The User Interface for *Interactive*_SIGHT

David Oliver
Dept of Computer Science, Millersville University
dtoliver@marauder.millersville.edu

Stephanie Elzer
Dept of Computer Science, Millersville University
elzer@cs.millersville.edu

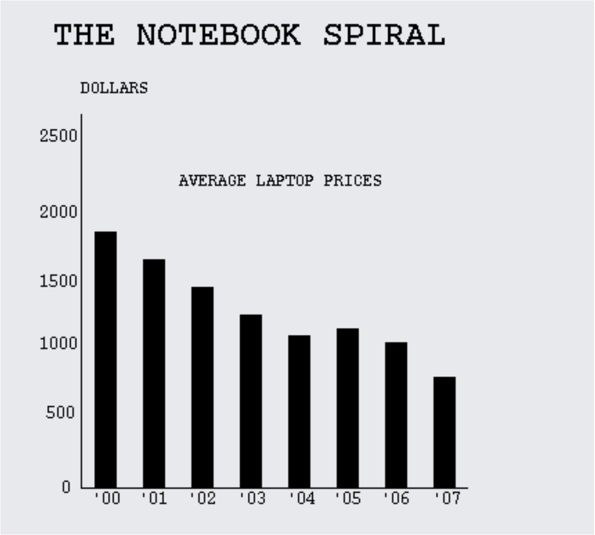
ABSTRACT

The *Interactive*_SIGHT (Summarizing Information GrapHics Textually) system is an ongoing effort that aims at providing visually impaired persons with greater accessibility to information graphics found on the Web. Originally named SIGHT, the system uses a Browser Helper Object (BHO) and screen reading software to enable visually impaired internet users to learn the underlying message of information graphics (specifically bar charts), which would have previously been skipped over by the screen reader [1]. Recently, work has been done to facilitate extensive user interaction with the SIGHT system. This extension now allows users to of the system to attain follow-up responses to requests for more information, allowing users to gain deeper understanding of the graphic [2]; the system is now called *Interactive*_SIGHT. Expanding the system at the API or code-level obviously calls for a more extensive user interface to support the additional functionality. Since *Interactive*_SIGHT's users are visually impaired, there is the challenge of designing an accessible interface that allows these users to navigate through the program with as much ease and as little confusion or frustration as possible. This paper describes an interface for *Interactive*_SIGHT that enables visually impaired users to gain knowledge of information graphics found on websites.

1. Introduction

Despite the development of assistive technologies such as screen readers and haptic devices, many blind and visually impaired persons become frustrated when using the internet. Web designers do have the ability to provide alternative text ("ALT text") to graphics on their sites, but very few ever implement this feature [1]. Further, poor site design including cluttered pages and non-intuitive layouts cause users with sight impairments to often be unproductive on the Web. In fact, according to a Nielsen Norman Group study in 2001 blind users were six times less successful and users aided with screen magnification were three times less successful than sighted users in accomplishing the study's prescribed online tasks [3]. The U.S. Government passed Section 508 of the Rehabilitation Act in 1998. This amendment required Federal agencies to make information technology more accessible. However, the rules outlined by this amendment do not apply to private sector websites [4]. Consequently, the private sector has not taken as active role in making the internet accessible to those who have disabilities. Still, the lack of well-designed websites has been acknowledged by several groups, and they have constructed accessibility guidelines for web-designers [5] [6].

It has been our observation [1] that the vast majority of information graphics that appear in popular media sources such as newspapers and magazines have an underlying communicative message that the graphic designer intended to convey to the viewer. For example, the bar chart in Graph 1 appeared in its original form in BusinessWeek on September 5, 2005 (it has been recreated here). A viewer will be able to see that the dollar value of average laptop prices fell from 2000 to 2003 and then fell more slowly until 2007. It is the overall purpose of the Interactive_SIGHT sytem to provide access to the underlying message and content of information graphics such as this one for visually impaired viewers. The core methodology of the Interactive_SIGHT system for recognizing the intended message, generating follow-up responses, and presenting the initial summary within a Browser Helper Object is described elsewhere [7] [2] [1] [12]. This paper describes the accessibility aspects of *Interactive*_SIGHT's user interface for presenting follow-up responses and how it maintains the system's goal of providing visually impaired users with an effective source for gaining the underlying message of information graphics.



Graph 1: Information Graphic Example

2. Approach

The user interface implemented for *Interactive*_SIGHT takes many of the recommended guidelines for accessible websites and applies those to the system's dialog box driven interface. It is assumed that the user will be utilizing screen reading software and/or screen magnification software to access the interface. The description in this paper is geared towards users utilizing screen reading software, such as JAWS by Freedom Scientific (the most widely used screen reading software for Windows,) since our system has been tested using JAWS.

Following accessibility recommendations, our user interface includes an intuitive TAB order that allows users to quickly move through the different buttons. Also, each dialog box has a descriptive title bar, making it easy for users to orient themselves with their current location within the system's interface. These two elements, along with several more, preserve and enhance the accessibility of *Interactive*_SIGHT. Throughout the rest of this paper we will describe the accessibility elements of *Interactive*_SIGHT and why we chose to implement these elements. We will provide an overview of the interface, and finally we will discuss the users' feedback on the interface of *Interactive*_SIGHT.

3. Accessibility Elements

This section explains the layout and navigational elements implemented in the *Interactive*_SIGHT interface and the accessibility reasons behind their inclusion.

3.1 Layout Elements

The success and utility of a program lies heavily upon how well the user interface is constructed. Web and software designers are constantly trying to tailor products to suit their clienteles' usage habits and abilities in the best possible way. Research done by the American Foundation for the Blind (AFB) revealed several important facts about blind and visually impaired users [8]. First, sight impaired users commonly navigate through webpages either through searching or scrolling. That is, some users like to find a particular piece of information and then move on (searching), whereas others would rather have the entire screen read to them (scrolling) before choosing a navigational button. Consequently, *Interactive*_SIGHT had to have an interface that met both of these methodologies. Our solutions to the searching method are discussed later (in section 3.2). Because some users like to hear an entire screen read to them before proceeding, windows with too many links or buttons can take a significant amount of time to traverse [9]. Our interface solves this problem by breaking up windows by their content, thus limiting the number of buttons per window to eight or less. A second point of consideration in our design was the consistent placement of the navigational buttons. The AFB's research showed that blind and visually impaired users rely heavily upon their ability to remember the layout of a site. Thus, if the layout is consistent from window to window users will be able to quickly become familiar with the design and begin to move easily and efficiently through the system's interface. Another layout consideration was making sure users were aware of their location in the interface. The AFB's research once again shows that blind and visually impaired users can quickly become confused as to where they are -- often because the same navigational links are located at the top of every page and are read by the screen reader each time a page loads, even when the pages are different. To help solve this problem we implemented two solutions. First we gave each dialog box a distinct name which would be read by the screen reader before any other content. Second, we did not put our main navigational links at the top of any of the dialog boxes.

In addition to the research done by AFB, Kara Pernice and Jakob Nielsen have written an extensive manual on building accessible websites called "Beyond ALT Text" [9]. In this paper Pernice and Nielsen recommend that website text should not be smaller than 11 point font. When we designed our interface for the *Interacive*_SIGHT system we used 16 point font throughout the interface. 16 point fonts allow visually impaired users working with a screen magnifier to read the text easily without it becoming distorted due to magnification.

There was one final layout consideration when we designed our interface: button placement. This design aspect was considered primarily for the benefit of those visually impaired users who use screen magnifiers. When using a screen magnifier the user is often unable to see the entire screen all at once and with large windows this can make it hard to navigate around the site because not all the navigational buttons can be seen at once. The first step to meeting this problem was to use dialog boxes that were well-sized, while not full-screen windows. Once we decided on the window size we then determined to layout all our buttons in a linear fashion. Depending on the window we either chose a vertical layout or a horizontal layout. A linear layout allows a person using a screen magnifier to move effortlessly across all the buttons. Those using screen magnifiers should be able to see the entire window and all of the buttons at once (unless significantly magnified), enabling them quickly to choose the one they want and move on.

3.2 Navigation Elements

The navigational portion of the interface furthers the system's accessibility by adding greater functionality to the layout elements. As we researched accessible web-designs both the AFB's article [9] and the paper

“Testing the Effects of Web Usability Criteria for Vision Impaired Users” [11] by Barbara Leoprini and Fabio Paterno noted how visually impaired persons use the TAB key to navigate through web pages. Users who defer to this kind web surfing are called “searchers” for the way they search for particular pieces of information. In the AFB article author Dr. Elaine Gerber recommends that a site’s TAB order should be logical. A user should be able to anticipate where the TAB order is going next; it should not jump erratically from one position in the window to another. Leoprini and Paterno’s paper affirms Gerber’s view and suggests that web-designers set the TAB order so that higher priority elements are tabbed to first. Following these suggestions we set TAB orders for each of our dialog boxes.

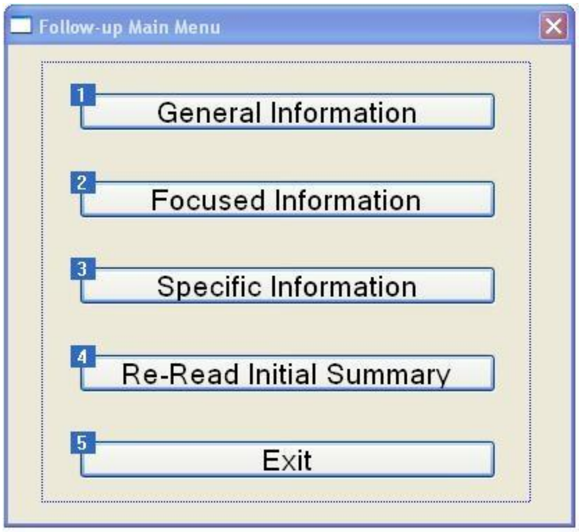
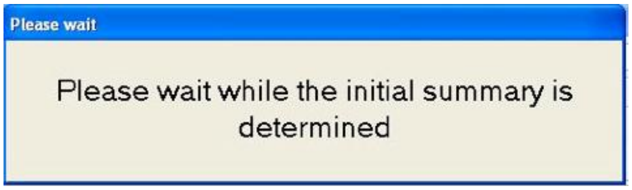


Figure 1: The TAB order of the Follow-up Main Menu

The more important navigational buttons (those that users will use more frequently) received a higher priority in the TAB order (Figure 1), this way users do not have to listen to all the buttons being read just to get the one they want. Setting the TAB order as we did allows users to efficiently move through all buttons in a methodical manner.

The other primary mode of navigation that our interface has is the use of keyboard accelerators. Keyboard accelerators are key combinations that are connected to menu items or buttons, so that when the keyboard combination is pressed it is as though the user clicked on a particular button (e.g. ALT-G for the “**G**eneral Information” button). Each of our buttons has a keyboard accelerator associated with it. Users can press a keyboard combination to exit a window, to access a different section of the interface, or to ask the system for more information on the graphic. Also, to aid our users in memorizing the keyboard accelerators, buttons that do the same thing from window to window have the same key combinations. Further, pressing F1 will open a help window that has a listing of all the keyboard accelerators for the current window. A logical TAB order and keyboard accelerators provide our users with tools to move them through *Interactive*_SIGHT’s interface without having to wait for a screen reader to read everything within the window.



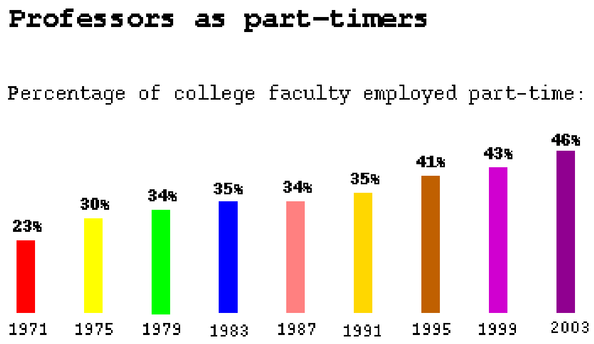
Please wait while the initial summary is determined

Figure 2: Please Wait dialog box

4. Interface Overview

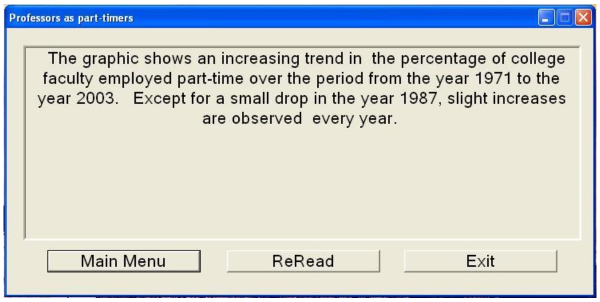
The following portion of the paper is a brief walk-through of how the interface works.

When a screen reader encounters a bar chart, such as the one in Graph 2, within a web-page it will notify the user by stating: “This is a bar chart, press CONTROL-Z to determine the caption.” If the user presses CTRL-Z a new dialog box labeled “Please Wait” will appear on the screen (see Figure 2).



Graph 2: Professors as part-timers

Soon after this dialog box has appeared, the screen reader will read the box’s textual content to the user. Meanwhile, the system is analyzing and determining the caption of the bar chart. When the caption is obtained the “Please Wait” dialog box automatically closes, and the initial summary dialog box appears on the screen (see Figure 3). As Figure 3 shows, the initial summary dialog box has a title at the top of it. This title is the title of the graph that was analyzed – in this case, it is the title of the bar chart in Graph 2. We placed the title here to help orient our users and remind them of the graph being analyzed. The initial summary dialog box contains the overall underlying message of the selected bar chart as inferred by the *Interactive*_SIGHT system [7].



The graphic shows an increasing trend in the percentage of college faculty employed part-time over the period from the year 1971 to the year 2003. Except for a small drop in the year 1987, slight increases are observed every year.

Figure 3: Initial Summary dialog box with the overall message of the bar chart.

The initial summary window is the access point to the rest of the interface. From this window the user can go to the main menu, have the initial summary re-read to them, or close the interface altogether. For the sake of this overview let us suppose the user selects the "Main Menu" button. As soon as the user clicks that button the initial summary dialog box will close and the dialog box named "Follow-up Main Menu" will appear on the screen. Before we describe the workings of this window, it is important to notice the layout elements discussed earlier (Section 2.1). Particularly take note of the linear arrangement of the buttons and the size of the text (see Figure 4).

There are many options presented to the user at this point. This overview will summarize the general function of each button. If a user presses the "General Information" button the "Follow-up Main Menu" disappears and the "General Summary" window opens.

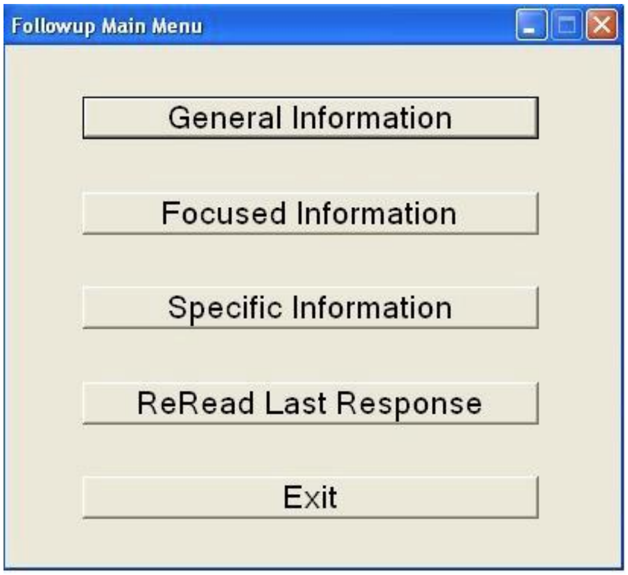
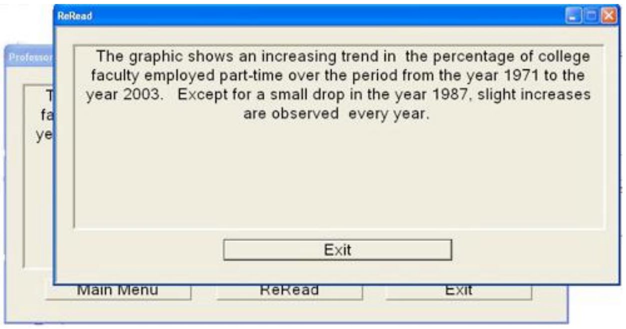


Figure 4: Follow-up Main Menu

The "General Summary" window looks almost exactly like the initial summary window seen in Figure 3. There are, however, two differences. The text displayed in the "General Summary" dialog box is not the same as in the initial summary dialog box; it is updated text which gives more detail on the information graphic that has been analyzed. The other difference is the "General Summary" window has an added button called "More General." When selected, the "More General" button causes the system to generate further general information about the bar chart and updates the current text with what is generated.

If we go back to the "Follow-up Main Menu" we see several other buttons. At this point the user may realize that each window so far has had a "ReRead" or "ReRead Last Response" button. The "ReRead" button basically performs the same action whenever it is pressed. If pressed the current window will not close, but a "ReRead" window will open over top of it (see Figure 5). The "ReRead" window will display the last summary that was generated. For example, if the user presses this button in the "General Summary" window the "ReRead" window will open and display the most recently read general information. The re-read function is designed to ensure that our users adequately hear the summary; the computer generated voices used by screen readers can sometimes be hard to understand or may mispronounce certain words [8]. Also, it can be used to help users decide whether they want more information or if they have heard enough and are ready to exit the program.

The next route we will take from the "Follow-up Main Menu" is through the "Focused Information" button. When a user clicks this button the current menu window closes and a new menu window opens. The new window is the "Focused Follow-up Menu" shown in Figure 6.



The graphic shows an increasing trend in the percentage of college faculty employed part-time over the period from the year 1971 to the year 2003. Except for a small drop in the year 1987, slight increases are observed every year.

Figure 5: Re-Read window

Same as the "Follow-up Main Menu", the "Focused Follow-up Menu" is a directory of options. As shown in Figure 6 there are four buttons that provide focused information about the graph; they are labeled: "Overall Change in Trend," "Max and Min Changes," "Stats about the Chart," and "Periods with unusual drops." Examining this dialog box also reveals a gap in between the first four buttons and the rest of the buttons. This gap is provided so that a fifth button can be placed there. The "Focused Follow-up Menu" is designed to generate the applicable buttons based on the specific graph being analyzed. Different graphs allow for different focused options, therefore, to maintain a manageable number of buttons, and to avoid user frustration, only the available/applicable buttons are presented to user. This design again aligns with the accessibility guidelines we have implemented thus far.

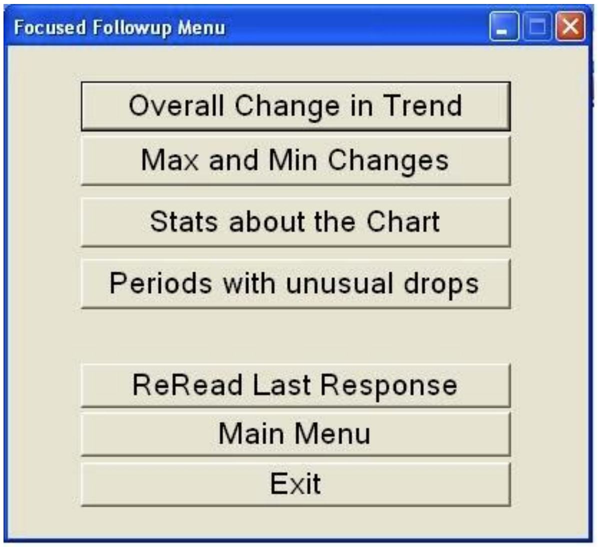
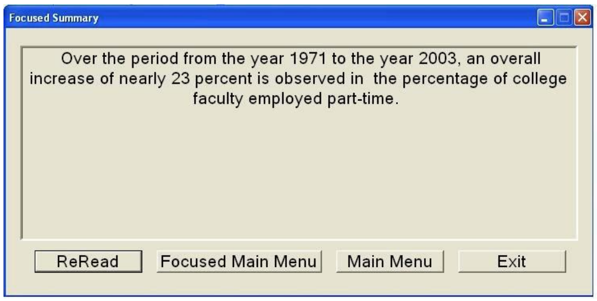


Figure 6: Focused Follow-up Menu window

Pressing "Overall Change in Trend" will generate a summary about the average change shown in the graph. For example, if the graph compared the average price of computer prices in the past tens years and the prices were gradually decreasing, then the "Overall Change in Trend" summary would talk about this overall decreasing trend. Pressing "Max and Min Changes" will give the user the largest and smallest changes. Pressing "Stats about the chart" outputs a summary of all different aspects of the graph; it may include the largest and smallest values in the graph. Finally, the "Periods with unusual drops" would display a summary that notes any drastic drops in the graph.

Each of these four buttons takes the user to the same place, but produces a different focused summary [2]. All four of these buttons close the current menu and open the "Focused Summary" dialog box (see Figure 7).



Over the period from the year 1971 to the year 2003, an overall increase of nearly 23 percent is observed in the percentage of college faculty employed part-time.

Figure 7: Focused Summary dialog box

The "Focused Summary" window closely resembles the other output windows. The "Main Menu" button takes the user back to the "Follow-up Main Menu" window and the "Focused Main Menu" button takes the user to the "Focused Follow-up Menu".

Now the final button to push on the "Follow-up Main Menu" is the "Specific Information" button. The "Specific Information" section of the interface enables the user to get familiar with the minute aspects of the graph they are examining. Consequently, the user interface could be very complex if not designed well.

When a user clicks the "Specific Information" button, a new dialog box will appear with the following menu buttons: "Values of All Bars in Graph," "Values of All Bars in Period," "Values and Rank of Single Bar," "Difference Between Two Bars," "ReRead Last Response," "Main Menu," and "Exit" (Figure 8).

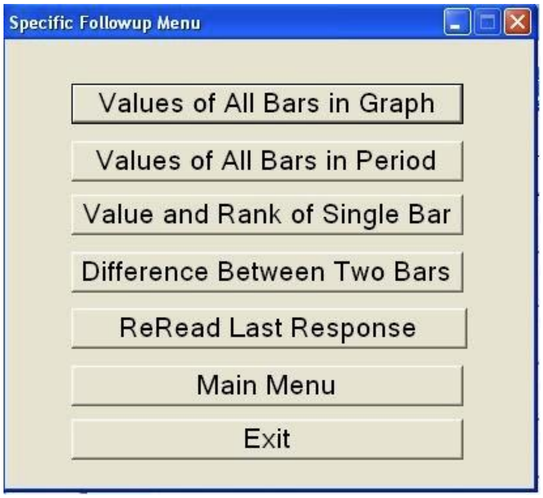

Specific Followup Menu
Values of All Bars in Graph
Values of All Bars in Period
Value and Rank of Single Bar
Difference Between Two Bars
ReRead Last Response
Main Menu
Exit

Figure 8: Specific Follow-up Menu window

At first this dialog box may seem overwhelming. There are many different options that all do very different tasks. The process of hearing and selecting a choice is simplified, however, by the features discussed earlier (TAB navigation and uniformity of layout). The user can easily cycle through the options as many times he or she wants want before selecting one.

Selecting the first button, “Values of All Bars in Graph” will immediately close the menu window and a “Generating follow-up information” (Figure 9) dialog box will appear. Once the graph’s detailed summary is realized a “Specific Summary” dialog box will appear with the resulting caption (see Figure 10).

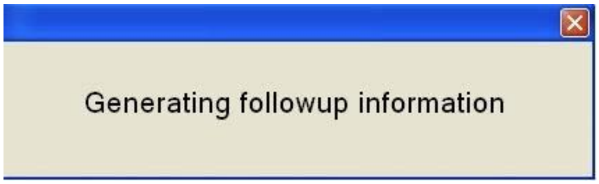
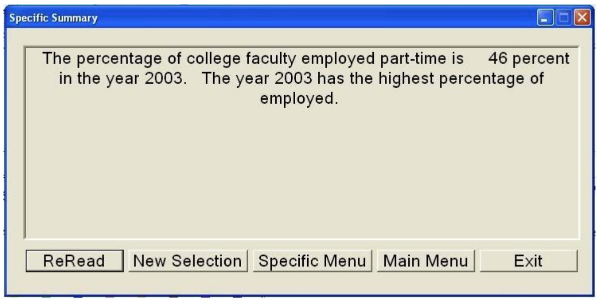

Generating followup information

Figure 9: Generating follow-up information window

The summary will include very detailed information about the entire graph, possibly highlighting one bar or another as the highest or lowest. The screen reader will first read the name of the dialog box, then the summary and finally the buttons available. After the user listens to the summary, they have several options. They may choose to hear the summary read again. The user may also go back to the “Specific Follow-up Menu” so that they can receive different specific information. In the case that the user decided on learning specific information about the graph before general or focused information there is a “Main Menu” button that will bring the user back to the first menu. As in all the other dialog boxes there is an “Exit” to close down the entire system.

By clicking on the next “Specific Information” button, “Values of All Bars in Range,” a dialog box full of radio buttons appears on the screen. The radio buttons are broken up into two groups. The first group is used to select the first bar in the desired period, and the second group is for selecting the second bar in the period. Each of the radio buttons corresponds to a bar in the current graph. So, for example, if the bars labels were years then the buttons might have dates like 1999 or 2003 next to them (Figure 11).



The percentage of college faculty employed part-time is 46 percent in the year 2003. The year 2003 has the highest percentage of employed.

Figure 10: Specific Summary window

We decided to use radio buttons for this window to avoid any confusion about which bar in the chart was being selected. Since in a group of radio buttons only one can ever be selected at a time, the user could never select more than one bar. After the user chooses the bars they would like they can then press the “Submit” button present and after analyzing the graph, the system will display the specific information about the range of bars queried. Using the same dialog box style as earlier makes it easy for the user to know how to navigate around the window. The only difference with this window is there is one extra button. The button called “New Selection” allows the user to select a new range of bars without going through the menu again.

Although the next two buttons in the “Specific Follow-up Menu” (“Value and Rank of Single Bar” and “Difference Between Two Bars) differ from each other and from the last button discussed, the process is exactly the same. Both buttons, when pushed, open a dialog box with an array of radio buttons. The “Value and Rank of Single Bar” button only asks the user to select one bar as Figure 12 shows, and the “Difference Between Two Bars” button brings the user through a process entirely the same as the “Values of All Bars in Range” button did.

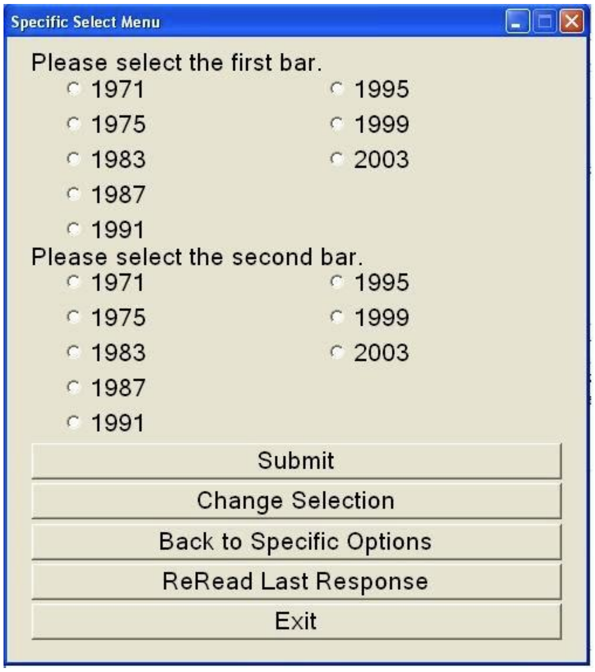


Figure 11: Specific Select Menu window when choosing two bars

The robustness of the “Specific Information” portion of the interface protects blind and visually impaired users from confusing situations. For example, if a user chose the “Difference Between Two Bars” button the user would be asked to select two bars in the graph. If only one bar was selected, the underlying system would not crash; however, the resulting textual response would contain nonsense. If the user interface did not protect against this, the user could wonder what they (user is not plural) did wrong or think that the graph is incorrect. The way the user interface is designed prevents users from pressing the "Submit" when selecting the bar unless they have checked a radio button from all the required fields. Thus, a user cannot move forward in the selection process without completing the current step.

In addition, notice that although the layout of the "Specific Information" section of the interface is more involved than any other area of *Interactive*_SIGHT's interface, it still maintains the key features of a linear button layout. The uniformity of the button placement is also maintained from window to window.

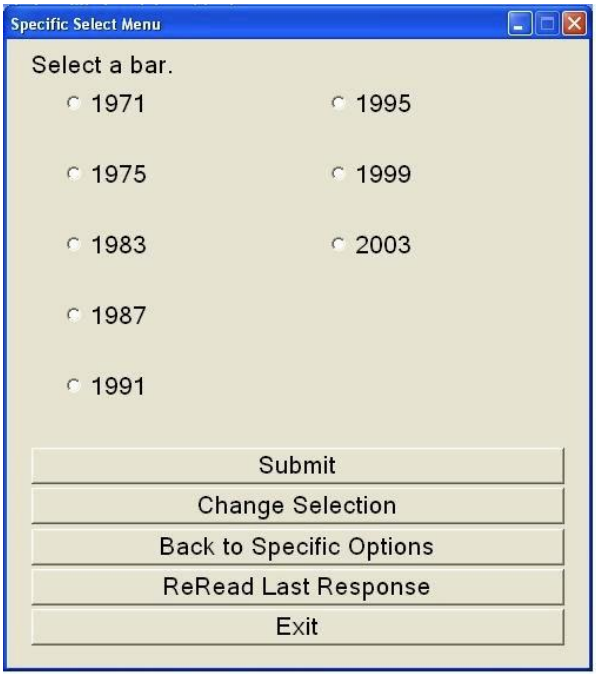


Figure 12: Specific Select Menu window when choosing one bar

The "Specific Information" interface simplifies a process that could be complex. Using fail-proof selection methods like radio buttons and uniformity in the button layout, getting specific facts about graphs is a logical and sequential process that allows the user to access the information that they want without frustration.

5. User Evaluations

User evaluations of the overall *Interactive*_SIGHT system were conducted with four visually impaired persons by a colleague at the University of Delaware [12]. Two of the users were not completely blind and used screen magnification. The other two were blind and used the JAWS screen reading software. After being shown how to use the *Interactive*_SIGHT system, the participants were presented with three different information graphics and asked to answer three questions about each of these graphs. The questions had been identified by sighted users in a separate study [12]. The participants were then asked to evaluate the system both on the ease of use of the interface as well as the usefulness of the content being disseminated.[1]

The research we did in order to create an accessible system proved effective. When asked "On a scale of 1-10 with 1 being extremely difficult and 10 being extremely easy, how easy was it to use the system?", the users gave the system an average of an 8.25 for ease of use. They thought that the layout of the dialog boxes was well designed and they liked that they could navigate using different methods. One user remarked "Your system was fantastic! It made it very easy for me to understand the graph and interpret the information. I did not have to worry about navigating around the screen because all of the information I needed was printed right there in front of me." The same user also gave some helpful feed-back regarding the "Specific Information" section of our interface. The user noted that when using screen magnification the radio buttons were hard see. They suggested that we make them a little bigger and possibly change their color so that they are accented in the dialog box. Overall, all four subjects were satisfied with the effectiveness of the *Interactive*_SIGHT system.

6. Conclusion

*Interactive*_SIGHT is designed to allow visually impaired and blind persons to have as full of an experience with information graphics as a sighted user given their disability. This system opens that door in a way that is useful and simple. The *Interactive*_SIGHT program gives the user almost total control over what information they want to know about a particular graph. It does not run continuously like certain programs do, thus reducing the lag on the user's computer due to the intense task of natural language processing. Also, it does not seek to recreate the image of the graph in the user's mind using textual descriptions. Instead, *Interactive*_SIGHT seeks to present the information that most sighted users would naturally gain when they viewed the image [7]. It aims at shrinking the internet accessibility divide that stands in the way of many visually impaired and blind users.

Currently, the *Interactive*_SIGHT does have some limitations. The system only has the capability of translating simple bar graphs into text, though it is currently being expanded to handle line graphs and grouped bar charts. Further, not even all simple bar graphs are accepted by the system. The text contained in the graphs

[1] The content being disseminated to the user is not the focus of this paper, and therefore, we will not discuss those results here, but they can be found in [12].

has to be a certain font for the system to correctly analyze it. Research is underway to help eliminate these image processing barriers. In addition, the *Interactive*_SIGHT system was designed to work on Internet Explorer and to be used with JAWS screen reading software. The methodology should apply with other web browsers, but further development work would be necessary. Alternative screen readers should currently work, though they have not been tested.

The applications for this software are varied. Many different websites use graphs to provide their readers with information. Political, sports, business, and medical sites all employ information graphics. This software would allow visually impaired and blind persons in all stages of life to understand more information than before. Both the college student doing research and the elderly person checking political polls would be aided by this software. In all, the Interactive_SIGHT software makes the world of information graphics more accessible to the visually impaired and blind. It takes their internet experience one step closer to that of the sighted internet user.

Acknowledgments

This material is based upon work supported by the NIDRR under Grant No. H133G080047.

This paper endeavors to focus on the work completed by David Oliver, an undergraduate computer science student at Millersville University, under the supervision of Dr. Stephanie Elzer. This work is part of a larger collaborative project between Millersville University and the University of Delaware, and much of the underlying system has been developed by researchers at the University of Delaware.

References

[1] S. Elzer, E. Schwartz, S. Carberry, D. Chester, S. Demir, & P. Wu. A Browser Extension for Providing Visually Impaired Users Access to the Content of Bar Charts on the Web. A software overview, *Proc. of International Conference on Web Information Systems (WEBIST),* 2007, p. 59-66.

[2] Seniz Demir, Sandra Carberry, and Stephanie Elzer. Effectively Realizing the Inferred Message of an Information Graphic. *Proceedings of Recent Advanced in Natural Language Processing (RANLP)*, 2007, p. 150-156.

[3] J. Nielsen. Beyond Accessibility: Treating Users with Disabilities as People. *Alertbox,* 2001. <http://www.useit.com/alertbox/20011111.html>.

[4] 1998 Amendment to Section 508 of the Rehabilitation Act. United States Congress, 1998. <http://www.section508.gov/index.cfm?FuseAction=Content&ID=3>

[5] J. Abramatic, C. Adams, A. Anderson, et. al. Web Content Accessibility Guidelines (WCAG) 2.0. 2008. <http://www.w3.org/TR/WCAG20/>.

[6] R. Bailey, C. Barnum, J. Bosley, et. al. *Research-Based Web Design & Usability Guidelines.* U.S. Government Printing Office Official Editions, 2006.

[7] Elzer, S., Carberry, S., Zukerman, I., et. al. *A probabilistic framework for recognizing intention in information graphics*. In Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence (IJCAI), 2005, p. 1042–1047.

[9] E. Gerber. Surfing By Ear: Usability Concerns of Computer Users Who Are Blind or Visually Impaired. *AccessWorld*, *3*(1), 2002.

[9] E. Gerber. Conducting Usability Research With Computer Users Who Are Blind or Visually Impaired. A research overview, *Proc. of 17th Annual International Conference of California State University Northridge (CSUN) "Technology and Persons with Disabilities"*, Northridge, CA, 2002.

[10] K. Pernice, J. Nielsen. *Beyond ALT Text: Making the Web Easy to Use for Users with Disabilities.* Fremont, CA: Nielsen Norman Group, 2001.

[11] B. Leporini, F. Paternó. Testing the Effects of Web Usability Criteria for Vision Impaired Users. *ACM*, 2004.

[12] S. Demir. Sight for Visually Impaired Users: Summarizing Information Graphics Textually. PhD Dissertation, University of Delaware, Spring 2010.