The Wiimote with SAPI: Creating an Accessible Low-Cost, Human Computer Interface for the Physically Disabled

Aqil Azmi†, Nawaf M. Alsabhan†† and Majed S. AlDosari††

†King Saud University, Riyadh, Saudi Arabia
††King Abdulaziz City for Science and Technology, Riyadh, Saudi Arabia

Summary
In this paper we report on an inexpensive device that helps a person with physical disability to interact with the Web. The human computer interface is made through a head tracking pointer. The head tracking pointer consists of the Wii controller (Wiimote), a low cost and readily available game controller, and infrared (IR) LED. The Wiimote is used to detect the position of the user's head. The Windows API (Application Programming Interface) has been used to move the pointer to the proper position according to the position given by the Wiimote. Microsoft SAPI (Speech Application Programming Interface) has been used to input commands via speech. The browser has been developed to provide a simple easy-to-use graphical user interface (GUI) to be used by the physically disabled. The total cost of the system is less than $50, much less than any commercial product with the same functionality. The system received high marks in the field test.

Key words:
Accessibility, physical disability, input and interaction technology, object tracker, Wii Remote.

1. Introduction

In less than ten years, the Internet has become indispensable. Consequently, physically handicapped people require an efficient, user-friendly Web browser application with innovative technologies, and most importantly an inexpensive software and hardware interface that ease the process of surfing the Web and which is accessible to general users. New user interfaces have been developed that allow direct control of pointers using body gestures [1]-[4]. Previously available technologies, for users with physical disabilities, are developed with expensive technologies, making them inaccessible to general users. In this paper, a new innovative head tracking pointer system for people with major defects, such as people with spinal injuries, was introduced and coupled with simple user-friendly Web browser software.

Gaze or head pose is a potentially powerful and intuitive pointing cue if it can be obtained accurately and unobtrusively. Eye movement is an important topic of study in evaluating user interface performance and potentially as an input cue [5]. However, the dynamics of involuntary human eye movements limit its accuracy for independent fine pointing control, although it can provide a coarse reference frame [6]. In this paper, we focus on head tracking, which has been a topic of active interest to many people in the computer vision community [7], and which appears to be less affected by involuntary control movements than eye gaze. Several authors have built prototypes which incorporate head pose tracking into user interface controls [8]-[11].

The objective of this paper is to propose simple user-friendly head pointer system and Web browser software for people with physical disability. The goal of the head pointer system and Web browser is to ease the process of surfing the Internet. To validate the viability of the proposed new head pointer system and Web browser, the Web browser software and the head tracking pointer system will be built and tested.

In the following section we discuss related work. Some assistive technologies and adaptive strategies are discussed in Section III. In Section IV, we describe both hardware and software components and how the system is put together or designed. Then, we go on to discuss the technologies used in the implementation and describe how the system is used in Section V. The paper concludes with the present status of the project and a description of further work.

2. Related Work

The proposed head pointer system and Web browser has a foundation in previous works. The MBROLA, BrailleSurf, Access Gateway and NEDO projects were designed for handicapped people, such as blind people and people with speech impairment [12].

- The MBROLA Project
  The aim of the MBROLA project, initiated by the Circuit Theory and Signal Processing (TCTS) Lab, is to obtain a set of speech synthesizers for as many languages as possible. A speech synthesizer takes a list of phonemes as input, together with prosodic information, to produce a speech sample. (It converts text to speech "TTS.")

Manuscript received December 5, 2009
Manuscript revised December 20, 2009

- BrailleSurf
  An Internet browser for users with vision impairment, which allows a simplified reading of the information available on the Web. BrailleSurf shows this information in a text form [13], [14]. This information then can be displayed on a Braille bar, or it can be spoken out by a speech synthesizer. The text also can be presented on the screen for people with low vision, and can be used to provide a fast review of the accessibility level of a Web site for visually impaired people.
- Access Gateway
  An online "browser within a browser" providing more control over how Web sites are displayed (regardless of which browser is being used). The Gateway's main purpose is to make the Web easier to access for people with a print disability (such as users with weak vision or dyslexia) [13]. It also translates Web pages into other languages and displays them as an image, so they can be displayed when the encodings are not supported by the user's browser.
- NEDO (New Energy Development Organization) IT Barrier-Free Project of disabled persons and other users. This project promotes the development of verification and evaluation experiments on a navigation system having portable user terminals that can be shared among all people, including the disabled, and which are particularly easy to operate [15].

3. Assistive Technologies and Adaptive Strategies

The proposed head pointer system and Web browser is intended to make use of assistive technology and adaptive strategies. An understanding of what assistive technologies and adaptive strategies are and what kinds are available will assist in the comprehension of the system and browser design.

Assistive technologies are products used by people with disabilities to help accomplish tasks that they cannot accomplish otherwise or could not do easily otherwise [16]. When used with computers, assistive technologies are sometimes referred to as adaptive software or hardware.

Adaptive strategies are techniques that people with disabilities use to assist in using computers or other devices [16]. For instance, someone who cannot see a Web page may tab through the links on a page as one strategy for helping skim the content.

Following is a *non-comprehensive* list of assistive technologies and adaptive strategies available [16].

- Alternative keyboards or switches
- Scanning software
- Screen reads
- Speech recognition
- Speech synthesis
- Tabbing through structural elements
- Text browsers
- Visual notification
- Voice browsers

4. System Synthesis

In designing the head pointer system prototype, Commercial-Off-The-Shelf (COTS) hardware and software were used. This helped in focusing on the functionality of the system. The head tracking pointer system is designed to be comfortable, simple, user friendly and most importantly cost efficient. In addition, it is designed to require a minimum effort from the user to interact with computer and the browser. To validate the viability of this project, a model system was built, implemented and tested.

An in-depth look at the hardware and software of the system is provided in the following sections, as is a discussion of the system as a whole.

4.1 Hardware Components

An extensive search for a hardware that can be used to sense the movement of a user keeping cost and reliability of system in mind. We decided to integrate the following two hardware components for the head-tracking pointer system: (1) Wiimote; and (2) IR Glasses. Below is further detail of both of these devices.

1) Wii Remote (Wiimote as it is commonly known*):* The dimensions of the Wiimote body are 148 mm long, 36.2 mm wide and 30.8 mm thick [17] (Fig. 1). It is considered to be a low-cost controller, costing less than $35.

The controller communicates wirelessly with the console (in our case a PC) via short-range Bluetooth radio, with which it is possible to operate up to four controllers as far as 10 meters (approx. 30 ft) away from the console. However, to utilize pointer functionality, the Wiimote must be used within five meters (approx. 16 ft) of the Sensor Bar [17], [18].

A main feature of the Wiimote is its motion sensing capability, which is particularly relevant to the proposed head pointer system because all such systems have some movement as a basis for their operation. The Wiimote has the ability to sense acceleration along three axes through the use of an ADXL330 accelerometer [17]-[20]. The Wiimote also features a PixArt optical sensor, allowing it to determine where the Wiimote is pointing [17], [18], [21]. Unlike a light gun that senses light from a television screen, the Wiimote senses light from the console's Sensor Bar, which allows consistent usage regardless of a screen's type or size. The Wiimote's image sensor [18], [21] is used to locate the Sensor Bar's points of light in the Wiimote's field of view. The Sensor Bar is about 20 cm (8 in) long and features ten infrared (IR) light-emitting-diodes (LEDs), five at each end of the bar [18], [21] (Fig. 2). Innovative users have used other sources of IR light as Sensor Bar substitutes, such as a pair of flashlights and a pair of candles [22]. Such substitutes for the Sensor Bar illustrate the fact that a pair of non-moving lights provides continuous calibration of the direction that the Wiimote is pointing and its physical location relative to the light sources.

Another feature of the Wiimote is its expandability through the use of attachments. In the future, attachments for the Wiimote could be designed that would expand or improve the capabilities of the proposed head pointer system.

Fig. 1: Wii Remote.

Fig. 2: Sensor Bar IR LEDs.

2) IR Glasses: The proposed head pointer system was designed to use a pair of LED Light Vision V2 Safety Glasses which come with two LEDs that emit bright white light (Fig. 3). These LEDs were replaced with IR LEDs, since the Wiimote can track sources of IR light (IR LED). With the LEDs replaced, the location of the user's head can be tracked accurately and the cursor can be made to respond to the user's head movements. This will result in full control of the curser. The only consideration left is how the user would click (see *4.2. Software Components).*

Fig. 3: LED Glasses.

4.2. Software Components

The system uses three software components. The first component is the Windows API (or formerly known as Win32 API), which is used to create and manage the user interface. For instance, in our case managing the position of the mouse pointer and its functions. The second component is the WiiLib, which is the library used to control and communicate with the Wiimote. The last software component is the SAPI (Speech Application Programming Interface) (Fig. 4).

SAPI, an interface designed by Microsoft, supports dynamic speech input and output, and is integrated in Microsoft's current operating system. With the application programming interface (API), it is possible to develop speech enabled applications without caring about the details of synthesis and recognition.

In general, all versions of APIs have been designed such that a software developer can write an application to perform speech recognition and synthesis by using a standard set of interfaces, accessible from a variety of programming languages [23]. In addition, it's possible for a third party company to produce their own Speech Recognition and Text–To–Speech engines, or to adapt existing engines to work with SAPI.

Typically, SAPI is freely redistributable component which can be shipped with the Windows application that is intended to use Speech technology. Many versions (although not all) of the Speech Recognition and Synthesis engines are also freely redistributable [23]. Since Windows is the most popular operating system throughout the world, this API was chosen to give the user of the proposed head pointer system the ability to click, right click and double click, using a predefined vocabulary.

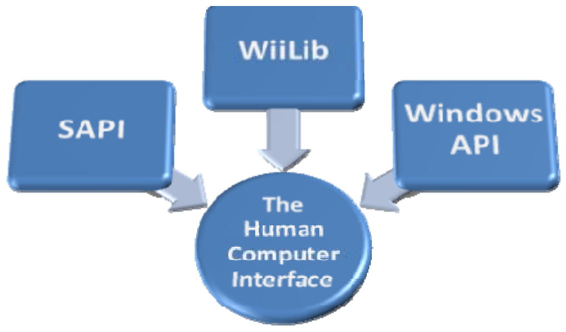

Fig. 4: Software Integration.

5. Implementation

Implementation of the proposed head pointer system was possible after completion of the system design. The technologies used in the implementation, as well as component integration and system use, are discussed in this section.

5.1 Used Technologies

For software writing we decided to use C# to write the desired classes, interfaces and the business logic, which must use .NET environment. Our system was built using some of Microsoft's development products (e.g., Visual Studio 2005).

1) C# (C Sharp): C# is a multi-paradigm object oriented Programming language developed from C and C++. It was created to run on the Microsoft .NET platform. C# was chosen based on our experience and preference. Other programming languages can be used. However, since both SAPI and WiimoteLib work either under or with the .NET framework, it is required to use a programming language supported by the .Net framework.

2) WebBrowser Class: Microsoft introduced the WebBrowser class (System.Windows.Forms), which enables applications to incorporate Web browsing capabilities [24]. It can be used, for example, to provide integrated HTML based user assistance or Web browsing capabilities in an application. Additionally, it can be used to add existing Web based controls to Windows Forms client applications. This control has several properties, methods, and events related to navigation. The following members can be used to navigate the control to a specific URL, move backward and forward through the navigation history list, and load the home page and search page of the current user [25].

- URL
- Navigate
- GoBack
- GoForward
- GoHome
- GoSearch

5.2 Initialization Set

For the set-up, a subject (or user) sits in front of the presentation monitor. The user must wear the IR glasses and make sure that the IR LEDs are working. The Wiimote IR camera next to the presentation monitor tracks the user's head activity.

A few seconds are taken initially to calibrate the particular user and their posture in this head tracking system. Then, the speech recognition functionality from SAPI is enabled to recognize a few words that we predefined to allow the user to click, right click, and double click. The Wiimote IR camera locates the position of the IR LEDs in 3D coordinates and sends them to the head tracking system Control Unit System (the application on the PC) via Bluetooth. The system Control Unit processes the information it receives from the Wiimote connected to it, and calculates head position in both vertical and horizontal in 3D coordinates. This information can be used in real time to convert 3D coordinates received from the IR camera to 2D coordinates of the screen. The result of this conversion is used in real time by the system to move the mouse pointer according to the given 2D coordinates.

The application was prototyped with the following properties:

- The Bluetooth is used to communicate with the Wiimote IR camera.
- Information is read continuously from the Wiimote IR camera.
- The information coming from the Wiimote is decoded to get the head coordinates in real time.
- The coordinates of the data stream coming from the Wiimote continuously are compared.
- If the subject says voice command "click," for instance, then a mouse click is made.

5.3 How to Use

As soon as the program runs, a small GUI form will pop up. This form consists of the following (Fig. 5):

- Bar indicator:
  Indicates the battery level of the Wiimote, which is retrieved from it using its API.
- Simple button:
  This button can be used to start the calibration, or calibration can be started just by clicking on the A button on the Wiimote. When the

calibration starts, a code segment will be executed to predefine the screen borders.

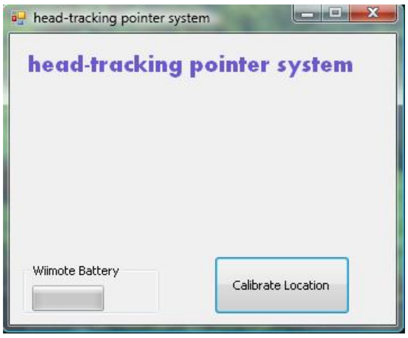

Figure 5: First Form.

Now, the user can interact freely with the computer via voice and head movements. With this ability to interact, the user freely can choose the browser that was developed with the HCI application to ease the process of browsing Web sites with the use of a virtual keyboard. This virtual keyboard considers the most frequently accessed Web pages and most frequently inputted terms (Fig. 6).

In order to test our system usability and get some feedback, users from different backgrounds, were invited to try it. All users found the system acceptable and most of them learned to use it in a minute. It is worth noting that most users were very pleased with the entire system.

Figure 6: Browser Form.

6. Conclusions and Future Works

One of the main ideas about this project was to develop a system that costs much less than any commercial products with the same functionality. Consequently, the total cost of this project results in less than $50, which is the price of the Wiimote (Fig. 1) and the IR glasses (Fig. 3).

An innovative system that combines head tracking and mouse input to allow a user to control a conventional GUI has been presented for people with major defects, such as people with spinal injuries. Both latency and obtrusiveness were found to be minimal in this system. Head tracking can be accurate, robust, automatically initialized and fast with this system, and the system can work efficiently on any current computer system. The developed browser 1) provides a virtual keyboard with large keys and font to make system use easier, and 2) provides a certain features, such as shortcut keys for the most frequently entered terms and most frequently visited Web sites.

To augment this system, new techniques and technologies for future work are being explored. Integrating some of the previously discussed assistive technologies and adaptive strategies may result in a better result or application. In future work, we will enhance this prototype and investigate how we can proceed towards mobile and robust use of this innovative technology.

Acknowledgment

The authors of this work would like to express gratitude and thanks to Dr. J.C. Lee, a HCI researcher at Carnegie Mellon University, for all his Wiimote projects. They were really the inspiration for this project. The authors are also grateful to B. Peek for providing the Wiimote library (WiimoteLib) on the Web, which was used for reading data from the Wiimote.

References

[1] A. Pentland, "Perceptual Intelligence," *Comm ACM*, Vol. 43, No.3, pp. 35-44, Mar. 2000.

[2] J. Crowley, J. Coutaz and F. Berard, “Things That See: Machine Perception for Human Computer Interaction,” *Comm ACM*, Vol. 43, No.3, pp. 54-64, Mar. 2000.

[3] IBM Almaden Research Center, "Creating computers that know how you feel," *Int. Business Machines (IBM) Corp.*, 2006.[Online]. Available: http://www.almaden.ibm.com/cs/BlueEyes/index.html. [Accessed: Oct. 10, 2009]

[4] M. Turk and G. Robertson, "Perceptual User Interfaces," *Comm ACM*, Vol. 43, No.3, pp. 32-34, Mar. 2000.

[5] R. Jacob, “What you look at is what you get: Eye movement-based interaction techniques,” *Proc. CHI '90* (1990), pp. 11-18.

[6] S. Zhai, C. Morimoto and S. Ihde, “Manual and gaze input cascaded (MAGIC) pointing,” *Proc. ACM CHI*, 1999, pp. 246-253.

[7] K. Schwerdt and J. L. Crowley, “Robust face tracking using color,” in 4th *IEEE Int. Conf. on Automatic Face and Gesture Recognition*, Grenoble, France, Mar. 2000, pp. 90-95.

[8] F. Berard, “The Perceptual Window: Head Motion as a new Input Stream,” in 7th *IFIP Conf. Human-Comput Interaction (INTERACT)*, 1999, pp. 238-244.

[9] M. Jones and T. Poggio, “Multidimensional Morphable Models,” *Proc. ICCV98*, 1998, pp. 683-688.

[10] R. Kjeldsen, "Head Gestures for Computer Control", *Proc. 2nd Int. Workshop on Recognition, Analysis Tracking Faces Gestures Real-time Systems, ICCV*, IEEE Press, 2001, pp. 62-67.

[11] ABiSee Inc., [Online]. Available: www.abisee.com. [Accessed: Oct. 20, 2009]

[12] T. Dutoit, X. Ricco, "Towards a Free Multilingual Speech Synthesis Software for the Vocally Handicapped," *Österreichische Gesellschaft für Artificial Intelligence*, vol. 20, Jul. 2001, pp. 36-38.

[13] “Software for people with disability,” [Online]. Available: http://www.ebility.com/links/software.php. Jul. 13, 2009. [Accessed: Sept. 18, 2009].

[14] Web Alternative Initiative. “Alternative Web Browsing.” [Online]. Available: http://www.w3.org/WAI/References/Browsing#1. Oct. 18, 2005. [Accessed: Sep. 10, 2009].

[15] New Energy and Industrial Technology Development Organization (NEDO). “IT Barrier-Free Project for Disabled Persons and Other Users.” [Online]. Available: http://www.nedo.go.jp/english/activities/1_sangyo/1/p04006e.html. [Accessed: Sept. 12, 2009]

[16] W3C, “How People with Disabilities Use the Web.” [Online]. Available: http://www.w3.org/WAI/EO/Drafts/PWD-Use-Web/#tools. May 5, 2005. [Accessed: Sept. 11, 2009]

[17] H. O. Wiggins, "Gesture Recognition of Nintendo Wiimote Input Using an Artificial Neural Network," Henry Owen Wiggins webpage, pp. 2, Apr. 17, 2008. [Online]. Available: http://sites.google.com/site/kindjie/GestureRecognitionofNintendoWiimoteI.pdf. [Accessed: Sept. 14, 2009].

[18] H. Filippi, “Wireless Teleoperation of Robotic Arms,” M.S. Thesis, Dept. Space Science, Luleå Univ. of Technol., Kiruna Helsinki, Finland, 2007, http://epubl.ltu.se/1653-0187/2007/079/LTU-PB-EX-07079-SE.pdf.

[19] C. Guo and E. Sharlin, “Exploring the use of tangible user interfaces for human-robot interaction: a comparative study,” in *Proc. 26th Annual SIGCHI Conf. on Human Factors in Computing Systems*, 2008, pp. 121-130, Florence, Italy.

[20] E. L. Wong, W. Y. F. Yuen and C. S. T. Choy, “Designing Wii controller: A powerful musical instrument in an interactive music performance system,” in *Proc Int Conf Advances in Mobile Computing and Multimedia*, New York: ACM, Linz, Austria, 2008,pp. 82-87.

[21] T. Schou , H.J. Gardner, “A Wii remote, a game engine, five sensor bars and a virtual reality theatre,” in *Proc. of the 19th Australasian Conf. on Computer-Human Interaction: Entertaining User Interfaces*, Adelaide, Australia, 2007, pp. 231–234.

[22] I. A. Suhardi, “Large Group Games With A Motion And Orientation-Sensing Game,” M.S. Thesis, Dept. Mathematics and Computer Science, Univ. of Bremen, Germany, 2008. [Online]. Available: http://www.abiamy.com/abiyasa/thesis/thesis-suhardi.pdf. [Accessed: Sept. 18, 2009]

[23] H. Shi and A. Maier, “Speech-enabled Windows application using Microsoft SAPI,” *International Journal of Computer Science and Network Security*, Vol. 6, No. 9A, pp. 33-37, Sep. 2006.

[24] H. Deitel and P. Deitel, *Visual Basic: How to Program*, Prentice Hall, 2005.

[25] MSDN. “WebBrowser Class.” [Online]. Available: http://msdn.microsoft.com/en-us/library/system.windows.forms.webbrowser.aspx. 2009. [Accessed: Sep. 12, 2009].

Aqil M. Azmi received the B.S.E degree from Univ. of Michigan, Ann Arbor and the M.Sc and Ph.D from Univ. of Colorado, Boulder. Currently he is an assistant professor at the Dept of Computer Science, King Saud University. His research interests include application of computers in religious studies, Arabic language processing, HCI and algorithms in general.

Nawaf M. AlSabhan received the B.Sc degree in Computer Science from King Saud University in February, 2009. Currently, he is an academic researcher at King Abdulaziz City for Science and Technology (KACST). His research interests include artificial intelligence and pattern recognition, ubiquitous computing, HCI and robotics. He is a member of ACM.

Majed S. AlDosari received the B.Sc in Computer Science from King Saud University, in February, 2009. He has been a software developer at King Abdulaziz City for Science and Technology (KACST) since June, 2009. His research interests include artificial intelligence, algorithm design, HCI and robotics.