MouseLupe

An Accessibility Tool for People with Low Vision

Luciano Silva¹, Olga R. P. Bellon², Paulo F. U. Gotardo² and Percy Nohama¹

¹CPGEI - CEFET/PR - Curitiba-PR, Brasil.

²Universidade Federal do Paraná - Curitiba-PR, Brasil.

Abstract: This paper presents a new accessibility tool developed to help people with low vision disabilities. The tool was initially designed to aid in Web navigation, but it can also be useful in other applications. First, we describe the importance of the Web as a source of information and knowledge and explain the main difficulties that disabled people have in accessing Web pages. We present relevant software applications for accessibility of people with low vision and explain their features and limitations. The developed tool named, *MouseLupe*, its characteristics and main contributions are presented in detail. Finally, we show a comparison of the developed tool with other related applications highlighting its efficiency and usefulness.

Key words: web navigation, usability, low vision disabilities, magnifier lens

1. INTRODUCTION

The World Wide Web has become the largest source of information, knowledge and entertainment in recent years, used mainly by people involved in teaching activities, scientific research and electronic commerce. Originating in the academic world, it is already considered an indispensable work tool for researchers in many different areas.

Although there is a ready access to many subjects provided on the Web, this does not avoid the common situation that a simple search for specific information may be a quite difficult task. Frequently precious time is wasted finding and visualizing significant content on some Web sites. For people with visual disabilities, the problem may be worse than that and practically unbearable.

Initially, Web pages had only text and their evolution to complex pages with images, sounds, graphics, and animations provided great benefits for many users. However, they also created difficulties for people with disabilities and have become practically inaccessible for the disabled (Head, 2000; Taucsher, 1997).

Even with the complex and constant transformations of Web organization, the number of disabled people using computers has surely been increasing (Paciello, 2000).

The World Wide Web Consortium (W3C) (Berners, 2000), an organization interested in all aspects of the Web, sponsors the most extensive set of programs and initiatives devoted to the issue of Web accessibility for people with disabilities. The W3C has created the Web Accessibility Initiative (WAI, 2000). It has published many documents and provided a single most comprehensive and useful set of resources for Web designers who wish to make their sites more accessible to people with disabilities.

There are four main categories of disabilities that affect a person's ability to use the Web:

- Mobility, including inability to move, insufficient dexterity to operate a mouse or a keyboard, inability to control unwanted movement, and lack of limbs.
- Hearing, ranging from inexact hearing, or diminished hearing, to no hearing at all.
- Cognition and learning, including various difficulties in reading, understanding, staying focused, remembering, and writing.
- Vision, including blindness, low vision and colour-blindness.

For people with mobility disabilities, there are several hardware devices and software applications that provide Web navigation and make computer usage possible: mouth devices to control cursor movements, Web cameras to capture eye or head movements and guide the cursor movements, speech recognition software to write words, execute commands, and others. Using these devices, people with mobility disabilities can access every Web page (Carter et al, 2001).

Sounds created as part of Web pages are normally irrelevant in the page's context except when they are songs or speeches. People with hearing impairments usually have easy access to the Web because most of the Web pages do not contain sound information.

Cognition and learning problems are very complex. Many researchers have been studying the relationship between the structured content of a Web page and people with these disabilities in order to propose efficient solutions and accessibility rules for designing Web pages. It is known that a page with a simple and intuitive interface, with clearly formulated text, and a consistent

navigational scheme between pages can facilitate access to those disabled people (Paciello, 2000).

Vision disabilities are considered a permanent concern in the research and development of accessibility tools. Many research groups have been concentrating on this subject and proposing new alternative techniques to aid those people.

Screen-magnification software and high contrast displays are commonly used by people with low vision and help them to read the text and interpret the images on websites. Such software improves the readability of small text, but limits the visible area of a document. Enlarged graphics that contain text may be difficult to read.

People who are blind do not use the screen display or mouse. Instead of reading Web pages or viewing images, they listen to the Web through software that acts as a screen reader. These screen readers convert text into synthesized speech and are unable to interpret the graphical content of images. Because of that, some pages may be very difficult or impossible for a blind person to understand or navigate.

In the United States approximately 3.5% of the population have low vision and only 1% of these uses a computer. If we consider this equivalent percentage for the whole world population, probably the percentage of people with impaired vision that use computers will be even smaller (Paciello, 2000).

Today, the knowledge about biological vision is extremely limited. Some research groups in computer vision have been proposing systems using procedures similar to human vision processes to explain its properties and to develop new computer vision applications (Cutzu, 1998; Georgeson, 1998). Projects are being developed to insert accessibility tools in many computer applications or in widely used computer operating systems such as Microsoft Windows and more recently LINUX (LINUX, 2001).

Section 2 of this paper discusses related works on accessibility tools for low vision disabilities. In Section 3 the developed tool is described and shown through the number of examples. The conclusion is presented in Section 4.

2. RELATED WORK

There are many accessibility tools for each disability category. In this paper, we highlight and compare just those accessibility tools for people with low vision, not for blind people. For blind people, we would have, for example, IBM's software named, *Home Page Reader*, a text reader that uses the company's *Via Voice Out loud* text-to-speech synthesizer.

A visually impaired user can use the IBM software to speak text, describe images, and other fields of Web pages (IBM, 2000). Accessibility tools for people with low vision are dedicated to help people in visualizing the content of a computer display more easily, as well as to aid in Web navigation.

Apple is an important company that develops a number of products for people with disabilities to view websites, including screen-magnification software, text-to-speech synthesis, voice recognition, and so on (Apple, 2000).

Most of the accessibility applications for people with low vision make zoom amplification of a certain area of the computer display to facilitate the visualization of its content. These applications are commonly named as screen magnification and generally the centre of the amplification area is the mouse cursor position. The zoomed area is shown in a different window of the computer display, overlapping other application windows. Whenever the user moves the mouse, the amplification screen is refreshed automatically.

In the most recent versions of its Windows operating system, Microsoft added an application named *Magnifier* (Microsoft, 2000), (Figure 1) that divides the computer screen horizontally in two parts. The top half, composed generally of 20% of the total screen area, is used for the amplification and the other is used for the Windows desktop.

Figure 1. Example of the Microsoft Magnifier application.

Because of the restriction on the amount of viewable area of the screen, a large, high resolution monitor is required for the screen magnification software to effectively increase the viewable area on the screen.

The main problem of this application is that, besides the reduced display area, the user can be confused when moving the cursor in the applications window, while looking at the magnified part of the screen. That is because of the difficulty in identifying the relative mouse position in the amplification window and its relationship to the application window. This problem is worse when the application is a Web browser with which the user reads the content of a Web page and moves the mouse to continue reading. In this case, the user may become disoriented and waste time to find the required position again.

In many applications, the option of changing the size of the text characters (font size) is considered an accessibility option. However, to enable this option, a disabled person initially needs the help of another person. In Microsoft Windows, there is an accessibility wizard (Windows, 2000), which guides the user through a series of Windows appearance options. These include adjustments to the font size, screen resolution, scroll bar size, icon size, colour scheme, mouse cursor appearance and mouse cursor blink rate.

The *Loupe* (Loupe, 2001) shown in Figure 2, is a small utility that displays a magnified view of whatever is beneath the mouse cursor, much like a jeweller's or printer's loupe. The captured image can be pasted into any graphics program and most word processors. The *Loupe* also includes options to monitor a specific area of the desktop instead of normal mouse tracking.

Figure 2. The Loupe application applied on Microsoft Windows Desktop.

The *BIGSHOT* (Bigshot, 2001), see Figure 3, is a screen magnification program that enables users working with large monitors and high screen

resolutions to quickly magnify part or the whole screen to display small pictures, icons and text in a more comfortable way.

Figure 3. The BIGSHOT applied on a Web browser.

The *Lunar* (Lunar, 2001) is a software magnification system that magnifies the screen from 2 to 32 times, at any screen resolution and colour depth. It allows horizontal and vertical magnification to be adjusted separately, and incorporates image smoothing, to make characters more readable particularly at higher levels of magnification.

The *ZoomText Xtra* (Xtra, 2001) software combines screen magnification software with a built-in document reader and is supplied in modular form, where level 1 provides magnification and level 2 adds speech output using any sound card. A speech synthesizer can also be used to provide the speech output (Figure 4).

Figure 4. Example of the ZoomText Xtra application.

Most of the available accessibility software is for Windows. There are some LINUX packages of accessibility tools for example, the *gmag* and *xlupe* (LINUX, 2001). The *gmag* is a screen magnification software similar to the *Magnifier* of Microsoft Windows. The *xlupe* software is a simple application that has a window of variable size which shows the amplification image of the region where the mouse cursor is. Software for LINUX is in its greater part open source, free software, which means no acquisition cost.

The main limitations of these applications and their implications for the user are:

1. The amplification window limits the useful display area.
2. The windows overlapping may hide significant parts of an application.
3. The user's attention is focused in a small region covered by the amplification window.
4. The user can be disoriented when visualizing the relative position of the mouse cursor in the amplification window.
5. The user may not have a perception of the global aspect of the application because they are always visualizing the small amplification window whose location is fixed.
6. In most of the applications in which it is possible to change the font size, there is no way to change the font size of the toolbars or option menus. In this case the user has to request the help from another person.
7. The change of the amplification level is a difficult task when this option is on the amplification window.

Following this initial study of the current accessibility applications, and their limitations and their virtues, we have proceeded to develop a new and more efficient accessibility tool.

3. DEVELOPMENT OF THE MOUSELUPE ACCESSIBILITY TOOL

We had difficulties obtaining a specification of the Microsoft Windows window manager. Thus, we decided to develop the *MouseLupe* for the LINUX operating system. As LINUX distributions have no acquisition cost, the adoption of this operating system made it possible to develop accessibility applications at an insignificant cost.

Our objective was to develop a tool like the traditional screen-magnification software, but not having a static window. The application needed to be fast and dynamic and show quickly whatever the user would like to enlarge.

We envisaged a tool integrating the mouse cursor movement and the displacement of the amplification window. The result was the development of *MouseLupe*, see Figures 5 and Figure 6.

Figure 5. Example of the *MouseLupe* amplification result over the Netscape browser.

MouseLupe is an original tool that simulates perfectly a magnifying glass (Lupe) guided by the mouse movement. It preserves the functions of the mouse buttons allowing a common Web navigation (Figure 5). It makes a great general purpose screen magnification utility for computer users with sight impairments.

The development guides of *X Window System* were used in this work (Nye, 1992a; Nye, 1992b) and made it possible to consult all the sub-routines of the *X window server*, mainly the window and mouse events. The *X* controls the graphic interface of LINUX and a window manager is executed as a first X client application, for example *WindowMaker*, *KDE*, *IceWM*, and so on.

When *MouseLupe* starts, it initially creates a transparent window, without borders, to contain the magnifying glass. The window borders of an application are attributes of the window manager. In order to create a window without its borders we needed to use the resource *OverrideRedirect* of *X*. This resource makes the application answer only to the *X server* commands and not to the window manager ones. This way, the window is directly controlled by the *X* events.

Figure 6. Example of the MouseLupe with high amplification result (zoom 8x).

The mouse and window events, obtained by the *XQueryPointer* and *XMoveWindow* methods, were used to move the amplification window following the mouse cursor position. The methods *XUnmapWindow* and *XMapWindow* were used to quickly remove the amplification window, copy the content of the display on the mouse position, and show the amplification window again. When doing that, a blink in the magnifying glass is noticed, facilitating the *MouseLupe* identification on the screen.

A circle of 2 pixels wide in a light green colour was added to the border of *MouseLupe* to improve its limits identification in the screen. The *MouseLupe* content is refreshed every 5 seconds by default configuration, however the refreshing time may be decrease to allow the visualization of image animations shown in a Web browser. The size of the glass magnifying region is proportional to the amplification factor and its shape is a circle.

There is a 1-pixel hole in the centre of *MouseLupe* circle to allow mouse clicks to function. The mouse events are applied in the window below the *MouseLupe*.

The *MouseLupe* was tested by several users that checked its efficiency not only for people with visual impairments but also for people that need to magnify regions of its applications in a fast and dynamic way. The users' main comment was that they found this tool to be an efficient screen magnification tool that could be used in most computer applications.

The developed tool was compared with the other accessibility tools and the results showed the efficiency of the *MouseLupe* tool. The test was

accomplished equally for each accessibility tool. A group of 20 users of different ages (between 20 years and 50 years) was consulted. Testers were asked to rate each tool on 8 different criteria between 1 (very unsatisfied) and 5 (very satisfied).

The chosen criteria for the test were:

1. Easiness for web navigation.
2. Easiness to start and stop the tool.
3. Easiness to change the tool configurations (e.g. zoom factor).
4. Easiness in the use of the interface.
5. Increase of the productivity in the use of the computer.
6. Time for adaptation with the tool.
7. Graphic quality.
8. Number of tool options.

Figure 7 presents a graph of the comparative results of the tools with the average of the evaluations.

Figure 7. Comparative results.

The *MouseLupe* tool is efficient and simple and it does not require any resource of special hardware. In this first version, the only requirement is a computer with a LINUX operating system.

The main advantages of the *MouseLupe* in relationship the other tested applications are:

1. It eliminates the overlapping window of traditional screen-magnification software.
2. It eliminates user confusion when the screen is split in 2 windows as is shown in the Microsoft Magnifier application.
3. It allows normal mouse functions, move and button clicks.

4. It is possible to see an animated Web page because of the continuous refresh, even when the mouse cursor is not moving.
5. It is possible to move the mouse and quickly see an amplified region of the screen quickly.
6. The user can change the *MouseLupe* configurations using their own *MouseLupe* magnifying region to see the configuration options enlarged.

In the next version, other features are being developed for *MouseLupe*. These are:

- **Colours reduction mode** to reduce the number of colours shown in the amplification window.
- **High contrast view mode** to show the amplification image in high contrast colours.
- **Voice command recognition** to allow changes of magnification level, or Lupe window shape, or to enable or disable the *MouseLupe*.
- **Lupe window shape** to allow different shapes of the *MouseLupe*, such as a circle, cube, rectangle, Lupe, and so on.
- **Text mode** so that the image on the amplification window has only white or black pixels.
- **Image smoothing** to make characters more readable particularly at higher levels of magnification.

4. CONCLUSION

In this paper, we have reported on the results of our research on accessibility tools to aid people with low vision to access computer applications, mainly Web navigation ones. We have also developed an accessibility tool, named *MouseLupe*, for people with low vision. This tool allows disabled users to access the Web in a more comfortable and pleasant way, stimulating their social inclusion and improving their access to information and, thus, their quality of life.

The *MouseLupe* is an original tool that simulates perfectly a magnifying glass (Lupe) guided by the mouse movement and preserves the functions of the mouse buttons allowing a common Web navigation. It makes a great general purpose screen magnification utility for computer users with sight impairments.

For our future work, we intend to finish the new features of *MouseLupe* and make it available for other operating system platforms, including the Microsoft Windows versions.

5. REFERENCES

- Apple (2000), Apple Computer, Inc., Disability resources, Apple Computer Website, <http://www.apple.com/education/k12/disability>
- Berners, L. T. (2000), Web accessibility initiative, World Wide Web Consortium, <http://www.w3.org/WAI/>
- Bigshot (2001), The BIGSHOT screen magnification software, <http://www.bigshotmagnifier.com>
- Carter, J. & Markel, M (2001), Web Accessibility for People With Disabilities: An Introduction for Web Developers. *IEEE Transactions on Professional Communication*, (44)4:225-233.
- Cutzu, F. & Edelman, S. (1998), Representation of object similarity in human vision: psychophysics and a computational model, *Vision Research*, (38):2229-2257.
- Georgeson, M.A. (1998), Edge-finding in human vision: a multi-stage model based on perceived structure of plaids, *Image and Vision Computing*, (16):389-405.
- Head, M. & Archer, N. & Yuan, Y. (2000), World Wide Web navigation aid, *International Journal of Human-Computer Studies*, (53):301-330.
- IBM (2000), The voice of the world wide web, <http://www-3.ibm.com/able/hpr.html>
- LINUX (2001), The blind LINUX project, <http://leb.net/blinux/>
- Loupe (2001), Loupe package for accessibility of people with low vision, <http://www.gregorybraun.com/Loupe.html>
- Lunar (2001), The Lunar magnifier, <http://www.dolphinuk.co.uk>
- Microsoft (2000), Microsoft Corporation, Accessibility policy and strategy, <http://microsoft.com/enable/microsoft/policy.htm>
- Nye, A. (1992a), Xlib Programming Manual, vol. 1., O'Reilly and Associates, Inc.
- Nye, A. (1992b), Xlib Reference Manual, vol. 2., O'Reilly and Associates, Inc.
- Paciello, M. (2000), People with disabilities can't access the Web, <http://www.webable.com/mp-pwdca.html>
- Tauscher, L. & Greenberg, S. (1997), How people revisit Web pages: empirical proceedings and implications for the design of history systems, *International Journal of Human Computer Studies*, (47):97-137.
- WAI (2000), Policies relating to Web accessibility, <http://www.w3.org/WAI/>
- Xtra (2001), The "ZoomText Xtra" application, <http://www.aisquared.com>