use the lowest level of magnification that he/she can comfortably see in order to maximise the amount of useful information on the screen. Some magnifiers, for example Magnus++, allow the magnification levels for the horizontal and vertical directions to be independently controlled. This may be particularly useful in text reading applications when the vertical magnification level can be set to be greater than the horizontal. This has the effect of increasing text size without reducing the number of characters visible on the same line, at the expense of relatively minor character distortion. Other magnifiers, for example Ai Squared's Big Shot provide very modest magnification levels to provide support for people who require a small degree of screen enlargement, e.g. between 1.1 and 2.0 [6]. This type of magnifier is not widely used, Ai Squared report that their ZoomText system (a screen magnifier with 'standard' levels of magnification) outsells Big Shot in the ratio of around 9:1 [7], perhaps due to the wider range of magnification it offers.

4.3 SMOOTHING

An unfortunate consequence of enlargement is that the magnified image can appear blocky. This is because enlargement typically works by pixel copying and consequently replication of pixels over several lines can lead to an image that is aliased. Fig. 7 provides a comparison between un-smoothed and smoothed images. To reduce the 'blockiness', smoothing (anti-aliasing) techniques are commonly applied to the image.



Fig 7 Text with (right) and without smoothing (left). Taken from [8].

4.4 MAGNIFICATION MODES

Most screen magnifiers offer a range of magnification modes to the user. These are illustrated in Fig 8 and described below. Fig. 8(a) shows the original image.

- Full screen – in which the entire screen is taken over with the magnified image (Fig. 8(b)). The magnified area is roughly centred on the cursor position. The advantage of this mode is the maximum area of the screen is used for the information content. The disadvantage is users can only view a part of the information without being able to link it into the global context. Navigation around the screen in this mode is straightforward because there is only one area to consider.
- Lens mode – in which the area under the cursor is magnified, but only a certain proportion of the screen is taken up with the magnified image (Fig. 8(c)). As the cursor moves, the lens moves. Outside the lens, the original image is displayed. The advantage of this mode and the subsequent modes is the provision of the contextual information in the form of the unmagnified image. The disadvantage of this mode and the subsequent modes is that there is a need to design the screen carefully so that sufficient space is allocated for the contextual information and the content information. Users have control of both the lens size and the magnification within the lens. However, users have to mentally integrate the contextual information and the content information, which may be a challenging cognitive processing task for some users.
- Fixed area magnification – in which an area of the screen (typically the bottom or the top) displays the magnified image and the rest of the screen displays the original image (Fig. 8(d)). As the cursor moves, the area of the screen displayed in the magnified area is updated. This mode is similar to the early magnification systems such as the LLP described in Section 3.3.2. This style of magnification is demonstrated by Microsoft Magnifier. The advantages and disadvantages of this mode are similar to those described in lens mode. However, since the contextual and content information are consistently located in certain areas, we argue that it may be easier for users to integrate the contextual and content information.
- Moveable area magnification – this is similar to fixed area magnification, but differs in that the magnified area moves with cursor movement (Fig. 8 (e)). In many respects it is similar to Lens mode, but the magnified area covers a band across the screen, which puts the control of the focus window in users' hands at the expense of higher cognitive demand when integrating the contextual and content information.
- Moveable area magnification offset – this mode is the same as the previous except that the magnified area is offset from the cursor position (Fig. 8(f)). The reason for the offset is because the user may be able to

⁶ This assumes full-screen magnification. With other modes the proportion of the screen will be further reduced.

determine some additional contextual information from the location of the unmagnified cursor in the unmagnified original image.

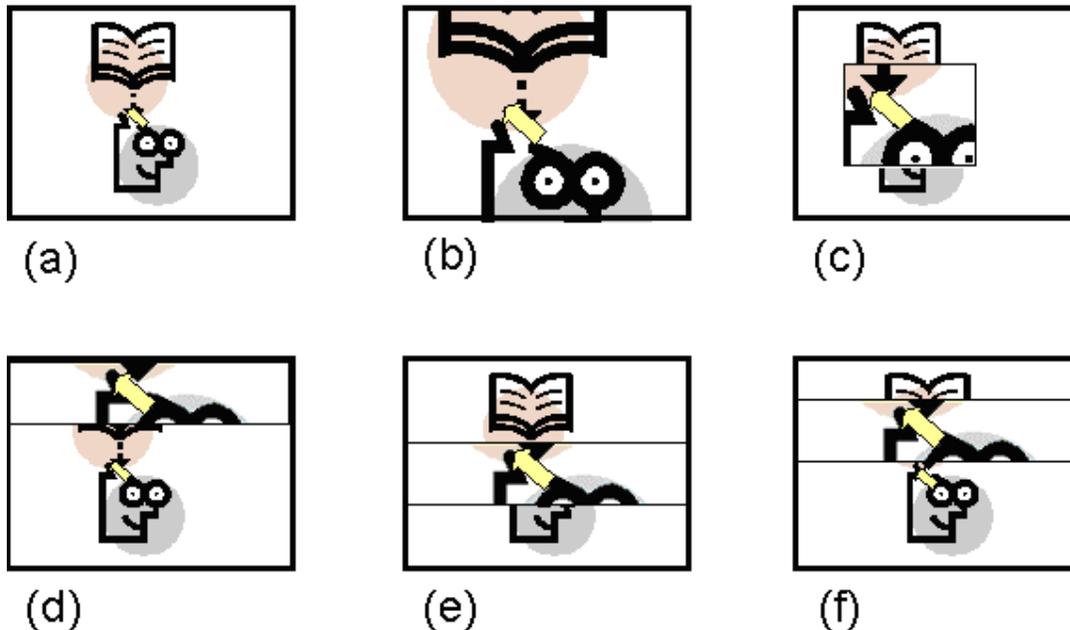


Fig. 8 Magnification modes. (a) Unmagnified image, (b) Full-screen magnification, (c) Lens magnification, (d) Area magnification (fixed), (e) Area magnification (moveable), (f) Area magnification (moveable and offset).

All the modes that present contextual information (i.e. all modes except full screen magnification) may potentially reduce the risk disorientation because of the representation of the cursor position in the original image. However, this idea is predicated on the user being able to see the original image in sufficient detail to determine precisely his/her position, and this is unlikely to be true for many users. As a consequence, and despite the fact that screen magnifier manufacturers have developed a number of new magnification modes over the past few years, the vast majority of users predominantly use full screen magnification.

4.5 MULTIPLE VIEWS

Some systems also enable multiple magnified views. The earliest of these was the EGA Soft Window system. Here two or more magnified views are presented on different parts of the original screen. This technique enables, for example, a word processor user to edit the body of their text using one view, say at the top of the screen, and to view of the status bar of the word processor at the bottom of the screen. Another, more complex example is when a spreadsheet user wishes to view three parts of a spreadsheet taken from different parts of the screen: the current cell; the name of the current cell; and the formula for the current cell. In theory therefore Multiple Views allows the user to select different areas of the screen that provide useful contextual information. However, these areas must first be identified and selected. One approach is for the user to select the context-providing areas. However, this approach is probably only of use to sophisticated users of applications who have a good understanding of which areas will provide them with appropriate contextual information, who have the ability to find and select these areas, and who can manage their positioning on the screen. The second approach is for the magnifier to select which areas are to be magnified. This can be done by magnifiers that have application-specific scripts that determine which areas should be magnified based on the application and the user's current tasks. We believe that a small number of commercial magnifiers are scripted in this way and the approach is quite common in screen readers.

5 USER STUDIES OF A JOYSTICK-CONTROLLED FULL-SCREEN MAGNIFIER

Despite the fact that screen magnifiers are widely used by visually impaired people, there are few studies that investigate the usability of screen magnifiers. We performed two user studies of screen magnifier. The primary goal was to determine how easy or difficult visually impaired people found joystick control of a screen magnifier developed at UMIST. The study also aimed to evaluate with visually impaired users some of the basic features commonly provided by commercial magnifiers.

There are several reasons for testing a joystick as an input device to the screen magnifier. A mouse, the most commonly used input device for a screen magnifier, is a relative positioning device; hence its position cannot be

used as a spatial reference. The magnifier can use a joystick as an absolute positioning device so that its movement corresponds directly to the location of the magnified image within the original image (called *the absolute positioning mode*). That is, to view the bottom right corner of the document, the joystick is tilted toward the user and to the right. The joystick also offers additional benefits, i.e. the buttons can be used to control the operation of the magnifier (see Fig.9 and 10). The magnifier can also treat the joystick as a *relative positioning* device, where a displacement of the joystick from its centre position causes the magnifier to move its focus in the corresponding direction. The image will continue to move in the direction of the displacement until the joystick is again centred, the edge of the screen is met, or the joystick is moved to indicate a new direction of travel. To maintain the magnified image as the same portion of the original image the user must move the joystick back to the centre position. This contrasts with the absolute positioning mode, where the user keeps the joystick in a particular position and the focus rests at the location indicated by that position.

Because full-screen magnification is the most commonly used mode, these studies only tested this mode. Using only this mode also allows the investigation of whether the use of a joystick as an absolute positioning device (rather than the provision of the contextual information in the form of the unmagnified image) helps users in their interaction with the screen magnifier. Magnus was used as the screen magnification program. The studies were carried out using a laptop computer with a 1GHz Pentium processor, 512 MB memory, running Windows 2000 Professional. The screen was a 15" active matrix LCD display with resolution set to 1280 x 1024 pixels with 32-bit colour resolution. To verify that users still have usable vision, they were asked to fill in a paper questionnaire, printed in 24pt Arial font, concerning their demographic data and past computer experience.

5.1 THE FIRST STUDY

In this study, a Microsoft ForceFeedback 2 joystick was used because it is one of the most popular joysticks on the market. The mapping between features and buttons of the joystick is shown in Fig. 9.



Fig.9. The mapping of buttons to magnifier features in the joystick used in the first study (taken from [8]).

Seven registered blind users (two females and five males) with some residual vision participated. The participants' mean age was 50.9 years (S.D. = 17.80 years) with a mean of 9.2 years of prior computer experience. Six of the users had used a screen magnifier before, but none had used Magnus; four of the users had prior experience of using a joystick with computer applications.

The participants used Magnus to seek answers to comprehension-based questions from two sets of documents (taken from <http://www.aarp.org>) presented in Microsoft Word XP. The questions were chosen carefully to ensure that the users would not be able to answer the questions without reading the articles. The document was displayed onscreen in 10pt Arial font, black text on a white background, as is commonly found in online newspapers. The participants followed think-aloud protocol.

The following comments on the usability of the system are derived from the experimenters' observations, the participants' comments recorded during the experiment and the participants' answers to usability-related questions. Overall, most participants indicated that they felt that, with some practice, controlling the screen magnifier with the joystick would be easy to master. They also felt that it was a good idea to assign the screen magnifier's features to the joystick's control and buttons. There were some complaints about the buttons of the joystick. The buttons were located toward the back of the joystick's base, which interfered with the throttle or the stick when they tried to reach those buttons. The buttons were also very similar in shape and colour, making them difficult to distinguish.

There were a number of problems reported with the joystick itself. The joystick was somewhat imprecise and this often caused participants to overshoot their desired location (in absolute positioning mode) or move too fast (in relative positioning mode). One elderly participant was concerned for other users of his age group, whose trembling hands may amplify this problem. Indeed, a previous study comparing the use of a mouse versus a joystick found that the physical construction of the joystick made it susceptible to tremor in a manner that the mouse was not [9]. Two solutions to this joystick problem were recommended: to use a joystick with more

resistance; or to provide the user with a directional lock to limit movement to one direction only. Both suggestions were implemented in the second study and are described below.

As mentioned previously, the second goal of the user studies was to evaluate some of the basic features commonly provided by commercial magnifiers with visually impaired users. Most of our participants preferred to view the screen in negative contrast (white text on a black background), while most online materials are presented in positive contrast. Therefore, the colour inversion feature was highly appreciated. The most intuitive and favoured feature was the use of the joystick's throttle to control the magnification level.

All participants (except one) commented favourably on smoothing, especially those that used high levels of magnification, although some did not notice the difference, possibly because of the extent of their visual impairment. It should be noted, however, that smoothing requires more computer processing, which can result in a longer delay between a user's movement of the joystick and the corresponding movement of the screen focus. One user noticed the resulting degradation in responsiveness but did not consider it a problem.

The relative positioning mode was originally provided to give users finer control to complement the relatively coarser control provided by the absolute positioning mode. Interestingly, the participants viewed the two positioning modes as different scrolling speeds (relative = slow, absolute = fast). Participants suggested that the modes had different uses. One user stated that the absolute positioning mode would be particularly useful for novice users because this mode was spatially easy to understand, while the relative positioning mode may be more useful for expert users for finer control over the joystick.

A panning window, a common feature that, when activated, indicates the current focus by means of a highlighted block on the original image, was provided to allow the user quickly to obtain an idea of where the current magnified portion of the screen fits into the original screen. In addition, users could move quickly to other parts of the screen by selecting the position that they desired. This feature was not considered useful because users' visual impairment prevent them to see the original image in enough detail to enable them to perform meaningful selection of the area to view or move to.

5.2 THE SECOND STUDY

In this study, A Saitek Cyborg 3D joystick was chosen by the experimenters based on feedback from participants of the first user study. This joystick offered more resistance than the first joystick, its buttons were larger, better located (the throttle was located further away on the far side of the joystick buttons, thus reducing the possibility of accidental movement of the throttle) and had different shapes (which for this study were also painted in different colours, making it easier for the participants to recognize the buttons and recall their associated functions). The mapping between features and buttons is shown in Fig. 10.

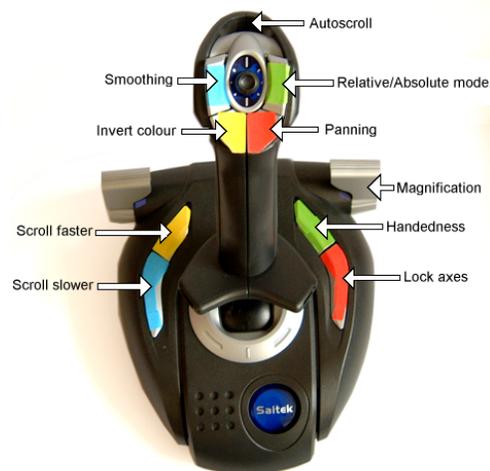


Fig.10. The mapping of buttons to magnifier features in the joystick used in the second study (taken from [8]).

Six other registered blind users with some residual vision participated in the study. There were four females and two males with a mean age of 39.0 years (S.D. = 20.83 years) and a mean of 8.3 years of prior computer experience. Three of the users had used a screen magnifier before, but none had used Magnus; two users had prior experience of using a joystick with applications.

Magnus was used to find an image, a word and some data in a table and to find and click links in locally stored Web pages (adapted from <http://www.bbc.co.uk>) presented by the Mozilla Web browser. As in the first study, think-aloud protocol was adopted.

From the experimenter's observation, users had fewer problems controlling the new joystick. From various comments in this study, we speculate that the problems faced by the participants of the first study were not a result of joysticks in general being difficult to control but a result of the particular joystick used.

Based on the feedback from the participants of the first study, a directional lock was provided. However, there were participants in the second study offered mixed comments on this feature. Some participants said that it was a good feature but could not think of the applications where it would be particularly useful. Some stated that it was not useful because they were confident they could control the joystick without the help of the directional lock. One user said that the lock would help users in maintaining their orientation. In general, most participants used this feature with few problems. Users in the second study were also quite positive about most features offered. Colour inversion, the use of the joystick as the throttle control, smoothing and the two positioning modes were considered useful and favourable. Users were still negative about the panning window feature.

6 CONCLUSIONS

The paper discusses a wide range of issues related to screen magnifiers and their user interfaces. Reviews of historical systems and contemporary approaches of screen magnification systems reveal that controlling the features commonly provided by commercial screen magnifiers is quite complex.

In this paper, a knowledge of HCI is used to speculate on the trade-offs between the usefulness and the cognitive processing demand associated with these features. We tested some of our hypothesis by performing a usability evaluation of a screen magnifier. In this study, we also tested our intuition that using an absolute positioning device will help users orienting and navigating, which was supported to some degree.

Screen magnifiers have evolved over the years and will probably continue to evolve. Evolution in this sense means that the new generation of screen magnifiers may become more feature-laden (but not necessarily more effective). Iterative, user-centred design in which prototypes are evaluated and improved in successive cycles has enabled us to understand users' perception on the effectiveness of various features. We would point to the approval of colour inversion and smoothing and the disapproval of panning window (which has been traditionally included in commercial screen magnifiers) as an example. Usability test also frequently creates unexpected results. Users preferred both cursor control modes (relative and absolute) and gave us insight into how they may be applied in expert and novice modes. This presents us with a further design problem of setting user modes either explicitly or implicitly via an adaptive user interface. For control devices users preferred the joystick with more haptic feedback and larger, more obvious button controls. This suggests a new design principle in which the granularity of perceptual modalities should be matched when perceptual modalities and control cues are coordinated. That is, users with poorer vision need coarser grained haptic feedback and more visually salient control cues (i.e. joystick buttons) to navigate more effectively. In our future work we will extend our study by using other modality (e.g. audio) as feedback. In summary, we believe that user involvement can help designers to develop a more effective screen magnifier.

REFERENCES

- [1] P. Blenkhorn, D. G. Evans. The Architecture of a Windows Screen Reader. In C. Marinček, C Bühler, H. Knops, & R. Andrich (Eds.) *Assistive Technology – Added Value to the Quality of Life (AAATE 2001)*, IOS Press, Amsterdam, pp 119 – 123, 2001.
- [2] M. Gibson and N. Reiser, “Lyon Large Print Program”, *Senus*, pp 32-34, Summer 1987.
- [3] A. Baude, P. Blenkhorn, D. G. Evans. The Architecture of a Windows 9x Full Screen Magnifier using DDI Hooking. In C. Marinček, C Bühler, H. Knops, & R. Andrich (Eds.) *Assistive Technology – Added Value to the Quality of Life (AAATE 2001)*, IOS Press, Amsterdam, pp 113 – 118, 2001.
- [4] P. Blenkhorn, D. G. Evans and A. Baude. Full Screen Magnification Using Direct X Overlays. *IEEE Transactions on Neural Systems and Rehabilitation Engineering* (in press).
- [5] Microsoft. Microsoft Active Accessibility Version 2.0. Available at: http://msdn.microsoft.com/library/default.asp?url=/library/en-us/msaa/msaastart_9w2t.asp. Current at: 05/12/02.
- [6] Ai Squared. Introducing BIGSHOT®. The first software magnifier designed specifically to combat computer eyestrain. Available at: <http://www.bigshotmagnifier.com/>. Current at 05/12/02.
- [7] B. Weiss, *private communication*, 2002.
- [8] S. Kurniawan, A. King, D. G. Evans, P. Blenkhorn. A usability evaluation of a joystick-operated full-screen magnifier. *Lecture Notes in Computer Science: State-of-the-art surveys* (in press).
- [9] A.K. Mithal, S.A. Douglas. Differences in Movement Microstructure of the Mouse and the Finger-Controlled Isometric Joystick. *Proceedings of CHI'96* (Vancouver, Canada, April 1996). ACM Press, 300-307.