LOUGHBOROUGH UNIVERSITY

DOCTORAL THESIS

# An investigation into gaze-based interaction techniques for people with motor impairments

*Author:*
Howell Owen ISTANCE

*Supervisor:*
Dr. Peter Alan HOWARTH

*A thesis submitted in fulfillment of the requirements for the degree of Doctor of Philosophy*

*in the*

Environmental Ergonomics Research Group
Loughborough Design School

January 27, 2017

LOUGHBOROUGH UNIVERSITY

# *Abstract*

Loughborough Design School

Doctor of Philosophy

**An investigation into gaze-based interaction techniques for people with motor impairments**

by Howell Owen ISTANCE

The use of eye movements to interact with computers offers opportunities for people with impaired motor ability to overcome the difficulties they often face using hand-held input devices. Computer games have become a major form of entertainment, and also provide opportunities for social interaction in multi-player environments. Games are also being used increasingly in education to motivate and engage young people. It is important that young people with motor impairments are able to benefit from, and enjoy, them.

This thesis describes a program of research conducted over a 20-year period starting in the early 1990's that has investigated interaction techniques based on gaze position intended for use by people with motor impairments. The work investigates how to make standard software applications accessible by gaze, so that no particular modification to the application is needed. The work divides into 3 phases. In the first phase, ways of using gaze to interact with the graphical user interfaces of office applications were investigated, designed around the limitations of gaze interaction. Of these, overcoming the inherent inaccuracies of pointing by gaze at on-screen targets was particularly important. In the second phase, the focus shifted from office applications towards immersive games and on-line virtual worlds. Different means of using gaze position and patterns of eye movements, or gaze gestures, to issue commands were studied. Most of the testing and evaluation studies in this, like the first, used participants without motor-impairments. The third phase of the work then studied the applicability of the research findings thus far to groups of people with motor impairments, and in particular,the means of adapting the interaction techniques to individual abilities.

In summary, the research has shown that collections of specialised gaze-based interaction techniques can be built as an effective means of completing the tasks in specific types of games and how these can be adapted to the differing abilities of individuals with motor impairments.

# *Acknowledgements*

There are many people to whom I am very grateful for the input and support they have provided, both direct and indirect, to the research described in this thesis.

First, I want to thank my friend and colleague Peter Howarth who has supervised this PhD. for his persistance and encouragement to complete this. It was Peter who obtained the original pupillometer at Loughborough University that was used as the eyetracker in the early work. Without it, it is quite possible that none of the research described here would have ever taken place.

I wish to thank my PhD students Steve Vickers and Richard Bates and acknowledge the huge contribution each has made to the work described in the thesis. Both are very talented individuals and I have learned a lot from working with them.

I'd like to thank Christian Spinner, and particularly acknowledge his work as my MSc project student on the second research output. I'd also like to express my thanks to the other students from De Montfort University, the University of Tampere and Loughborough University, who have participated in or contributed to the work in different ways. These are Santtu Mansikkamaa, Lauri Immonen, Thiago Chaves, Nasmi Ali, Sean O'Connor, Ikena Okeke, Chris Schrimpl, Simon Paterson, Darren O'Connor and Simon Layton..

A very big thanks is due to Jane Lynch and Helen Newton, both staff at Ash Field Academy in Leicester for contributing their enthusiasm and expertise in teaching students with special needs so willingly to the projects we undertook with them. A special thanks go to all of the Ash Field students who helped with the evaluation work so enthusiastically.

I wish to extend my gratitude to Kari-Jouko Räihä for giving me the opportunity to come and work in the University of Tampere, and to live in Finland. He also led the COGAIN EU Framework 6 Network of Excellence that had a profound impact on my work, and which I enjoyed enormously. I'd like to thank Alastair Gale, who I learned to know through COGAIN for being the second supervisor of this thesis.

Lastly, my special thanks to my colleague, co-author and soul mate Aulikki Hyskykari, who has made me realise how much fun life and work can be, and what the work-life balance really means.

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

## 1.1 Background

At the beginning of the 1990's, eye tracking systems generally either formed part of bespoke communication systems primarily aimed at people with motor impairments and communication difficulties, or were research tools intended for the detailed study of eye movements in different contexts. Since then, the techniques for measuring eye position have changed remarkably little, but instead have been refined. The applications of eye tracking however have expanded enormously. Often these use gaze position rather than eye position. Gaze position refers to the location where someone is looking, either in relation to some kind of display surface, or in relation to objects in the real world, and *gaze tracking* refers to the measurement of these positions. *Eye tracking*, in contrast to gaze tracking, refers to the measurement of eye position with the head as a frame of reference.

The interest in, and application of, gaze tracking looks set to continue expanding with the advent of wearable displays, which can be equipped with eye tracking capability. There are, however, issues still to be overcome with the provision of mobile image processing capacity, and of power supply to the cameras. These near-eye displays may be see-through, allowing data to projected onto the view of the world through the display, or they may not be as in the case of head-mounted VR systems. In each case, however, gaze tracking offers the opportunity for both hands-free interaction with the displayed information, and for continuous tracking of the objects of overt visual attention, either in the real world or in the virtual world.

Another important driver in the expansion of applications of gaze tracking is the recent advent of low cost reliable eye trackers from commercial manufacturers, which have a retail price in the region of hundreds of euros, rather than tens of thousands of euros. An important consequence of this will be the availability of gaze tracking to support access to computers for people with different types or motor impairments in far greater numbers than has been the case so far. Whilst accessibility for people with severe and complex needs can be gaze-based, for those with less severe and complex needs an alternative approach is to use gaze-assisted software to complement mainstream input devices, such as the mouse and keyboard.

## 1.2 Aim of the research

The research described in this thesis covers the author's contribution to the field of human-computer interaction during the period from the mid 1990's to the present time. **The overall aim of the research was to investigate**

**gaze-based interaction techniques suitable for people with motor impairments**. The investigation began with the view that standard office applications could be made more accessible by means of gaze-based interaction techniques. This focus shifted subsequently to studying such techniques with multi-player games and virtual communities, again for people with motor impairments. A explanation of the terms 'gaze-based', 'interaction techniques' and 'motor impairments' is given in Appendix A.

The eight research outputs are grouped into three phases that are chronological in order (see Figure 1.1). Each phase has an associated research question, and this thesis will examine the extent to which the outputs have contributed to the answering of each of the questions.

## 1.3 Phases of the research

### 1.3.1 Gaze beyond personal communication systems, enabling access to unmodified software through gaze interaction

Until the early 1990's, research in the field of interaction with computers by gaze was directed particularly towards text entry via different on-screen (or soft) keyboards. A period then followed of exploring how gaze could be used in a broader context. The approach adopted by the author and his colleagues in this phase of the research was to examine how gaze could be used by motor impaired users to access standard office applications (word processors, spreadsheets, web browsers). This was by means of 'middleware' designed and built in the research projects described in this thesis. See Appendix A for an explanation of 'middleware'. Three of the research outputs from this phase are presented in Section 2.

### 1.3.2 Novel gaze-based interaction techniques for multiplayer virtual worlds and games

The focus of the research moved from office applications to facilitating interaction with immersive multi-player games and social environments, such as 'Second Life'. The goal of this was to enable people with disabilities to take part in activities without the fact they were impaired in the real world being apparent to others in the virtual world. As much of the emphasis was on speed of interaction, techniques that rely on dwell-time (triggering an event by an extended stare at a location) were found to be limited. Consequently, gaze gestures (deliberate patterns of eye movements) were investigated as interaction techniques for common tasks in these virtual environments. Three of the research outputs from this period are presented in Section 3.

### 1.3.3 Adapting gaze interaction techniques for motor-impaired users

Although the target group for the research has been motor-impaired users, evaluative testing in the research so far had largely (although not exclusively) used able-bodied participants. To address this deficiency, a collaboration began between Ash Field School in Leicester (now Ash Field Academy) and the research group at De Montfort University. Ash Field is a special needs school with approximately 120 students all with motor

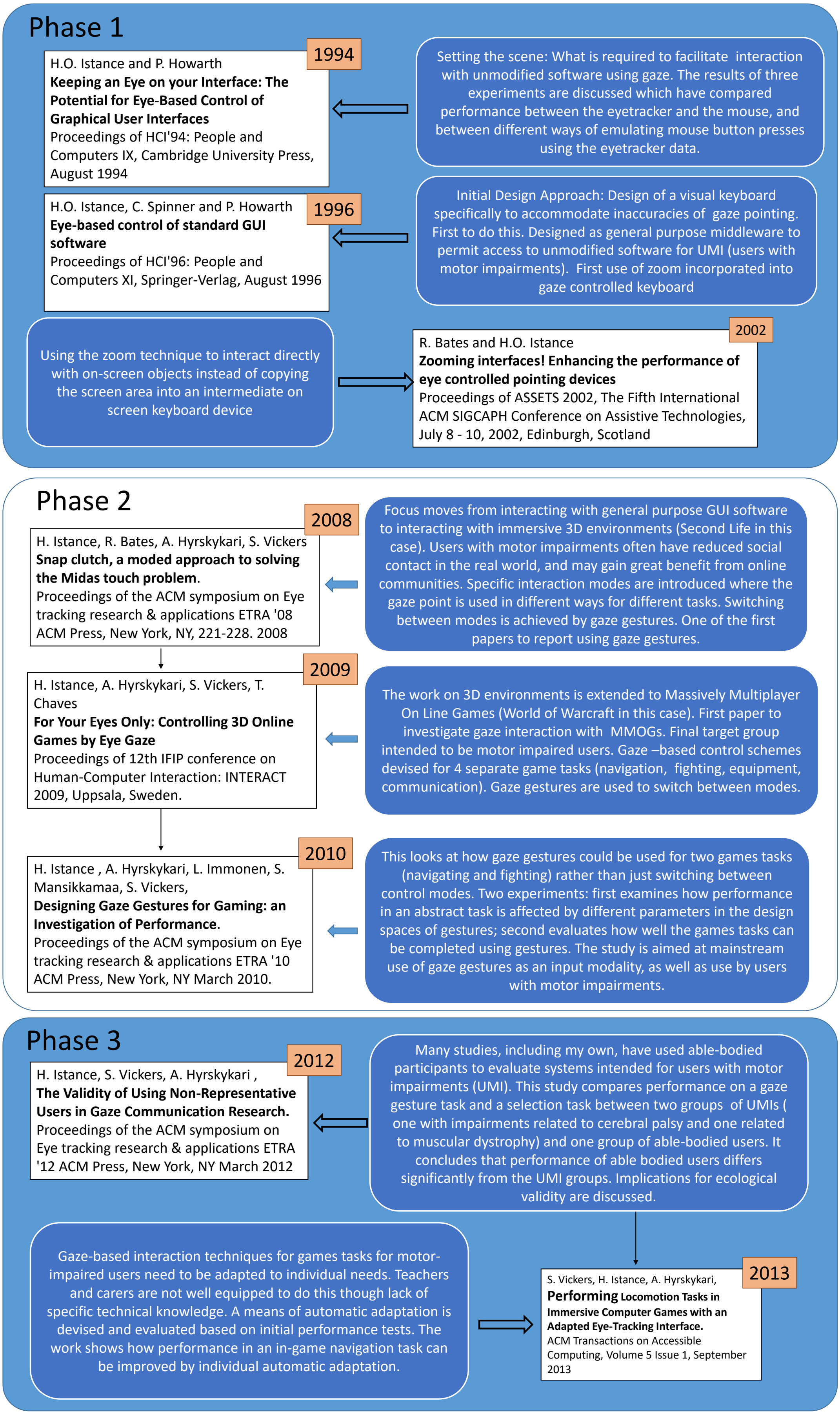


FIGURE 1.1: Chronology and content of the 8 research outputs.

impairments, and most with communication problems. This phase of research has involved testing gaze interaction techniques developed in the second phase with students with different types of motor impairments. It necessitated revising a number of key assumptions previously made about the validity of testing interaction techniques, and devices, intended for motor impaired users by able-bodied students. It also necessitated questioning the emphasis on gaze-based interaction adopted in the work so far, and whether greater emphasis should be placed on gaze-assisted interaction instead. Two of the research outputs from this period are presented in Section 4.

## 1.4 Research Outputs, co-authors and the author's contributions

### 1.4.1 Researchers who have contributed to the work submitted

All of the submitted research outputs are co-authored, and the author has been fortunate enough to work with a number of very talented people in the production of the research outputs. These are, in approximate chronological order:

- **Peter Howarth**, colleague from the Loughborough Design School, Loughborough University.
- **Christian Spinner**, student and graduate of the MSc Human-Computer Systems course, De Montfort University, whose project work the author supervised.
- **Richard Bates**, PhD student and graduate in the Faculty of Technology, De Montfort University, for whom the author was first supervisor.
- **Steven Vickers**, PhD student and graduate in the Faculty of Technology, De Montfort University, for whom the author was first supervisor.
- **Aulikki Hyrskykari**, colleague from the Tampere Unit for Computer Human-Interaction (TAUCHI), University of Tampere, Finland.

### 1.4.2 Contribution of the author to the research outputs

The contribution of the author to each of the research outputs is described below.

#### Research Output 1

H.O. Istance and P. Howarth
*Keeping an Eye on your Interface: The Potential for Eye-Based Control of Graphical User Interfaces* Proceedings of HCI'94: People and Computers IX, Cambridge University Press, August 1994

- generated the original idea for the 3 experiments, and co-designed these
- supervised data collection

- co-analysed data
- wrote the paper

**Research Output 2**

H.O. Istance, C. Spinner and P. Howarth
*Eye-based control of standard GUI software* Proceedings of HCI'96: People and Computers XI, Springer-Verlag, August 1996

- original idea for expanded keyboard solution
- co-designed the solution
- planned and carried out the evaluation study
- wrote the paper

**Research Output 3**

R. Bates and H.O. Istance
*Zooming interfaces! Enhancing the performance of eye controlled pointing devices* Proceedings of ASSETS 2002, The Fifth International ACM SIGCAPH Conference on Assistive Technologies, July 8 - 10, 2002, Edinburgh, Scotland

- co-generated idea
- supervised/ directed research carried out
- reviewed and revised paper

**Research Output 4**

H. Istance, R. Bates, A. Hyrskykari, S. Vickers
*Snap clutch, a moded approach to solving the Midas touch problem.* Proceedings of the ACM symposium on Eye tracking research & applications ETRA '08 ACM Press, New York, NY, 221-228. 2008

- co-generated the original idea
- co-designed modes and evaluation study
- co-collected and analysed experimental data
- reviewed and revised paper

**Research Output 5**

H. Istance, A. Hyrskykari, S. Vickers, T. Chaves
*For Your Eyes Only: Controlling 3D Online Games by Eye Gaze* Proceedings of 12th IFIP conference on Human-Computer Interaction: INTERACT 2009, Uppsala, Sweden

- co-originated the idea
- co-designed solution and evaluation tasks
- co-analysed data
- co-wrote the paper

**Research Output 6**

H. Istance , A. Hyrskykari, L. Immonen, S. Mansikkamaa, S. Vickers
*Designing Gaze Gestures for Gaming: an Investigation of Performance* Proceedings of the ACM symposium on Eye tracking research & applications ETRA '10 ACM Press, New York, NY March 2010.

- originated idea for the experimental investigation of gesture design space
- designed the experiment
- co-analysed data
- co-wrote the paper

**Research Output 7**

H. Istance, S. Vickers, A. Hyrskykari
*The Validity of Using Non-Representative Users in Gaze Communication Research*
Proceedings of the ACM symposium on Eye tracking research & applications ETRA ’12 ACM Press, New York, NY March 2012

- originated idea for comparing performance differences between groups to investigate validity
- analysed the data
- wrote the paper

**Research Output 8**

S. Vickers, H. Istance, A. Hyrskykari
*Performing Locomotion Tasks in Immersive Computer Games with an Adapted Eye-Tracking Interface* ACM Transactions on Accessible Computing, Volume 5 Issue 1, September 2013

- co-originated idea for adapting gaze interface to individual abilities
- supervised/ directed research carried out
- reviewed and revised paper

# Chapter 2

# A broader view of gaze-based interaction: away from text entry

The initial direction of the research was to extend what was being done on gaze-based text entry in the early 1990's (Majaranta and Räihä, 2002) to interaction with other objects common in graphical user interfaces. The purpose of this was to investigate making unmodified office software accessible to people with motor impairments, principally to enable them to work in office environments without the need for the employer to acquire bespoke software. Means were sought to activate menus and then select items from them, to select targets within the client window (such as hyperlinks on a web page), to change settings within a dialog box, as well as to enter alphanumeric data.

## 2.1 Influences on the research

Three papers at the end of the 1980's and early 1990's were particularly influential on the work contained in this thesis. These were Jacob's paper on eye movement-based interaction techniques (Jacob, 1990), Ware and Mikaelian's paper evaluating the performance of an eye tracker as input device (Ware and Mikaelian, 1987), and Starker and Bolt's gaze responsive self-disclosing display (Starker and Bolt, 1990). In addition, influential work on the notion of specifying software usability in terms of measurable criteria was published by DEC and IBM (Whiteside, Bennett, and Holtzblatt, 1988). These criteria mostly related to speed of task completion, errors made during task completion and the subjective assessment of factors associated with task completion, such as workload and comfort. This led to the idea that the usability of a software system could be engineered to a specified level in the same way as other aspects of system performance, such as reliability or response time.

A further influence was the important idea within usability engineering of impact analysis (Gilb, 1984). In this approach, in situations where metrics related to efficiency were not met, task completion time can be partitioned into 'productive time' and 'non-productive time' (or time spent in errors). The non-productive time represented a potential saving in overall task completion time. If the cause of a particular error could be designed out or removed, then overall task completion time would be reduced by the non-productive time associated with that error, and the efficiency metric would improve accordingly. The cost of designing out individual errors could be directly set against the benefits anticipated of doing so.

## 2.2 Focus of the research

A gaze-based mouse emulator was written that monitored and filtered incoming gaze position coordinates in real-time and updated the system cursor position. Mouse click events could be generated by a dwell or a prolonged stare in the region of 500 – 1000ms. The length of dwell period was chosen to prevent unintended click events being generated just from looking at objects on the screen.

The emulator was intended to be used with a 'soft' keyboard, which was a window containing buttons corresponding to keys or commands. A click event on one of these buttons would cause a keyboard event corresponding the character key, or a sequence of key events to be sent to the target application (such as a word processing program). The research question for this phase of work was:

**Research Question 1: How can gaze-based emulation of a mouse and interaction techniques embedded in middleware best be suited for common operations with graphical user interfaces (GUIs)?**

There were (and are) a number of well-known problems with gaze-based interaction that the solutions sought to overcome. The first was the 'Midas Touch problem'. As keeping the gaze position still in one position is used to signal a command, then unintentional commands may be generated by looking at objects naturally. The second problem is that natural variations in gaze position occur when looking at the same location. This positional tolerance arises because the fovea of the eye, which gives clear vision, subtends a visual angle of approximately $1^\circ$ arc of the retina (Carpenter, 1988). Hence when fixating a target the eye only needs to be within approximately $1^\circ$ of the target position to see the target clearly. This gives an inaccuracy in measured gaze position. In addition, there is high frequency jitter inherent in eye movement. The limited sample rates of the available eye tracking devices (in the region of 60Hz) means that the position of fixations extracted from the stream of sampled data may not be accurate. The third problem is that visual feedback as a result of a command on a GUI object often occurs at a different location from that of the control object that initiates the command. If gaze position is used to initiate a command at one location it may not be possible to observe simultaneously the feedback resulting from the command at another location.

## 2.3 Research into gaze-based interaction techniques and design solutions

The studies reported in the first 2 papers used a binocular Micromeasurements 7000 pupilometer. This required the use of a head rest and was built as a device to study pupil size. As the coordinates of the pupil centres were available it was possible to use this as an eye tracker. The study in the 3rd paper used a desktop eye tracker (the Sensor Motoric Instruments Remote

Eye tracking Device or SMI RED II) used without a head rest. Both devices sampled eye position at 60Hz.

The first study **(Research Output 1, 1994)** reported three experiments that examined performance differences in tasks carried out with a mouse and with a gaze-driven mouse emulator. In the first of these, target acquisition by mouse and by gaze was compared, and target selection was made with a hardware button in both cases. Target *acquisition* refers to identifying which target to send an event to, and target *selection* refers to sending an event (such as a left button click) to that target. The independent variables were pointing device and target size. In the second experiment, as eye position and pupil size data was available from both eyes simultaneously, the option of pointing with one eye and closing the other to emulate the 'mouse button down' condition was studied. This was compared with performance using a monocular dwell to select ,and with performance using a hardware mouse. In the third experiment, the selection of a piece of text was compared between the same binocular protocol and a 'moded wink' protocol. In the latter, the first wink generated a 'button down' event and then a second wink generated a 'button up' event. The error rates in particular found with the gaze protocols suggested that there would be serious accuracy problems using gaze as a mouse emulator even with access to binocular data and the constrained head position.

One solution to this was to base interaction around the use of short-cut keys that achieved the same outcomes in terms of commands as mouse pointing and clicking on GUI objects. Many menu-based commands also had keyboard shortcuts (such as CTRL-P to print). Also, key-bindings existed for interacting with objects within dialogue boxes, such as the Tab key to move between objects. The design solution then became the building of one or more soft keyboards adapted to accommodate the accuracy limitations of gaze position measurement.

The second study in this phase **(Research Output 2, 1996)** investigated the design and evaluation of a solution using specialised soft keyboards to address the problems of direct mouse emulation by gaze. The keypads were customised to the needs of different types of GUI object and interaction task (text entry, numeric data entry, zoom, dialog box, menu and system). The main innovations were the preloading of keypads according to the state of the application, moving text from menu items in the target application window into the menu keypad, automatically detecting and moving overlapping windows, and the zoom keypad (see Figure 2.1 ). To overcome the difficulty of selecting small targets accurately, an area from the client window could be copied, magnified and displayed in the zoom keypad. An extended stare or dwell at a location in the magnified view on the keypad would cause a mouse event to be sent to the corresponding location in the client window. Evaluation trials used a set of word-processing tasks that formed an integrated exercise, and web browsing tasks formed a follow-up. Text entry rates using the text keypad were slow in comparison with eye typing rates reported elsewhere at the time. The zoom keypad was reported as the most preferred means of command selection in the word processing tasks.

The idea of zooming into an area of the client window to compensate for the inaccuracy in gaze point measurement worked well, but there was

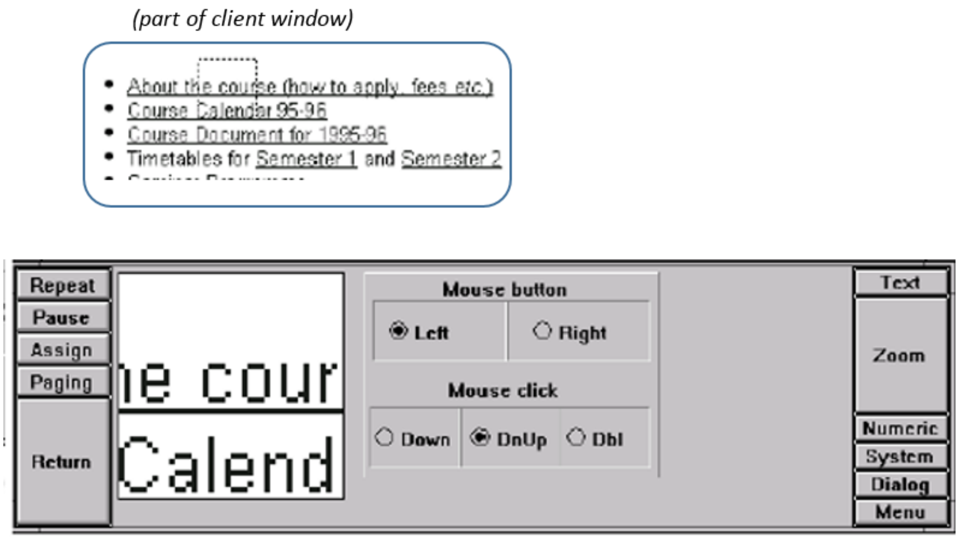


FIGURE 2.1: Zoom keypad of indirect GUI control solution (Research Output 2). A dwell on the right hand Zoom key loads the keypad, then another dwell in the client window copies an area into the zoom keypad, then another dwell within the zoom keypad area sends an event to the corresponding area of the client window.

a clear time overhead as multiple dwell events were required to produce a click event on one GUI object.

The intention in the next output **(Research Output 3, 2002)** was to study direct interaction with the client window by temporarily zooming or magnifying the entire screen under user control in order to make an on-screen object large enough to select reliably with the gaze based mouse emulator. This was instead of indirect zooming by copying a magnified region of the screen into a separate application, as was the case in Research Output 2. This study paid particular attention to partitioning task completion time into different components in the manner of Gilb's impact analysis. Efficiency as a usability metric was defined using these components. The number of incorrect commands, the number of target misses and the number of control corrections were used to define a quality of interaction metric, and this value divided by time was used as a measure of device efficiency. A suite of 150 test tasks was devised that related to word processing and web browsing. The selection target in each task was categorised into 4 sizes, the smallest being $0.3°$ of visual angle and the largest being $1.2°$.

Three devices were compared for pointing at the target, and the selection event was generated by a hand held switch in all cases. The devices were a head mouse (a commonly used assistive pointing device), an eye mouse, and the eye mouse together with the screen zoom facility. Four zoom levels were provided (1x, 2x, 4x, 8x), and zoom in and zoom out was controlled by hand held switches. A hand held mouse was included for comparison purposes. The provision of the zoom facility raised the performance of the eye mouse to above that of the head mouse, largely by reducing the need for cursor position corrections when selecting the targets. Unsurprisingly, it did not exceed performance with the hand held mouse.

All of the testing used able bodied participants.

## 2.4 Reflection on the research outputs

The work was innovative in that it was the first time (to the author's knowledge) that zooming or magnifying all or part of the screen to compensate for the poor pointing accuracy of an eye mouse was investigated and reported. Lankford (2000) later reported a similar device as part of the ERICA system. A US patent for the idea was applied for and obtained by those working with that system, but this was considerably later than the work carried out here. The use of the binocular protocol for mouse emulation was also a first (again to the best of the author's knowledge) but this was not studied in any depth. Another innovative aspect of the work was the application, in the context of gaze-based interaction, of the ideas underpinning usability engineering and impact analysis to define dependent variables.

The rights to visual keyboard described in Research Output 2 were sold to SMI in exchange for an eye tracker. SMI is a Berlin-based company and is presently one of the 2 leading international eye tracker manufacturers in terms of sales (Tobii of Sweden being the other). SMI shipped the visual keyboard with some of their devices for some years after they acquired the software.

The limitation of the research, which later became apparent, was the lack of focus on the real target user group, namely motor-impaired users. The assumption was made that if able-bodied participants had problems using gaze-based interaction techniques with unmodified applications, then users with motor impairments would also experience these. While this is probably true, it does not follow that a lack of interaction problems for able-bodied participants implies a similar lack of problems for motor-impaired users.

# Chapter 3

# Novel gaze-based interaction techniques for multiplayer virtual worlds and games

The next phase of the research investigated the use of gaze-based interaction to enable people with motor impairments to access on-line multiplayer games and virtual communities. Young people with motor impairments often experience isolation in the real world due to mobility and communication issues, but these restrictions can disappear in a virtual world.

## 3.1 Background

By the mid 2000's, virtual communities such as World of Warcraft and Second Life had become very popular with millions of active users worldwide. By 2008, World of Warcraft had 11 million monthly subscribers (Blizzard, 2010). These communities offered the possibility of interaction with other people without the extent of an individual's disability needing to be apparent. Alternatively, a person could choose to reveal the extent of their disability in a virtual community. Conventional assistive devices could be too slow, too fatiguing or both. Critical to successful interaction was the speed with which a person could react to other players in the game. If response latency was too great, other people in the community who were unaware of the person's disability may assume that the person was not interested in communicating or interacting.

What a person does in Second Life and in World of Warcraft are different, but share many common features.

In Second Life, there were generally virtual equivalents of real world activities. A person may have visited, say, a virtual university built as a 3D graphical model and attended a virtual open day. The person would be represented by their own avatar, and they would communicate with avatars of other people in the same virtual space.

World of Warcraft is a massively multiplayer online role playing game (MMORPG), and a person would also be represented by their own avatar in a 3D graphical fantasy world. The activities focus however on fighting other characters, acquiring strengths, skills and weapons. There are common tasks in Second Life and World of Warcraft: moving the player avatar or locomotion; controlling the camera; interacting with objects in the world; communicating with others; and accessing application commands.

## 3.2 Focus of the Research

Gaze-based interaction offers the prospect of fast interaction techniques. The issue is how well can these be fitted to common tasks in multi-player games and communities. The research question for the second phase of the research can be summarised thus:

**Research Question 2: To what extent can gaze-based interaction techniques be designed for tasks in multiplayer games and communities, so that the pace of interaction is not noticeably different to people using other conventional input devices?**

The groups of tasks studied in this phase of the research were categorised into a) locomotion and camera control, b) object manipulation, c) application control and d) communication with other player characters and non-player characters. The first 2 research outputs in this phase used dwell as a command selection technique. The conclusion was that dwell was too slow to enable the objective stated above to be achieved, and that alternative gaze-based selection mechanisms were needed. The use of gaze gestures as an alternative to dwell was studied in the 3rd research output.

## 3.3 Research into techniques to support multiple tasks

The fourth research output **(Research Output 4, 2008)** described an architecture (Snap Clutch) within which gaze-based interaction techniques can be situated. The approach here was to find solutions to two of the general problems of gaze interaction identified in the Introduction. These were the Midas Touch problem and the problem of input action and feedback occurring at different places.

As before, a middleware application was built and the target application was Second Life. Different modes of interaction were devised that used the constant stream of gaze data in different ways (See Appendix A for a description of a 'mode').

Four modes were available at any one time and gaze gestures were used to switch between these modes. Glancing from the target window, over the edge of the screen and back to the target window constituted a gesture. Glancing over the 4 edges of the screen each constituted a different gesture, which were associated with 4 different modes. Switching between modes was achieved simply by glancing over one of the screen edges and back again.

A solution to the Midas Touch problem was to disable active gaze control temporarily (or conversely to enable it temporarily), thus one mode was no gaze control. Dwelling on an object had no effect in this mode. In Second Life, two transparent control panels could be used, one to support locomotion and one to support camera control. Two control modes were devised to enable these panels to be used so that feedback from the virtual world could be seen as the commands were given. One of these ('Park it here') allowed the cursor to dropped at a location with a dwell action, Subsequent dwells sent the appropriate mouse event to the location where the cursor had been parked. In the third mode ('Drag from here'), the first dwell starts the drag operation (a mouse down event) from that location. Subsequent dwells had no effect and the drag was ended by a mode change or looking


FIGURE 3.1: Gaze-based locomotion in Second Life using the 'Drag from here' mode in the SnapClutch middleware tool

back at the start of the drag (see Figure 3.1). The fourth mode was a dwell click where dwelling at a particular location would cause a click event at that point. Thus, in each of the different modes, dwelling and moving the gaze position had either different effects or no effect.

These modes were evaluated in trials involving 4 tasks. The tasks were: moving the player avatar along a defined path; changing the camera position and field of view; creating an in-world object; and using the application commands to change the colour of the avatar's hair. Initially performance using a hand held mouse was compared with a simple gaze-based mouse emulator using dwell time as the means of selection. The locomotion and camera movement tasks in particular were very difficult to complete. The second phase of testing used the Snap Clutch modes, and considerable performance improvements were obtained such that these tasks could be completed without problems.

The inherent problems of gaze interaction (lack of pointing accuracy, disassociation of input location and feedback location, and the Midas Touch issue) have been addressed in the previous outputs. Compensating for pointing inaccuracy was addressed with a zoom facility (Research Output 3) and the other two problems were addressed with specific modes of interaction (Research Output 4). The next research output (**Research Output 5, 2009**) examined the extent to which the combination of these could be used to enable someone to play World of Warcraft using gaze input only.

The objective was to achieve a beginner's level of play only and there was no intention or expectation that gaze-only interaction would enable the same level of performance as hand held input devices for experienced players. The zoom facility was included by means of a magnifying glass tool (see Figure 3.2). This was a fixed screen magnification rather being variable as was the case in Research Output 3.

In this study, the locomotion task was enabled by a specific mode. Data was collected about where experienced players looked during normal game play. This region extended to the left and right of the player's avatar, which was always located in the centre of the screen. Other areas of the screen were rarely looked at. The 'locomotion' mode used gaze positions in these outer areas to turn the character's direction while moving. The 'no action' mode was replaced with a 'no movement' mode. The only response to gaze in this mode was to rotate the camera and the character if the user looked to the extreme left or right of the window.

The evaluation of the complete interface was based around 4 related tasks that were considered to be representative of a beginner's level of play. These were a locomotion task, fighting a non-player character, collecting equipment, and communication with another player. Communication was supported by a gaze-driven predictive text keypad (similar to T9, previously used on mobile phones to generate text using the number keys). The keypad was activated in the same way as the magnifier, i.e. by dwelling on an icon overlaid at the top of the game window. In the evaluation trials, performance using the gaze interface was compared with performance using a hand held mouse to complete the same tasks. Of interest was the ratio of task completion times and errors between the gaze and hand held mouse conditions. Expected performance ratios were generated on the basis of previous trials with Second Life (Istance et al., 2008).

All of the participants were able to complete all of the tasks, which was encouraging. The locomotion task controlled by gaze exceeded expectations and was found to be a very natural means of controlling the movement of the player's character.

The magnifier gaze on the other hand did not work well as the overhead of a dwell to activate the tool, a dwell to drop it and a dwell to generate the event, was too effortful and time consuming. Participants would try to click directly on small objects and would use the magnifier as a final resort. This led to long task completion times for some tasks. The conclusion was that, in this context, dwell was too slow as a means of signaling a deliberate intention with gaze. An alternative to dwell was needed in order to make selections more quickly and robustly. The alternative chosen was gaze gestures, which were already being used very successfully in Snap Clutch to switch between modes.

The next output (**Research Output 6, 2010**) investigated the use of gaze gestures to give specific commands. During normal gameplay using a mouse, commands could be given by clicking on icons arranged along the edge of the window. With gaze-based interaction, dwelling on these icons would mean diverting overt visual attention from the centre of the window where most of the game play took place. Ideally, a means of activating commands by gaze was needed that did not require the player to look away from the centre of the window. The gesture scheme developed for this purpose recorded fixations in specific areas of the window (referred to as zones). These were shown as semi-transparent regions overlaid on the game window (see Figure 3.3). Gestures then consisted of valid sequences of zones. All sequences began and ended with fixations recorded in the centre zone, which was overlaid on the player's character. There were made up either

FIGURE 3.2: Magnifying glass tool dropped over spell icon in World of Warcraft to enable dwell events to be targeted more accurately

2 eye movements or strokes (centre zone – other zone – centre zone), or 3 movements (centre zone – other zone 1 - other zone 2 – centre zone). Research Output 6 reported an experiment to study performance when making gestures, and an evaluation study of using the chosen scheme to play World of Warcraft. The gesture interface consisted of 3-stroke gestures and 2-stroke gestures. This can be seen in Figure 3.3, together with an example of one of the 3-stroke gesture patterns.

The 2-stroke gestures were used to control locomotion and the 3 strokes gestures were each mapped to specific commands. The evaluation consisted of 12 able-bodied experienced game players who were given set of objectives to accomplish during a period of game play. The outcome of the study was very positive in that all participants were able to achieve the objectives set for them. Gestures were found to be effortful and time consuming for controlling locomotion, but they were effective for issuing discrete commands.

Part of the game play evaluation study examined the extent to which the patterns of eye movement that represented gestures were made during normal game play. This enabled the likelihood of normal patterns of eye movements being mistakenly recognised as gestures to be estimated. Importantly, none of the 3-stroke gesture patterns occurred in eye movement during normal game play.

## 3.4 Reflection on the research outputs

The work was innovative in that it was among the first investigations of gaze only interaction with immersive 3D graphical environments such as


FIGURE 3.3: Example of a 3 stroke gaze gesture performed over the grey semi-transparent active zones. Gesture mapping to commands is shown on the light grey panel on the right of the window

World of Warcraft and Second Life. Tanriverdi and Jacob (2000) had previously investigated gaze-based techniques for selecting objects in virtual environments. Castellina and Corno (2008) reported an evaluation of gaze-based and gaze-assisted interaction techniques for games and virtual environments that included techniques for locomotion and camera control.

The work examined different gaze-based interaction techniques to achieve a range of tasks to enable beginner level of play with these games. Although there was no specific testing of the interfaces with people with motor impairments, the design space was explored and design options that would be ineffective for gameplay for the target user group were identified. The third phase of research in the next section examines how these gaze-based interaction techniques can be used and adapted to different levels of abilities.

# Chapter 4

# Adapting gaze interaction techniques for motor-impaired users

Motor impaired users are very different in terms of their abilities not only from non-impaired users but also from each other. A single person's abilities may also vary considerably between days or during the day as a result of fatigue or illness, or may gradually change as a result of a degenerative condition. It is often difficult to generalize about people abilities, or lack of them, even within the same category of disability. How then can the gaze-based interaction techniques be adapted to the needs of individual users?

## 4.1 Influences on the Research

COGAIN (Communication by Gaze Interaction), which was a Framework 6 European Network of Excellence had a significant influence on this work. The focus of the Network was the use of gaze to enable communication and use of computer systems by people with motor impairments (Bates et al., 2007). The author was a steering committee member for its duration (2004 – 2009), and organized the four annual conferences held after the first year.

The work carried out by Wobbrock and Gajos has been useful and influential. One paper is a discussion of ability based design (Wobbrock et al., 2011), as opposed to design for disability, and presents 7 principles for this. The other describes a tool, Supple, for automatically adapting the design of a GUI dialogue box to individual abilities and preferences (Gajos, Wobbrock, and Weld, 2008). The abilities were obtained from a simple diagnostic test.

A major influence on the work in this phase was a conversation with Anthony Hornof in 2008, who had developed a system called EyeDraw and had evaluated this with children with severe motor impairments (Hornof and Cavender, 2005). He explained how he had worked as a volunteer in a local centre for children with disabilities, and had later been able to enlist their help with the evaluation. He said their participation had been invaluable in obtaining insights into how the target user group responded to the system. He also said they had been very willing to help. The consequence of this was a collaboration with Ash Field Academy, a special needs school in Leicester, which has been the single most significant influence on the whole research program.

## 4.2 Focus of the Research

The final 2 research outputs examine the relationship between ability and the design of the gaze-based interaction techniques described in the previous section. The research question in the third phase of the research can be summarized thus:

**Research Question 3: How does the variability between people abilities affect the use of gaze interaction techniques in virtual worlds and games, and what extent can these techniques be adapted to individual needs?** During the first period of the collaboration with the school, students evaluated the interaction techniques in the Snap Clutch tool for World of Warcraft (Research Output 5). The game was run on a private server, so there was no other players present in the game, and most of the monsters were removed.

## 4.3 Research into abilities and adaptation of techniques to different abilities

**Research output 7, 2012** examined differences between a group of able-bodied participants and 2 groups of motor impaired users, and in doing so, raised questions about the ecological validity of only using able-bodied participants in evaluation studies of applications and devices intended for people with disabilities. Two studies were carried out.

The first required participants to make patterns of eye movements that could be recognized as gaze gestures. The second contained tasks that required gaze-based selection using dwell. The members of one group had some form of cerebral palsy, while the members of the other had some form of muscular dystrophy. Both groups were recruited primarily from students at Ash Field Academy. The first study investigating gaze gestures showed a large difference in performance between the two groups with motor impairments on one hand, and the able bodied group on the other. There were also significant differences between the 2 groups with motor impairments. Some participants were not able to complete gesture sequences because involuntary head movements caused the tracker to lose the image of the eyes, and some others were unable to fixate reliably in different parts of the screen. The picture that the second dwell time study presented was somewhat different and the group with muscular dystrophy performed much better than the cerebral palsy group, and there was no significant difference between them and the able-bodied group.

The final output (**Research Output 8, 2013**) examined in detail the response of participants to the original (and unmodified) set of interaction techniques. It described how one of the gaze-based interaction techniques could be adapted to the abilities of an individual user, and the basis on which this could be done automatically.

The response to being able to move the player's character under gaze control was generally very positive. Some students found this to be very empowering as, due to communication and mobility difficulties, their ability to act independently was very limited. They found that they could explore the game world and decide themselves where the player avatar went. The gaze-based locomotion technique was learned quickly and the gestures

FIGURE 4.1: An example of an adapted interface during the training phase where the location of the different locomotion controls is shown. After training was completed these were switched off.

used to change modes were effective. There were significant issues, however, in how well the eye tracker could be located in relation to the seated participant.

There was considerable variability in performance between individuals even within the same nominal category of disability. The gaze-based locomotion technique designed for use with immersive environments was taken as the target for automatic adaptation. The question was whether a person's performance with an individually adapted interface was better than the unmodified interface. The approach taken was analogous to Gajos and Wobbrock's Supple system, although their work was not directed to gaze-based interaction.

A diagnostic test of a person's ability to fixate on different parts of the screen was devised. The adaptation was based on allocating the individual locomotion controls to those parts of the screen that a person could reliably fixate upon (see Figure 4.1). Most of the participants in the muscular dystrophy group and all of the able-bodied participants did not require any adaptation as they had the ability to fixate reliably on all parts of the display surface. However nearly all of the group with cerebral palsy lacked this ability. The individually adapted interface was used in a second test, and the performance of all participants was better using the modified interface than when using the unmodified interface.

## 4.4 Reflection on the research

The research in this phase has focused on how the gaze based techniques for interacting with immersive games and virtual environments can be used by the target users of the work. It has highlighted the considerable challenges when designing solutions to fit such a diverse population as motor-impaired users. It has also pointed to the danger of only conducting evaluation studies with able-bodied users. The performance of able-bodied participants and the lack of problems they experienced was not representative of the target user group. One source of problems giving rise to the need for adaptation was the fixed position of the desk-mounted eye tracker. Wearable eye trackers offer a potential solution to this problem. Thus rather than only fitting the interaction technique to the individual, fitting the input device (the eye tracker in this case) to the individual becomes part of the overall solution to the question of individual adaptation.

A limitation of the research is that it has focused heavily on one interaction technique, namely gaze-based control of locomotion in immersive 3D environments. However, the justification for this was that, it was a very successful in enabling a rapid and intuitive means of controlling the movement of the avatar and thus enabling independent exploration of the virtual worlds. Some students found this to be very empowering. In one case, the ability to explore the virtual world independently led to the student wanting to learn to drive a powered wheelchair, something she had previously been unwilling to do. This in turn led to a significant improvement in self-confidence and a far greater engagement with education.

An important realisation in relation to the collaboration with the school was that it was a two-way process. In return for Ash Field's participation in the research, groups of older students from the school visited the university regularly for games design and building workshops over several years. The school thought this was a very valuable activity to encourage these students to believe that they too could apply for places in higher education. This is a valuable lesson for university research groups. Schools and other groups of people with special needs are likely to be willing participants in research projects such as those described here if there is a direct benefit to them of doing so.

# Chapter 5

# Conclusions

The work described in this thesis covers a period of 20 years from early investigations into making standard applications accessible by gaze-based interaction, through to the evolution of different methods of making immersive games and virtual communities available to the motor-impaired. This has been realised by the development of interaction techniques that can be adapted to the individual needs of motor impaired users.

## 5.1 Research Methods

The research has adopted largely a positivist view of research. In each of the papers presented as research outputs, there is at least one experimental study reported in which the effect of one or more independent variables on one or more dependent variables was studied. In Research Output 6, there is a less controlled empirical study of game play using gaze gestures. In this, participants used gaze gestures to issue commands during periods of game play based on a set of general instructions, rather than in carefully controlled conditions. In Research Output 8, there are descriptions of the individual participant's reactions to gaze based locomotion in World of Warcraft that lean to the phenomenological approach to research (Moustakas, 1994). With hindsight, greater use should have been made of qualitative research methods in the 3rd phase of the research to be able to describe single case studies. The diversity of abilities amongst people with motor impairments means that creating groups of participants in sufficient numbers for positivist experimental studies is challenging. This can be seen from the range of data collected in the group of participants with cerebral palsy in Research Output 7. A consequence of this diversity is that it is difficult to generalise from the results of studying a sample of participants to a larger population.

## 5.2 Achievements of the Research

The original reason for the focus on immersive games and virtual worlds was to enable people with motor impairments to participate in online communities such as Second Life and World of Warcraft to reduce the social isolation in the real world that can accompany movement and communication difficulties. However, over the period of the work the educational value of games in the classroom for developing problem solving skills, for motivation and for engagement has become apparent. A report published by Futurelab, now part of the UK National Foundation for Educational Research, reported as a result of a survey carried out in 19 schools that game-based approaches present an excellent opportunity to engage students in activities which can enhance learning and produce a range of educational benefits (Groff, Howells, and Cranmer, 2010). In addition to 2D games, 3D games and game environments, such as Minecraft, are used as educational tools (Walsh, Donahue, and Pease, 2016). Students with motor impairments need to be enabled to make full use of these opportunities, both at home and in the classroom.

In this context, the achievements of the research can be summarized into 4 points.

First, the work has contributed *innovative gaze-based interaction techniques*. The integration of zoom into gaze interaction was the first time this approach had been used to offset the difficulties associated with selecting small targets. A zoom keypad enabled part of the screen to be enlarged and copied into a separate area for gaze selection (Research Output 2), the whole screen could be magnified under gaze control (Research Output 3) and a magnifier tool was incorporated with the interaction techniques with immersive role playing games (Research Output 5). The value of this was recognized by a European manufacturer who acquired the commercial rights to the visual keyboard and distributed it with their eye trackers. The work was at the forefront of applying gaze gestures to games, being the first to use them in the context of immersive games and virtual worlds (Research Outputs 4, 5 and 6). The use of binocular control with a gaze-based mouse emulator was also innovative at the time. One eye was used to point at the target, and closing the other eye acted as a mouse button control (Research Output 1). This technique was used to enable selection and the dragging and dropping of piles of cards when playing Windows Solitaire by gaze.

Second, the work has *enabled access to immersive games and virtual worlds for people with motor impairments by eye gaze*. This has been achieved by collecting gaze-based interaction techniques into a single framework that allows rapid switching between modes that use gaze position and gaze movement in different ways. This allows the player to perform different in-game tasks without interruption or re-configuring any part of the interface. This was the first time that such a comprehensive approach to supporting multiple games-related tasks simultaneously by gaze had been used. (Research Outputs 4 and 5). The benefits of this have been demonstrated by single-case studies at Ash Field Academy, where students have obtained significant improvements in self-esteem through the sense of empowerment that independent interaction with virtual worlds has bought.

Third, the work has explored means of *automatic individual adaptation of gaze-based interfaces* for people with motor impairments (Research Output 8). The author is not aware of other work that has reported this. If the opportunities for using games in teaching and learning described above are to be realised, then this adaptation is important. Even in special-needs schools, teaching staff do not generally have the time or the expertise to be able to configure gaze-based systems for individual needs. For children with motor impairments who are in mainstream schools, the opportunities for manual configuration could be even less. Consequently, having the means of automatic adaptation to individual needs becomes is important if the benefits of gaze-based interaction for a wide range of students with motor impairments are to be fully realised.

Fourth, the work has made *significant methodological contributions*. The results of the study of the differences in performance between groups of able-bodied participants and groups of participants with either cerebral palsy or muscular dystrophy question the ecological validity of previous studies that only tested groups of able-bodied participants (Research Output 7). In many studies of eye-typing, for example, text entry rates are often reported without any reference to the rates that groups of motor-impaired users might achieve, even though the latter are the target user group. The work has also applied usability engineering ideas to the evaluation of gaze-based interaction such that predictions can be made about the improvement in performance expected if certain types of error have been removed in a subsequent design iteration. Finally, resampling statistical techniques have been applied to the comparison of data from small sample sizes. This removes the need to apply parametric techniques where the assumptions about acceptable sample sizes and underlying data distributions are difficult to justify.

## 5.3 Looking to the future

Eye tracking is becoming cheaper and the prospect of large numbers of low cost trackers in the classroom is realistic. Low cost head mounted eye trackers are also becoming available [1]. Near-eye displays with gaze tracking are also likely to be affordable for class room use in the near future. The question is how to make best use of this opportunity for students with motor-impairments in the classroom, especially if greater use is made of games for education in the future.

Work at Ash Field Academy suggests that the emphasis on gaze only interaction is misplaced. Understandably, research to date has generally focused on interaction being achieved solely by gaze because of the large variation in abilities amongst the target groups. Most people can use gaze, it is argued, but assumptions about other abilities are difficult to make.

The author and colleagues at the school conducted an unpublished survey of abilities of 110 of the pupils in 2012. This showed that those students with sufficient cognitive ability to use software applications for education or entertainment all had the ability to use a switch of some description. Those students who were unable to use a switch generally did not have the cognitive abilities to use this kind of software. There certainly are types of motor impairment, such as motor neuron disease and locked-in syndrome, where a person is unable to use a switch but has normal cognitive abilities. There were no people with these categories of impairments in the school. Also, people with these types of impairments are relatively few in comparison with those with cerebral palsy and muscular dystrophy (Jordansen et al., 2005 and Appendix A). In addition, teachers and assistants at the school reported that using eye trackers in classes was very tiring for the students. The sources of this tiredness are not well understood, although it is tempting to speculate that this is due to loading all interaction onto the location of and movement of the gaze point.

[1] https://pupil-labs.com/

So given the ability to use switches in addition to gaze, and the level of fatigue currently induced by using eye trackers in classes, one future direction of work is to investigate further gaze-assisted interaction, as opposed to gaze-based interaction with games, which includes automatic individual adaptation of the techniques.

Another topic within the same direction is the investigation of the extent to which head mounted eye trackers can be used to overcome the problems experienced by some students that are the result of the desktop trackers failing to track their eyes well. This failure can occur when students have difficulty sitting in a position that enables reliable tracking of their eyes. However, wearable eye trackers should overcome many of these problems.

## 5.4 Conclusion

To conclude, the work covered in this thesis by the author and his colleagues at De Montfort University, Loughborough University and University of Tampere has made a significant contribution to understanding how best to use gaze-based interaction techniques to allow people with motor impairments to control computer game software. Games will, in all likelihood, be used more and more in educational contexts. With the advent of cheaper eye tracking devices, and as a result of the research path followed over the 20 year period covered by and described in this thesis, these will now be more accessible to groups of students with motor impairments.

# Bibliography

Bates, R. et al. (2007). "Introducing COGAIN: communication by gaze interaction". In: *Universal Access in the Information Society* 6.2, pp. 159–166. DOI: 10.1007/s10209-007-0077-9.

Blizzard (2010). *World of Warcraft Subscriber Base Reaches 12 Million Worldwide*. Internet (Retrieved March 2011). Blizzard. URL: http://us.blizzard.com/en-us/company/press/pressreleases.html?101007.

Carpenter, Roger HS (1988). *Movements of the eyes (2nd rev*. Pion Limited.

Castellina, Emiliano and Fulvio Corno (2008). "Multimodal Gaze Interaction in 3D Virtual Evironments". In: *Proceedings of the 4th Conference on Communication by Gaze Interaction: COGAIN '08*. Prague, Czech Republic. URL: http://www.cogain.org/cogain2008/COGAIN2008-Proceedings.pdf.

Foley, James D. et al. (1990). *Computer Graphics: Principles and Practice (2nd Ed.)* Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc. ISBN: 0-201-12110-7.

Gajos, Krzysztof Z., Jacob O. Wobbrock, and Daniel S. Weld (2008). "Improving the performance of motor-impaired users with automatically-generated, ability-based interfaces". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '08. Florence, Italy: ACM, pp. 1257–1266. ISBN: 978-1-60558-011-1. DOI: 10.1145/1357054.1357250. URL: http://doi.acm.org/10.1145/1357054.1357250.

Gilb, T (1984). "The "impact analysis table" applied to human factors design". In: *First IFIP Conference on Human-Computer Interaction: Interact '84*, pp. 97–101.

Groff, J., C. Howells, and S. Cranmer (2010). *The impact of console games in the classroom: Evidence from schools in Scotland*. Tech. rep. Futurelab, Bristol. URL: https://www.nfer.ac.uk/publications/FUTL25.

Hornof, Anthony J. and Anna Cavender (2005). "EyeDraw: Enabling Children with Severe Motor Impairments to Draw with Their Eyes". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '05. Portland, Oregon, USA: ACM, pp. 161–170. ISBN: 1-58113-998-5. DOI: 10.1145/1054972.1054995. URL: http://doi.acm.org/10.1145/1054972.1054995.

Istance, H et al. (2008). "User Performance of Gaze-Based Interaction with On-line Virtual Communities". In: *Proceedings of the 4th Conference on Communication by Gaze Interaction: COGAIN '08*. Prague, Czech Republic. URL: http://www.cogain.org/cogain2008/COGAIN2008-Proceedings.pdf.

Jacob, Robert J. K. (1990). "What you look at is what you get: eye movement-based interaction techniques". In: *Proceedings of the SIGCHI conference on Human factors in computing systems: CHI '90*. Seattle, Washington, United States, pp. 11–18. ISBN: 0-201-50932-6.

Jordansen, I.K. et al. (2005). *D7.2 Report on a market study and demographics of user population*. Tech. rep. Communication by Gaze Interaction (COGAIN), IST-2003-511598: Deliverable 7.2. URL: http://www.cogain.org/results/reports/COGAIN-D7.2.pdf.

Lankford, Chris (2000). "Effective eye-gaze input into Windows". In: *Proceedings of the 2000 Symposium on Eye Tracking Research and Applications: ETRA '00*.

Majaranta, Päivi and Kari-Jouko Räihä (2002). "Twenty years of eye typing: systems and design issues". In: *Proceedings of the 2002 symposium on Eye tracking research & applications*. ETRA '02. New Orleans, Louisiana: ACM, pp. 15–22. ISBN: 1-58113-467-3. DOI: 10.1145/507072.507076. URL: http://doi.acm.org/10.1145/507072.507076.

Moustakas, C. (1994). *Phenomenological research methods*. SAGE Publications Ltd. DOI: http://dx.doi.org/10.4135/9781412995658.

Starker, India and Richard A. Bolt (1990). "A gaze-responsive self-disclosing display". In: *CHI '90: Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 3–10. ISBN: 0201509326.

Tanriverdi, Vildan and Robert J. K. Jacob (2000). "Interacting with Eye Movements in Virtual Environments". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '00. The Hague, The Netherlands: ACM, pp. 265–272. ISBN: 1-58113-216-6. DOI: 10.1145/332040.332443. URL: http://doi.acm.org/10.1145/332040.332443.

Walsh, Greg, Craig Donahue, and Zachary Pease (2016). "Inclusive Co-Design Within a Three-Dimensional Game Environment". In: *Proceedings of the The 15th International Conference on Interaction Design and Children*. IDC '16. Manchester, United Kingdom: ACM, pp. 1–10. ISBN: 978-1-4503-4313-8. DOI: 10.1145/2930674.2930721.

Ware, Colin and Harutune H. Mikaelian (1987). "An evaluation of an eye tracker as a device for computer input". In: *Proceedings of the SIGCHI/GI Conference on Human Factors in Computing Systems and Graphics Interface*. CHI '87. Toronto, Ontario, Canada: ACM, pp. 183–188. ISBN: 0-89791-213-6. DOI: 10.1145/29933.275627. URL: http://doi.acm.org/10.1145/29933.275627.

Whiteside, John, John Bennett, and Karen Holtzblatt (1988). "Chapter 36 - Usability Engineering: Our Experience and Evolution". In: *Handbook of Human-Computer Interaction*. Ed. by MARTIN HELANDER. Amsterdam: North-Holland, pp. 791 –817. DOI: http://dx.doi.org/10.1016/B978-0-444-70536-5.50041-5.

Wobbrock, Jacob O. et al. (2011). "Ability-Based Design: Concept, Principles and Examples". In: *ACM Trans. Access. Comput.* 3.3, 9:1–9:27. ISSN: 1936-7228. DOI: 10.1145/1952383.1952384. URL: http://doi.acm.org/10.1145/1952383.1952384.

# Appendix A

# Explanation of terms used

In this Appendix, the terms that appear in the title of the work are explained, together with other related terms.

It is usual to distinguish between the user interface to a software application, and the internal representation and processing of the objects making up the application. Applications are often constructed such that the software that displays information about application objects to the user and obtains information from the user is quite separate from the software that represents and processes these application objects.

## A.1 Interaction techniques and interaction tasks

Foley et al., 1990 made the distinction between *interaction task* and *interaction technique*. Interaction techniques are ways to use input devices to enter information, while interaction tasks classify different types of information entered with the interaction techniques. Consequently, many different interaction techniques can be used for the same interaction task. He identified 4 basic interaction tasks: 'position' or entry of a location (x,y or x,y,z); 'text' or entry of a string of characters; 'select' or entry of an element from a choice set (a command, an attribute value, an underlying application object); and 'quantify' or entry of a numeric value.

An example of a common interaction technique is a pop-up menu appearing over an object in response to a right mouse click, allowing the user to highlight different commands by moving the mouse over these, then either selecting one by left clicking on an item or none by clicking outside the menu. This is associated with the 'select' interaction task.

### Components of an Interaction Technique

- User action with an input device
- Input device response and associated events and feedback to user
- Supporting visual (and auditory) display of action, range of values or options available
- Feedback of current action, value, option via supporting display
- (Feedback through display of application object is not included as part of an interaction technique)

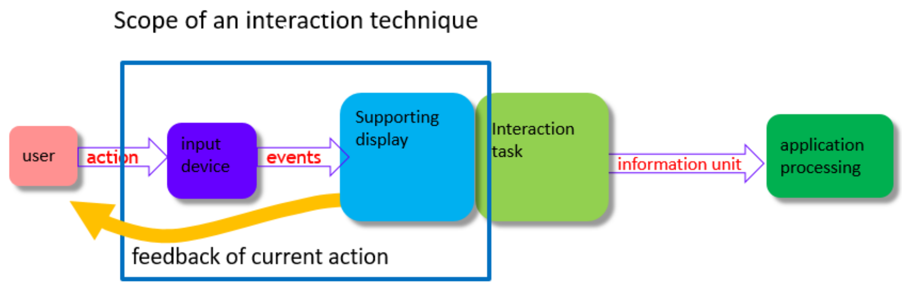


FIGURE A.1: The scope of an Interaction Technique

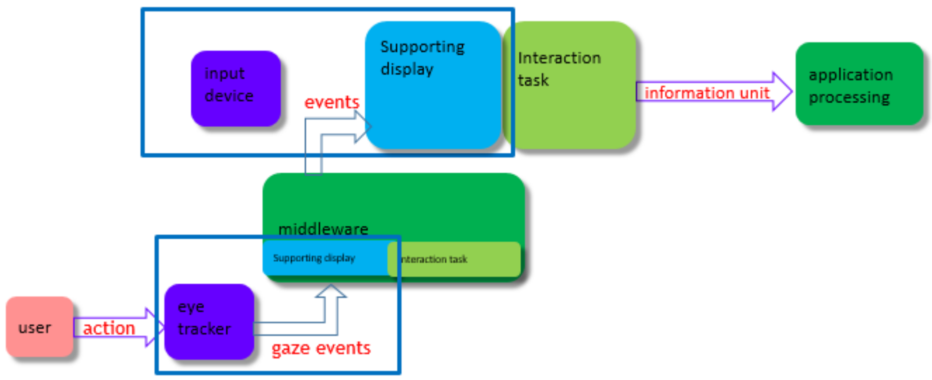


FIGURE A.2: Gaze-based Interaction, all events are generated by the middleware

## A.2 Gaze-based interaction techniques

Here a separate program shown, in the diagram as 'middleware,' takes input from the eye tracker, generates input events that normally come from the input device, say a mouse or a keyboard, and sends these to the target application. The target application treats these as if they had originated from the input device. With gaze-based interaction, all input events are generated in this way (see Figure A.2). The middleware has its own interaction techniques, which are overlaid on top of the target application's window.

## A.3 Gaze-assisted interaction techniques

Gaze-assisted interaction means that events generated by the input device are used together with events generated by the middleware. A mouse-move event in this scenario might be generated by the middleware in response to a change in gaze position, while a mouse-left-button-click event might be generated by the user clicking the mouse button (see Figure A.3. With gaze-based interaction, both events would have been generated by the middleware.

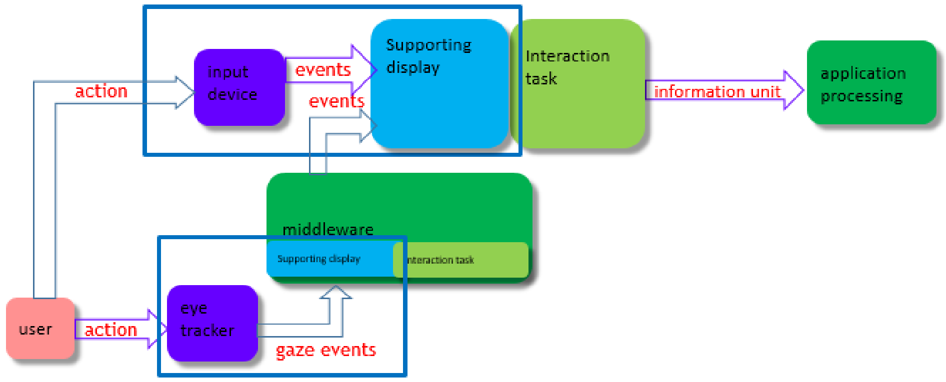


FIGURE A.3: Gaze-Assisted Interaction, some events are generated by the middleware, others originate directly from input devices, such as mouse and keyboard

## A.4 Middleware

This term refers to a piece of software that takes input from a device, such as an eye tracker, and generates events that are posted to the event queue of an applications. The event queue is managed by the operating system. These events might be mouse events, such as 'mouse left button click at location (x,y)'. The target application takes the event from its event queue and processes it no differently than if it had originated by a mouse action. The middleware may also update the system cursor position directly.

## A.5 Mode

In human-computer interaction, a 'mode' is a mapping between an input action and a system response or output. So, in different modes, the same input will have different effects or outputs. Clear feedback is important if modes are to be used so that the user knows what the current mode is. For example, if the 'format paint' mode is selected in a word processor, then selecting a piece of text, has the effect of applying previously selected format instructions to the text. Feedback about the mode is often shown by the shape of the cursor. In other modes, selecting the same piece of text would have other effects (or possibly no effect). In this research, a gaze event as an input action had different outcomes depending on the mode selected. The modes were defined in and selected from the middleware, so that different modes resulted in different events being sent to the target application.

## A.6 Motor impairments

In this research, the term means any restriction on the use of the mouse or keyboard. The restriction may be related the range of movement, precision of movement, or the duration over which effective operation of input devices can be sustained. It relates too to the ability to view a display in a fixed location over a period of time. It does not however include visual

| Groups of motor impairments | Prevalence | Estimated number in EU |
|---|---|---|
| Amyotrophic lateral sclerosis (AMD)/Motor Neurone Disease (MND) | 6 per 100,000 | 27,000 |
| Multiple scleroses (MS) | 30 per 100,000 | 135,000 |
| Cerebral palsy (CP) | 200 per 100,000 | 900,000 |
| Spinal cord injury | 8 per 100,000 | 36,000 |
| Spinal muscular atrophy (SMA) | 12 per 100,000 | 54,000 |
| Rett syndrome | 6.66 per 100,000 | 29.970 |
| Muscular dystrophy (MD) | 28 per 100,000 | 126,000 |
| Brainstem stroke | 153 per 100,000 | 688,500 |
| Traumatic Brain injury (TBI) | 150 per 100,000 | 675,000 |
| | Total | 2,671,470 |

TABLE A.1: Prevalence of different diagnoses and estimated total number in Europe, Table taken from Jordansen et al., 2005, p.11

impairments that make perceiving the information on the display difficult. Motor impairments may be the result of traumatic injuries or of diseases or congenital conditions. An estimate of the numbers of people with different types of impairment in the EU, as of July 2005, is shown in Table A.1.

# Appendix B

# Research Output 1

H.O. Istance and P. Howarth
*Keeping an Eye on your Interface: The Potential for Eye-Based Control of Graphical User Interfaces* Proceedings of HCI'94: People and Computers IX, Cambridge University Press, August 1994

*Reprinted with permission of Cambridge University Press*

# *Keeping an Eye on your Interface: The Potential for Eye-Based Control of Graphical User Interfaces (GUI's)*

**Howell O Istance[†] & Peter A Howarth[‡]**

[†] *Imaging and Displays Research Group, Department of Computing Science, De Montfort University, Leicester, UK.*
EMail: *hoi@de-montfort.ac.uk*

[‡] *Vision and Lighting Research Group, Department of Human Sciences, Loughborough University of Technology, Loughborough, Leicestershire LE11 3TU, UK.*
EMail: *P.A.Howarth@loughborough.ac.uk*

**This paper examines the issues surrounding the use of an eyetracker, providing eye-movement data, as a general purpose input device for graphical user interfaces. Interacting with computers via eye-movements is not in itself new, however previous work in the area has been directed towards interaction with purpose-built software which can take into account device limitations such as accuracy. This work investigates how one can interact with unmodified graphical interface software which normally requires mouse and/or keyboard input. The results of three experiments are discussed which have compared performance between the eyetracker and the mouse, and between different ways of emulating mouse button presses using the eyetracker data. The experiments as a whole consider a range of tasks from simple button presses to the more complex and demanding operations of selecting text, and they indicate the feasibility of using the eyes to control computers.**

**Keywords:** physically-challenged, eye-control, input device.

## 1. Benefits of Controlling Graphical User Interfaces by Eye

### *1.1. Overview*

The use of the eyes as a primary means of controlling input is appealing for a number of reasons.

First, it can be considered as a 'natural' mode of input and by-passes the need for learned hand-eye co-ordination to effect operations such as object selection. The user simply looks at a screen object they wish to select rather than using a hand-held pointing device, such as a mouse, to position a screen cursor over the object.

Second, one can expect performance benefits. If a user need only look at an object to acquire it, rather than having additionally to control and position a cursor by hand, speed of selection will be increased. Eye-based cursor control, in combination with other input devices for effecting a selection, holds the promise of increasing the narrow communication bandwidth from user to machine.

Third, it allows for interaction in situations where the user is unable to use their hands. This restriction may be a result of the task itself, where both hands are used for other purposes, or may be due to physical disability on the part of the user. The most commonly available input devices make no provision for users with severe disabilities, many of whom may have little or no control over their limbs and may only be able to make controlled movements with their eyes. A low-cost eyetracker, using eye position as a means of issuing commands via the user interface, offers a means of enfranchising this group with respect to access to modern graphical interface software.

### *1.2. Previous Research in Relation to the Anticipated Benefits of Eyetracking*

#### *1.2.1. Naturalness of Eye Movement*

The advantage of 'naturalness' in the use of eye-gaze data is that some insight is gained into the user's current intention during interaction with a computer, because before carrying out any action, you first look at what you are interested in. However, without an independent selection protocol, the difficulty lies in whether to make the inference that a user also wishes to select the item that he or she is currently looking at, or whether to assume that they are simply browsing.

In his work on naval command and control systems, Jacob (1993) investigated the advantages of natural interaction, via eye movement tracking, in the provision of 'non-command' interfaces. Non-command techniques attempt to facilitate interaction with the system without the need for the user to give it specific commands. This requires that the system makes some kind of inference about what the user wants to do by monitoring some data associated with user behaviour, such as eye movement, and responding accordingly. Consequently, Jacob rejected all eye-based selection mechanisms other than dwell-time (an extended fixation on a target) as he considered interactions that required external training, such as blinking, to be 'unnatural'. He studied the use of eye positioning of the cursor, and used both dwell time and an external hardware button as selection mechanisms. He presented and evaluated interaction techniques for object selection, menu item selection, control of scrolling text and switching the active window. The techniques described were based on a combination of eyetracker and mouse-button commands, and are not transferable for a general purpose input device for a 'command driven' GUI. This is because the target software was purpose-built to accommodate the different command protocols used.

Eye-based control has been used in another example of non-command interaction using eye position. Starker & Bolt (1990) provided viewers of a picture with information about particular features by assuming that the parts of the image of interest to a particular user would be revealed by whereabouts on the picture they were looking. The features of interest were determined by an index of 'focus of attention' which was calculated from matching eye-gaze position to features in the underlying image. Again, the problem here lies in producing decision rules about whether a person wants information about that part of the picture, or whether they are simply looking at it.

### *1.2.2. Anticipated Performance Benefits*

Ware & Mikaelian (1987) examined performance differences in terms of speed of response and error rates for simple target acquisition tasks. The screen cursor was moved under eye-control and different ways of issuing selection commands (screen button, hardware button and dwell-time) were compared for several different target sizes. Selection speed increased, and error rates decreased dramatically, when target size increased from 0.45 to 1.5 degrees of visual angle. Selection times of less than 1 second were recorded, and increasing the target size above 1.5 degrees produced no further improvement in performance. Ware and Mikaelian found that the dwell time and the hardware button selection protocol were equally fast and both were superior to the screen button condition in terms of mean error rates (22% in the screen button condition, 8.5% in the hardware button condition and 12% in the dwell time condition). They also observed that the eyetracker could be a faster selection device than the mouse for targets above 1.5 degrees. This conclusion was arrived at, however, by comparison with data reported by Card, English & Burr (1978) for mouse performance and not with data they had collected themselves for the same tasks. Hence this comparison is unlikely to provide a reliable indicator of the relative merits of the devices. One of the issues addressed in the research described in this paper is a direct comparison between mouse and eyetracker.

### *1.2.3. Special-needs User Devices*

The potential of the use of eye tracking for specialist input devices for physically-challenged users has already been recognised (Downing, 1985; Hutchinson et al., 1989). These systems do not necessarily use eye-position data to drive a screen-based pointing device, but may give feedback in different ways, such as highlighting the object being looked at on the screen. Downing developed an eye-gaze communicator and control system which also used dwell-time as a selection mechanism. He estimated that selection speeds with the device would enable a word generation rate of between 10 and 20 times that achievable with other devices used by a physically-challenged user group.

Ten Kate et al. (1980) studied different methods of providing communication aids controlled by eye. One used an eye switch for the selection of letters whilst the other used eye position recording and fixations on screen-targets corresponding to letters. Shaw et al. (1990) presented a low-cost control device, based on the sensing of eye-winks, for controlling a wheel chair. Different combinations of lid positions for each eye were used, and each mapped onto specific command actions. Rosen & Durfee (1978) used encoding of sequences of changes in eye-gaze directions and mapped these onto a vocabulary of communication items.

A significant feature in each of these systems was that the application or device being controlled by eye position was specially constructed for the purpose of improving communication or mobility, and consequently any limitations in the performance of the eyetracking device could be taken account of in the design. The challenge addressed in this research is enabling interaction with any software application running on a particular windows platform (in this case, MS-Windows™ ).

### 1.2.4. Challenges of using Eye Control for GUI's

There are many challenges which must be met if we are to produce a usable general-purpose device emulator for mouse and keyboard which allows interaction with a wide range of available, unmodified software. To do this, the following must be provided:

- A sufficient degree of pointing accuracy with the eye-based device to permit a realistic level of interaction with system and application objects.
- A means of signalling commands equivalent to key or button presses which is comfortable and will not interfere with device pointing accuracy.
- A means of disabling control of the cursor (equivalent to taking one's hand off a mouse).
- A degree of comfort which facilitates use of the input device for extended periods of time.
- A device which is of low cost and as portable as a PC.

In meeting these challenges, two points need particular consideration.

## 1.3. Implications of General Purpose Input Device Emulation

The requirement to enable interaction with unmodified software has two consequences. The first of these is that the angular size of the targets on the screen will have been determined independently of the capabilities of the eyetracker. These are determined by the size and resolution of the monitor used, the viewing distance from the subject to the screen, the actual graphical user interface used, and the size and type of window interaction objects it contains (such as buttons, menus bars, scroll bars). The second consequence is that the eyetracker device should permit the emulation of all of the commands and operations used to interact with the software. Simply emulating single mouse button clicks is not sufficient. It should also be possible to emulate click-and-drag and double-click actions with different mouse buttons, and to emulate keyboard operations to facilitate text input and key-based command input. Several screen-based keyboard products already exist, and in these products buttons representing keys appear on screen. The buttons are activated by a simple click operation with a mouse or other pointing device and cause a keystroke event to be sent to the underlying window system. With suitable calibration, it would be possible to look at a keyboard template, located off-screen, either above or below the display monitor and map the recorded eye position to the corresponding key. The advantage of an on-screen keyboard is that the cursor can give feedback as to where the system thinks the present point of gaze is whilst this information would not be available in the off-screen version. The disadvantage of an on-screen keyboard is that part of the screen must be given over to the keyboard application, unless a two-screen solution is considered.

## 1.4. Signalling the Equivalent of Mouse Button Presses

At present, there are two options for selecting the object at which the cursor points. One is to use an external device, such as a blow tube or a muscle-activated switch. The alternative is to use an eye-based protocol, where an 'unnatural' event is used to signal selection. Options available here are to stare or 'dwell' on a item for an abnormally long time, to close both eyes for a period longer than a natural blink, or else to use a wink. The challenge is to find a usable selection protocol which is relatively rapid, has a high hit-rate and a low false-positive rate.

Jacob (1993; Jacob et al., 1993) described the 'Midas Touch' problem associated with the use of eye-movement data in the context of 'non-command' interfaces, where dwell time is used to signal target selection. Essentially, the user can not look at anything for too long without also selecting it, as it is not possible to distinguish between the intention to select an object or merely to browse. This problem disappears when the user is required to make some explicit action in order to select an object.

The approach adopted in this research is that in the absence of information about which faculties any individual possesses, it should be possible to achieve all interaction via the eyes alone. However, it should also be possible, if desired or necessary, to add external devices to the system for signalling commands so that the eyetracker is used for cursor positioning only.

## 2. Hardware Requirements and Usability Issues

In this section, usability problems will be discussed which have been identified during the research carried out to date. In summary, the main usability issues for the eyetracker input device concern:

- Ease and speed of initial device calibration.
- Ease of setting up the cameras to obtain stable, well-focused images of individuals' irises and pupils.
- The need for subsequent adjustment of initial calibration values and whether this can be done during use of the device by the user.
- Postural constraints, tolerance to head movement and consequent discomfort and fatigue.
- The level of training necessary to use the device effectively.
- Individual characteristics of the user, such as droopy eyelids or long eyelashes.

### *2.1. Eyetracker Systems and Problems with Eyetracking*

In most activities, people are normally able to, and do, move their head whilst continuing to look at objects in their field of view. The consequence for a system attempting to record where the eye is looking is the disambiguation of eye movements needed to keep the object of regard the same when the head moves, and eye movements resulting from changing the object of regard. Eyetracker systems use a variety of techniques including corneal reflection and processing video images of the pupil to measure eye position with respect to a set of optical components. Cameras and other optical devices can be head-mounted and eye position measured with respect to the head. Head-mounted devices do not constrain head position, but are obtrusive in that the subject is required to wear a helmet or headband. If the line of gaze from the eye to some external reference point, such as a display screen, is needed, then the orientation and position of the head has to be measured as well as the position of the eye, and the two sets of data aggregated.

If, on the other hand, the cameras and optical components are fixed in relation to the room then eye position can be measured directly with respect to the room and objects within it, such as display screens. However, room-mounted systems require the head to be kept within the field of view of the camera or optical measuring device. Normally the head would need to be constrained to enable statically mounted cameras to maintain a reasonable image of the pupil, and this is usually effected by means of a head-rest. Whilst acceptable in a laboratory, this degree of constraint is not going to be generally acceptable in an input device, and will inevitably lead to postural fatigue after periods of extended use.

Room-mounted systems do exist which allow a certain degree of head movement. The more expensive of these use servo-controlled mirrors to track the eye and maintain a stable camera image, although this is usually accompanied by some loss of measuring accuracy. The range of movement which can be tolerated is typically in the region of ±15 cm in the horizontal and vertical planes and even in the fore and aft plane (ASL Inc, 1992). Generally, the greater the degree of constraint on head position, the greater the possibility for more accurate measurement. Room-mounted systems will also have vertical and horizontal limits to the angular range over which the eye can be tracked. In some cases, the vertical range of measurement can be prohibitively small (a 15-degree range at a viewing distance of 500 mm translates into a vertical distance of 134 mm, smaller than the height of most display screens).

### 2.2. Device Accuracy and Precision

Device *accuracy* can be considered as the distance between the computed line of gaze and the true line of gaze and is expressed in terms of visual angle. Device accuracy of 1°, for example, at a viewing distance of 500 mm to a display screen would translate into approximately 9 mm on screen.

Device *precision*, on the other hand, is related to the distribution of measurement points about a mean whilst the subject is fixating on an object, the mean of the distribution corresponding to the accuracy. This variation around the mean comes about because, in addition to any measurement device imprecision, the eye will naturally fixate on a number of different points around a object of regard between saccades.

In trying to drive readily-available software the resolution of the output from the eyetracker to a device driver generally needs to be comparable with that of a mouse, although the actual pointing resolution required will depend on the underlying application. For example, pointing to the start of a word within a piece of normal-sized text is clearly more demanding than selecting a command button in a dialog box. In systems which use a processed video image of the pupil, a relatively large image is necessary to obtain the maximum resolution from the eyetracker as the range of horizontal and vertical eye movement corresponding to the bounds of the screen has to occupy as much of the range of the measuring instrument as possible. The consequence of this is that small head movements could cause the pupil image to go outside the field of view of camera, and data (and control) would be temporarily lost.

All of the above factors will influence the stability of the cursor under eye control with respect to where the subject is looking. Two problems are manifest which make accurate control of the cursor difficult, and these correspond to the descriptions of device accuracy and precision. The first is an offset between the cursor and the point of regard, and the second is jitter in the cursor position. Clearly, the smaller the target to be selected, the greater the effect these problems will have. The extent to which these affect task performance will be discussed in Section 4.

### 2.3. The System Used in the Current Research

The eyetracker system (Micromeasurements System 7000 pupillometer) is a binocular, infra-red video-based machine, which measures eye position 60 times per second. The video image of each pupil is processed to enable the horizontal and vertical co-ordinates of the pupil centre to be determined. In addition the pupil area is computed. To compensate for head movements a compensation algorithm examines the position of an image of a small infra-red light source reflected in the subject's cornea. The position of this image is then compared with the position of the pupil centre. Similar movements of the two occur during translational movements of the head, whilst differential movements of the two images occur during eye rotation.

### *2.3.1. Device Emulation Under MS-Windows*

The signal from the eyetracker is received on the serial port of the PC running the target application software. The incoming data is filtered and scanned before being processed by the device emulator software. The filtering is achieved by means of a simple buffering system, which can be adjusted to damp the jitter in the observed cursor position. The extent to which filtering techniques can be applied are constrained by the need for real-time control of the screen cursor. If the user perceives a lag in the cursor position then he or she is likely to look back at the cursor, causing the cursor to be 'chased' around the screen. The filtered data is mapped onto a screen co-ordinate, clipped to the boundary of the screen and used to update the cursor position. The stream of incoming position and pupil size data is scanned for temporal changes used to characterise a dwell or a wink. If such an event is detected then an appropriate window system message is generated corresponding to, for example, a mouse button press.

### *2.3.2. Binocular Versus Monocular Systems*

The fact that the eyetracker system is binocular is significant in that data from both eyes can be used firstly, to increase bandwidth by using data from each eye for different purposes and, secondly, to improve accuracy by having two estimates of eye position. The former advantage is utilised in facilitating click-and-drag actions (see Section 4.2). With respect to the latter point, a measure of convergence can be obtained which offers the prospect of introducing corrections for fore and aft movement of the head. This can be done through knowledge of the positions of both eyes when fixating on the same point, at a known viewing distance, when the device is calibrated. Fore and aft movements of the head would give rise to a discrepancy between the two eyes in the respective calculated eye positions, which will be a function of distance from the screen to the eyes.

The advantage of a monocular system is that it is likely to be less expensive, as it would only need the one camera. Also, it is likely to be simpler to set-up and calibrate initially. The advantage of the binocular system is that it can provide additional information about the eyes, which can be used for increasing system accuracy and for facilitating selection. For example, with a monocular system one needs additional feedback to the user about 'button-up' or 'button-down' status, (see Section 4.3) during a click-and-drag operation, whereas this is not needed with a binocular system.

### *2.3.3. System Calibration*

Calibration involves establishing a zero point for the eyetracker data with reference to the PC screen, and a scaling factor to map from eyetracker co-ordinates to screen co-ordinates. The usability issues in initial calibration partly relate to the time needed to establish a stable image for both eyes and the time taken to execute the calibration procedure. The latter is usually very short (40–60 seconds) in comparison with the former (4–5 minutes for a new user). A number of individual features will influence the ease with which satisfactory images can be produced. These include eye colour, the normal position of the upper lid relative to the iris, and in some cases the use of eye make-up.

If the user is to initialise the use of the device without help from another person then they will need to be given feedback about the quality of the pupil images available to the eyetracker in relation to the relative positions of the head and camera. The difficulty in subsequent correction by the user of drift from the initial calibration values is partly dependent on whether a badly located screen cursor under eye-control will be used for changing these values.

### 2.3.4. *Head Movement, Head Position and Postural Fatigue*

At present, the system used for the research described here is run with the two cameras, mounted on tripods, each at an angle of about 25 degrees to either side of the head. This allows the subject an uninterrupted view of the host computer screen. The subject sits with their chin placed on a chin-rest, and their brow against a curved, padded, brow-bar. This arrangement allows the subject to make small lateral head movements (which, as explained above, do not affect cursor position) but does not allow excessive fore and aft head movement. Trials carried out with certain categories of disabled users have shown that the constrained posture necessary to maintain a stable pupil image is currently a major usability problem. This problem will be alleviated by using a redesigned head-rest.

### *2.4. Implications for a Low Cost Device*

In specifying a low-cost device, two solutions suggest themselves. The main problem to contend with in any system is that of head movement. A light-weight head-mounted binocular camera system with separate head-tracking to enable a world gaze position to be computed could be used. The technology to produce these relatively inexpensively has recently been developed for VR applications. Alternatively, one or two small video cameras may be mounted on a frame that fits onto the monitor of the machine to be controlled. This mount could also contain devices such as LEDs which could be used to give continual feedback to the user about the position of their head with respect to the cameras.

### *2.5. Usability Issues in Comparison with Other Eye-Based Systems*

The problems encountered during the use of the present system have clear analogues with those reported elsewhere (Jacob, 1993). Shaw et al. (1990) reported that their eye-wink control device took 30 minutes to set up, which included establishing the measurement thresholds between the open and closed states of the eye. Jacob et al. (1993) made particular reference to the stability and repeatability of eye position measurements and concluded that the currently available commercial eyetrackers were still not adequate for use in practical applications with disabled users.

## 3. Summary of User Performance Investigations with the Eyetracker

The concerns for the research programme to date have been as follows:

i. The feasibility of using the eyetracker as a pointing device. This investigation examined performance when using the cursor under eyetracker control to point at simple targets on screen, and enabled a comparison (in terms of speed and accuracy) to be made with a mouse.

ii. The different means of emulating mouse button actions using the eyetracker.

iii. The use of the eyetracker for click-and-drag operations in the context of text selection, compared with a mouse.

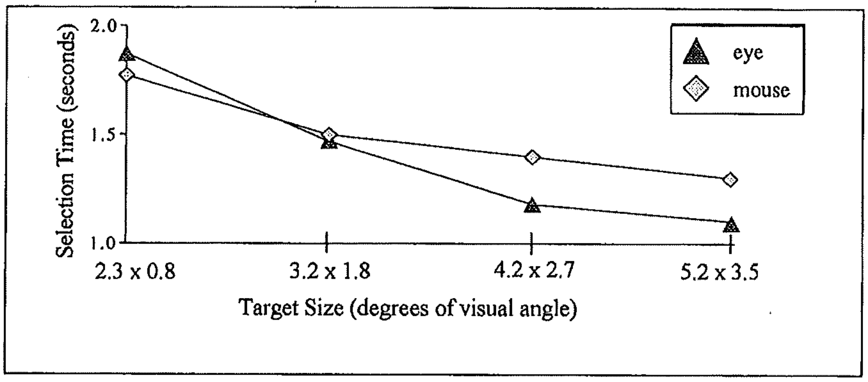


Figure 1: Selection times for different target sizes.

### *3.1. Comparison Between Eyetracker and Mouse for Pointing at Targets*

An experiment was conducted to compare selection speed when a cursor was positioned on a rectangular on-screen button using a mouse and the eyetracker respectively. For both means of cursor positioning, *selection* of the target was achieved by pressing a mouse button. Six subjects took part in the experiment, all of whom were novice mouse users. These subjects were used so that unskilled eyetracker users were compared with unskilled mouse users. Each subject completed 5 trials with a mouse and 5 trials with the eyetracker as the means of positioning the cursor. A trial consisted of 12 runs, 3 for each of the 4 target sizes. Each run was initiated by the subject selecting a 'start' button on the screen. The target then appeared at a random position and distance from the 'start' button, and the time taken to select the target was recorded.

Mean target selection times are shown in Figure 1 averaged across all subjects and all selection distances. Eye control of the cursor leads to faster acquisition times where accuracy is less important (i.e. larger target sizes) but performance declines in comparison with the mouse as demands on accuracy increase and target sizes become smaller. The selection times in the eyetracker condition are significantly faster for the two larger sizes ($p < 0.001$ and $p = 0.002$ respectively) but not for the two smaller sizes.

Eyetracker selection times in this experiment are longer than those reported by Ware & Mikaelian (see Section 1.2.2). Two factors can account for this difference. First, the distance to target appears to be much shorter in Ware & Mikaelian's experiment. In their procedure, all targets were permanently visible on-screen and were adjacent to each other in a column, with the start button situated in the centre of the column. Using this configuration one would expect shorter selection times. Second, the level of previous experience amongst subjects may have been different. One of their subjects was one of the authors and presumably was not a novice eye-tracker user. Again, one would expect better performance because of this.

In conclusion, this first experiment has demonstrated that the eyetracker-based device *can* be used to point at simple screen targets, and that performance is broadly similar to that of a mouse for these targets. The eyetracker does then hold the promise of enabling faster acquisition than a mouse, providing that the targets are sufficiently large.

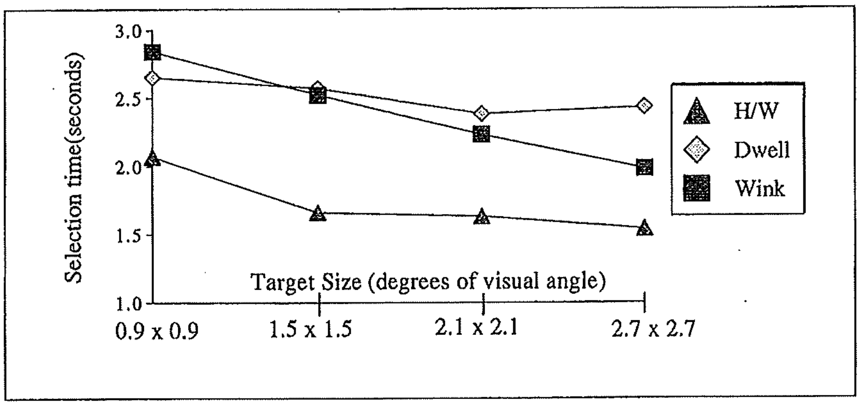


Figure 2: Selection times for different selection protocols.

### *3.2. Investigations of Protocols for Emulating Mouse Commands*

The second set of experiments investigated performance differences between alternative ways of emulating mouse commands. The target selection task was similar to the experiment described above. The following protocols were investigated:

i. Wink: one eye was used for cursor positioning whilst the other eye was used to emulate mouse button states. Subjects could choose which eye they wanted to use for cursor control and which for button control. The eye open corresponded to button up and the eye closed corresponded to button down. Using the binocular system, this protocol offers the possibility of emulating click-and-drag operations as the cursor can be moved with the other eye whilst the 'button eye' is either open or closed.

ii. Dwell: an extended fixation at the same place on the screen for a specified dwell time was treated as the equivalent of a mouse click. During the experiment three different dwell conditions were investigated, representing different combinations of dwell duration (from 100 msec to 200 msec) and sizes of area within which the cursor was considered to be stationary. In comparing different protocols we have only shown the results when the smallest area and the longest time (200 msec) were used, as the numbers of errors in this condition was significantly lower than the other two dwell conditions. It will be seen later that accuracy is a critical feature of user performance with the eyetracker.

iii. Hardware Button (H/W): selection in this case was made using the mouse button. This condition was included to enable comparison with the results obtained in the first experiment and to enable the relative efficiencies of the eye-only protocols to be assessed.

Four subjects took part in experiment 2. Each subject completed 5 blocks of 10 trials. A trial consisted of 3 runs of each of the 4 different sized target buttons, as in experiment 1. Within a trial, the order of target size presentation was randomised over the 12 runs. No initial training was provided, as part of the purpose of the experiment was to investigate performance changes over an extended number of trials.

The data in Figure 2 shows selection time for each target size averaged over error-free trials and all subjects. Data from the first 20 trials has been excluded as performance improved

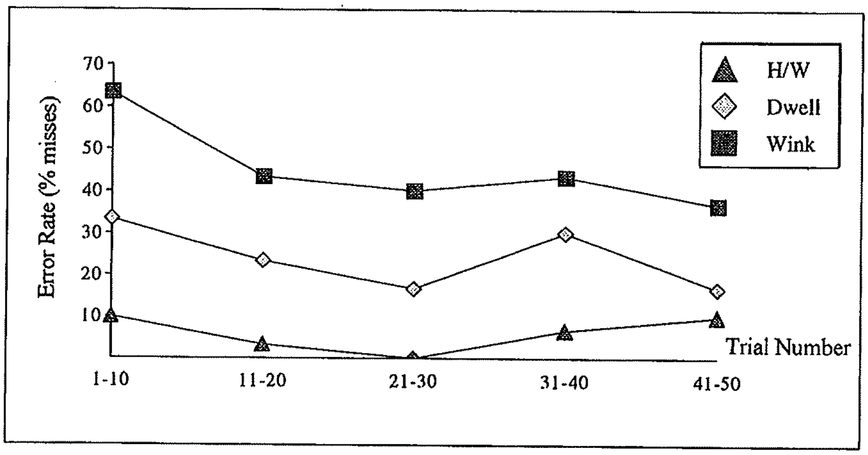


Figure 3: Error rates for selecting smallest target (size 0.9 degrees) over repeated trials.

over these trials. It can be seen in this figure that selection times in the Hardware button condition were significantly faster than both of the eye-based conditions, for all target sizes. The selection time in the Wink condition was affected most by target size.

Turning now to selection errors, it was found that there was no significant difference in error rates between the three protocols for the two larger targets, but performance did decrease for the smallest target. Figure 3 shows the error rates for the smallest target (0.9 $\times$ 0.9 degrees) used in the experiment for the three protocols across a total of 50 trials. Error rates are defined here as the percentage of trials where the *first* selection was not on target (i.e. the target button was missed).

The data is averaged for blocks of 10 trials across subjects. There is a clear improvement in performance across all three protocols during the first 20 trials, after which there is no apparent further improvement. There is a fixed ordering of error rates between protocols, and averaging over the last 30 trials (after performance had stabilised) the error rates were Hardware Button 5.5%, Dwell 21% and Wink 40%.

Although these error rates would seem high, we must view the performance in the light of the fact that this is the worst performance encountered, and that the error rate for larger targets was low. The higher error rate in the Wink condition and the increase in selection time for this condition is possibly due to movement induced in the 'cursor controlling' eye by closing the 'button' eye, which was observed to cause the cursor to be moved off target before the selection event was detected. The closure of one eye usually results in the lowering of the upper lid of the other eye, and if this partially obscured the pupil then the cursor on the screen would also be lowered — thereby causing an increase in errors. This problem can be overcome easily by simply taking the cursor position prior to the wink onset as being the screen point to be selected.

Concerning the dwell protocol, there is trade-off between the size of area on screen within which the cursor must remain for it to be considered as 'stationary' and the risk of missing the target if it is small in comparison with the 'stationary' area. Selection times for the other dwell conditions investigated which used larger acceptance areas were faster than those shown

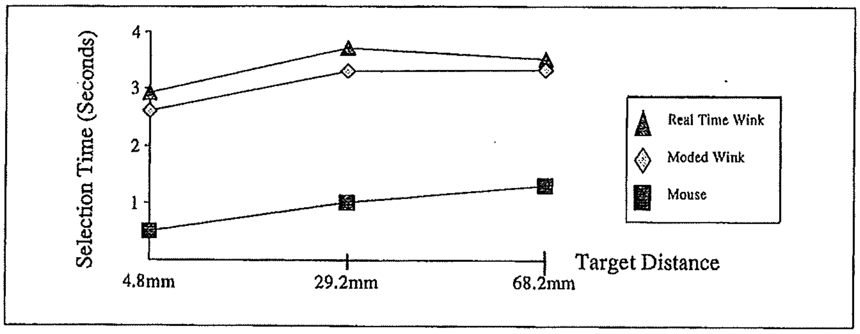


Figure 4: Selection times for different dragging distances.

in Figure 2, but the number of error-free trials was much lower and error rates consequently much higher.

In conclusion, the Hardware button shows the best overall performance. The Wink protocol offers the prospect of a usable protocol with large target sizes and where the cost of a miss is small. It is relatively easy to play (and complete) the Windows game of Solitaire, which requires clicking and dragging piles of cards, using the Wink protocol. However, both of the eye-based protocols are associated with higher error rates for smaller targets, which presently limits the usability of the eyetracker. On the other hand, the relatively low error rate with the Hardware Button is encouraging for the use of the eyetracker for cursor positioning together with an external selection device, such as a blow tube, for target selection.

### *3.3. Investigation of Text Selection Under Eye-Control*

This experiment compared performance between the eye-tracker and the mouse for a more complicated task involving click-and-drag to select a piece of text.

Using these two input devices, the effect of selection protocol ('real-time' wink and moded wink) and target size (font size) were studied. The 'real-time' wink protocol was the same as the Wink protocol studied in the second experiment. The moded wink is a monocular variation of this in which a first wink sends a button-down event and a second wink sends a button-up event. Moving the eye between winks with both eyes open is equivalent to moving the mouse with the button down. This means that only one eye need be tracked, and hence a monocular tracking system is sufficient, although as pointed out earlier additional feedback is necessary to indicate to the user whether the device is currently in 'button-up' or 'button-down' mode.

Nine subjects completed 3 trials of 63 runs each, using both eyetracker and the mouse. Each trial was partitioned into two sets of runs containing 27 and 36 runs respectively. The first set contained 3 levels of distance to the start of the target text from the 'start' button and three levels of target text size. The second set contained two levels of selection protocol, three levels of distance of target text to be dragged over and three levels of target text size. Each subject was given a brief practice session of 20 minutes with each device.

The subject was required to select a section of highlighted text from within a body of text, each trial being started with an on-screen button outside the text body. Selection required the cursor to be moved to the start of the target text, followed by a click action, followed by dragging the cursor across the text with the 'button' down and finally releasing the 'button' at

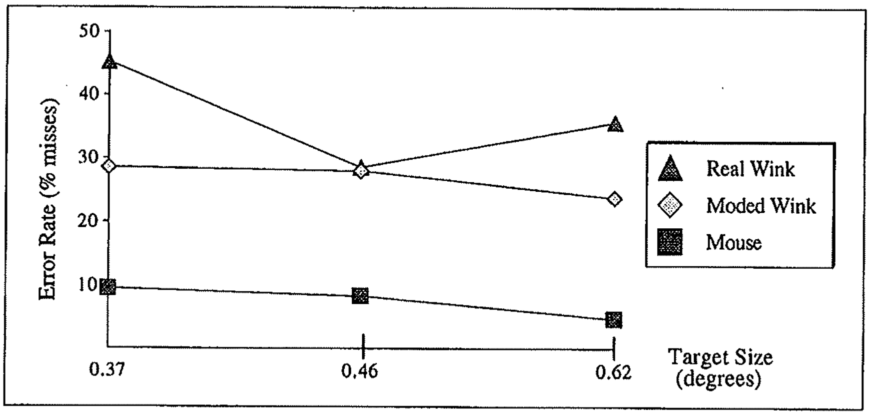


Figure 5: Error rates during dragging for different font sizes.

the end of the piece of text. Figure 4 shows the effect of the length of the text to be selected on the selection times for the input devices and protocols. The times are averaged over the three different font sizes used in the experiment.

Considering speed of dragging (selection of the far end of the target text) the mouse was significantly faster than the eyetracker for all dragging distances. The moded wink condition for the eyetracker produced slightly, but significantly, shorter selection times than the real-time wink condition. It was noted during the trials that the closed state of the 'button' eye during dragging caused the fixation point of the cursor eye to be lowered slightly. The subject had to compensate for this which made the dragging procedure more difficult and time consuming. This is a similar artefact to that observed previously during selection of targets with the Wink protocol, in experiment 2.

The 'target width' was the width of the area within which the first or the second 'click' action could be made. This area increased with font size. Although not shown in the figure, it was also found that selection times for the eyetracker conditions were reduced by increasing the target width whereas these times for the mouse were unaffected.

In this experiment, as before, error rates are defined as the percentage of trials where the *first* selection attempt was outside the target area. Error rates for both devices and protocols are shown in Figure 5 for the different font sizes used and include errors made both in selecting the beginning of the text section and its end. The trials were conducted at a viewing distance of 600 mm so that the target sizes in Figure 5 correspond to target widths of 3.9 mm, 4.8 mm and 6.5 mm.

The cost of a miss is, relatively speaking, very high in the context of text selection. If the error is made when anchoring the end of the selected text then the whole operation, including anchoring the start of the selected text, has to be repeated. For applications such as this one, the error rates with the eye-based protocols and the eyetracker are, at present, very high — even with large font sizes. This constitutes the main obstacle to the use of a mouse-emulator for text selection. However, areas for system modification have been identified which will result, it is anticipated, in a reduction of these error rates.

## 4. Conclusions

A major influence on the error rates recorded for the eyetracker is the deviation between the point of gaze and the position of the cursor on the screen. This can be attributed to a combination of factors including noise in the measurement system, errors and drift in the initial calibration as well as jitter in the eye movements themselves. In the first two experiments, subjects were able to adopt strategies to compensate for small disparities, such as fixating to the side of the target and observing in peripheral vision whether the cursor was bounded by the target button. Use of a similar strategy was much harder in the case of text selection as it was necessary to ascertain which two letters bounded the cursor, which was far more difficult using peripheral vision. The accuracy and stability of the screen cursor with respect to the point of gaze will limit the extent to which one can achieve successful interaction with small application, and window system, objects. Work is at present underway to examine the use of adaptive filtering techniques to improve stability.

An alternative route is to accept the inherent device inaccuracies and concentrate on virtual keyboard emulation as a means of interacting with GUI's. In general, authors of GUI software provide key-based alternatives to the mouse for interacting with control objects. Typically, within a dialog box, it is possible to use the tab key to move between control elements and use the cursor control keys for operations such as changing list selections. Menu items can be selected via modified character identifier (accelerator) keys (e.g.[alt] + [c]). Small, customised, on-screen keyboards which enable direct access to these commands offer an alternative to the primary normal means of interacting with GUIs, namely via the mouse.

Using this approach may be far more fruitful than attempting to develop a more accurate and precise system which would enable direct emulation of mouse input. Here, one is able to accept the lower pointing accuracy and precision of a low-cost eye-tracker together with a greater degree of tolerance to head movement. Using this approach, however, one has to accept the restriction that most interaction is accomplished by screen-based keys which map directly onto equivalent keyboard commands, rather than having direct control over the graphical interface.

The ergonomic problems associated with the relatively high levels of visual and postural fatigue currently associated with extended periods of device use need careful attention. This presents a greater challenge if one is to satisfy the goal of a low cost system, whilst ensuring a sufficient degree of device accuracy and allowing the user greater freedom of head movement to alleviate postural and fatigue problems. This requirement makes the use of a head-mounted device attractive in spite of the additional obtrusiveness imposed by the device itself. Finally, the issue of initial calibration and correction of calibration values by the user needs to be addressed if a person is to be able to use the input device largely unaided. Although problems remain to be solved, encouraging progress has been made with respect to each of these challenges and the authors believe that eye-gaze interaction at low cost with GUI's is feasible. It now remains to determine the optimum approach to realise this.

## Acknowledgements

Several research students have made valuable contributions to the work described in this paper, in terms of ideas as well as implementation and running experiments. The contributions of Simon Paterson and Chris Schrimpl (Department of Computing Science, De Montfort

University) and Darren O'Connor and Simon Layton (Department of Human Sciences, Loughborough University) is gratefully acknowledged.

## References


ASL Inc (1992), *Eyetracker Instruction Handbook, 3rd Edition*, Applied Science Group Ltd.

Card, S, English, W K & Burr, B (1978), "Evaluation of Mouse, Rate-Controlled Isometric Joystick, Step Keys and Text Keys for Text Selection on a CRT", *Ergonomics* 21, pp.601–613.

Downing, A R (1985), "Eye-Controlled and Other Fast Communicators for Speech Impaired and Other Physically Handicapped Persons", *Australian Physical and Engineering Sciences in Medicine* 8 (1).

Hutchinson, T E, White, P, Martin, W N, Reichert, K C & Frey, L A (1989), "Human–Computer Interaction Using Eye-Gaze Input", *IEEE Transactions in Systems, Man and Cybernetics* 19 (6).

Jacob, R (1993), "Eye Movement-based Human–Computer Interaction Techniques: Towards Non-Command Based Interfaces", in *Advances in Human–Computer Interaction*, R Hartson & D Hix [eds.] #4, Ablex.

Jacob, R, Leggett, J, Myers, B & Pausch, R (1993), "Interaction Styles and Input/Output Devices", *Behaviour & Information Technology* 12 (2), pp.69–79.

Rosen, M J & Durfee, W K (1978), "Preliminary Report on EyeCom: An Eye Movement Detection and Decoding System for Non-Verbal Communication", in *Proceedings of the Annual Conference on Systems and Devices for the Disabled*, pp.167–171.

Shaw, R, Crisman, E, Loomis, A & Laszewski, Z (1990), "The Eye-Wink Control Interface: Using the Computer to Provide the Severely Disabled with Increased Flexibility and Comfort", in *Proceedings of the 3rd Annual IEEE Symposium on Computer-Based Medical Systems*, IEEE Publications, pp.105–111.

Starker, I & Bolt, R A (1990), "A Gaze-responsive Self-disclosing Display", in *Proceedings of CHI'90: Human Factors in Computing Systems*, J C Chew & J Whiteside [eds.], ACM Press, pp.3–9.

Ten Kate, J, Frietman, E, Stoel, F & Willems, W (1980), "Eye Controlled Communication Aids", in *Medical Progress Through Technology* #8, Springer-Verlag, pp.1–21.

Ware, C & Mikaelian, H (1987), "An Evaluation of an Eyetracker as a Device for Computer Input", in *Proceedings of CHI+GI'87: Human Factors in Computing Systems and Graphics Interface*, J M Carroll & P P Tanner [eds.], ACM Press, pp.183–188.

# Appendix C

# Research Output 2

H.O. Istance, C. Spinner and P. Howarth
*Eye-based control of standard GUI software* Proceedings of HCI'96: People and Computers XI, Springer-Verlag, August 1996

*Reproduced with permission of Springer*

# *Eye-based Control of Standard GUI Software*

**Howell Istance†, Christian Spinner† & Peter Alan Howarth‡**

† *Department of Computer Science, De Montfort University, The Gateway, Leicester, Leicestershire LE1 9BH, UK.*

EMail: *hoi@dmu.ac.uk*

‡ *Department of Human Sciences, Loughborough University, Loughborough, Leicestershire LE11 3TU, UK.*

EMail: *P.A.Howarth@lboro.ac.uk*

**This paper discusses the design and initial evaluation of a visual on-screen keyboard, operated by eye-gaze, intended for use by motor-impaired users. The idea of an on-screen keyboard controlled by eye or by other modalities is not new. However, the keyboard presented here is different in two important respects. First, it enables interaction with unmodified standard Graphical User Interface (GUI) software written for able-bodied users, and provides eye-based control over menus, dialogue boxes, and scrollers; it is not solely designed around the need to enter text. Second, the software architecture enables the keyboard to respond to events generated in the windows environment by the application it is controlling. This allows the keyboard to adapt automatically to the application context by, for example, loading a specific set of keys designed for use with particular menus whenever a menu is displayed in the target application. Results of initial evaluation trials are presented and the implications for improvements in design are discussed.**

**Keywords:** eye-control, visual keyboard, physically-challenged, disability, handicapped.

M. A. Sasse et al. (eds.), *People and Computers XI*
© Springer-Verlag London 1996

# 1 Introduction

An important element in enabling people with various forms of motor impairment to work in office environments is to provide the means whereby they can use the same software tools at the workplace as their able-bodied colleagues. This requires building interaction devices which not only enable effective interaction with a single application but are sufficiently flexible to allow the user to switch quickly between different software applications without the need to reconfigure the device. Eye-based interaction is attractive for a number of reasons. First, it offers the prospect of reducing learning time by providing a 'natural' means of pointing at a displayed object on-screen. Second, moving the eye is fast and positioning a pointer on a required object can be done quickly, even if other interaction components may be relatively time-consuming compared with normal usage of keyboard and mouse. Third, users with severe degrees of motor impairment often retain good ocular motor control, and so devices based on eye-movement may be used by a larger range of users with motor impairments than devices relying on other muscle groups.

The advantages have, however, to be offset against the known problems of eye-based interaction. These include the level of accuracy with which eye position can be measured, the degree of fine control that the user can exercise over eye movement and the need to be able to disengage eye-control whenever the user wishes to look at the screen without issuing commands. To overcome these problems, eye-based control may be combined with other input modalities, such as speech, so that eye-gaze is used for pointer positioning only and other modes of input are used to make selection actions (equivalent to those normally made by the mouse button). However, at this stage we have restricted ourselves to the use of the eyes alone for both cursor control and selection, rather than investigating such combinations. Previous work (Istance & Howarth, 1994) investigated the use of eye-gaze for emulation of a mouse to interact directly with standard Graphical User Interface (GUI) applications, and it was concluded then that an indirect 'soft' control device, such as an on-screen keyboard,* offered a means of overcoming many of the problems associated with the direct eye-based interaction approach. If these problems are overcome, then one has no need to invoke additional input modalities, and one does not then have to rely upon the user having any form of motor control other than over their eyes alone.

The idea of a visual keyboard displayed on a screen and operated by some external device is certainly not new. Indeed, visual keyboards have been developed in the past for use with a variety of interaction devices, such as joysticks and mice, as well as eye-gaze control. Some of these have emulated normal keyboards, but in doing so have been restricted almost entirely to text entry. These have not allowed control over the variety of objects, such as those contained in menus and dialogue boxes, found in modern direct manipulation GUI software. There are many applications, such as Web browsers, which have very limited requirements for text entry but instead require interaction with displayed documents, such as scrolling or clicking on links, as well as

*In this paper, '*virtual keyboard*' is used as a general term to describe alternative keyboards (which do not necessarily have a visual presentation on the screen), whereas the terms *visual keyboard* and *on-screen keyboard* (used interchangeably) always refer to a keyboard displayed on a screen. A keyboard can be made up of a number of different key configurations, each of which is termed a *keypad*.

interaction with menu items. Other visual keyboards have been developed to control specific software applications only, for example communication and control systems, and can not be used as general purpose input devices.

# 2 Functions and Design of Existing Virtual Keyboards

## 2.1 *Single Application Keyboards*

An example of an eye-controlled text-entry system is ERICA, an application for PC-DOS systems with a menu-based interface (Hutchinson et al., 1989; White et al., 1993). The interface of ERICA was limited in that it could not display the complete screen-based keyboard within one window (Frey et al., 1990). This is a recurrent problem with applications designed for eye-based interaction, in which interface objects like buttons and menus need to be made sufficiently large to overcome the limitations of pointing accuracy inherent in eye-control systems. EyeScan (Eulenberg et al., 1985), BlinkWriter (Murphy & Basili, 1993) and EyeTracker (Friedman et al., 1981) were further examples of applications which supported eye-gaze controlled text entry, but which similarly could not be used as a general input device for standard software.

### 2.1.1 *Keyboard Layouts*

Several studies have examined the respective efficiencies of alphabetical or QWERTY arrangements of keys — e.g. (Roussos, 1992; Quill & Biers, 1993; Douglas & Happ, 1993; Mackenzie et al., 1994). The recommendations for key arrangements vary from study to study and depend heavily on assumptions about the user and the pointing device to be used. The studies have generally neglected the flexibility available in presenting on-screen keyboards in comparison with the normal hardware keyboard. Effective and successful keyboard emulation requires more than simply duplicating the key arrangements of conventional keyboards on-screen. The screen representation of the WiViK visual keyboard of Nantais et al. (1994) matches, more or less, a conventional keyboard with QWERTY layout. Their design has the advantage that all keys are selectable with a single step. However, WiViK is intended to be operated by a head motion input device which permits a higher spatial resolution of the key matrix than in the case of an eyetracker. A single interface which accommodates *all* the keys of a conventional keyboard remains problematic for an eye-controlled on-screen keyboard.

### 2.1.2 *Techniques for Alphanumeric Text Input*

Early text entry devices for physically-challenged users arranged letters according to their frequency of occurrence in text and allowed indirect selection by stepping through a matrix (e.g. MAVIS system in Schofield (1981); HandiWriter in Ten Kate et al. (1980). This technique has the advantage of accommodating a large set of characters and other symbols within a single template. Demasco & McKoy (1992) proposed a virtual keyboard model which, besides scanning single characters, also supports scanning of words. This technique is unnecessarily restrictive for eye-based pointing and it is better suited to devices reliant on more limited forms of motor control. Nevertheless, laying out character keys on the basis of frequency of use is an important design consideration. As selecting larger targets is easier than selecting smaller targets when using an eyetracker, frequently used keys can also be within the keypad (static sizing) or they can be expanded temporarily when the pointer moves within the key (dynamic sizing), as well as being positioned appropriately.

Operating a visual keyboard by eye is comparable to the one-finger approach of a novice user who also delays each keystroke after the position of a desired key has been recognised. It is a highly sequential task as text must be entered character-by-character with forced delays between the 'eyestrokes'. Consequently, text entry becomes tedious and keystroke savings through a word prediction system not only have the potential to improve speed but also to reduce errors and user fatigue.

The study of Koester & Levine (1994) examined user performance in text entry tasks with word prediction by considering the trade-off between the number of keystrokes and the additional cognitive and perceptual loads imposed by having to select from the presented word choices. They concluded that the cognitive cost of presenting a set of word choices, and explicitly selecting one, largely eliminated the performance advantage of keystroke savings. However, their results depended very much on the degree of disability and the chosen method of controlling the keyboard. For instance, there may be a different trade-off when using a mouthstick for typing on conventional keyboard than in the case of a visual keyboard, where the input and output channel is the same. Indeed, the absence of the need to switch between external keyboard and screen suggest that there will be a lower cognitive load when a visual keyboard is in use.

Swiffin (1988) reported keystroke savings of between 30% and 60% through the use of a *Predictive Adaptive Lexicon* (PAL). This is based on prediction of words which have been entered so far in a document and is therefore useful when word redundancy occurs. This approach is probably a good choice for programming tasks. Demasco & McKoy (1992) used a technique in their keyboard model which took a compressed message and expanded it into a sentence.

With the exception of PAL, word prediction as well as word selection techniques have the advantage of helping to avoid typing errors. In the context of the visual keyboard, where text input is a laborious task, any additional correction aid becomes an important consideration because of the laborious nature of the correcting task. If the visual keyboard is used to control a state-of-the-art word processor which incorporates a spell checker, then the performance lost as a consequence of inputting incorrect text is greatly reduced.

## 2.2 General Purpose Keyboard Emulators

EyeTyper, developed by Friedman et al. (1985) was an eye-gaze controlled keyboard which could be used in place of the standard keyboard of an IBM-PC. Hence, it was transparent to the host computer's software but its architecture was based on a single keypad template which did not allow more than a very simplistic keyboard emulation.

More recently, the visual keyboard developed by Bishop & Myers (1993) at the University of Iowa supports control by an eyetracker device. The arrangement of character keys is advantageous and groups frequently-used characters towards the centre of the screen. In our experience, a discrepancy between visible cursor position and actual eye position is more likely in the screen corners, and the keys and button sizes of their visual keyboard are arranged to take this into account. However, the

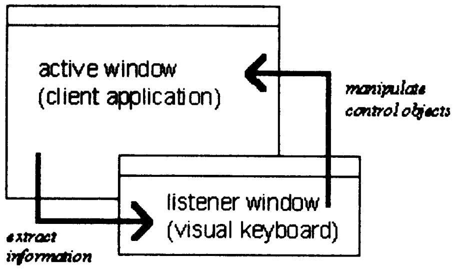


**Figure 1:** Active window/listener window architecture.

support for editing text is quite limited. Key combinations to highlight text, such as <*Ctrl+Shift+Right*> are not possible, and whereas characters are available for text input, they are not available for command selection (e.g. <*Alt+F*>). Thus it is not possible to make use of the rich functionality of a word processor like MS-Word, thereby loosing the advantage of services such as spell-checking.

As a general purpose device for use with a text processor, a visual keyboard has to be able to satisfy the following requirements:

- Text entry.
- Command selection.
- Error correction.
- Controlling the visibility of text document.
- Key combination support.
- Precise caret location with the text.

However, when used with other client applications, the priorities for support can be different. For instance, browsing through hypertext documents requires point-and-click actions on links in documents, selection of menu items and scrolling through documents. Editing text, for example, is far less important with this application, and a visual keyboard primarily designed for text entry tasks would no longer be optimal. Furthermore, space limitation makes it impossible to display all keys within a single window if they are to be of adequate size for eye control.

## 3 Issues in Designing the Visual Keyboard

### *3.1 Communication Between the Client Application and the Visual Keyboard*

Virtual keyboards which were primarily designed to provide an alternative form of text entry and control of a character-based user interface typically lack support for effective interaction with GUI objects. Shein et al. (1991) discussed the development of virtual keyboards for GUI environments and identified functional text entry and manipulation of GUI objects as the two main problem areas.

Jacob (1993) presented a number of eye-based interaction techniques for components of the WIMP interface. Unfortunately, most design features of the underlying user interface are not transferable to standard GUI environments. However, the architecture of that system, which is based on an *active* and *listener* window is suitable for transfer. The visual keyboard and standard client applications can enter into the same relationship as it is possible to extract information from the client applications (Figure 1). Multitasking GUI environments and client-server architectures offer this possibility through their event-driven communication. Events associated with the target application (active window) can be inspected by the visual keyboard application (listener window).

Jacob's (1993) concept of active and listener window requires separating *object selection* from continuous display of that object's *attributes*. For instance, if a menu is displayed by the client application (such as a word-processor) in the active window, information about the currently-highlighted menu item can be extracted and displayed in the listener window as part of the keyboard.

The listener window, in addition to displaying attributes, must also support the need for manipulation of the selected control object in the active window. This can be done by presenting the attributes in a form which is more appropriate for eye-controlled interaction. One example based on this concept is to present a part of the active window as an enlarged view in the listener window. A similar zoom-in interaction technique was proposed by Starker & Bolt (1990) for use with an eyetracker system and is also used for overcoming device inaccuracies for pixel operations within bitmap editors.

## 3.2 Supporting Current Task Context

Graphical User Interface objects such as menus, buttons, list boxes and dialogue boxes occur in nearly all applications. These objects are controlled in the same way regardless of the tasks for which they are used. Most objects can be manipulated with a small subset of keys. For example, when interacting with a menu, the:

*<Alt>*, *<Left>*, *<Right>*, *<Up>*, *<Down>*, *<Esc>* and *<Enter>*

keys are sufficient. This characteristic allows the provision of keypads which are best suited to a particular group of interface objects. If the visual keyboard can detect when the client application has a dialogue box displayed over its main window, then the visual keyboard can automatically load and display the appropriate 'dialogue box keypad'. Similarly, if the keyboard can detect the currently highlighted item in a client application menu, then the text of that menu item can be used within the visual keyboard itself.

## 3.3 Supporting Different Interaction Methods

Typically, interface objects can be controlled with a variety of combinations of input devices. For instance, a single menu command can be selected through:

- Selecting menu item through a mouse-click.
- Selecting menu item through a sequence of key-based shortcuts.

- Mouse-click at smarticon of toolbar.
- Keyboard accelerator (e.g. CTRL-S to save).

The preferred interaction method depends very much on the experience and preference of the user, and ideally the visual keyboard should support all interaction methods in order to address the needs and preferences of different *types* of user. Smarticons, tool- and colour-palettes however are frequently restricted to mouse input.

## 3.4 *Emulating Command Actions by Dwelling*

Using this technique, the user has to fixate, or dwell, on a key for some specified time in order to select a key. The duration used has a major impact both on performance and on user errors in cases of unintentional command execution. In previous studies the dwell time has varied from 100ms (Istance & Howarth, 1994) over the more common 500ms (Cleveland & Cleveland, 1992; Shaw et al., 1990) up to 1000ms for menu commands (Jacob, 1993). Nevertheless, even long dwell times cannot eliminate the risk of unintentional commands.

The performance loss caused by long dwell times settings has recently been addressed by Nantais et al. (1994). They developed a visual keyboard for MS-Windows which was controlled by head motions in conjunction with the dwell time protocol. Instead of assigning an overall dwell time for all the keys, a key selection probability model was introduced with very short dwell times for those keys which are likely to be chosen next, based on lexical probability calculations from the previous input. Dwell times were reduced by between 20% and 60% while the error rate increased not more than 3%. This technique is therefore potentially of interest for eye-based text entry. We can conclude that different *types* of keys can also be assigned different dwell times depending on whether speed or preventing unintentional operation is important. In addition, we can assign different dwell times depending upon the size of the keys.

## 3.5 *Adjusting the Size and Position of Client Window and Keyboard Window*

In order to make best use of the workspace available, the visual keyboard should not be of a fixed size. The arrangement of the listener keyboard window and the size of client application windows are likely to change during interaction. As part of the screen will be occupied by the visual keyboard window, it is necessary to provide some functionality to facilitate easy rearrangement of the relative size and position of these windows. Some applications benefit from the use of a maximised window whereas other applications require only a small part of the screen (e.g. calculator, character table). This will also accommodate differences in the user's pointing ability as enlarging the keyboard window will increase the size of the keys contained within it.

## 3.6 *Customizing the Keyboard*

The requirements discussed so far indicate the importance of end-user customisation, however it is necessary to limit the amount of customisation necessary during normal interaction to prevent the keyboard becoming overloaded with customisation functionality. The problem can be minimised through:

- Providing suitable configurable defaults for the keyboard (e.g. dwell time settings).
- Providing an off-line keypad editor to enable new keypads to be created which are suited to different applications.
- Providing the option to create keys which invoke macros. For example, a keypad intended for use with Netscape might contain a key to display a particular page, and this would invoke a macro with the keystrokes necessary to display the URL prompt and enter the characters defining the data.
- Self-adaptation of visual keyboard to the current application context (e.g. automatically loading a menu keypad).
- Configuration of client application to enhance access (e.g. enlarge font, show toolbar).

### *3.7 Summary of Design Requirements*

In summary, there are a number of requirements that a visual keyboard controlled by eye needs to satisfy:

- It should support both keyboard and mouse emulation.
- It should support effective interaction with GUI components such as the scrollable lists, text fields and buttons found with dialogue boxes and not be solely designed around the need for text entry.
- It should provide mechanisms to compensate for the inaccuracy inherent in eye-based control.
- It should enable the individual to customise the device to suit individual preferences concerning tasks within specific applications, but should allow the user to switch between different applications without the need for device reconfiguration.

## 4 Design and Operation of the Visual Keyboard

### *4.1 System Architecture*

The eyetracker provides raw data on the positions of the left and right eye to the host machine (Istance & Howarth, 1994). This data is processed completely separately from the visual keyboard. The software responsible for this uses the current gaze position to move the mouse pointer. The visual keyboard runs as an application, and is sensitive to the position of the mouse pointer. Its window is overlaid on top of the active client window. If the mouse pointer remains within the area of the key for a specified time (dwell time), a Windows event corresponding to the key is generated. In this way, it is possible to set different dwell times for different types of key. The

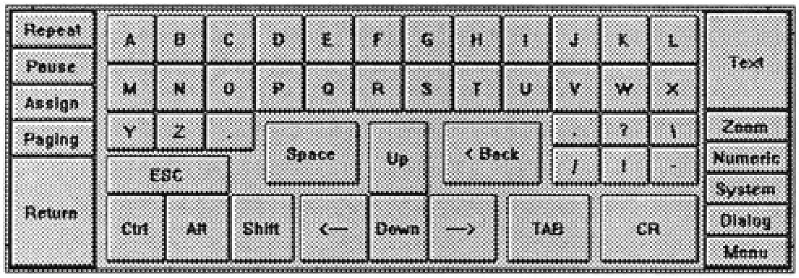


**Figure 2:** Visual keyboard interface (text keypad).

---

visual keyboard is thus completely separate from the eyetracker system providing the data, and although we have concentrated on eye-control, the keyboard could perfectly well be used with other types of driver device, such as a joystick or mouse.

## *4.2 Overall Design of the Visual Keyboard*

The keyboard itself can be considered to be made up of a number of distinct sections, the centre section of which is referred to here as a keypad. The design of each keypad is based around the need to support different types of command, and each keypad can be selected from a keypad menu. As an example, Figure 2 shows the keyboard with the text keypad loaded, and the three sections of the keyboard are:

1. General keyboard system commands menu (left).

2. Text keypad (centre).

3. Keypad selection menu (right).

Of the general system commands, *<pause>* engages and disengages eye control of the pointer, *<assign>* selects a keypad (Section 4.4) *<paging>* moves a client application window (Section 4.6) and *<return>* closes the keyboard and displays the keypad editor.

In the keypad selection menu and the system command menu, each key expands when the pointer first moves into it and reverts to its original size when the pointer moves over another key. When the cursor moves off either menu, the last key expanded remains enlarged. In Figure 2, the user has just loaded the text pad. This makes dwelling with the eye within the key area easier without the penalty of taking up window space for keys which are not being used (the same principle as a pull-down menu). Additional keypads may be created in the keypad editor and added to the menu of standard keypads or may replace them in the menu.

## *4.3 Keypad Design*

### *4.3.1 Text Keypad*

The current text keypad (shown in Figure 2) incorporates two important design decisions. First, it is based on an alphabetical arrangement of characters (although this could be replaced by a Dvorak or QWERTY arrangement. Quill & Biers (1993) recommend a QWERTY arrangement, but this is rejected for this prototype because

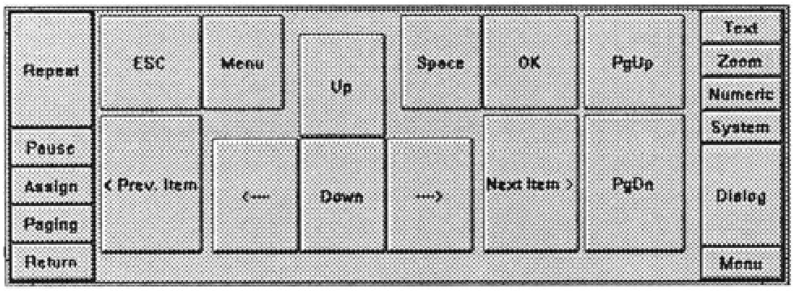


**Figure 3:** Dialogue keypad.

the physically-challenged user is not likely to be familiar with it. Second, the key arrangement gives prominence to the *<cursor>* keys due to their relative importance not only during text entry, see also (Gould et al., 1985), but also during text editing.

### *4.3.2 Dialogue Keypad*

The dialogue keypad shown in Figure 3 contains the keys necessary to control dialogue boxes. There are fewer keys here than in the text keypad and therefore the individual keys are larger. The main keys are the *<cursor control>* keys, the *<next item>* and the *<previous item>* keys (corresponding to *<tab>* and *<shift-tab>* respectively) and the *<escape>* key.

### *4.3.3 Menu Keypad*

The *menu keypad* (not shown) is another context-sensitive keypad for interaction with the menu system. The main keys here are the *<cursor control>* keys, the *<menu>* key (corresponding to the *<alt>* key), the *<escape>* key and a key which contains the text of the currently highlighted menu item. Selecting this key selects the menu item (and is equivalent to pressing the *<carriage return>* key).

### *4.3.4 Zoompad*

Many users may have difficulty in precisely positioning the cursor, and so a zoompad has been incorporated (Figure 4). The zoompad shown here emulates mouse commands. It incorporates the equivalent mouse command selection by means of radio buttons. The user looks at a region of the client window and after a brief dwell interval has expired, the region is copied and enlarged into the keypad. The user effects a click action within the zoom area and the event is sent to the corresponding part of the client window.

## *4.4 Overriding Automatic Keypad Selection*

The visual keyboard automatically loads and displays keypads appropriate to the current task context as, for example, when the user accesses the menu-system in the client application or when a dialogue box is displayed. However, this self-adaptation mechanism is not always desirable. For instance, if the first control object in the dialogue box is a text element, the user may prefer the text keypad for the initial activation. In such cases, the user can make explicit assignments using the *<Assign>* button in the left hand part of the keyboard system command menu. Subsequently,

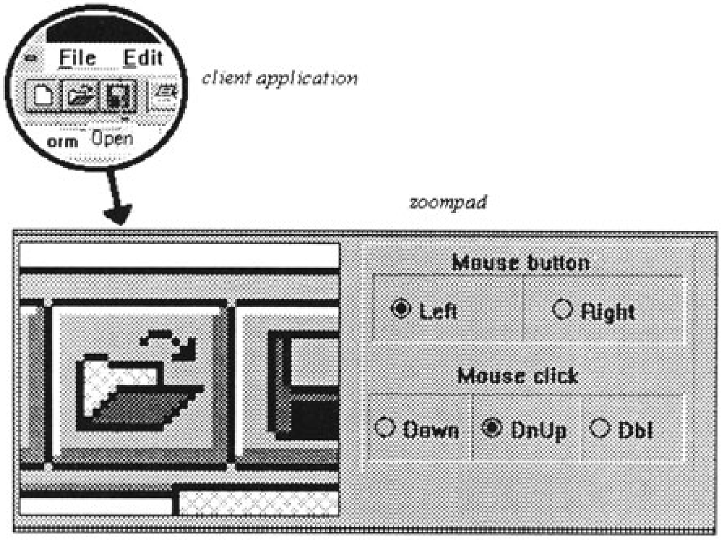


**Figure 4:** Zoompad.

| Dwell object | Dwell time (ms) |
|---|---|
| <Cursor control> keys | 1000 |
| <Enter> key (menu keypad only) | 2500 |
| System command menu | 1000 |
| Keypad selection menu | 500 |
| Target selection (zoompad) | 2000 |
| Any other key | 1500 |

**Table 1:** Dwell time settings.

the assigned keypad will be loaded automatically when the client application context is the same.

## *4.5 Object Dwell Time*

Dwell times for different types of keys can be set individually and these can be altered by the user to reflect personal preference. The values in Table 1 show the different settings used as initial values in the evaluation trials (described in Section 5). The user is warned about expiration of the dwell time by a change in the cursor. The mouse pointer is represented by a circular cursor within the keyboard and this changes to show a black spot in its centre just prior to the end of the dwell interval. If the user does not wish to select the key, they may look away at this point and selection is then inhibited.

## *4.6 Window Arrangement*

The window arrangement of the client application and visual keyboard is supported by three different approaches. In combination, these attempt to compensate for the fact that the on-screen keyboard has to take up a finite part of the available screen area, thereby reducing the area available for the application.

- *Paging*: In the case that the window of a client application is partially overlapped by the visual keyboard or even requires the whole screen, the *<paging>* command moves the client application window so that either its top half or bottom half is displayed in the area of the screen above the keyboard window.

- *Heading*: Windows and in particular dialogue boxes which appear initially in the centre of the screen will be automatically moved to the top of the screen if there is overlap. In the case that the remaining screen space is too small, the *<paging>* command will be selected.

- *Bounding*: Re-sizing the visual keyboard will automatically re-size the client application window.

## 5 Outcomes of Initial Evaluation Trials

This section reports results from initial evaluation trials with a modern word processor (MS-Word 6.0). More evaluation work remains to be done, both with this application and with other types of application. The trials do, however, give a good indication of the success, or otherwise, of some of the design ideas that have been included in the keyboard. The input device was the eye-control system described previously (Istance & Howarth, 1994).

### *5.1 Selection of Tasks*

A set of five tasks, which constituted an integrated exercise, was completed by each subject:

1. Run application and open text document.

2. Enter a few lines of text (an address).

3. Save the file.

4. Edit the existing text.

5. Require help about a specific problem.

The tasks, and keystroke level actions required to execute them, are transferable to most word processors. Some tasks, such as loading a file via a 'file open' dialogue box or saving a file, are standard across many applications. The means of completing all tasks (except the last) were prescribed to the subjects to ensure that alternative input styles were used. Collectively, the tasks incorporated several forms of command selections requiring the usage of the menu system, toolbar and accelerators and consequently resulting in the use of different keypads. Usage of the help system was included as a task for two reasons. First, many MS-Windows applications have associated help systems. Second, the help system is based on the hypertext concept, and so allows some assessment of the usability of the visual keyboard for hypertext systems.

## 5.2 Usability Issues

One objective of the evaluation trials was to judge whether the visual keyboard was effective enough to perform the specified tasks. Task completion, and the support of the visual keyboard to enable users to recover from errors, were considered to be major indicators of keyboard efficiency, and keyboard effectiveness could thus be measured by the number of tasks completed and by the number of uncorrected errors made. Measuring effectiveness makes it possible to determine whether eye-based visual keyboard interaction with off-the-shelf GUI software is feasible, even if improvements in efficiency are needed.

In addition, in order to evaluate the efficiency with which the tasks were performed the effort required for the user to correct errors, or to edit text, serves as a useful performance indicator. Comparisons could be made between (virtual) key-based interactions (e.g. navigation with cursor control keys) and the mouse emulation supported by the keyboard in the form of the zoompad. Subjective data was collected in a post-trial interview on issues such as error handling and satisfaction with the functionality provided through different keypads.

## 5.3 Subjects

Five able-bodied users, recruited from academic staff and students, acted as subjects. It was felt that, at this initial stage, the evaluation trials did not require physically-challenged users as subjects. One subject used spectacles and another subject normally used contact lenses, and all subjects had either normal or corrected-to-normal vision. All subjects reported having previous experience with the GUI of MS-Windows and all were familiar with MS-Word. Two subjects had already had practice in using an eyetracker but none had previously operated an eye-controlled visual keyboard.

## 5.4 Results and Discussion

All subjects were able to complete the tasks which were given to them. Moreover, all subjects were able to recover from errors and consequently the usability objective of an effective visual keyboard was met in this context. The mean time spent on the completion of all tasks was 19 minutes per subject whereas the total time, including recalibration and repetition of tasks due to eyetracker problems, required on average 38 minutes. The task completion time of the whole task sequence did not vary across subjects by more than three minutes either side of the mean.

### 5.4.1 Text Entry and Feedback

On average, subjects were able to enter their address in about seven minutes (task 2) which corresponds to a text entry rate of only one word per minute. This is very inefficient and it was observed that most errors occurred during this task. Subjects frequently unintentionally entered a character twice and consequently the time spent on correcting those errors had a major impact on text entry efficiency. The reasons for this lay partly in the techniques used by subjects to get feedback, partly due to a lack of training, partly due to the dwell time being too short, and partly due to a delay in updating cursor position.

Two subjects looked for feedback in the text document after each character was entered whereas the other subjects entered a sequence of characters before checking. When the subject's task completion times were compared, the first approach was found to be more efficient. This was because errors were recognised earlier and consequently the subjects required less cursor control keystrokes to return to the site of the error. However, one might expect reductions in key location and selection times in the second approach as the user's attention remained on the keyboard for longer periods of time. There is clearly room for improvement in the rate of text entry, however even the present rate may be acceptable if text entry is limited to a few letters, such as when entering a file name or a help item to search for.

### 5.4.2 Interacting with Dialogue Boxes

During tasks 1 and 5, (see Section 5.1) navigation problems occurred when interacting with dialogue boxes. In some cases, subjects mistook the control object with the input focus and so manipulated interface elements by keystrokes which were actually designated for a different object. The visual feedback showing the current input focus was not always clear. These problems are caused by a lack of feedback when moving the input focus from one control to another. When an ordinary keyboard is used, it is possible to press a key and observe changes in the visual appearance of a control simultaneously. Using the eyes to 'press' a virtual key prevents the user from observing the effects of the command as it is taking place. However, during these tasks, unintentional key presses occurred far less frequently than with the text entry task, as one might expect given that the dialogue keypad had fewer, and therefore larger, keys.

### 5.4.3 Combinations of Keystrokes

The fourth task incorporated the use of shortcut keys for stepping from one word to the next in order to select and highlight text. Most subjects had no difficulties in combining two or three keys and hence were able to move the caret with shortcut keys rather than by a series of cursor control keystrokes.

### 5.4.4 Dwell as a Means of Activating Commands

Problems associated with interference between mouse events generated directly by the eyetracker software and similar events generated by the visual keyboard were apparent. These arose on occasions when subjects were reading a document or browsing a dialogue box. While subjects were aware that there is some form of response when dwelling too long within the visual keyboard area, the possibility of an event happening inside the client window was not apparent. Whilst it would be possible to disable all event generation in the client window by the eyetracking software, there is a case for letting the user interact directly with the client application (by the eyetracking software sending events directly to it) as well as by using the visual keyboard. The advantage lies in greater flexibility and not always having to use the keyboard on-screen. The disadvantage lies in the possibility of generating unwanted events in the client window. The feedback provided by the change in the cursor, which was intended to warn that dwell time was about to expire, was generally misinterpreted by subjects who thought that the change signified that the key had been pressed.

### 5.4.5 Feedback on Modifier Key States

All subjects were observed to have difficulties in remembering the current state of the modifier keys, for example, forgetting that the <*Shift*> key was locked, and this issue needs addressing. If another keypad was selected after locking the modifier key, then feedback indicating that a subsequent keystroke would lead to a key combination was lost. Errors were also made even though the keypad remained visible, but the effect of the locked key was not obvious. For example, moving the carat with the cursor control keys to correct a typing error with the <*Shift*> key still locked resulted in the text becoming highlighted rather than the carat simply being moved. A strategy such as altering the colour of the keypad during the time the shift key was enabled would provide the user with feedback about its status.

### 5.4.6 Mechanisms for Changing Keypads

All subjects were able to operate the menu-based keypad selection mechanism, although frequently this required more than one attempt. This is acceptable because the keypad selection is based on a short dwell time (500ms). Once the desired keypad button had been 'acquired' subjects could easily keep the pointer within that button because of its large size. The use of expanding, 'fish-eye' buttons has been shown to be particularly successful for this type of eye-based interaction. Furthermore, the self-adaptation by the visual keyboard by loading the appropriate keypad automatically was also successful in reducing user input.

### 5.4.7 User Preferences for Interaction Styles with the Visual Keyboard

In general, the most preferred means of command selection was the use of the zoompad to select the smart icons of the toolbar, followed by menu item selection. The least preferred option was using shortcut keys and accelerators. Preferences appeared to depend on the expertise with MS-Word. For instance, one subject reported frequently using shortcut keys for text editing and preferred to use the keypad supporting use of the shortcut keys. A follow-up trial where all tasks, excepting the text entry task, were performed using the zoompad showed considerable increases in speed.

## 5.5 Additional Trials using a Web Browser

Following the trials with the word processor, trials using Mosiac™ to browse documents located on the Internet were carried out using the same version of the visual keyboard without any modifications. Subjects were required to go to the home page of a computer science department, to find the home page of a particular person, to find a link to a paper from the page and finally to find the conclusions section of the paper. There was no prescribed means of completing the task.

Subjects were able to complete the task without difficulty. In this case, navigating through the document was an important issue and subjects used either the <*page-up*> and <*page-down*> keys or used the zoompad to zoom in on the scroll-bar at the side of the client window. These informal trials demonstrated the utility of the keyboard with a completely different piece of software and the benefits of supporting different ways of completing tasks using the keyboard.

## 6 Conclusions

This work has demonstrated how a visual keyboard controlled by eye can be used to interact with standard software produced for the able-bodied user and thus allow the physically-challenged user to benefit from the wealth of software produced for modern GUI environments. This includes being able to access the Internet using existing browsers without the need for any modification either to the browser or to the keyboard.

Furthermore, many GUI applications require precise pointing ability which is often lacking in eye-controlled systems (Istance & Howarth, 1994) and the visual keyboard is capable of assisting with these tasks. Adequate solutions for non-keyboard sensitive GUI objects have been considered problematic in the past (Shein et al., 1991; 1992). The zoompad has been shown to provide an effective solution here.

Further work is required to improve the text entry rate achievable with this keyboard. Part of the problem here lies with the eyetracking system and its associated data processing software, rather than with the keyboard itself. A major problem was the need to correct unintentional keystrokes rather than the time required to locate and press an individual key. The next phase in the development will look closely at the causes of this and will examine means of improving text input rates. In addition, means of editing and interacting with existing text using the visual keyboard will be studied more closely. At the workplace, the visual keyboard is perhaps more likely to used for editing existing documents than for original document creation.

Future work with the visual keyboard will focus on the issue of feedback and examine ways of overcoming the visual separation between the keyboard and the region of client window providing feedback on the effects of commands. Additionally, it is intended to examine how direct eye-based interaction with the client application could be integrated with the use of the visual keyboard.

The major conclusion of this project is that the visual keyboard can be considered as a valuable low-cost enhancement to the eyetracker with the capability to compensate for its limitations as a pointing device. It has demonstrated that effective interaction with standard modern software applications using eye-based interaction techniques is entirely possible. This will greatly enhance the possibilities for physically-challenged users to work with the same software products and on a more equitable basis with their able-bodied colleagues.

## References


Bishop, J. B. & Myers, G. A. (1993), "Development of an Effective Computer Interface for Persons with Mobility Impairment", *Proceedings of the Annual Conference on Engineering in Medicine and Biology* **15**(3), 1266–1267.

Cleveland, D. & Cleveland, N. (1992), Eyegaze Eyetracking System, *in* "Imagina: Images Beyond Imagination, Proceedings of the 11th Monte-Carlo International Forum on New Images", pp.1–7.

Demasco, P. W. & McKoy, K. F. (1992), "Generating Text From Compressed Input: An Intelligent Interface for People with Severe Motor Impairments", *Communications of the ACM* **35**(5), 68–77.

Douglas, S. D. & Happ, A. J. (1993), Evaluating Performance, Discomfort, and Subjective Preference between Computer Keyboard Designs, *in* G. Salvendy & M. J. Smith (eds.), "Proceedings of the 5th International Conference on Human–Computer Interaction (HCI International '93)", Elsevier Science, pp.1064–1072.

Eulenberg, J. B., King, M. T. & Patterson, H. (1985), Eye-Controlled Communication, *in* "Proceedings of Speech Technology '85 — Voice Input and Output Applications", pp.210–211.

Frey, L. A., White, K. P. & Hutchinson, T. E. (1990), "Eye-Gaze Word Processing", *IEEE Transactions in Systems, Man and Cybernetics* **20**(4), 944–950.

Friedman, M. B., Kiliany, G. & Dzmura, M. (1985), An Eye Gaze Controlled Keyboard, *in* "Proceedings of the 2nd International Conference on Rehabilitation Engineering", pp.446–447.

Friedman, M. B., Kiliany, G., Dzmura, M. & Anderson, D. (1981), The Eyetracker Communication System, *in* "1st National Search for Applications to Aid the Handicapped", John Hopkins University, pp.183–185.

Gould, J. D., Lewis, C. & Barnes, V. (1985), Effects of Cursor Speed on Text-Editing, *in* "Proceedings of CHI'85: Human Factors in Computing Systems", ACM Press, pp.7–10.

Hutchinson, T. E., White, K. P., Martin, W. N., Reichert, K. C. & Frey, L. A. (1989), "Human–Computer Interaction Using Eye-Gaze Input", *IEEE Transactions in Systems, Man and Cybernetics* **19**(6), 1527–1534.

Istance, H. O. & Howarth, P. A. (1994), Keeping an Eye on your Interface: The Potential for Eye-based Control of Graphical User Interfaces (GUI's), *in* G. Cockton, S. Draper & G. Wier (eds.), "People and Computers IX (Proceedings of HCI'94)", Cambridge University Press, pp.195–209.

Jacob, R. J. K. (1993), Eye Movement-based Human–Computer Interaction Techniques: Toward Non-command Interfaces, *in* R. Hartson & D. Hix (eds.), "Advances in Human–Computer Interaction", Vol. 4, Ablex, pp.151–190.

Koester, H. H. & Levine, S. P. (1994), "Modeling the Speed of Text Entry with a Word Prediction Interface", *IEEE Transactions on Rehabilitation Engineering* **2**(3), 177–187.

Mackenzie, S., Nonnecke, B., Riddersma, S., McQueen, C. & Meltz, M. (1994), "Alphanumeric Entry on Pen-based Computers", *International Journal of Human–Computer Studies* **41**(5), 775–792.

Murphy, R. A. & Basili, A. (1993), Developing the User–System Interface for a Communications System for ALS Patients and Others with Severe Neurological Impairments, *in* "Designing for Diversity; Proceedings of the Human Factors and Ergonomics Society", Vol. 2, Human Factors and Ergonomics Society, pp.854–858.

Nantais, T., Shein, F. & Treviranus, J. (1994), "A Predictive Selection Technique for Single-digit Typing With a Visual Keyboard", *IEEE Transactions on Rehabilitation Engineering* **2**(3), 130–134.

Quill, L. L. & Biers, D. W. (1993), On-screen Keyboards: Which Arrangements Should Be Used?, *in* "Designing for Diversity; Proceedings of the Human Factors and Ergonomics Society", Vol. 2, Human Factors and Ergonomics Society, pp.1142–1146.

Roussos, P. (1992), Effects of Keyboard Layout on Children's Performance and Interaction with Computers, Master's thesis, University of Leeds.

Schofield, J. M. (1981), Chapter ?, *in* P. A. Samet (ed.), "Microcomputer-based Aids for the Disabled", Heyden.

Shaw, R., Crisman, E., Loomis, A. & Laszewski, Z. (1990), The Eye-Wink Control Interface: Using the Computer to Provide the Severely Disabled with Increased Flexibility and Comfort, *in* "Proceedings of the 3rd Annual IEEE Symposium on Computer-Based Medical Systems (CBMS'90)", IEEE Computer Society Press, pp.105–111.

Shein, F., Hamann, G., Brownlow, N., Treviranus, J., Milner, M. & Parnes, P. (1991), WIVIK: A Visual Keyboard for Windows 3.0, *in* "Proceedings of the 14th Annual Conference of the Rehabilitation Engineering Society of North America (RESNA)", pp.160–162.

Shein, F., Treviranus, J., Hamann, G., Galvin, R., Parnes, P. & Milner, M. (1992), New Directions in Visual Keyboards for Graphical User Interfaces, *in* H. J. Murphy (ed.), "Proceedings of the 7th Annual Conference on Technology and Persons with Disabilities", California State University, pp.465–469.

Starker, I. & Bolt, R. A. (1990), A Gaze-Responsive Self-Disclosing Display, *in* J. C. Chew & J. Whiteside (eds.), "Proceedings of CHI'90: Human Factors in Computing Systems", ACM Press, pp.3–9.

Swiffin, A. L. (1988), A Predictive and Adaptive Communication System for the Handicapped, Master's thesis, University of Dundee.

Ten Kate, J. H., Frietman, E. E. E., Stoel, F. J. M. L. & Willems, W. (1980), "Eye-Controlled Communication Aids", *Medical Progress through Technology* **8**, 1–21.

White, K. P., Hutchinson, T. E. & Carley, J. M. (1993), "Spatially Dynamic Calibration of an Eye-Tracking System", *IEEE Transactions in Systems, Man and Cybernetics* **3**(4), 1162–1168.

# Appendix D

# Research Output 3

R. Bates and H.O. Istance
*Zooming interfaces! Enhancing the performance of eye controlled pointing devices*
Proceedings of ASSETS 2002, The Fifth International ACM SIGCAPH Conference on Assistive Technologies, July 8 - 10, 2002, Edinburgh, Scotland

*Copyright 2002 ACM, Included here by permission*

# Zooming Interfaces! Enhancing the Performance of Eye Controlled Pointing Devices

**Richard Bates, Howell Istance**
Human-Computer Interaction Research Group
De Montfort University
The Gateway
Leicester, LE1 9BH, UK
+44 116 250 6295
rbates@dmu.ac.uk, hoi@dmu.ac.uk

**ABSTRACT**
This paper quantifies the benefits and usability problems associated with eye-based pointing direct interaction on a standard graphical user interface. It shows where and how, with the addition of a second supporting modality, the typically poor performance and subjective assessment of eye-based pointing devices can be improved to match the performance of other assistive technology devices. It shows that target size is the overriding factor affecting device performance and that when target sizes are artificially increased by 'zooming in' on the interface under the control of a supporting modality then eye-based pointing becomes a viable and usable interaction methodology for people with high-level motor disabilities.

**Keywords**
Eye-tracking, pointing devices, assistive technology, zoom screen, graphical user interfaces

## INTRODUCTION

Eye-based pointing devices, or eye mice, have been in existence for many years within the motor-disabled community, with a small but significant number of disabled people using these devices to access computers and communication devices. Anecdotal evidence suggests that eye-based pointing is an inefficient means of pointing in assistive technology due to the inaccuracy of eye-tracking systems, making direct interaction with standard graphical user interfaces very difficult. To overcome this difficulty, most systems in use typically interact indirectly with standard graphical interfaces via soft devices or secondary interfaces specifically designed to allow for the limitations of eye-based interaction. Although these custom interfaces allow interaction, it is indirect and often laborious and cumbersome, reducing the benefits of direct pointing and manipulation of the interface. The aim of this paper is to investigate the performance of direct manipulation using an eye mouse on a standard graphical user interface and to show how the performance of an eye mouse can be dramatically improved by the addition of a 'zoom screen' facility.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
*Assets 2002*, July 8-10, 2002 Edinburgh, Scotland
Copyright 2002 ACM ISBN 1-58113-464-9-02/07 ...$5.00

### The Benefits of Eye-Based Pointing

Firstly, eye-gaze has the potential to be a very natural form of pointing, as people tend to look at the object they wish to interact with [11, 17]. Secondly the speed of eye-gaze to locate a target can be very fast when compared to other pointing devices [23, 18]. Thirdly, due to the specialised nature of the muscles controlling the eye, natural eye movements exhibit little detectable fatigue and offer near fatigue-free pointing [16]. Finally, eye-tracking technology can be non-encumbering, as users are not required to wear or hold any device.

### The Costs of Eye-Based Pointing

Firstly the eye is not a highly accurate pointing device as it exhibits a positional tolerance [5, 12]. The fovea of the eye, which gives clear vision, covers a visual angle of $\approx 1°$ arc of the retina, hence when fixating a target the eye only needs to be within $\approx 1°$ of the target position to clearly see the target. This gives an inaccuracy in measured gaze position. Secondly, since eye gaze position cannot easily be consciously controlled or steered, as it is driven by subconscious interest [28], the eye tends to fixate briefly on targets of interest before jumping to other points of interest. Thus it requires effort to fixate steadily on a target for any extended period of time. Thirdly the eye is being employed as both an input modality to the user, so the person can see feedback from the interface, and an output modality from the user to the interface, indicating the pointing intention of the user on the interface. This convergence of interaction point and gaze point means that, without recourse to an additional command, the pointing cursor cannot be parked or left at a position on the screen whilst the eye momentarily looks away. This results in unwanted pointing movements at the feedback point on the computer screen as the cursor follows the eye wherever it gazes [12, 23]. Finally, eye-trackers are not widely available, and can be expensive.

### A 'Zoom Screen' Facility

There is a clear relationship between target size and the performance of eye-based pointing devices, with the smaller targets found on common graphical user interfaces presenting considerable selection difficulties [9, 23]. Target magnification, such that the user can increase the effective size of target objects by temporarily 'zooming in' on the interface during a single interaction task, has been suggested in order to overcome the difficulties with smaller targets [10, 13]. However these 'zoom' devices have been based on *indirect* interaction with the interface, which is often a poor solution due to the additional interaction overheads of indirect interaction in comparison with direct interaction. Previous work [1] has confirmed with abstract target acquisition tests that adding a transparent magnification function that allows continuous *direct* interaction, with no visible on-screen device between the user and the interface, does increase the performance of eye-based pointing devices. Hence employing such a zoom screen facility retains the benefits of direct interaction and should also increase eye-based interaction performance. To date, this approach has not been tested during direct interaction with a standard graphical user interface.

## TESTING EYE-BASED POINTING

A series of comparative 'real world' experiments were conducted to assess the performance and usability of direct eye-based interaction on a standard graphical user interface with and without zoom enhancement.

### A Baseline for Comparison

In order to place the performance of eye controlled pointing devices in the context of other assistive technology pointing devices, an eye mouse with and without a zoom screen facility was compared to a standard head mouse. Head mice are very commonly used by people with high-level motor disabilities and are widely accepted as 'usable' pointing devices. If the performance of an eye mouse could approach or surpass that of a head mouse then it would offer a usable alternative pointing device for disabled users, without the fatigue often associated with head pointing. A standard desktop hand mouse was also included in the tests to give a known benchmark performance.

### Test Apparatus

A standard PC running Windows was used for the tests. For the eye mice a Senso-Motoric Instruments [26] infrared video-oculography eye-tracker was used to measure eye-gaze position with a software driver used to move the cursor in response to the eye-gaze of the test participants. A Polhemus Isotrack [25] electromagnetic motion tracking system was used to measure the head position of the test participants for the head mouse and a second software driver was used to move the cursor in response to head position. Target selection was by a hand held micro-switch and text entry was via a WiViK [27] on-screen keyboard. A zoom screen facility was implemented by controlling a specially modified Dolphin Computer Access Ltd. 'Supernova' [24] commercial screen zoom application, originally designed to magnify the screen for users with low vision, via a custom driver. The zoom level was controlled by two hand-held micro-switches, one to increment the zoom level and one to decrement the zoom level. Other supporting modalities, for example a multi-state sip-puff switch, could be used equally well for users with high-level motor disabilities. Four zoom levels were possible: ×1, ×2, ×4 and ×8. During a zoom the complete screen was magnified, with the magnified area centred on the current cursor position. Participants were seated with a head or eye to monitor screen distance of 60cm on a seat with a backrest and head support to help participants to steady their head position and to increase seating comfort.

### A 'Real World' Test

A 'real world' experimental test sequence, rather than an abstract target acquisition test, was used to test the performance of the devices. The test consisted of a series of simple tasks in two domains, word-processing with Microsoft Word and web browsing with Internet Explorer, that formed a natural flow of interaction. Two different domains were used so that any performance differences caused by the different nature of interaction in each domain could be identified. A total of 150 test tasks were constructed, with approximately half of the tasks comprising a word-processing sequence and half a web browsing sequence. The proportions of object usage, target sizes and interaction techniques in the two sets of tasks mimicked as closely as possible 'real world' interaction based on previous observation of users. The 150 test task objects that comprised the test, such as a button or menu item, were then assigned one of four size categories (0.3°, 0.6°, 0.9°, 1.2°) based on the smallest visual angle subtended by the screen object central to the task at a distance of 60cm from the screen.

### Measuring Performance

The usability of the mouse systems was assessed in terms of objective device *efficiency* and subjective user *satisfaction* based on the European ESPRIT MUSiC performance metrics method [3, 15] and the recommendations outlined in the ISO 9241 Part 11 'Guidance on Usability' International Standard [19]. These metrics were defined as follows:

- *Efficiency*: the objective performance of the device, expressed in terms of the amount and quality of interaction with the device and the time taken to perform that interaction.

- *Satisfaction*: the subjective acceptability of the device, expressed in terms of the user workload and comfort when using the device and the ease of use of the device.

*Efficiency*

The efficiency of interaction with the devices was calculated by measuring the quality of interaction during the tasks and the time taken for the tasks. Quality was assessed by counting the number of incorrect commands generated (such as hitting the wrong target), the number of intended targets missed (with no command generated), and the number of cursor position corrections. A cursor position correction was defined as a path variation or unnecessary pause of cursor movement during the task [14]. These variations and pauses indicate a lack of control when compared to an idealised 'perfect' cursor movement. Tasks were initially given a quality rating of 5 (perfect) [20], with subsequent errors reducing the quality until the task was completed or failed, and the next task started. To reflect the consequences of generating each error type, the quality factors were weighted, giving a simple formula for quality (Figure 1). Tasks were declared failed when the quality was reduced to 1. The time taken to complete the tasks was measured from the start of a task until the task was finished or abandoned and included all activities during the task, including any time taken controlling the zoom facility. Device efficiency was calculated by a simple formula (Figure 1). The formula was constructed so that completed tasks that have the highest level of quality and take no time would give a performance of 100%, with any reduction in quality or increase in time degrading the measured efficiency.

$$\text{Quality of interaction} = 5 - (3\times \text{ count of incorrect commands} + 2\times \text{ count of target misses} + 1\times \text{ count of control corrections})$$

$$\text{Device efficiency} = \frac{\text{Quality of interaction (1-5)}}{5 + \text{Time taken for interaction (secs)}}$$

*Figure 1.* Calculation of Efficiency

*Satisfaction*

Device satisfaction was measured using a multidimensional device assessment questionnaire based on the ISO 9241 Part 9 'Non-keyboard Input Device Requirements' International Standard [19] and the NASA task load index questionnaire [8]. The questionnaire consisted of three rating sections: workload, comfort, and ease of use, with each giving a multidimensional score comprised of ratings from the factors within each section (Table 1).

The comfort and ease of use factors were chosen specifically to examine issues related to eye and head pointing satisfaction. 7-interval fully labelled scales suitable for input device assessment were used for rating all of the individual questionnaire factors [4, 2].

| Satisfaction Questionnaire Sections | Section Factors (each rated 1-7) |
|---|---|
| Workload | Physical effort<br>Mental effort<br>Temporal pressure<br>Frustration<br>Performance |
| Comfort | Headache<br>Eye comfort<br>Facial comfort<br>Neck comfort |
| Ease of use | Accuracy of pointing<br>Speed of pointing<br>Accuracy of selection<br>Speed of selection<br>Ease of system control |

*Table 1.* Satisfaction Questionnaire Factors

**Test Subjects**

Six able-bodied test participants were chosen for the experiment. The participants were selected to give a wide range of experience using the head and eye mice from very experienced users through to novice users with little previous experience of the devices. Each participant was given training and practice to become familiar with the test tasks before the tests were started. A hand held mouse was used for the practice sessions and all participants were familiar with this device. The number of test participants required to identify the usability problems of a system can be quite small [22]. From this work, only six test participants were required to determine 100% of 'high severity' usability problems and at this number of participants 95% of 'medium severity' usability problems and 60% of 'low severity' problems were also found.

**Data Collection**

All data was obtained by capturing the complete contents of the test computer screen, including the cursor position, at a rate of 5 frames per second. The data was analysed by stepping through the video files and recording the quality and time taken to perform each task. In addition, the time taken by any non-productive actions during the task was measured and the nature of the non-productive action was recorded. The pointing accuracy of the participants with the devices was recorded after device calibration and before each test by asking the subjects to point at 9 equally spaced targets on the screen, with the overall mean distance of the cursor from the targets recorded. From this, tests were only conducted with calibrations exceeding 75% of the accuracy obtained by expert users with the devices. This removed the possibility that a poor calibration would affect the test results. The head mouse and standard eye mouse tests were conducted in a random order with the zoom eye mouse tests conducted after development of the device. To avoid order effects, a gap of several months was left between the head and standard eye mouse tests and the later zoom eye mouse tests, with no intermediate practice with the devices permitted. Statistical comparisons were made using Mann-Whitney two-sample rank tests, with any significant differences ($p<0.05$) shown on plots where appropriate.

## RESULTS

### Task Domains and Efficiency

Figure 2 shows box-plots of the efficiency metrics for all tasks in each domain contributed by the 6 participants. There were no differences found in device efficiencies for the head mouse and standard eye mouse between the domains, indicating that the use of these devices was not affected by the nature of the tasks

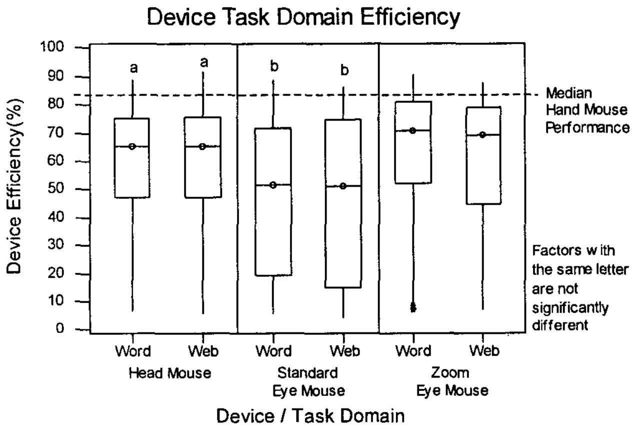


*Figure 2.* Device Efficiency by Domain

However, there was a small significant difference in performance between the domains for the zoom eye mouse. Pooling the efficiencies for the devices for both task domains gave an efficiency of 52% for the standard eye mouse, 65% for the head mouse and 70% for the zoom eye mouse. None of the assistive technology pointing devices rivalled the performance of the hand mouse baseline at 83%. The standard eye mouse performed poorly in comparison with the head mouse. However, the addition of zoom increased the eye mouse performance by 36% such that the performance of the enhanced eye-mouse outperformed the head mouse.

The small, 70.9% vs. 69.2%, but significant difference between the Word and Web domains for the zoom eye mouse can be attributed to the reduced time taken in controlling the zoom facility in the word processing domain. The mean time taken for controlling the Word domain zoom level was 138ms, and for the Web domain zoom level, 236ms. It was observed that participants held the same zoom level over several tasks during typing. Participants would zoom in on the on-screen keyboard and type several characters before zooming out and observing the text generated, thus spreading the time to control the zoom facility over several tasks. The amount of text entry required for the word processing tasks was greater than that required for the web browsing tasks.

### Target Size and Efficiency

Figure 3 shows box-plots of the efficiencies for the 4 target size categories, pooled across participants and task domains. A similar pattern of efficiency increases with increasing target size exists for the head mouse and the standard eye mouse. Note that the efficiency of the standard eye mouse on the 0.3° target is very low, to the extent that it is probably unusable. This confirms the difficulty these devices have with smaller interface objects.

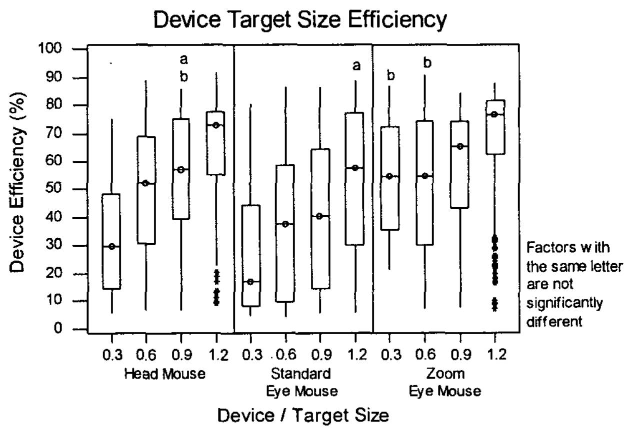


*Figure 3.* Device Efficiency by Target Size

| Target size | Proportion of tasks in which zoom was used | Proportion of task time used to control zoom | Mean eye mouse zoom level used | Effective mean eye mouse zoomed target size |
|---|---|---|---|---|
| 0.3° | 94.4% | 19.8% | 5.39 | 1.62° |
| 0.6° | 85.9% | 13.6% | 2.80 | 1.68° |
| 0.9° | 62.7% | 10.1% | 2.03 | 1.83° |
| 1.2° | 42.7% | 2.2% | 1.43 | 1.72° |

*Table 2.* Zoom Levels and Equivalent Target Sizes

There was a clear relationship between target size and the use of zoom, with zoom increasingly being used as target size decreased (Table 2). The use of zoom was most pronounced with the smallest target size with almost all interactions using zoom at high levels. Translating the zoom levels into the effective zoomed target sizes used by the participants shows a consistency in effective zoomed target size for all targets, translating to an overall preferred size of 1.73° for all targets (Table 2). It is notable that the effective mean zoomed target size is just larger than the mean pre-test measured pointing accuracy of the eye mouse at 1.61°. This strongly suggests that participants magnified targets until they were just larger than the pointing accuracy of the device and hence could be selected accurately and reliably. The zoom levels used can be modelled by zoom level = 1.59 (target size)$^{-0.89}$. This model is supported by a previous abstract target acquisition test using a similar direct interaction zoom method that found zoom level = 1.15 (target size)$^{-0.99}$ [1]. The slightly higher zoom levels found in this experiment compared to the previous abstract test are probably due to the higher consequences of error on the real interface causing participants to zoom targets to reliably acquired sizes. This again suggested that the use of the zoom enhancement with the eye mouse was affected by the nature of the tasks it was required to perform.

**Target Size and Quality**

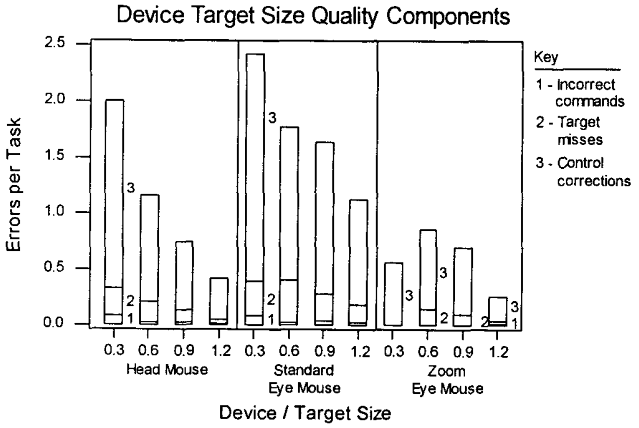


*Figure 4.* Device Quality by Target Size

Figure 4 shows a breakdown of the types of errors occurring for each target size, for tasks across both domains and for all participants. The task quality elements of the efficiency metric showed steady decreases in errors, and hence increases in quality, with increasing target size for the head mouse and standard eye mouse. All of the devices have low counts of incorrect commands, suggesting that the devices can be accurate when the consequences of error, such as correcting the outcome of an incorrect command, are high. The higher counts of target misses for the standard eye mouse in comparison to the head mouse indicate some difficulty in maintaining the cursor over the intended target during selection. This resulted in a 'machine gun' approach to selection with multiple button presses close to the intended target but not hitting adjacent targets. The high rate of control corrections for the standard eye mouse in comparison to the head mouse indicates some considerable difficulty in manoeuvring and positioning the cursor onto a target – the rate is equivalent to more than one control correction per interaction. Of all of the quality metrics, it is clear that the number of control corrections generated by the standard eye mouse causes the most impact on the efficiency of the device.

Examining the results for the zoomed eye mouse we see the effect of the zoom facility, with a large reduction in control corrections. It is notable that the smallest target size has a disproportionately lower error rate than the larger targets, this is probably due to participants nearly always (94.4% from Table 2) zooming these targets so that errors rarely occur.

**Target Size and Task Time**

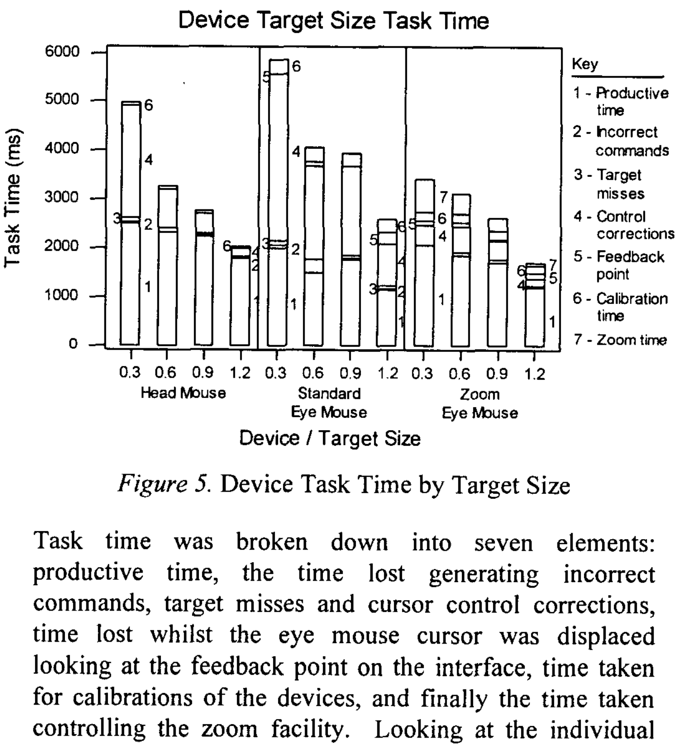


*Figure 5.* Device Task Time by Target Size

Task time was broken down into seven elements: productive time, the time lost generating incorrect commands, target misses and cursor control corrections, time lost whilst the eye mouse cursor was displaced looking at the feedback point on the interface, time taken for calibrations of the devices, and finally the time taken controlling the zoom facility. Looking at the individual elements of task time for the head mouse and standard eye mouse first (Figure 5), it was clear that time lost in cursor control corrections was by far the most non-productive element for these devices, indicating that considerable time was wasted correcting the cursor position onto targets. A comparison of the productive times shows that the standard eye mouse had shorter productive times (was more time efficient) than the head mouse, indicating that it has the potential to be superior to the head mouse if the non-productive elements can be reduced. The time lost in incorrect commands and misses was not significantly different between the devices.

Examining the zoom eye mouse results showed that it too had shorter productive times than the head mouse, indicating the potential to outperform this device, although it had slightly but significantly longer times than the standard eye mouse. This additional time is due to participants taking a little more time, without producing errors, during positioning of the cursor before zooming and may be due to participants planning the best cursor placement before zooming. The effect of the zoom facility on task time is marked, with a large reduction in the non-productive time due to control corrections to below that of the head mouse. The cost of controlling the zoom facility is shown in Figure 5, with the addition of a zoom non-productive time element indicating the time taken changing the zoom level. Zoom time, at an average of 8.2% of task time for all tasks, now becomes a significant non-productive time factor and approaches the time lost in cursor control corrections at 12.7%. This overhead is particularly important for smaller targets (Table 2), where zoom time becomes the largest non-productive time element. This indicates the need to find a more efficient method of controlling zoom, such as an automatic zoom based on user intent derived from gaze patterns or timings [6, 7]. In addition, based on the preference for an overall zoomed target size of 1.73°, all targets could simply be zoomed to this size using a single command, removing the need for multiple commands to step through each available zoom level. However, even with the overhead of zoom time, it is clear that the benefits of zoom outweigh the costs.

**Device Satisfaction**

Figure 6 shows box-plots of the average of the individual ratings within each subjective assessment category (Table 1) to give the overall workload, comfort and ease of use ratings for the devices.

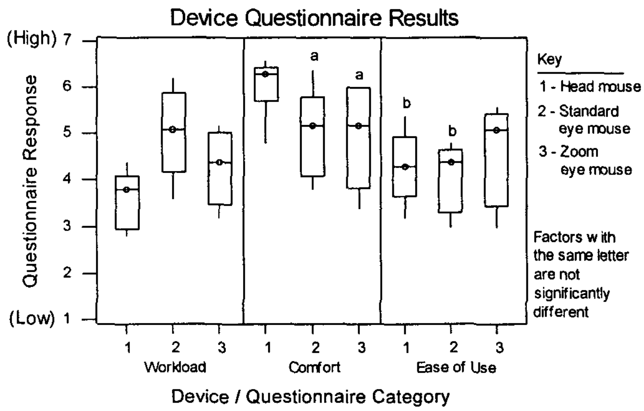


*Figure 6.* Device Satisfaction Questionnaire Results

*Aggregated factors*

Examining the results for the head mouse first, it is clear that the device has the lowest workload and highest level of comfort, indicating that the head mouse is probably the most sustainable to use over longer periods of time. However, its ease of use rating is no different from the standard eye mouse. The standard eye mouse had the highest workload and was less comfortable to use than the head mouse. The relatively low subjective acceptability of the standard eye mouse, together with the relatively low efficiency of the device supports the anecdotal evidence as a reason why eye-based pointing devices are not commonly used.

The addition of the zoom facility halved the difference in workload between the standard eye mouse and the head mouse, in spite of any additional workload caused by controlling the zoom facility. The zoom facility did not change the level of comfort, with the zoom eye mouse having the same comfort rating as the standard eye mouse. However, the zoom eye mouse showed the highest ease of use of all devices with a 16% improvement over the standard eye mouse.

*Individual factors*

Table 3 shows the individual satisfaction ratings within the workload, comfort and ease of use categories and shows the eye mouse differences from the head mouse baseline. For all factors except pointing speed and clicking speed, the standard eye mouse was rated as poorer than the head mouse.

With addition of zoom, the eye mouse has retained superior ratings of pointing speed and has also gained superior ratings of pointing accuracy compared with the head mouse. This improved rating for pointing accuracy is supported by the reductions in the number of control corrections generated by the zoom eye mouse (Figure 4). However, the addition of the zoom facility has reduced ratings of clicking speed, as participants needed to spend time controlling the zoom level before clicking a target. This change in rating is supported by the addition of a zoom level control time to the task time (Figure 5). For the other factors, there is a small reduction in mental workload for the zoom eye mouse compared to the standard eye mouse, a potentially greater reduction is possibly offset by additional mental effort caused by operating the zoom facility. Both physical workload and frustration show marked reductions, probably due to the increased accuracy provided by the zoom facility allowing participants to relax more and feel confident of hitting targets. The addition of zoom has no effect on the comfort ratings of the eye mouse compared with the head mouse.

Overall, the addition of the zoom facility has reduced the eye mouse workload and increased the ease of use but has not affected the physical comfort experienced when using the device. Much of this discomfort seems to be due to participants being required to remain still in front of the eye-tracking camera. Eye-based interaction should be natural and fatigue-free. However, from these results, it is clear that the eye-based devices are less comfortable to use than the head-based device. The head tracking facility of the eye tracking equipment was not used, as it was found to be ineffective for large changes in posture during trials. Consequently, participants were required to maintain a static posture whilst using the eye-mouse in both its forms. This is the probable cause of the relatively high discomfort ratings.

| Factor / Device | | Head Mouse | Standard Eye Mouse* | | Zoom Eye Mouse* | |
|---|---|---|---|---|---|---|
| Workload (low=good) | Physical | 3.8 | 5.5 | (+1.7) | 4.5 | (+0.7) |
| | Mental | 3.8 | 5.7 | (+1.9) | 5.3 | (+1.5) |
| | Temporal | 2.7 | 4.3 | (+1.6) | 3.7 | (+1.0) |
| | Frustration | 3.7 | 5.0 | (+1.3) | 4.2 | (+0.5) |
| | Performance (inv.) | 4.2 | 4.7 | (+0.5) | 3.8 | (-0.4) |
| Comfort (high=good) | Headache | 6.5 | 5.5 | (-1.0) | 5.5 | (-1.0) |
| | Eye | 6.2 | 4.7 | (-1.5) | 4.3 | (-1.9) |
| | Facial | 6.2 | 5.0 | (-1.2) | 5.0 | (-1.2) |
| | Neck | 4.7 | 3.8 | (-0.9) | 3.7 | (-1.0) |
| Ease of Use (high=good) | Pointing Accuracy | 3.8 | 2.2 | (-1.6) | 4.7 | (+0.9) |
| | Pointing Speed | 3.5 | 4.5 | (+1.0) | 4.5 | (+1.0) |
| | Clicking Accuracy | 4.5 | 4.5 | (0.0) | 4.8 | (+0.3) |
| | Clicking Speed | 4.5 | 5.1 | (+0.6) | 4.2 | (-0.3) |
| | System Control | 5.2 | 4.1 | (-1.1) | 5.0 | (-0.2) |

*Figures in brackets indicate difference from head mouse baseline

*Table 3.* Individual Satisfaction Factors

If, by the addition of reliable and accurate head tracking and re-calibration, the user was allowed to move more freely whilst operating the eye mouse, then it is likely the eye mouse comfort ratings would improve. It is notable that the intended users of eye-based pointing are often quite severely motor-disabled, making movement difficult. It is possible that the levels of physical discomfort experienced by these user groups would be somewhat lower than those experienced by the able-bodied participants used in these trials, making the zoom eye mouse a more attractive device.

## CONCLUSIONS

These experiments have investigated and compared the usability of a standard eye mouse and a zoom eye mouse to the baseline of a head mouse for direct interaction on a standard graphical user interface. Not surprisingly, it was found that none of the assistive technology pointing devices performed as well as a standard hand mouse when tested with able-bodied users who were all experienced users of hand mice. The performance of the standard eye mouse shows that direct interaction on a standard graphical user interface is more difficult with this device than with a head mouse. The addition of the zoom enhancement controlled by a supporting modality, however, lifts the performance of the device to the extent that its efficiency exceeds that of a head mouse. This has been achieved largely by reducing unnecessary cursor position corrections when selecting targets. The provision of the zoom enhancement reduced the high subjective workload ratings of the standard eye mouse considerably and lifted the ease of use of the mouse to higher then the head mouse. The measured performance improvement takes into account the cost of using the facility in terms of the additional time taken to use it. The provision of the zoom facility has not however resulted in a measurable improvement in the ratings of physical comfort for the eye-mouse in comparison with the head mouse. The static posture required to use the eye-tracking equipment effectively, thought to be responsible for the comfort ratings, may be much less of a problem when it is used by groups of users with severe motor impairments

The work has shown the value of measuring the various components of efficiency and showing in detail how the provision of the zoom enhancement has improved device performance. It also indicates where further improvements can be made.

Future work will investigate more efficient methods of controlling the zoom level, particularly in view of the constancy of the zoomed target size, and will also examine how the performance metrics of efficiency and satisfaction are changed when eye mice are used by high-level motor disabled users.

## ACKNOWLEDGMENTS

We would like to thank all of the participants who took time to become trained with the devices and volunteered to sit through our tests. Also, we would particularly like to thank Dolphin Computer Access Ltd. for developing a modified version of their 'Supernova' screen magnifier specifically for these tests.

## REFERENCES

1. Bates, R. Multimodal Eye-based Interaction for Zoomed Target Selection on a Standard Graphical User Interface, in *Proceedings of Interact'99 vol. II* 7-8 British Computer Society, 1999.

2. Bates, R., Bierton, R. Experimental determination of quantifiers for usability questionnaire design, in *Proceedings of Human Computer Interaction 2000 vol. II*, 57-58, British Computer Society, 2000.

3. Bevan, N., Kirakowski, J., Maissel, J. What is usability?, in *Human Aspects in Computing: Design and Use of Interactive Systems and Work with Terminals*, Elsevier Science, 1991.

4. Bierton, R., Bates, R. Experimental determination of optimal scales for usability questionnaire design, in *Proceedings of Human Computer Interaction 2000 vol. II*, British Computer Society, 55-56.

5. Carpenter, R.H.S. Eye movements, MacMillan, 1991.

6. Edwards, G. New Software makes eye tracking viable: You can control computers with your eyes, in *Proceedings of CSUN Technology and Persons with Disabilities Conference 1998*, California State University Northridge.

7. Goldberg, J.H. Schryver J.C. Eye-gaze control of the computer interface: disrimination of zoom intent, in *Proceedings of the Human Factors and Ergonomics Society 37th Annual Meeting*, Human Factors Society, 1993.

8. Hart, S. G., Staveland, L. E. Development of the NASA-tlx (Task Load Index): Results of empirical and theoretical research, in *Human Mental Workload*, 139-183, Elsevier, 1988.

9. Istance, H. Howarth, PA. Keeping an eye on your interface: The potential for eye-based control of graphical user interfaces (GUIs), in *Proceedings of HCI'94 People and Computers IX* 195-209, Cambridge University Press, 1994.

10. Istance, H. Spinner, C. Howarth, P.A. Eye-based control of standard GUI software, in *Proceedings of People and Computers XI HCI '96* 142-158, Springer, 1996.

11. Jacob, R.J.K. The use of eye movements in human-computer interaction techniques: What you look at is what you get, *ACM Transactions on Information Systems* 9, 3, 152-169, 1991.

12. Jacob, R.J.K. Eye tracking in advanced interface design, in *Advanced Interface Design and Virtual Environments*, 258-288, Oxford University Press, 1995.

13. Lankford, C. Effective eye-gaze input into windows, in *Proceedings of the Eye Tracking Research and Applications Symposium 2000,* 23-27 ACM press, 2000.

14. MacKenzie, I.S., Kauppinen, T., Silfverberg, M. Accuracy measures for evaluating computer pointing devices, in *Proceedings of CHI 2001*, ACM Press.

15. Macleod, M., Bowden, R., Bevan, N., Curson, I. The MUSiC performance measurement method, in *Behaviour and Information Technology* 16, 4/5, 279-293, Crown, 1997.

16. Saito, S. Does fatigue exist in a quantitative measurement of eye movements? *Ergonomics* 35 5/6, 607-615, 1992.

17. Salvucci, D.D. Anderson, J.R. Intelligent gaze-added interfaces, in *Proceedings of CHI 2000*, 2, ACM Press, 273-280.

18. Sibert, L.E. Jacob, R.J.K. Evaluation of eye gaze interaction, in *Proceedings of CHI 2000*, 2, ACM Press, 281-288.

19. Smith, W. J. ISO and ANSI ergonomic standards for computer products: A guide to implementation and compliance. Prentice Hall, New York, 1996.

20. Szczur, M. Usability testing - on a budget: a NASA usability test case study. In *Behaviour and Information Technology*, 13, no.1 and 2, 106-118, 1994.

21. Velichkovsky, B. Sprenger, A. Unema, P. Towards gaze mediated interaction: Collecting solutions of the "midas touch" problem, in *Human-Computer Interaction: INTERACT'97*, Chapman & Hall, 1997.

22. Virzi, R. A. Refining the test phase of usability evaluation: How many subjects is enough? *Human Factors*, 34, 4, 457-468, 1992.

23. Ware, C. Mikaelian, H.H. An evaluation of an eye tracker as a device for computer input, in *Proceedings of CHI 1987*, 183-188, ACM Press.

24. Dolphin Computer Access Ltd. Technology House, Blackpole Estate West, Worcester, WR3 8TJ, UK. www.dolphinuk.co.uk

25. Polhemus Incorporated, 40 Hercules Drive, P.O. Box 560, Colchester, VT 05446, USA. www.polhemus.com

26. SensoMotoric Instruments GmbH (SMI), Warthestraße 21, D-14513 Teltow/Berlin, Germany. www.smi.de

27. Prentke Romich Company, 1022 Heyl Road, Wooster, OH 44691, USA. www.wivik.com

28. Yarbus, A.L. Eye Movements and Vision, Plenum, New York, 1967.

# Appendix E

# Research Output 4

H. Istance, R. Bates, A. Hyrskykari, S. Vickers
*Snap clutch, a moded approach to solving the Midas touch problem.* Proceedings of the ACM symposium on Eye tracking research & applications ETRA '08 ACM Press, New York, NY, 221-228. 2008

*Copyright 2008 ACM, Included here by permission*

# Snap Clutch, a Moded Approach to Solving the Midas Touch Problem

Howell Istance*, Richard Bates*, Aulikki Hyrskykari**, and Stephen Vickers*,
* Human-Computer Interaction Research Group
De Montfort University
The Gateway
Leicester, LE1 9BH, UK
+44 116 250 6295
{hoi, rbates, svickers}@dmu.ac.uk
** Human-Computer Interaction Unit (TAUCHI)
Department of Computer Sciences
FIN-33014 University of Tampere, Finland
ah@cs.uta.fi

## Abstract

This paper proposes a simple approach to an old problem, that of the 'Midas Touch'. This uses modes to enable different types of mouse behavior to be emulated with gaze and by using gestures to switch between these modes. A light weight gesture is also used to switch gaze control off when it is not needed, thereby removing a major cause of the problem. The ideas have been trialed in Second Life, which is characterized by a feature-rich of set of interaction techniques and a 3D graphical world. The use of gaze with this type of virtual community is of great relevance to severely disabled people as it can enable them to be in the community on a similar basis to able-bodied participants. The assumption here though is that this group will use gaze as a single modality and that dwell will be an important selection technique. The Midas Touch Problem needs to be considered in the context of fast dwell-based interaction. The solution proposed here, Snap Clutch, is incorporated into the mouse emulator software. The user trials reported here show this to be a very promising way in dealing with some of the interaction problems that users of these complex interfaces face when using gaze by dwell.

**CR Categories**: H.5.2 [Information Interfaces and Presentation]: User Interfaces – Evaluation/methodology; Input devices and strategies

**Keywords**: eye tracking, gaze control, gaze gestures, feedback, disabled users

## 1 Introduction

A vision of many researchers who work with gaze based interaction techniques is that some day most mainstream applications will use an eye tracker as a standard input device. At that point we will need standard ways in which interface components react to gaze if an eye tracker is detected to be in use and active. Gaze information can be used in different ways. For able-bodied users, applications could use gaze data passively (i.e. gaze-aware attentive interfaces), whereas the motor-impaired community could choose to use gaze as a command-based primary input modality.

Online-communities, games and other internet-based services are becoming increasingly important in society as a whole. For gaze-based interaction to be used with these services, this is much work still to do in terms of research and subsequent standards. Meanwhile, a reasonable way of using gaze as an input device would give a large community of motor-impaired people the possibility to control electronic media and provide access to an increasing variety of services.

The problems hindering the use of eye tracking can be divided into three categories: (1) hardware problems including the usability and cost of eye trackers, (2) the inherent inaccuracy of eye tracking and (3) the Midas Touch problem.

Progress towards resolving the first category of problems has been encouraging e.g. [Babcock and Pelz 2004; Hansen et al. 2004; Amir et al. 2005], see also [IPRIZE 2006].

Considerable research effort has been devoted to the second class of problems, to overcome the inaccuracy of eye tracking. Approaches include a variety of different zooming solutions, e.g. [Istance, et. al 1996; Lankford 2000; Bates and Istance 2002; Ashmore and Duchowski 2005; Kumar et al. 2007], using semantic information acquired from the application to correct the inaccuracy, e.g. [Salvucci and Anderson 2000; Beymer et al. 2005; Hyrskykari 2006] or using additional operations to refine the inaccurate gaze point, e.g. [Zhai et al. 1999; Yamato et al. 2000; Miniotas et al. 2006].

This paper attempts to find a solution for the Midas Touch problem. The Midas Touch problem arises from the fact that the eyes are an always-on device; they are a perceptual organ meant for looking at objects rather than deliberately controlling them [Jacob 1993]. When eyes are used for control this leads to unintentionally initiated actions. There are several approaches to overcome the Midas Touch problem, most of which use a keyboard button, mouse button, or some other additional means to initiate actions, e.g. [Zhai et al. 1999; Fono and Vertegaal 2005; Kumar et al. 2007].

Copyright © 2008 by the Association for Computing Machinery, Inc.
Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions Dept, ACM Inc., fax +1 (212) 869-0481 or e-mail permissions@acm.org.
ETRA 2008, Savannah, Georgia, March 26–28, 2008.
© 2008 ACM 978-1-59593-982-1/08/0003 $5.00

If gaze is the only modality being used, the most common approach to overcoming Midas Touch is to use long deliberate dwell times. This is to ensure that any selection action is intended by the user before committing to a system event such as mouse click. However, long dwell times can be fatiguing and can result in the gaze point moving off the intended target before the end of the dwell period, which leads to slow, effortful interaction.

We approach the problem by first analysing what problems eye control causes in a certain application domain, (interacting with virtual communities) and we present one solution, Snap Clutch.

The rest of the paper is structured as follows. First we justify the decision to choose virtual communities, and specifically Second Life, as our test bed application. Second, we report briefly the results of user trials when we analysed problems when using the eyes as the principal input modality for interacting with on-line real-time communities. The last sections describe how Snap Clutch is used to address the problems identified, and give initial results from its use.

## 2 Virtual Communities

On-line virtual communities have become increasingly popular in recent years. For people with disabilities they provide an important opportunity to loose their disability temporarily in their interaction with others, to meet other members in a similar situation, and sometimes even help them to recover. Researchers are just starting to appreciate the impact of this phenomenon [Stein 2007].

Second Life (Figure 1) is one example of a very widely used 3D environment, which enables the users to create a rich virtual lives through avatars. It contains dozens of different support groups formed by and intended for those with different diseases and disabilities [Stein 2007].

In Second Life the user can walk and run (even fly and teleport) in an realistic environment containing lots of different settings like parks, homes, bars, shopping malls, etc. Figure 1 shows the user's avatar in the foreground with the camera (the viewpoint) placed behind the avatar. In the distance there is another avatar. Menu controls for the application are placed upper left, and camera placement and avatar movement controls are on the lower right on moveable transparent overlay panels. Frequently used commands are located on buttons along the bottom of the screen. Much of the scene is animated, including trees which move in the virtual breeze, as well as running water, video advertising boards, and a tram that has just passed from view from the station on the right and from which the distant avatar has just alighted. The avatar may interact with and manipulate many of these objects, creating quite a realistic scenario.

### 2.1 Interaction in Second Life

Typically, the interaction and control of an avatar in a virtual world is achieved by a desktop mouse and keyboard. Previously [Bates et. al, 2007] we used Hand's [1997] taxonomy as the base for identifying the following three different task categories and added a fourth at the end of the list:

- locomotion and camera movement - moving the avatar and moving the viewpoint in the world,
- object manipulation – creating and changing attributes of objects,
- application control – controlling functions of the application, menus etc., and
- communication – chatting, generating text.

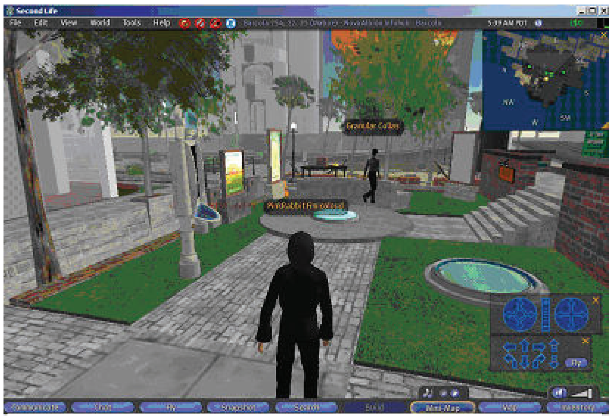

**Figure 1** The Second Life 3D virtual environment (Linden Labs, www.secondlife.com)

In virtual environments the first category of interaction tasks is important. The avatar movement is normally done by arrow keys (or clusters of keys such as WASD), but usually mouse control is also an option for movement control. This is the case in Second Life by clicking on the arrows of the transparent movement panel (on the bottom right in Figure 1). Also the camera movement can be performed by the mouse in a transparent control panel (positioned above the avatar movement panel in Figure 1).

Actually, tasks in all categories can be performed by the mouse. Object manipulation and application control are performed via standard menus, transparent pie-menus (Figure 2) and dialog panels. Pie-menus have been found to be promising interaction widgets for gaze control [Urbina and Huckauf 2007]. That is not surprising, as the expanding selection area radiating from the initial input point is an good solution for low accuracy eye input. Text entry for communication can be generated with a mouse and an on-screen keyboard.

Gaze interaction has been shown to be an effective means of computer control for users with high level paralysis [Bates 2002; Bates and Istance 2002b; Bates and Istance 2004] and has been used effectively in eye controlled games, e.g. [Isokoski 2007], and in immersive environments [Cournia 2003; Tanriverdi 2000].


**Figure 2** Pie menu that allows different actions performed to an object (chair in this case)

One approach to using eye-gaze is straight-forward mouse emulation, placing the mouse cursor where the user is gazing on the screen. The underlying application is unaware that the cursor movement and button events originate from a gaze-based device.

This means that in principle Second Life can be controlled by gaze by using an eye-mouse. For communication we need an external soft keyboard for entering the text by eye-mouse. This can be performed with on-screen keyboard that are readily available. Basic versions are also included with the accessibility tools of Windows

However simply emulating a mouse using the eye has its own problems. To get a better understanding of what these problems are, we have already experimented in Second Life to evaluate the usability issues arising from using a gaze driven mouse emulator [Bates et. al 2007].

## 3 Operating Second Life By Eye

### 3.1 Problems with straightforward mouse emulation

We designed four tasks that were to be conducted in our own purpose built Second Life environment. Two experts experienced and familiar with Second Life carried out four tasks using gaze control. The four tasks to be conducted were:

- *Locomotion* – the subjects were required to navigate their avatar along a predefined path
- *Camera movement* – the subjects were required to zoom the camera in to the back of the avatar, so that the t-shirt logo could be clearly seen. They were then required to rotate the camera around to the front of the avatar and then finally to zoom out
- *Object manipulation* – the subjects were required to create a cube primitive and scale it to a given size
- *Application control* - the subject was to navigate through a series of menus and change the appearance of the avatar.

Each series of tasks were conducted using the following modes:

- A conventional mouse
- A gaze driven mouse emulator using dwell click

The environment consisted of a continuous path with multiple turns and a number of distracting objects including fences, trees, houses and cars.

The two subjects sat approximately 60cm away from a 17” monitor and used an SMI REDII remote infrared eye tracker.

The completion times for each task were recorded along with a count of errors that occurred (User trials 1 in Table 1). The data for both users is shown within each cell separated by a hyphen.

### 3.2 Gaze control by task domain – performance times

In Table 1 the task execution times and the number of errors are displayed by task type for both users. The rightmost column relates to the second set of user trials, which are explained in Section 6. Next we will examine each of the tasks when performed using dwell-time-triggered eye control (or normal mouse emulation mode).

*Locomotion*
In principle, the subjects found that gaze control was effective as they could just look at where they wanted to go and the avatar would follow their gaze point. However, the task times were much longer than that of the mouse. The major issue was that users had to drag from the arrow control on the panel to the direction of motion. The drag was implemented with a dwell to start and a dwell to end the drag. Looking too long in one place during locomotion meant that the drag was ended prematurely and the arrow in the control panel had to be re-selected by dwelling on it.

**Table 1** Control requirements for task domains

| Task domain | Task times and error counts for two users (user 1 -- user 2) | | |
|---|---|---|---|
| | User trials 1 | | User trials 2 |
| | **Mouse** | **Gaze-dwell** | **Snap Clutch** |
| **Locomotion** | | | |
| time (s) | 54–47 | 113–93 | 36–69 |
| nr of errors | 4–4 | lots– 9 | 3–5 |
| **Camera movement** | | | |
| time (s) | 24–18 | failed–100 | 40–45 |
| nr of errors | 0 – 0 | lots–3 | 1–2 |
| **Object manipulation** | | | |
| time (s) | 28–24 | 51–36 | 44–38 |
| nr of errors | 0–0 | 0–2 | 1–1 |
| **Application control** | | | |
| time (s) | 21–16 | 61–67 | 63–64 |
| nr of errors | 0–0 | 1 – 0 | 3–1 |

*Camera Movement*
For this task too the execution times were considerably longer with gaze than with the mouse. There were two main reasons,

- inaccuracy errors due to the small size of the camera control widget, and
- feedback errors caused by not being able to see the effect of the camera movement while using the camera control widget.

In particular, feedback errors seemed to make this task difficult. The user would dwell on the arrow of the camera control but instinctively look at the avatar with the drag still active. This would cause the camera to spin. User 1 could not complete the task due to this problem.. We will discuss feedback errors more closely in Sections 3.3 and 4.2.

*Object Manipulation*
In object manipulation the difference in execution times between gaze and mouse was about the same as in the first task. The errors were partly due to the difficulty in placing the gaze point on the small handle controls of the object.

*Application Control*
The times for gaze dwell were again higher that that of the mouse mostly due to the inaccuracy of gaze pointing when interacting with the small buttons on the application interface.

The first user trials were performed using a simple dwell-based mouse emulation, and the gaze inaccuracies could be overcome in the same way as have been shown on a 2D interface e.g. by the use of magnification tools [Bates and Istance, 2002]. This would allow the temporarily magnification of a specific section of the user interface to allow easier manipulation of the smaller controls.

## 3.3 Error analysis

The slower execution with eye-mouse in the first user trials was expected as Second Life was not designed for eye control, and what we were really interested in was what caused the time difference. What kind of problems were encountered and what kind of errors did the test users make when they used eyes for operating in Second Life? We categorized the errors into the following error types. However the error count in Table 1 sums across all of these categories.

*Path Deviation*

This error was where the avatar left the path that was prescribed, which contained a number of right angled turns. In the gaze condition, the steering of the avatar's motion became very sensitive and more difficult to control. However this was also true in the mouse condition. One deviation from the path was counted as one error.

*Selection*

A selection error was defined as the activation of a control object other than the one intended.

Many of these were due to the fact that the controls designed for mouse use were simply too small for eye pointing; when trying to select a control an adjacent one was selected in error. Each occasion of erroneous selection was counted as one error.

*Distraction*

This type of error was typically caused by the user looking at an object (being distracted) when gaze was being actively used to do something else. This would cause erratic control on the object. This error occurred typically during the locomotion task. When the eyes are being used for continuous fixation on a few locations for steering the locomotion path (even in the region of 5 – 10 seconds), it is difficult not to become distracted. It is hard not to look, for example, at a suddenly appearing object, which you can see in the peripheral vision. Glancing at the distracter (which may be an automatic unconscious reaction) results in changing the avatar's path of motion into the distracted object. Every occasion of this was scored as one error.

*Feedback*

These errors arise when the eyes are looking at an control object to activate, but the effect of that control action appears somewhere on the display. The user has to look at the feedback to know whether the control action has been successful. Then the eyes keep darting from between the control point and the point of feedback. This type of errors were very common during the camera control task. The camera control panel was activated with a button down event but the user immediately looked at the avatar to see what the resultant effect of the camera move was. With the button in the down state, this caused unwanted effects on the camera position.

# 4 Taming the Midas Touch Problem with Snap Clutch

In this paper we focus on finding a lightweight solution to the problems rooted in Midas Touch. The inherent inaccuracy of the eye tracking system causes the pointer to move over objects close to the required object. Midas Touch turns this inaccuracy into selection errors if control actions are applied to these objects instead. Distraction errors, as we have defined them, are clearly a consequence of the Midas Touch problem. In the case of the camera control task, even feedback errors arise from the always-on Midas Touch. The effect of moving the gaze point to look at the result of the input, while the eye is in its active control state has drastic effects.

Snap Clutch is an attempt to provide a solution to the all of these error types by providing a fast way of disengaging gaze control when its not needed. It also provides a light weight means of disassociating control of the mouse pointer from the gaze point, but still allowing control events to be generated by gaze.

## 4.1 Snap clutching the eye-mouse on and off

If gaze is the only modality available to a user, dwell time is the most common way to generate events in eye controlled interaction. Generating an event over the wrong object (selection error) may happen if the dwell time is too short. Similarly, dwell times which are too long also have the problems noted earlier associated with them.

Most commercial gaze control systems have a means of allowing the user to engage and disengage control. Moving eyes is extremely fast and effortless, making a gesture by eyes would be one route to a simple solution. My Tobii[1] already uses this to some extent and the user is able to activate a popup menu by glancing left to an off screen target. However, fetching the menu and making the selections from that often just takes too much effort to keep switching gaze control on and off. Consequently users may simply choose not to use the control even when it is provided.

We aim to reduce the effort and cognitive load involved with 'snap clutching' and 'declutching' the gaze control via very simple gaze gestures. If doing so is both fast and effortless, then activating gaze-control when needed becomes equivalent to reaching for the mouse when the user needs to move the pointer. In addition to being able to browse at the interface without the risk of unwanted selection, this could also enable a decrease in dwell time and make the dwell-time-triggered eye control more fluent.

However, a fast way of toggling the eye control isn't enough. For example, it does not solve the problems associated with the feedback errors, which needs a means of separating the point of input and the point of feedback.

## 4.2 Snap clutching additional modes

Both distraction and feedback errors happen when the user wants to look at a point other than the point of active gaze input.

In the first case (distraction error) the user is distracted by some object in the world to look away from the control point. The other case (feedback error) is caused by the need of switching the point of regard between the point of input action and the point of feedback. One solution is to combine the points of regard. This could be done by moving the input point to the feedback point. Attaching a transparent pie menu to the gaze point would be an effective way to do this. The change at the feedback point can be observed through the menu.

However, the disadvantage of the transparent pie menus is that the interface to the application needs to incorporate these gaze-specific interaction techniques. Our focus here is on techniques outside the application, in the mouse emulation software, so that no modification to the target application is assumed.

---

1 http://www.tobii.com/products_-_services/communication_systems_for_disabled/ (4.11.2007)

It should be possible to drop the cursor (and the input point) in one place, then move the gaze point somewhere else, and then be able to cause the input events to be made at the cursor position from there. This could be achieved if a dwell drops, or parks, the mouse pointer. After the cursor has been parked, a dwell on another position (probably at the feedback point) causes a button down event to be generated. When the gaze point is moved outside the dwell threshold area, a button up event is generated. Thus the longer the dwell, the longer the button is held down for.

The need for parking the cursor was evident, for example, when operating the camera position. In Figure 3 the camera control panel is seen right beside the avatar's left arm.


**Figure 3** An avatar in Second Life. Camera control panel right beside her on her left.

The camera control panel has two circles and a scrollable slider between them (Figure 4). The circles contain four headed arrows, clicking on which makes the camera move to the selected direction. With the leftmost circle the camera can be rotated around the focus to the selected direction, and with the rightmost circle the camera is moved to the selected direction. The slider in-between the circles zooms the camera either towards the focus (mouse click on the upper end of the ruler) or away from it (click on the lower end of the ruler).

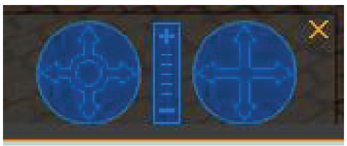

**Figure 4** Camera control panel for rotating, zooming and moving the camera position.

When using a standard mouse, the cursor is moved to the desired area of the control, and the feedback from pressing the mouse button down is given by a change in the whole scene. For example, if '+' in the slider is pressed (to zoom towards the target point of the move) the eyes are not looking on the control, but looking at the world to see the right time to stop the zooming. Without the possibility of parking the cursor over the control, these kinds of operations are difficult as moving the eye in its active control state leads to erratic and unpredictable behavior.

Thus far, we have identified the need for three modes:

- where the user browses the interface in a relaxed manner with no active mouse control,
- where a dwell causes a click to be generated at the gaze point (the normal dwell-click mode), and
- where the mouse pointer can be left (using a dwell) at a position on the interface and then a mouse down and up event generated there by dwelling anywhere else on the screen.

In addition to these three, there is a need to generate a mouse down event at the required input point, and move the pointer with the button down (dragging). In the tasks we studied dragging was used for the locomotion task. Therefore the fourth mode was added where

- the mouse pointer can be left (using a dwell) at a position and the next move of the gaze position emulates dragging the mouse from that point with the button down.

In years gone by, modes in the user interface were considered to be bad design practice due to the overhead of remembering which mode was currently in operation [Nievergelt and Weydert 1987]. Additionally the user has to know what the different modes are and how to move between them. Those issues were taken in account in the implementation.

## 5 Implementation of Snap Clutch

We implemented Snap Clutch to move quickly between the four modes by glancing off screen in four different directions, up, left right and down. The eye tracker used was the SMI REDII remote infrared eye tracker.

In order to prevent the confusion that poorly designed modes can cause, we tried to keep the modes as simple as possible. It is particularly important to give the user feedback about the active mode. In fact, in an eye aware environment, we have better opportunities to ensure that the presented information gets the user's attention. Since we know the user's point of visual attention, changing the shape and color of the gaze cursor gives the user very clear and immediate feedback of the current mode.

**Table 2** Attributes of the Snap Clutch modes

| Mode | Eye Control Off | Dwell Click | Park it Here | Drag from Here |
|---|---|---|---|---|
| Eye gesture to enter the mode | glance up | glance left | glance right | glance down |
| Pointer appearance | green pip | red pip | green crosshair | green arrow |
| Dwell effect and the changed pointer appearance | none | mouse click<br>system sound (no visual change) | parks the pointer<br>red crosshair (+system sound) | activates drag<br>red arrow (+system sound) |
| | | | Dwell when the cursor is parked (red) causes a click on the pointer position<br>Looking at the pointer picks it up again (turns pointer green) | Drag is stopped by changing the mode, or by looking at the original cursor position |

The modes, the chosen eye gestures, and the feedback of the mode given as the changed appearance of the mouse pointer are shown in Table 2 below.

In the current versions we use both the shape and color of the cursor to inform the user of the mode. Green stands for inactive state of a mode and red indicates that gaze is active; active, for example, in dragging or ready to send a click to the parked cursor[2].

In the 'Eye Control Off' mode, there is no mouse emulation although the gaze point is continually tracked. Feedback that the interface is in this mode is provided by a green pip. This mode can be activated at any time by a brief glance beyond the top border of the screen area.

In the 'Dwell Click' mode, the mouse pointer follows the gaze point. A dwell at any time generates a click event at that point. Feedback of this mode is a red pip. This mode can be activated at any time by a glance beyond the left hand border of the screen area.

In the 'Park it Here' mode, dwelling on a position allows the user to leave the mouse pointer at that position, say over an input control. The eyes can be moved anywhere else and a dwell action causes a mouse down and up event at the pointer position. The user can then "pick the mouse pointer up" again just by looking back at it. It can be dropped at some other point by dwelling at that point. The cursor is green when being moved by the gaze point, and red when parked and active. An example of this mode can be seen in Figure 5.

**Figure 5** 'Park it here' mode. The top screen shows the mode in an inactive state and the bottom screen shows the mode active and the cursor being "parked"

The 'Drag from Here' mode allows the user to leave the mouse pointer at a position (by a dwell), then start a mouse drag action from that position by moving the gaze point. The drag action can be finished by changing the mode (e.g. to Eye Control Off or to Dwell Click), or by looking back at the original start position. Feedback of this mode before the drag action is by a green up arrow. When this is an active dragging state, it turns red. This mode can be activated by glancing beyond the bottom region of the screen. An example of this mode can be seen in Figure 6.


**Figure 6** Drag from Here Mode. The leftmost screen shows the mode in an inactive state and the screen shows on right the mode active.

## 6 User experiences of Snap Clutch Eye Control

This section looks at the outcomes from a set of user trials to evaluate Snap Clutch, the feedback from the participants and the work for the future to extend the ideas.

In order to test whether the expected benefits of the new modes would be realized, a second set of user trials was carried out. The same tasks were used as previously and the same participants carried these out. The data from these is shown in the right hand column of Table 1. Given the few participants taking part in the trials, then obvious caveats are made about how representative the outcomes are. In time, a full set of formal evaluation experiments will be conducted when all of the important parameters for this gesture-based moded approach to mouse emulation have been identified.

The locomotion task was expected to benefit from the 'Drag from here' mode to the extent that gaze might even outperform the mouse in the guided path following. This was because of the naturalness of looking in the direction of travel. Also the risk of dropping active control by dwelling too long in the same place was removed.

'Drag from here' retained the dwell until the user looked back at the original arrow, highlighted in the panel. (see Figure 6) This worked very well and the eye worked as a joy stick during the locomotion task. Looking at where the path was going to go was an easy means of steering the avatar. For one user the completion time with Snap Clutch was less than the hand mouse and much less than the dwell condition. The second user improved their performance time compared with the dwell condition but not to the same level as the hand mouse. Error rates were reduced dramatically. For the first user the problems

[2] We chose green cursors for the inactive states as these are intended to be the default or normal states. The active control states are intended to be temporary or exceptional states and have been coded red.

caused by being distracted dwelling and loosing the dwell disappeared. For the second user although slower, they reduced their error rate by nearly half and the path errors were reduced significantly.

The camera rotation task was expected to benefit considerably from the 'Park it here' mode, as the participant could dwell anywhere (and particularly on the object related to the move of the camera position) and the parked mouse pointer would receive the dwell event. The first trials were very effective in demonstrating the problem. This was so bad that the first user did not complete the task. With Snap Clutch, this was completed in 40 seconds with only 1 error, and this compared favorable even with the hand mouse, which can easily be parked with little cognitive overhead. A similar pattern of times is shown for the second user, although he did complete the task the first time. The error count was low to start with and was not affected by Snap Clutch. The other feature of Snap Clutch was that it was easy to move the mouse pointer by dwelling on the pointer and 'unparking' it. This was used to quickly change the position of the pointer of the camera control panel.

The object manipulation task required the users to select 'create an object' from the main menu. The use of the right hand mouse button to pop-up the pie menu was not investigated here. The task required the user to create a cube, to select a 'stretch' command from a crowded dialogue box, to grab the handle of the box and stretch it to 2 meters (shown by in-world rulers). The main benefit of Snap Clutch here was the fast switching between modes by glancing. As can be seen, the completion times error rates between dwell and Snap Clutch were very similar for User 2. For User 1 a new error was introduced, namely a mode error. When the time in this state was discounted, this time too was very similar to the dwell condition in the first trials. The mode error arose by the user trying to get Snap Clutch to return the inactive cursor when the application (Second Life) had taken over the cursor in its own object manipulation mode. So here is a potential problem with the moded approach, i.e. that there may be conflicts between the emulator's modes and feedback about these and those of the application.

The application control task required a combination of actions, the selection of a pull-through item menu from the main menu (rather than the right hand mouse button), the selection of narrow vertical tab and then the manipulation of sliders with a panel to change the color of the avatars hair. Again the task times between the dwell condition and Snap Clutch were very similar, between the two users as well. Slightly more errors were made in the Snap Clutch condition for User 1.

In both object manipulation and application control the anticipated performance benefits of fast gestures was not realized. It could well be a consequence of the relatively short duration of the tasks.

*Subjective impressions of Snap Clutch.*
Some unexpected features emerged. During the camera rotation in the Park it Here mode, both subjects felt the interaction to be extremely natural where staring at the object of interest was translated into a rotation about the object. Simply looking away stopped the action. There was no need to worry about loosing the input position as the pointer was parked there, regardless of the gaze point. Also looking back at the pointer to re-acquire it was a very natural and lightweight control action. Both subjects thought the glance switches worked well as, again, lightweight controls and far preferable to searching for a control panel and looking at a button within it. This meant it was entirely feasible to switch gaze control off when it was not being used.
However the current version of Snap Clutch needs further work to ensure that spurious data and involuntary glances are filtered without compromising the responsiveness of the glance detection.

## 7 Conclusions

The user trials have shown that moded mouse emulation with gestures to switch between modes have much to offer as one approach to addressing the Midas Touch Problem as it becomes an issue in interacting with complex interfaces to graphical environments. Two particular tasks in Second life which are difficult to perform with standard dwell click emulation are made much simpler with the Snap Clutch device. Further analysis and trials are needed to establish how far this moded approach to interaction can be taken. It remains to be seen how many interaction techniques of the total set required can be accommodated before the familiar problems of moded interaction become apparent. The approach can be already be adapted for use with many 2D applications where problems rooted in Midas Touch need to be addressed. The significant advantage is that the implemented modes are independent of the target application.

One can imagine gaze interaction with 3D graphical worlds such as Second Life at a number of places along a continuum: outside the application altogether in the mouse emulator device; in interface elements overlaid on top of the target window (as on screen keyboards that float over desktop application windows.); as part of the interface to the application; as an in-world object (held or attached to the avatar). Snap Clutch operates at the mouse emulator extreme end of this continuum. It represents a simple, configurable and fast approach to the use of dwell while overcoming many of the familiar problems with it.

## References


AMIR, A., ZIMET, L., SANGIOVANNI-VINCENTELLI, A., and KAO, S. 2005. An Embedded System for an Eye-Detection Sensor. Computer Vision and Image Understanding, *CVIU Special Issue on Eye Detection and Tracking, 98* , 1, 104-123.

ASHMORE, M., DUCHOWSKI, A. T., and SHOEMAKER, G. 2005. Efficient Eye Pointing with a FishEye Lens. In *Proceedings of Graphics Interface (GI 2005)*, 203-210.

BABCOCK, J. S., and PELZ, J. B. 2004. Building a Lightweight Eyetracking Headgear. In Proceedings of the Eye Tracking Research and Applications Symposium (ETRA 2004), 109 – 113.

BATES, R. and ISTANCE, H. 2002. Zooming interfaces! Enhancing the Performance of Eye Controlled Pointing Devices. In *Proceedings of the Fifth International ACM SICACCESS Conference on Assistive Technologies (ASSETS 02),* ACM Press, 119-126.

BATES, R. ISTANCE, H. 2002. Why Are Eye Mice Unpopular? A Detailed Comparison of Head and Eye Controlled Assistive Technology Pointing Devices. *Universal Access in the Information Society,* Volume 2, Number 3, October 2003, 280 - 290 Special Issue on "Countering Design Exclusion", guest-edited by Simeon Keates and John Clarkson, Springer-Verlag Heidelberg ISSN: 1615-5289

BATES, R. and ISTANCE, H. 2004. Towards Eye Based Virtual Environment Interaction for Users with High-level Motor Disabilities.

In *Proceedings of International Conference on Disability*, *Virtual Reality and Associated Technologies* (ICDVRAT), 275-282.

BATES, R., ISTANCE, H., and VICKERS, S. 2008. Gaze Interaction with Virtual On-Line Communities: Levelling the Playing Field for Disabled Users. In *Proceedings of the 4th Cambridge Workshop on Universal Access and Assistive Technology (CWUAAT),* University of Cambridge, 13th-16th April 2008.

BEYMER, D., FARRELL, S. P., and ZHAI, S. 2005. System and Method for Selecting and Activating a Target Object Using a Combination of Eye Gaze and Key Presses. USA Patent 2005, International Business Machines Corporation.

COURNIA, N., SMITH, J.D., and DUCHOWSKI, A.T. 2003. Gaze vs. Hand-Based Pointing in Virtual Environments. In *Proceedings of International ACM Conference on Human Factors in Computing Systems* (CHI '03), Ft. Lauderdale (FL), April 5-10, 2003, Short Talks and Interactive Posters, New York: ACM Press, 772-773.

FONO, D. and VERTEGAAL, R. 2005. EyeWindows: Evaluation of Eye-Controlled Zooming Windows for Focus Selection. In *Proceedings of International ACM Conference on Human Factors in Computing Systems* (CHI '05), New York: ACM Press. 151-160.

HAND, C. 1997. A Survey of 3D Interaction Techniques. *Computer Graphics Forum 16*, 5, 269-281.

HANSEN, D. W., MACKAY, D., and HANSEN, J. P. 2004. Eye Tracking off the Shelf. In *Proceedings of ETRA: Eye Tracking Research & Applications Symposium.* San Antonio, Texas, USA: ACM Press. 58-58, 2004.

HYRSKYKARI, A. 2006. Utilizing Eye Movements: Overcoming Inaccuracy While Tracking the Focus of Attention During Reading. In *Computers in Human Behavior 22,* 4, Elsevier Science, 657-671.

IPRIZE, 2006. A Grand Challenge for Human Computer Interaction. Available at http://hcvl.hci.iastate.edu/IPRIZE/

ISOKOSKI, P., HYRSKYKARI, A., KOTKALUOTO, S., and MARTIN, B. 2007. Gamepad and Eye Tracker Input in FPS Games: Data for the First 50 Minutes. *Proceeding of the 3rd Conference on Communication by Gaze Interaction (COGAIN 2007),* 11-15. Available at http://www.cogain.org/cogain2007/COGAIN2007Proceedings.pdf (1.11.2007)

ISTANCE, H. O., SPINNER, C., and HOWARTH, P. A. 1996. Eye-based Control of Standard GUI Software. In *Proceedings of HCI on People and Computers XI,* M. A. Sasse, J. Cunningham, and R. L. Winder, Eds. Springer-Verlag, London, 141-158.

JACOB, R.J.K. 1993. Eye Movement-Based Human-Computer Interaction Techniques: Toward Non-Command Interfaces. In H.R. Hartson and D. Hix (Eds.) *Advances in Human-Computer Interaction*, Vol. 4, Ablex Publishing Co., Norwood, N.J. 151-190. Available at http://www.cs.tufts.edu/~jacob/papers/hartson.pdf (1.11.2007).

KUMAR, M., PAEPCKE, A., and WINOGRAD, T. 2007. EyePoint: Practical Pointing and Selection Using Gaze and Keyboard. In *Proceedings of the SIGCHI Conference on Human factors in computing systems (CHI '07).* ACM Press, New York, NY, 421-430.

MINIOTAS, D., O. ŠPAKOV, I. TUGOY, and I. S. MACKENZIE. 2006. Speech-Augmented Eye Gaze Interaction with Small Closely Spaced Targets. In *Proceedings of ETRA: Eye Tracking Research & Applications Symposium (ETRA 2006),* ACM Press. pp. 67-72.

NIEVERGELT, J. and WEYDERT, J. 1987. Sites, Modes, and Trails: Telling the User of an Interactive System Where He Is, What He Can Do, and How to Get to Places (excerpt). In *Human-Computer interaction: A Multidisciplinary Approach,* R. M. Baecker, Ed. Morgan Kaufmann Publishers, San Francisco, CA, 438-441.

SALVUCCI, D. D. and ANDERSON J. R. 2000. Intelligent Gaze-Added Interfaces. In *Proceedings of International ACM Conference on Human Factors in Computing Systems* (CHI '00), New York: ACM Press. 273-280.

STEIN, R. 2007. Real Hope in a Virtual World. Washington Post, Oct 6th 2007. Available at http://www.washingtonpost.com/wp-dyn/content/story/2007/10/05/ST2007100502446.html?hpid=topnews (1.11.2007)

TANRIVERDI, V. and JACOB, R. J. K. 2000. "Interacting with Eye Movements in Virtual Environments, In *Proceedings of ACM CHI 2000 Human Factors in Computing Systems Conference,* 265-272, Addison-Wesley/ACM Press

URBINA, M. H. and HUCKAUF A. 2007. Dwell Time Free Eye Typing Approaches. In *Proceeding of the 3rd Conference on Communication by Gaze Interaction (COGAIN 2007),* 65-69. Available at http://www.cogain.org/cogain2007/COGAIN2007Proceedings.pdf (1.11.2007)

YAMATO, M., MONDEN, A., MATSUMOTO, K.- I., INOUE, K., and TORII, K. 2000. Button Selection for General GUIs Using Eye and Hand Together. In *Proceedings of AVI*, ACM Press, 270-273.

ZHAI, S., MORIMOTO, C., and IHDE, S. 1999. Manual and Gaze Input Cascaded (Magic) Pointing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '99),* ACM Press, 246-253.

# Appendix F

# Research Output 5

H. Istance, A. Hyrskykari, S. Vickers, T. Chaves
*For Your Eyes Only: Controlling 3D Online Games by Eye Gaze* Proceedings of 12th IFIP conference on Human-Computer Interaction: INTERACT 2009, Uppsala, Sweden

*Reproduced with permission of Springer*

# For Your Eyes Only: Controlling 3D Online Games by Eye-Gaze

Howell Istance[1,2], Aulikki Hyrskykari[2],
Stephen Vickers[1], and Thiago Chaves[2]

[1] Human-Computer Interaction Research Group, De Montfort University,
The Gateway, Leicester, LE1 9BH, UK
{hoi,svickers}@dmu.ac.uk
[2] Human-Computer Interaction Unit (TAUCHI), Department of Computer Sciences,
FIN-33014 University of Tampere, Finland
ah@cs.uta.fi, thiago.chaves@uta.fi

**Abstract.** Massively multiplayer online role-playing games, such as World of Warcraft, have become the most widespread 3D graphical environments with millions of active subscribers worldwide. People with severe motor impairments should be able to take part in these games without the extent of their disability being apparent to others online. Eye gaze is a high bandwidth modality that can support this. We have developed a software device that uses gaze input in different modes for emulating mouse and keyboard events appropriate for interacting with on-line games. We report an evaluation study that investigated gaze-based interaction with World of Warcraft using the device. We have found that it is feasible to carry out tasks representative of game play at a beginners skill level using gaze alone. The results from the locomotion task part of the study show similar performance for gaze-based interaction compared with a keyboard and mouse. We discuss the usability issues that arose when completing three types of tasks in the game and the implications of these for playing of this type of game using gaze as the only input modality.

**Keywords:** Gaze interfaces, games, evaluation, virtual communities, MMOGs.

## 1 Introduction

The popularity of Massively Multi-player Online Games (MMOGs) has increased enormously in recent years. World of Warcraft, probably the most popular fantasy role playing game, has 11 million monthly subscribers [1]. This has been accompanied by a similarly massive increase on the graphics capabilities of home machines that run the clients for these games and online worlds.

People with severe motor disabilities can derive much enjoyment from playing these games and taking part in virtual communities. Participation can be challenging and fun, it gives opportunities for social interaction, and the extent of the player's disability need not be apparent to other players. For some groups of people, eye gaze offers the only input modality with the potential for sufficiently high bandwidth to support the range of time-critical interaction tasks required to play.

T. Gross et al. (Eds.): INTERACT 2009, Part I, LNCS 5726, pp. 314–327, 2009.
© IFIP International Federation for Information Processing 2009

There has been much work on eye gaze interaction with 2D desktop interfaces [2,3]; there has been only a little work on eye gaze interaction with virtual environments, and even less work on real-time interaction with multiplayer graphical worlds. There has been some work on how eye gaze can be integrated with other input modalities for games playing by able-bodied users, but no work to our knowledge on how far gaze can be used as the only input modality to play MMOGs. Understanding how gaze can be used as a single modality for motor impaired users will have a positive carry-over for understanding how to use it as an additional modality for able-bodied users too.

A number of general problems exist with using gaze-based interaction techniques developed for 2D desktop applications for the control of 3D worlds [4]. In addition, the player is under time pressure, which poses additional demands on gaze interaction techniques. Our aim is to design a software device that will enable game playing for the users who are not able to use traditional keyboard/mouse/gamepad input devices.

In this paper we report an evaluation study of the gaze interaction techniques developed so far carried out with World of Warcraft using able-bodied participants. We present a comparison of performance data from user trials with gaze and with keyboard/mouse as the input modalities. We also discuss the main usability issues associated with the gaze condition that arose during the game playing tasks.

## 2 Related Work

Interest in using gaze-input in games has been increasing due to the naturalness of pointing, and the potential for additional attentive input that the user's gaze can provide.

Isokoski et al. [5,6] used a first person shooter (FPS) style game in order to assess the performance of eye gaze as an extra input modality to mouse and keyboard. Their first findings showed that using eye gaze for aiming will not always improve the performance of the players when compared to using the game controller for aiming. However they did find that the number of hits from gaze is comparable to using the game controller alone, and that using gaze to play was more entertaining. The possibility of using eye gaze for controlling player direction was briefly examined but due to the necessity of the user constantly needing to change direction it was deemed not feasible.

Smith and Graham [7] performed an experiment using a similar control system on an open source port of the FPS Quake 2 called Jake2. Similar to Isokoski, the authors did not find any advantage in performance with gaze. However, their subjective user results showed that using eye gaze offered a much more immersive experience than using a mouse and keyboard.

Increased levels of immersion and enjoyment were also found by Jönsson [8] during trials using a combination of eye gaze and mouse within the FPS Half Life. Smith and Graham also performed trials using a version of the 80's arcade game Missile Command. Participants were required to use eye gaze to target missiles that were falling from the top of the screen and press a button to shoot them. They found that there is a need to fire ahead of the missile for a successful hit and this is easily achievable using a mouse. However, it is extremely difficult when using eye gaze to fire ahead as there is a constant distraction of the missile itself (the users looked at the missile rather than where they wished the missile to go). Thus, the majority of eye gaze shots missed and fell behind the missile, demonstrating the importance to disambiguate between a users attention and their intention when implementing interaction techniques.

Various implementations of different gaze driven paddle games (e.g. [9,10]) where simply following the ball by gaze gives the paddle the optimal coordinates show how effective gaze can be when used in a natural way. This point was made long ago by Jacob who advocated using gaze for non-command-based interaction, rather than deliberate command-based interaction [11]. Good task candidates are the ones where the user has to make a move to a point of interest (bat to ball in this case), or perhaps in World of Warcraft, move from ‘here’ to a target object, such as an enemy character, by simply looking at that target object.

Recently, Isokoski et al. [12] has reviewed the potential of using eye gaze in different genres of gaming as an additional modality for able bodied gamers. They identify features of each genre that are favourable or unfavourable for gaze control. They raise the important point that modifying a game to facilitate gaze control may remove some of the challenges and requisite skills that make playing the game interesting.

In context of immersive virtual environments (rather than games) Tanriverdi and Jacob [13] investigated gaze-based interaction techniques for selecting objects and compared performance using gaze with using a handheld pointer. Objects were assigned an index of interest determined by how long and often the user looked at them and were automatically selected and zoomed in upon. Significant performance benefits were found particularly for objects distant from the user in virtual space. They also found there was a cost in terms of poorer spatial memory of the locations of objects in the world in the gaze conditions. This is of particular interest for gaze-based interaction with games, when a significant amount of a user’s visual attention may be allocated to interacting with the game, rather than observing the environment.

## 3 Design of the Eye-Gaze Based Games Interaction Device

Our overall objective is to produce a software device that uses eye position and gaze patterns as input, and produces keystroke and mouse events as output. The game client reacts to these events as if they had come from the keyboard and mouse hardware devices. In this way the device can be used with any game that can be operated by a keyboard and mouse. When the user gets a new game or joins a new on-line community, it should be easy to configure the eye device for the new game. Consequently the device should not require any modifications to the game client software.

Pointing using gaze measurement is inherently inaccurate. The eye is being used for interacting with on-screen objects as well as looking at the game. Normal keyboard and mouse use utilizes both hands for very precise rapid movements in parallel with the use of the eyes. We have no expectation that gaze interaction will be as good as keyboard and mouse for all aspects of games playing and all skill levels of players. However we do want to understand which parts of playing a particular genre of game, and at which skill level of play, gaze based interaction comes close to conventional input devices. For these tasks and at this skill level at least, the player supported by gaze need not appear to be different when on-line from their able-bodied counterparts.

### 3.1 Principles of the Design Solution

Our device [4] maps patterns of gaze behavior into various keyboard and mouse events. Each mapping corresponds to a mode which defines how the user's gaze behavior is interpreted. We can select four modes at any one time, which are then assigned to the four edges of the screen (see Fig. 1). The user can switch between modes by glancing off a particular edge of the screen and back again. Feedback about the currently active mode is given by a green strip that appears along the edge of the screen indicating the active mode. Additional feedback is given by changing the system cursor but this is unreliable as some games will define their own cursors.

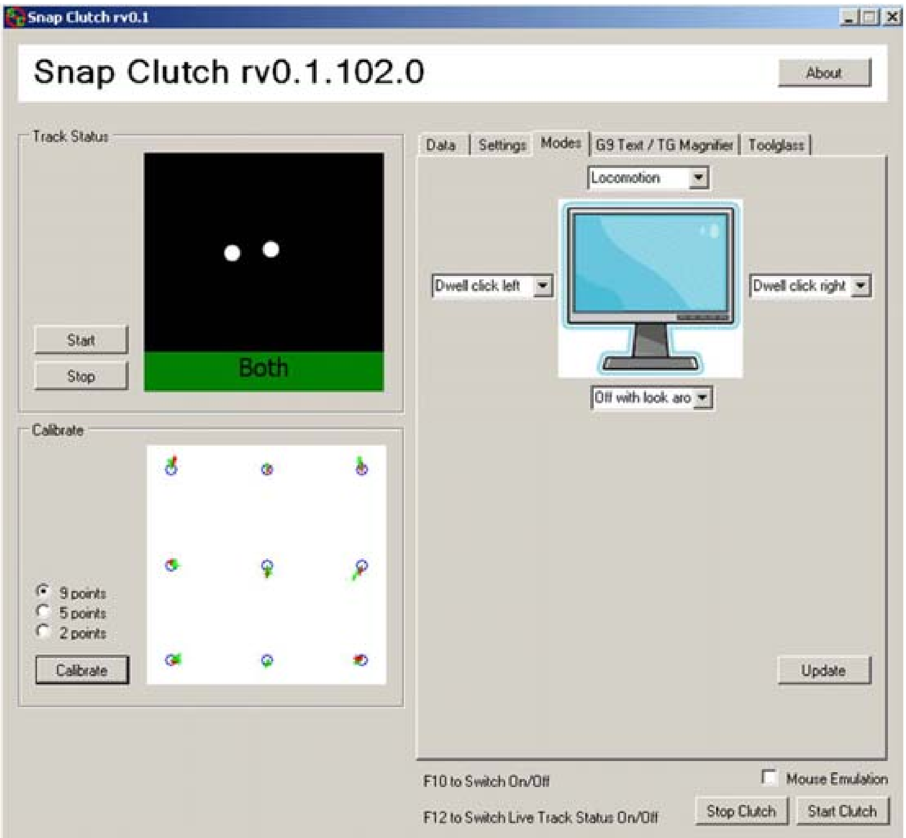


**Fig. 1.** The configuration window of the device in which the user defines the mapping of the desired modes

### 3.2 Gaze Interaction Performance Estimates

We obtained performance data for gaze interaction with a previous version of the device from an evaluation study using Second Life [14]. Twelve participants were required to do set of three tasks with keyboard and mouse and a similar set of three tasks with gaze. The tasks were designed to represent locomotion, object manipulation and application control.

The results were encouraging and showed that all participants were able to complete all tasks, after only a brief introduction and training with Second Life and with gaze interaction. Task times between gaze and keyboard/mouse were compared for each task, and these were partitioned into ‘error time’ and ‘non-error’ time. The proportions of each type of error enabled predictions to be made about the performance benefits that could be expected if the respective causes of each type of error were to be designed out. The main error types found were locomotion errors and accuracy errors. The first type resulted from a lack of sufficient control over the avatar’s direction and speed of movement in the virtual world. The second type resulted from difficulty in positioning the cursor over small targets in interface control objects long enough for the dwell period to expire and the click event to be generated. The latter problem is common with gaze-based interaction. Table 1 shows the ratios of the task times with and without the error time component. In the present study we wished to see whether the non-error time performance ratios were achievable following modifications to our software device when using World of Warcraft (as an example of a popular MMORPG).

**Table 1.** Ratio of task time components from first evaluation study

| Second Life Task | Total task times KB/Mouse : Gaze | Non-error time KB/Mouse : Gaze |
|---|---|---|
| Locomotion | 1 : 1.6 | 1 : 1.2 |
| In-World Object Manipulation | 1 : 4.6 | 1 : 2.0 |
| Application Control | 1 : 2.8 | 1 : 2.5 |

### 3.3 The Present Design of the Gaze Interaction Device

The *locomotion mode* uses ‘active regions’ of the screen. When the user is in this mode, different keystroke events − which control locomotion − are automatically generated and sent to the game client application. Many games use the convention that the ‘w’ key moves the character forward, the ‘a’ key to the left, the ‘d’ key to the right, and the ‘s’ moves the character backwards. The cursor control keys usually have the same function. We found that a player’s eye movements using a mouse and keyboard in World of Warcraft, stayed in quite a distinctive area in front of the avatar. In Fig. 2 there is a heat map visualization [15] of a player’s gaze positions during a period of movement around in an unfamiliar part of a world. On the basis of this we defined regions of the screen that the user does not usually look into during normal navigation. These are also shown overlaid on Fig. 2, although these are not visible to the player during use.

When the user looks at the regions, 'w', 'wa', 'a', 's', 'd', and 'wd' keystrokes respectively are streamed to the games client application. The first evaluation in Second Life showed that turning using gaze was very sensitive and often caused overshooting that required a steering correction in the opposite direction. To smoothen

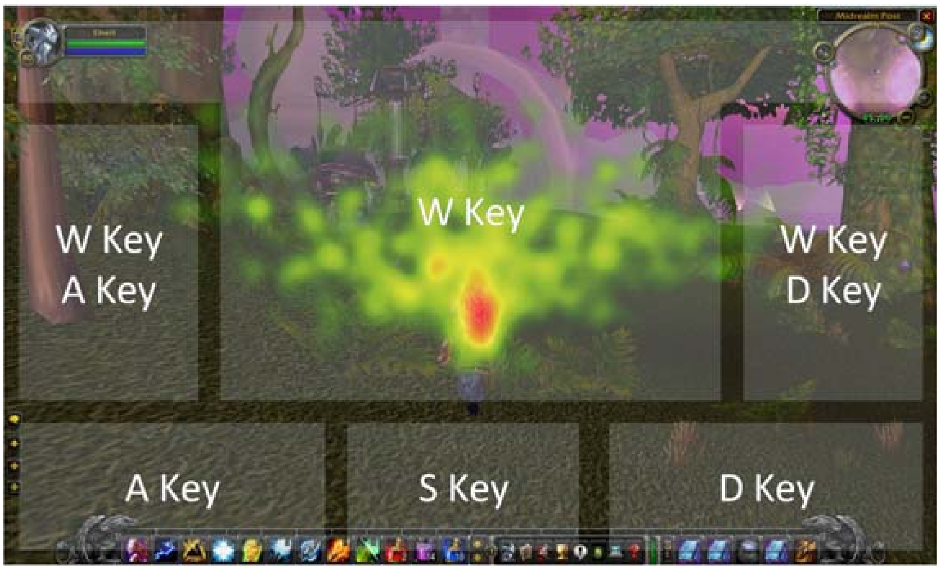


**Fig. 2.** Heat map illustrating the gaze behavior of a player when moving around in a part of the world in World of Warcraft that he was unfamiliar with

the turn, regions on the right and left send the 'w' key interleaved with the turning 'a' and 'd' keys. Looking down and left (/ right) still sent just a ( / d) keys to the application, and seemed to match surprisingly well the participants' intuitive expectations. To stop locomotion, the participant glanced down to switch into 'no action' mode.


**Fig. 3.** The magnifier glass can be dropped by dwell to a location where a close-up manipulation is needed

The *magnifier glass* (see Fig. 3) was designed to counteract the accuracy problem. The user can pick up the magnifier glass by a dwell on a semi-transparent icon placed on the game window. The magnifier glass then follows the point of gaze until the user drops it with another dwell. When the magnifier glass is dropped the user can then dwell within the magnifier area and send other interaction events within the magnifier glass area. A dwell outside the magnifier area moves the glass to a new position, and a subsequent dwell on the magnifier icon turns the magnifier off. The transparency, location and size of the magnifier icon can be configured from the device settings (Fig. 2) to reduce its interference with the underlying screen.

# 4 Testing the Modified Device with World of Warcraft

We carried out a series of user trials with the modified gaze interaction device to study its usability when playing World of Warcraft. Unlike the first study in Second Life, this study included time-constrained interaction with other characters. We also wanted to see whether we could obtain similar performance ratios of keyboard (and mouse) to gaze to those expected from the first study. World of Warcraft is an MMORPG in which the player's character or avatar plays alone or with other players to complete quests. The play involves fights with monsters or other players. These fights involve the use of hand-held weapons or spells which can be cast on opponents. The player has a collection of equipment which can be worn or sold and which can be taken, or 'looted', from opponents when they have been defeated in a fight. A player can have increasing levels of experience as a result of acquiring skills and using them to defeat opponents. In this study, we were only interested in tasks representative of beginners' level experience. The rationale here was that if these are achievable by gaze only, then we can progressively increase the difficulty of the tasks to establish the limits of what is possible using gaze interaction only. A character was created with a medium experience level (level 16) and all trials were carried with this character in the same virtual space around a village. We used a public server so there were other characters in the same space. We wanted the tasks to be conducted in a realistic play environment with a reasonable level of random distraction caused by external events in the game.

## 4.1 Device Configuration

For the user trials we used the following modes. **(1) Glance Up:** *'Locomotion'* mode, which functions as described in Section 3.3. **(2) Glance Down:** *'No action with look around'* mode, in which gaze dwell invokes no action, but the character rotates when the user looks inside the left and right hand edges of the screen. **(3) Glance Left:** *'Left mouse button click'* mode, in which a dwell causes a left button click event. **(4) Glance Right:** *'Right mouse button click'* mode, in which a dwell causes a right button click event. **(5) Glance at Magnifier icon:** '*Magnifier'* mode, whose operation was described in Section 3.3.

### 4.2 User Trials

**Tasks.** We designed four tasks representative of beginner level play in the game. The tasks were chosen following a task analysis of a period of beginners play in an MMORPG. These were:

- *Locomotion task* to walk to a location identified on the inset map, to turn around and return the starting point; and then to repeat the task running. There was no control over character speed and the participant was asked to stay on the path and complete the task as quickly as possible.
- *Fighting task* to find and fight a level 3 monster. The participant was asked to cast the same spell as many times as possible during the fight (by left clicking on an icon located on the shelf in the centre bottom of the screen). The difference in levels assured the participant would always win. After the fight, the participant was asked to loot the corpse (by right clicking on it) of one item (by left clicking on the list of treasure).
- *Equipment task* to put on or wear four items of equipment by opening a pouch (left clicking its icon in the bottom right of the screen); then opening the character sheet (left clicking its icon also in the bottom right of the screen); then selecting an item from the pouch (left clicking on its icon in the pouch); then selecting the highlighted slot in the character sheet which was open in the upper left part of the screen (again by left clicking in the empty slot); then closing both windows (left click in the close box in the top right of the window)
- *Communication task* to greet an object by typing a sentence using a predictive text keypad and then respond to the objects reply by typing another sentence and a closing abbreviated remark. We had designed and implemented the keypad to support communication with other players. There were a number of problems that arose during the trials with this part of the device and the outcomes of this task are not presented in this paper.

**Participants.** Ten participants were recruited for the trials, aged between 18 and 44. These were 9 males and 1 female, all were able-bodied, and all were students or staff at the computer science department at the university (biased gender distribution is justified on the basis that gamers are mostly males). None had taken part in the first experiment. Five had current extensive games playing experience with MMORPGs, three with World of Warcraft. All of the other 5 had played computer games, but did not consider themselves to be experienced MMORPG players. Participants were given cinema tickets in return for taking part.

**Procedure.** We carried out the trials in a usability laboratory equipped so that the trials could be observed from an adjacent room, separated by a one-way glass window. A Tobii T60 was used for the trials. The screen image from the trial machine was visible in the viewing room and it was recorded for subsequent video analysis.

Each trial consisted of a training phase (50 to 60 minutes), a break (20 to 30 minutes), and the data collection phase (about 30 minutes). The first part of the training covered the use of the gaze device, the magnifier, and locomotion mode. The second part of the training consisted of a structured introduction to World of Warcraft and completing a set of standard tasks. This was done first by keyboard and mouse, and then with gaze. After a break, all the four tasks were recorded with the keyboard and mouse. The same four tasks were then carried out using gaze. After the fourth task there was a 10 minute interview. The order of conditions during the trials was not counterbalanced as we wanted to increase the practice obtained before the gaze trial. We had no expectation that gaze would perform better than keyboard and mouse.

# 5 Results

## 5.1 Locomotion Task

In the present study, the locomotion task was carried out both running and walking. All participants completed the task in both conditions. Table 2 shows the means and standard deviations of the task completion times for the 9 participants. Data from one participant was omitted from the quantitative analysis but retained in the analysis of subjective data. This was due to problems calibrating the eyetracker. Willcoxen's Matched Pairs Signed Ranks Test shows the difference between the two conditions not to be significant ($p\ (\alpha) > 0.05$) when participants were walking. Running however took significantly longer in the gaze condition (($p\ (\alpha) \approx 0.01$) compared with the keyboard and mouse condition.

**Table 2.** Locomotion total task times for Kb/M and gaze

| **Locomotion** | | **Kb/M** (s) | **Gaze** (s) | **Kb/M:gaze** (ratio) |
|---|---|---|---|---|
| | mean | 80.9 | 83.2 | 1 : 1.0 |
| **walk** | stdev | 2.6 | 7.5 | |
| | n | 9 | 9 | |
| | mean | 29.6 | 32.7 | 1 : 1.1 |
| **run** | stdev | 1.4 | 2.5 | |
| | n | 9 | 9 | |

In both cases the keyboard and mouse to gaze performance ratio was better than expected from the locomotion task in the Second Life trials (Table 1).

In the subjective evaluation, 7 participants of the 10 participants said controlling the rate of turn of the character was especially difficult in the gaze condition. Fine control of changes in direction was said by some participants to be much easier with the keyboard than with gaze. The other control issue reported by 3 was the difficulty in starting and stopping movement quickly in the gaze condition (by glancing over the bottom edge of the screen). Also searching for a type of monster required reading the labels over the heads of characters as they appeared on screen. If these appeared on the right or left sides, reading the labels would cause unintentional turns in that direction. Another participant referred to the problem of feedback where it was difficult to see whether the characters had turned far enough when looking at the bottom left or right hand corners of the screen. Three participants rated gaze control of locomotion to be easier than keyboard and mouse as there was no need to keep pressing a key to move.

We tried to rectify the ‘turn overshoot’ problem identified in the Second Life evaluation study by interleaving forward and sideways key events during a turn. However we still observed many instances of this error, particularly in the gaze condition. These did not result in significant recovery time loss but they did lead to more deviations from the centre of a forward path movement. Another observed gaze specific error was a ‘distraction’ error, where another character took the participant’s visual attention to part of the screen which caused the own character to turn. This also caused path deviation, which had to be corrected.

### 5.2 Fighting Task

All participants completed the task in both conditions. The data from the fighting task is shown in Table 3. This shows the duration of the fight and the numbers of spells cast during the fight. The duration was measured from when the own character first engaged the monster until the monster died. The gaze fight lasted twice as long as the fight in the keyboard/mouse condition because the number of spells cast was fewer.

**Table 3.** Time taken and numbers of spells cast during the fighting task

| **Fighting** | | **Kb/M** | **Gaze** | **Kb/M:gaze** (ratio) |
|---|---|---|---|---|
| number of | median | 5 | 3 | |
| spells cast | n | 9 | 9 | |
| | mean | 15.1 | 31.69 | 1 : 2.1 |
| time | stdev | 2.67 | 17.82 | |
| | n | 9 | 9 | |

In this simplified fighting task, the main requirement was to click the spell icon continuously to cast as many spells as possible. Willcoxen’s Matched Pairs Signed Ranks Test shows the difference in the tasks times between the two conditions to be significant ($p\ (\alpha) < 0.01$)

In the subjective evaluation, 5 of the 10 subjects considered the size and location of the spell buttons to be a major factor with the difficulty of the task in the gaze condition. The magnifier was not used by any of the participants. When asked whether they considered using this to select the spell, one participant said that the number of actions to get the magnifier, drop it and then select the spell was simply too distracting from the action during the fight. This is an important indicator for the design of gaze interaction techniques for this type of task which involves interaction with other characters.

Two participants said it was difficult to control the character during the fight as it was not possible to do multiple actions at the same time, such as moving and casting spells. This is a requirement for some classes of character but not for others. There is one class of character that has an agent (a pet) that can fight on its behalf, which offers one type of solution to the issue of gaze-controlled fighting. Another participant pointed to the difficulty of gaze selecting a monster to engage in a fight while it was moving as the location for the dwell event has to be anticipated before the dwell begins. Another participant noted how difficult it was not to look at the battle while they needed to keep looking at the spell button in the tool bar at the bottom of the screen.

This task required rapid changing between modes to move, engage the character with a right click and then to cast spells with a left click. Three participants noted that they found changing modes quickly by glancing off screen difficult, although they thought the situation might improve with more practice.

### 5.3 Equipment Task

All participants completed the task in both conditions. The results are shown in Tables 4 and 5. The task has been split into 2 parts, opening the pouch window and the character sheet window (Table 4), and moving each of the four items from the pouch to the character sheet (Table 5). The icons to open the two windows were situated at the edge of the screen and some participants found selecting these by gaze particularly difficult due to the tracking accuracy near the edge of the calibrated area.

**Table 4.** Number of clicks and time taken for the first part of the equipment task

| **Opening 2 windows** | | **Kb/M** | **Gaze** | **Kb/M:gaze** (ratio) |
|---|---|---|---|---|
| number of clicks | median | 2 | 3 | |
| | n | 9 | 9 | |
| time | mean | 3.4 | 17.2 | 1 : 5 |
| | stdev | 1.1 | 14.6 | |
| | n | 9 | 9 | |

**Table 5.** Number of clicks and time taken for the second part of the equipment task

| **Moves 4 items** | | **Kb/M** | **Gaze** | **Kb/M:gaze** (ratio) |
|---|---|---|---|---|
| number of clicks | median | 8 | 14 | |
| | n | 9 | 9 | |
| time | mean | 17.4 | 45.4 | 1: 2.6 |
| | stdev | 5.4 | 23.0 | |
| | n | 9 | 9 | |

The keyboard/mouse to gaze performance ratios for the first and second parts of the tasks were 1:5 and 1:2.6 respectively. This gives a measure of the difference in difficulty between the two parts. Some participants used the magnifier in the gaze condition but only after they had tried to select the targets unaided. This resulted in long times on task and the standard deviations in both of the tables above reflect the large variability in task times. Also dropping the magnifier at the bottom of the screen meant that half of the magnifier was clipped, which could, in some cases, obscure the enlarged view of the target icon.

In the subjective evaluation, opinion was divided between those who thought the task was easy to complete and those who found the first part (opening the equipment windows) and consequently the whole task difficult. 4 of the 10 participants rated the ease of the task completion with gaze as being either as easy as or easier than with mouse and keyboard. There may be an order effect as this task always followed the fighting task in both the gaze and the keyboard and mouse conditions, and may have been considered easier overall.

## 6 Discussion

The outcomes of the trials have demonstrated the feasibility of gaze control of MMORPGs in as much that all participants were able to complete all of the tasks.

There is no universal definition of 'beginner' in terms of skills. Once a player knows what to do in the game, how to level the character, where to buy equipment and what spells do, he or she is no longer a beginner but a novice. Getting to that stage does not take very long (perhaps 30 minutes of play), but getting beyond this stage takes a much longer time. We believe that we demonstrated that gaze control of novice play is achievable.

We have used the ratio of task time using gaze to the time taken to complete the same task with keyboard and mouse as the main quantitative performance indicator. This allows some comparisons to be made between games (or worlds) provided the limits of similarities between the games and their tasks are recognized. The first experiment carried out with Second Life suggested that if the causes of identified problems in controlling locomotion could be designed out, then a performance ratio of keyboard/mouse to gaze in the region of 1 : 1.2 could be expected. We obtained performance ratios of 1 : 1.1 or better in these trials. The main problem with gaze control of locomotion is the lack of fine control over the rate of turn of the character. To some extent, this is a problem with the game client as well as with gaze, and there have been some discussions on forums about the need for better rate of turn control when using keyboard and mouse control with the World of Warcraft client. We recognise that the task given to participants was restricted to moving in a fixed path, and not moving in response to dynamic events in the game.

The fighting task shows some of the real limitations of using gaze to emulate normal mouse and keyboard without modifying the interface. The fighting task was deliberately chosen so that the participant character would always win and casting the same spell repeatedly is a very simplified view of fighting. The trials also revealed the limited nature of moded interaction in the present configuration of the interface, that is, that the player could either move, or cast spells, but not do both at the same time.

There was a 'midas-touch' like problem when the participant looked at a character that appeared at the edge of the screen when looking for monsters which also caused an unwanted change in direction.

The equipment and the fighting task were both hampered by the familiar problem of the difficulty of selecting small targets using gaze. A number of icons in the interface configuration we used were located right at the bottom of the screen, which lead to problems with the eye tracker calibration accuracy. The version of the magnifier that we developed as a means of overcoming accuracy problems apparent from the first experiment was not an effective solution to these. Some of these problems could be attributed to specific implementation issues and some to the lack of training the participants had with the interaction technique. However the main problem appeared to be the means invoking the magnifier, moving it, dropping it, and clicking through it were just too distracting and time consuming for it to be effective in a time-constrained game playing situation. An alternative means of using the magnifier needs to be found, or an alternative solution altogether to the accuracy issue is needed.

The equipment changing task shared some similarities with the appearance tasking changing task in Second Life. That experiment suggested that if the accuracy issues with gaze selection could be resolved then a performance ratio of 1 : 2.5 could be expected. The part of the equipment task involving object selection away from the edge of the screen in these trials had a keyboard/mouse to gaze ratio of 1 : 2.6. The similarity in these ratios gives encouragement to the idea that gaze performance across games can be quantified using the ratio as a metric, and that there is some consistency between similar types of task.

There are also broader interaction issues that the study has raised. In normal interaction in World of Warcraft, information about characters or equipment, for example, is displayed as text in a pop-up box in response to a mouse rollover. Dwell is fundamentally unsuitable as a means of rolling the mouse pointer over elements. The player will read what the box contains and in so doing will move the gaze point off the element. Alternative gaze actions for selecting elements, other than dwell, are needed.

The trials show that where we have time constrained game play, then gaze based emulation of mouse actions using dwell on standard interfaces is too limited. An interface configuration which allows the player to issue rapid commands with visual attention being diverted from the centre of the screen as little as possible is needed. Our ideas here involve using gaze based gestures, and a prototype gaze gesture driven interface to World of Warcraft has been built, and is currently being tested.

## 7 Conclusions

This work should be considered as a first step towards gaze-based game interaction for motor impaired users. We have not yet tested the interface with such users nor have we explored fully the range of design variables necessary to accommodate different types of motor impairment. We do, however, believe that the objective of total gaze control is achievable for a large proportion of users with motor impairments. The same interface works with an example of an MMORPG (World of Warcraft) and with an example of a multi-user virtual community (Second Life) and we expect it will work, with minor adjustments, with other games in each of these genres.

The difference between this project and others that have investigated eye gaze as a modality for game playing is the emphasis in this work on gaze as the sole input modality to enable motor impaired people to play MMORPGs. Others have studied how gaze can be used to complement other input modalities for use by able-bodied gamers. We have been able to demonstrate that it is feasible to carry simple locomotion, fighting and equipment manipulation tasks using gaze alone in World of Warcraft. From earlier work with gaze control of Second Life, we generated some expected performance differences between gaze and keyboard/mouse interaction using task time ratios for similar types of task. In this study we found good agreement with these expected values. The study has also highlighted the limitations of the current approach to using gaze for time-constrained interaction with World of Warcraft as an example of an MMORPG. If gaze-based interaction with MMORPGs is to be realised then interaction techniques which are lightweight, rapid and allow the user to maintain their attention on the centre of the screen are needed. This leads to alternative approaches to gaze interaction that embody these requirements, which are currently under investigation.

**Acknowledgments.** This work is supported by: Communication by Gaze Interaction (COGAIN) FP6 Network of Excellence, the Institute of Creative Technologies (IOCT) at De Montfort University, the Royal Academy of Engineering London and the Academy of Finland.

## References

1. WoW Insider (2008), `http://www.wowinsider.com/2008/10/28/world-of-warcraft-hits-11-million-subscribers-worldwide/`
2. Hyrskykari, A., Majaranta, P., Räihä, K.-J.: From Gaze Control to Attentive Interfaces. In: Proc. HCII 2005, Las Vegas, NV (July 2005), `http://www.cs.uta.fi/hci/gaze/docs/Hyrskykari-HCII2005.pdf`
3. Majaranta, P., Räihä, K.-J.: Twenty years of eye typing: systems and design issues. In: Proc. Symposium on Eye Tracking Research & Applications (ETRA 2002), pp. 15–22. ACM Press, New York (2002)
4. Istance, H., Bates, R., Hyrskykari, A., Vickers, S.: Snap clutch, a moded approach to solving the Midas touch problem. In: Proc. Symposium on Eye Tracking Research & Applications (ETRA 2008), pp. 221–228. ACM, New York (2008)
5. Isokoski, P., Martin, B.: Eye tracker input in first person shooter games. In: Proc. Communication by Gaze Interaction; COGAIN Conference 2006, pp. 76–79 (2006)
6. Isokoski, P., Hyrskykari, A., Kotkaluoto, S., Martin, B.: Gamepad and eye tracker input in fps games; data for the first 50 minutes. In: Proc. Communication by Gaze Interaction; COGAIN Conference 2007, pp. 11–15 (2007)
7. Smith, J., Graham, T.C.N.: Use of eye movements for video game control. In: Proc. ACM SIGCHI international conference on Advances in computer entertainment technology (ACE 2006). ACM, New York (2006)
8. Jönsson, E.: If looks could kill - an evaluation of eye tracking in computer games, Master's thesis, Royal Institute of Technology, Stockholm (2005)
9. LC Technologies, Eyegaze software (2009), `http://www.eyegaze.com/content/programs-eyegaze-edge-communication-system`
10. Dorr, M., Bohme, M., Martinetz, T., Barth, E.: Gaze beats mouse: a case study. In: Proc. Communication by Gaze Interaction; COGAIN Conference (2007)
11. Jacob, R.J.K.: Eye Movement-Based Human-Computer Interaction Techniques: Toward Non-Command Interfaces. In: Hartson, H.R., Hix, D. (eds.) Advances in Human-Computer Interaction, vol. 4, pp. 151–190. Ablex Publishing Co., Norwood (1993)
12. Isokoski, P., Joos, M., Spakov, O., Martin, B.: Gaze Controlled Games, to appear in UAIS. Journal of Universal Access in the Information Society (2009)
13. Tanriverdi, V., Jacob, R.: Interacting with eye movements in virtual environments. In: Proc. SIGCHI Conference on Human Factors in Computing Systems (CHI 2000). ACM Press, New York (2000)
14. Istance, H., Hyrskykari, A., Vickers, S., Ali, N.: User Performance of Gaze-based Interaction with On-line Virtual Communities. In: Proc. COGAIN Conference 2008, pp. 28–32 (2008)
15. Wooding, D.: Fixation maps: Quantifying eye-movement traces. In: Proceedings of the Symposium on Eye Tracking Research & Applications, pp. 31–36. ACM Press, New York (2002)

# Appendix G

# Research Output 6

H. Istance , A. Hyrskykari, L. Immonen, S. Mansikkamaa, S. Vickers
*Designing Gaze Gestures for Gaming: an Investigation of Performance* Proceedings of the ACM symposium on Eye tracking research & applications ETRA '10 ACM Press, New York, NY March 2010.

*Copyright 2010 ACM, Included here by permission*

# Designing Gaze Gestures for Gaming: an Investigation of Performance

Howell Istance
Department of Informatics
De Montfort University
Leicester, UK
hoi@dmu.ac.uk

Aulikki Hyrskykari
TAUCHI
Department of Computer Sciences
University of Tampere, Finland
ah@cs.uta.fi

Lauri Immonen
TAUCHI
Department of Computer Sciences
University of Tampere, Finland
lauri.immonen@cs.uta.fi

Santtu Mansikkamaa
TAUCHI
Department of Computer Sciences
University of Tampere, Finland
santtu.mansikkamaa@cs.uta.fi

Stephen Vickers
Department of Informatics
De Montfort University
Leicester, UK
svickers@dmu.ac.uk

## Abstract

To enable people with motor impairments to use gaze control to play online games and take part in virtual communities, new interaction techniques are needed that overcome the limitations of dwell clicking on icons in the games interface. We have investigated gaze gestures as a means of achieving this. We report the results of an experiment with 24 participants that examined performance differences between different gestures. We were able to predict the effect on performance of the numbers of legs in the gesture and the primary direction of eye movement in a gesture. We also report the outcomes of user trials in which 12 experienced gamers used the gaze gesture interface to play World of Warcraft. All participants were able to move around and engage other characters in fighting episodes successfully. Gestures were good for issuing specific commands such as spell casting, and less good for continuous control of movement compared with other gaze interaction techniques we have developed.

**CR Categories:** H.5.2 [Information Interfaces and Presentation]: User Interfaces – Evaluation/methodology; Input devices and strategies.

**Keywords:** eye tracking, gaze gestures, gaze control, feedback, gaze and gaming

## 1 Introduction

The context for this work is designing interaction with Massively Multiplayer Online Games by eye gaze only. In particular, we are interested in role playing games and virtual communities. The target user group is people with motor impairments who wish to play games, such as World of Warcraft, or participate in virtual communities, such as Second Life. A high bandwidth input modality is needed for this, and simple mouse emulation by gaze is not sufficient to facilitate an adequate range or pace of interaction.

New gaze-based interaction techniques have to be found that fit (i) the particular game, (ii) the user in terms of their particular impairments and preferences, and (iii) the eye tracking equipment the user has in terms of its accuracy. User interfaces to MMORPGs (Massively Multiplayer Online Role Playing Games) enable a player to control his or her character's locomotion through a 3D graphical world, fight other characters, communicate with other players, and manipulate objects at the interface, such as an equipment pouch. In addition to the well-established issues of gaze-based interaction, this situation requires time-constrained, if not real-time, interaction, which is not the case with 2D desk top applications.

If dwell-click techniques are used for selection by gaze where icons are located at the edges of the screen, then a number of issues arise. Visual attention is diverted away from the centre area of the screen where most of the action takes place. The player has to look at a "cast spell" icon until it times out and the spell is launched, and then the player has to look at it again to cast another. Furthermore the size of the icon may be small leading to the familiar issues of inaccuracy when dwell clicking on this kind of icon. An interaction technique is needed that allows the player to look at the centre of the screen, is fast, and is not constrained by the need to maintain the gaze point within small targets. Previously, we have studied various ways to address these issues ([Istance, Bates, Hyrskykari & Vickers, 2008 and Istance, Hyrskykari, Vickers & Chaves, 2009]. In this paper we investigate the use of gaze gestures as a means of overcoming these problems.

We define a gaze gesture as..

> *"A definable pattern of eye movements performed within a limited time period, which may or may not be constrained to a particular range or area, which can be identified in real-time, and used to signify a particular command or intent."*

Most actions or commands in a game like World of Warcraft have user definable key bindings. We have built a layer of software as 'middleware' that is capable of recognizing patterns of eye movements in relation to areas displayed on a screen, and which generates keyboard events in response to these. Gaze gestures have so far been mainly used as a means of entering

Copyright © 2010 by the Association for Computing Machinery, Inc.
Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions Dept, ACM Inc., fax +1 (212) 869-0481 or e-mail permissions@acm.org.
ETRA 2010, Austin, TX, March 22 – 24, 2010.
© 2010 ACM 978-1-60558-994-7/10/0003 $10.00

text by eye. We have extended their use into interaction with 3D graphical environments and games.

In this paper we report an experiment with 24 participants to investigate factors in the design space of gaze gestures and the impact of these on performance and skill building.

Prior to presenting this experiment and its results in detail, we review the previous work with gaze gestures. We also wanted to study how well the designed gaze gestures work when actually used in games context, so we asked the experiences of 12 gamers who used the gaze gestures to play World of Warcraft. In section 5 we report their subjective opinions of their experience and also a study of how easily the designed gestures are made accidentally. We finish the paper by giving conclusions on what we learned of gaze gestures in the experiment and from the test play session.

## 2 Previous work

Gestures are a familiar concept in the context of other input devices; gestures made for example by stylus, mouse, hand or even body have been used in giving commands or feeding information to a computer. Even if the notion of 'gaze gestures' is relatively new, there has been studies that track gaze paths which can in a broad sense be considered as using gaze gestures.

### 2.1 Entering text using gaze gestures

There have been several different approaches to using gaze gestures for text entry. In the traditional dwell based eye writing system the dwell time sets a determinate limit for the typing speed. Thus, gesture based systems have appeared to be one possible solution to get rid of this constraint.

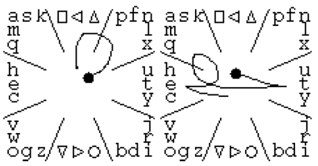

**Figure 1.** Quikwriting the letter 'f' and the word 'the' [Perlin, 1998].

Quikwriting [Perlin, 1998], a Graffiti-like stylus writing system, has been used as the basis for gaze sensitive writing systems. In Quikwriting the user enters characters with a stylus by drawing a continuous line on the surface. The characters are arranged around the starting position into 8 zones (Figure 1). A character is chosen by dragging the stylus from the centre to the zone where the needed character lies. If the character is the middle character in its zone, like 'n' in the top-right zone, the stylus is dragged back to the centre and 'n' is typed. To type other than the middle characters from a zone the stylus is homed via adjacent zones. Generating the letter 'f' is shown left in the Figure 1. To get 'p' the stylus should be 'home'd via the second adjacent zone, i.e. the top-left zone.

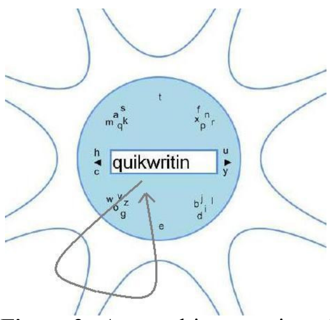


**Figure 2.** A gaze driven version of Quikwriting [Bee and Andrè, 2008]

In 2008 Bee and Andre built and tested a gaze driven version of Quikwriting (Figure 2). As gaze is used both as an input device and to view feedback, the hints for characters could not be displayed in the gaze sensitive zones, since the need of checking the hints would have disturbed making the gestures.

Another approach to using gestures for text entry is to make the shape of the gesture resemble the printed or handwritten shape of the character (Graffiti type of writing). This could make learning the gesture alphabets easier. Wobbrock (2008) built and evaluated such a system, EyeWrite (Figure 3). In their experiments with the system they found that it was somewhat slower than traditional on-screen keyboard dwell time typing, but it resulted in less mistakes. Thus, there seemed to be a speed-accuracy trade-off between these two approaches. However, the learning curve suggested that with practice the speed of using gestures approaches the speed of dwell time typing. In addition, EyeWrite was considered to be less fatiguing than on-screen keyboard typing.

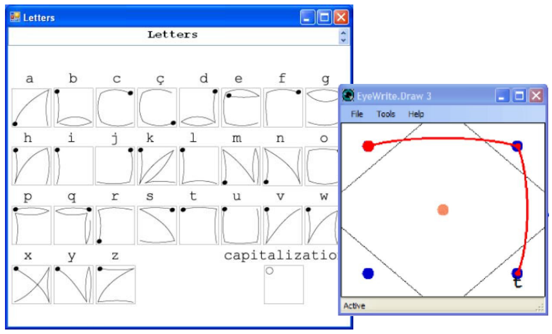


**Figure 3.** The gesture alphabet of the EyeWrite implementation of EdgeWrite and EyeWrite in action: writing a character 't '[Wobbrock et al., 2008].

Other text entry systems that share a gaze gesture approach include e.g. VisionKey [Kahn, Heynen, and Snuggs, 1999], pEYEwrite [Huckauf and Urbina, 2008]. For a review of these, see [Heikkilä and Räihä, 2009].

### 2.2 Gaze gestures in the interface

Beyond text entry systems, there have not been many studies of using gaze gestures to actively control the computer. However, Drewes and Schmidt (2007) built a general gesture recognizer and evaluated its performance. The gesture scheme they designed was inspired by FireGestures[1], a Firefox web browser plug-in, which recognizes mouse gestures that are composed of strokes into four directions left, up, right and down. The gestures in their system were composed from eight strokes consisting also the diagonal directions.

To find out how users are able to do these kind of gestures they made an experiment in which the users made a square clockwise and counter clockwise 4-legged gesture by looking at corners of a window (they could be interpreted e.g. as 'ok' and 'cancel'). Another set of gestures they experimented with, were the ones in Figure 4: two 6-legged gestures and one 4-legged gesture. Nine participants performed these gestures on different backgrounds, one having the square with helping lines giving support for the eye movements, another a spreadsheet document with a grid of lines and the third one was a blank (gray) background.

The participants were able to perform all the gestures so that the gesture recognizer registered them with the exception that four

[1] https://addons.mozilla.org/en-US/firefox/addon/6366

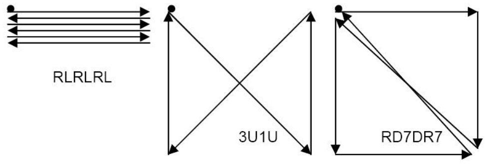


**Figure 4.** The three gestures chosen for an experiment [Drewes & Schmidt,2007]

of the nine participants failed to do the last, most complicated 6-legged gesture on blank background. It is well known that fixating "on nothing" is hard. The average time required to do one leg of a gesture was 557 ms.

An interesting feature in Drewes' and Schmidt's gesture recognizer is that the gestures are not location bound, which means that the algorithm is constantly "on watch" and a gesture can be launched from any position on the screen. Also, the algorithm is interpreting the direction between each recorded gaze position and if the direction is the same as the previous then the stroke is considered to be continuing. This makes the gesture sizes scalable and can be made in whatever size or aspect ratio.

In addition to the work above, there are at least two other studies on gaze gestures. Heikkilä and Räihä (2009) have been interested in using gaze gestures in the context of an eye driven drawing application. In their experiment the participants performed 2-legged, 3-legged and 4-legged gestures on both empty background and on a background with a visual guidance to do the gestures. The times per leg varied from 824 ms (in a 2-legged gesture forming a L shape on an empty background) to 1190 ms (in a 4-legged gesture with visual guidance). Mollenbach, Hansen, Lillholm and Gale (2009) discuss using simple single stroke gaze gestures combined with dwell buttons. In that context they studied single stroke gestures and the mean time they got for one leg gesture (a stroke from one side of a screen to the other side) was 334 ms.

## 3 Design of a gesture scheme

We used a scheme, which is a modified version of Perlins (1998) Quickwriting, and similar to that used by Bee and Andre (2008) in their work on gesture based text entry. Our version used a reduced number of regions or zones so that 12 different gestures could be recognized. We wanted the player to be able to make control gestures while looking generally in the central part of the screen. Thus, for us the use of active regions located in the centre of the screen is an acceptable restriction on where gestures would be recognized. The zones themselves were made semi-transparent so the player could see the avatar and surrounding part of the game world through the zones.

The gestures were made using 5 active zones. They were either 2-legged or 3-legged as shown in Figure 5 giving a total of 4 possible 2- legged and 8 possible 3 -legged gestures. The first of these target zones was called the major zone and the second, the minor zone. 2-legged gestures have major zones only. We wished to understand how different attributes of the design of the gesture scheme affected user performance, particularly in terms of the speed and reliability of performing the gestures.

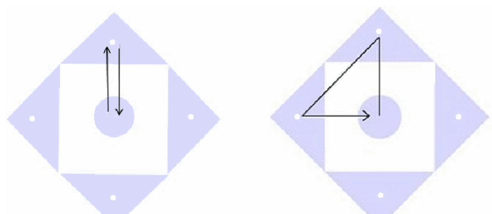

**Figure 5** 2-legged and 3-legged gestures, starting and ending in the centre.

### 3.1 Parameters investigated during pilot testing

*a) Size of the gesture zones and distance from the centre to the inner edge of the active zone.*

We opted not to test locating gesture zones at the edges of the window area as the amplitude of the gesture legs would be unnecessarily large. We did test the size of regions shown in Figure 6 against a set of regions which were 200 pixels greater than those shown. There was no significant difference in the pilot trial. We chose to continue with the smaller of the two sets.

*b) Impact of adding fixation targets within the zones.*

The initial trials showed that gaze points were clustered around the corners of the triangular zones in the absence of any other fixation lock. Adding small circular cutouts in the centre of the triangles had the effect of attracting gaze points to this feature.

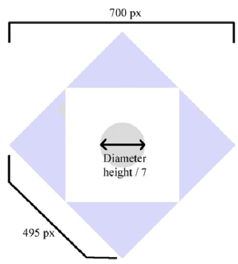


**Figure 6** Size of the used gesture area.

*c) Maximum allowable durations for gestures*

The time for both 2 and 3-legged gestures was studied and the timeout period was set to 2 seconds for the main experiment. This was further revised after the experiment when gestures were incorporated into the gaze interface for the game.

*d) Feedback*

We investigated the impact of providing visual feedback by changing the colour of the zones, but we found this to be too distracting as pilot participants reported waiting for the feedback for each leg. For the experiment, a simple click sound was given as feedback that the complete gesture had been recognized within the timeout period. No feedback was given for an incorrect gesture.

*e) Gesture timing*

The initial implementation of the gesture recognizer was based on an "eye mouse", where the cursor was attached to the gaze point. We then used the operating system timestamp of mouse events as the cursor moved in and out of the gesture regions. However, when studied more closely this was found to be far too unreliable in view of the very short time durations of the gestures. There was a substantial lag after the point of gaze entered a region, before a mouse over event was generated. Thus, this approach was abandoned in favour of the one described in section 3.2.

### 3.2 Implementing the gesture recognizer

A valid *gesture* was accepted as a sequence of fixations, which begins with the last fixation in the centre zone before leaving it, one or more fixations in the major zone, followed by one or more fixations in the minor zone (for 3-legged gestures), and terminated with the first fixation back in the centre zone. Any no-zone fixations were allowed in the sequence. Let us use references T, B, L, R, C and N references for top, bottom, left, right, centre, and no-zones, respectively. Multiple sequential fixations in the same zone are replaced with one fixation and all "N" zones are removed from the sequence. For example, "C-T-C" and "C-T-L-C" are the valid 2 and 3-legged gestures as seen in Figure 5. An invalid gesture sequence is one that does not start and end with "C", exceeds the sequence timeout period, or does not produce any of the defined valid sequences.

The sequence had to occur within a 2-second time period. The time for the sequence began with the time of the last gaze point in the first fixation in the sequence, to the time of the first gaze point in the last fixation. Without this constraint, a variable amount of time could be spent looking in the centre zone at the beginning and/or the end of the sequence.

A *fixation* was defined as being 5 or more gaze points falling within a tolerance region centered around the average x value and the average y value of the previous gaze points in the sequence of gaze points. The tolerance region was defined to cover a visual angle of one degree. The location of the fixation was defined as the rolling average of the x and y coordinates of its component gaze points. The location was hit tested at the end of fixation to whether or not it fell inside a zone. The gaze points were delivered to the application every 15 or 16 ms by the eye tracker.

The implementation produced 3 logs, one of individual gaze points, one of fixations, and one of gesture sequences.

## 4 Experiment – performance in making gestures

Within the gesture scheme described in Section 3, we chose to investigate the effects on performance of 3 factors. By understanding how these factors impact performance, we hope to be able to devise a reliable and efficient gesture system by reducing the impact of negative factors. The factors were the following three.

*a) The number of legs in a gesture: 2 or 3*

As stated earlier, we count from the end of the starting fixation to the start of the terminating fixation. So a minimal 2-legged gesture would consist of saccade-fixation-saccade, and a minimal 3-legged gesture would consist of saccade-fixation-saccade-fixation-saccade (see Figure 5). If we simply assume that fixation durations are much longer than saccade durations then we would expect the durations for 3-legged gestures to be slightly less than 2 times the durations of the 2-legged gestures (2 fixations and 3 saccades versus 1 fixation and 2 saccades ).

*b) Principal direction of the eye movements in the gesture: vertical/horizontal or oblique*

This is an important difference in the context of gaming. Vertical and horizontal gestures map well to directions of character movement, compared with diagonal (or oblique) movements. If the gestures are not used for movement then this natural mapping is less important (except perhaps in the case of camera control). However we would expect more accidental gestures where the principle components are vertical and horizontal compared with oblique eye movements. Here it is possible that a person's natural eye movements result in an unintentional gesture (a 'Midas' gesture). This factor can be manipulated by rotating the gesture detection regions by 45° resulting in a diamond and a square shape respectively (Figure 7).

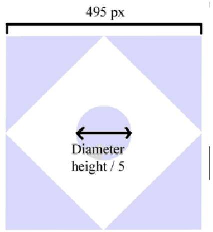


**Figure 7** By setting the height and width of 495 x 495 px of the used square, the length of required saccades for gestures stays the same than in the 700 x 495 size diamond.

*c) Direction of the first movement in the gesture: leftward or rightward*

We suspected that there could be an effect on performance due to the direction of the first movement due to reading behaviour. It has been found that the perceptual span field is asymmetric [Rayner, 1995]. The span extends 14–15 character spaces to the right of fixation, but on the left only to the beginning of the fixated word, or 3–4 character spaces. This depends on cultural background, and we thought that since out participants are western readers, that might result the right first movements being faster than left first movements.

These three factors are represented in the eight gestures shown in Figure 8.

### 4.1 Participants

24 participants were recruited from staff and students attending a summer school in the university. There were 13 male and 11 females, with an average age of 38. No participant reported any ocular muscular defects that would have adversely affected their performance in the experiment. 12 participants had uncorrected vision, 11 had vision corrected with spectacles, and 1 had vision corrected with contact lenses.

### 4.2 Task and procedure

Participants were not required to learn the gesture sequences. Instead, they were presented with the 5 regions (top, right, bottom, left, centre) against a blank white background in the centre of screen. The 8 gestures were displayed as images at the edges of the screen, 4 on each side, and each were identified by a number. After an initial short training period, the participant was asked to make one of the 8 gestures 5 times. They were asked to make each gesture as quickly as possible, but they were told it was not necessary to make the set of 5 as quickly as possible.

A click sound was given as feedback after each one of the set was recognized, and a ping sound was given after the 5th successful gesture. After all 8 gestures had been repeated 5 times in this way, the complete set of eight gestures was repeated in the same order on 2 further occasions with a short pause for recovery in between. The next gesture that a participant was required to make, was announced by the trial leader verbally as a number, and participants were able to see the required pattern at the edge of the screen.

Participants were advised that they could withdraw at any time. There was no reward given for participation. The complete data collection period, including introduction calibration, training and 3 blocks of 8 gestures took between 20 - 30 minutes. The order in which the 8 gestures were presented to a participant was counterbalanced using a Latin Square.

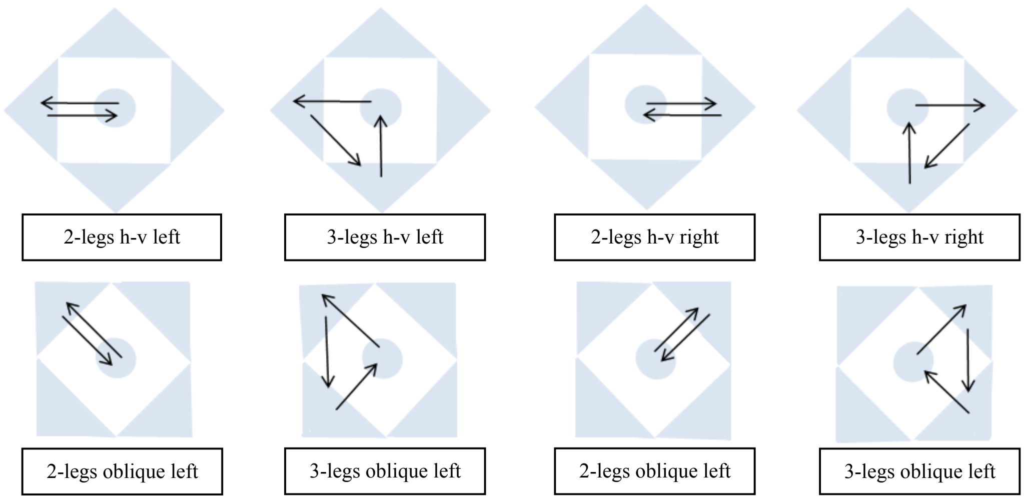


**Figure 8** Four examples of horizontal-vertical (square layout) and four oblique (diamond layout) gestures.

## 4.3 Apparatus

All trials were conducted in a research office. A Tobii X120 eye tracker was mounted beneath a Samsung SyncMaster 959nf 19" CRT display. The participant was seated approximately 60 cms from the display. As described earlier, the centre of each zone was marked with a small circle as a fixation feature. The visual angle subtended between the marker in the centre zone and the markers in each of the triangular zones was about 7°.

## 4.4 Design

To summarise, the experiment was a 2 x 2 x 2 within-participants design, with 24 participants, 3 blocks of 8 conditions, 5 trials per block, giving a total of 120 trials per participant and 2,880 trials in total. The dependent variables were time to complete each valid gesture and errors. Errors were counted as either valid gestures that took longer than the 2 second time out, or gestures that were not the one that a participant was being asked to make at that time.

## 4.5 Results

We observed during the trials that frequently, when asked to make a new gesture, a participant would check the required pattern before and also during the first gesture. This led to timeout errors before the first gesture was successfully made and to very long times for the first gesture in the set of 5 compared with the other 4. We decided to remove the data from the first of the 5 gestures in all sets in all conditions. The average time to complete the remaining 4 gestures was used as the single score from a gesture in a given block. The data from each participant consisted of 24 scores, 1 for each of the 8 gestures in each of the 3 blocks.

### *4.5.1 Effect of practice*

To gauge the learning effect, the times in the 3rd block were compared with those of the 1st block across all gestures and all participants. There were 8 data points in each block for each of the 24 participants giving 192 values for each block. Although the order in which gestures were completed was counterbalanced between participants, each participant performed the gestures in the same order. Therefore we can examine the differences between the blocks using a paired t-test.

There was a significant improvement in time to complete a gesture between block 1 and block 3 (Table 1). As a consequence, it was decided to discard the data from blocks 1 and 2 from subsequent analysis, and use the data from block 3.

**Table 1** Time to complete a single gesture.

| Time (ms) per Gesture | Block 1 (8 Gestures) | Block 3 (8 Gestures) |
|---|---|---|
| **n** | 192 | 192 |
| **Mean** | 719 | 687 |
| **Stddev** | 332 | 290 |

Paired t-test, $p=0.04$

### *4.5.2 Main effect of number of legs in a gesture*

As stated earlier, we would expect a difference in performance such that the time to complete a 3-legged gesture is slightly less than 2 times that required for a 2-legged gesture.

**Table 2** Comparison of time to complete 2 and 3-legged gestures (block 3 only)

| Time (ms) per Gesture | 2 Legs (4 Gestures) | 3 Legs (4 Gestures) |
|---|---|---|
| **n** | 96 | 96 |
| **Mean** | 493 | 880 |
| **Stddev** | 332 | 290 |

Paired t-test p (1 tail) $< 0.0001$

The difference between the times for a 2-legged and a 3-legged gesture was highly significant, as expected (Table 2). The average time for a 2-legged gesture was about 0.5 second which compares favourably with dwell times commonly used in gaze communication for experienced users. The 3-legged gestures take longer and the ratio between them is 1.78 to 1, which was similar to what we expected on the basis of a simple comparison of the minimal number of saccades and fixations (less than 2 to 1).

### *4.5.3 Main and simple effects of the primary direction of eye movement in a gesture.*

The main effect of the primary direction (direction of the first gesture to the major zone) was not significant (Table 3).

**Table 3** Main effect of primary direction of movement (block 3 only)

| Time (ms) per Gesture | Oblique (square) (4 Gestures) | H/V (diamond) (4 Gestures) |
|---|---|---|
| **n** | 96 | 96 |
| **Mean** | 689 | 684 |
| **Stddev** | 282 | 298 |

Paired t-test p (1 tail) ≈ 0.69

However, the main effect includes both 2 and 3-legged gestures. The effect of the primary direction of movement is likely to be more pronounced for 2-legged gestures than for 3-legged. It is more interesting to look at the simple effect of direction of primary movement for 2-legged and 3-legged gestures separately.

**Table 4a** Simple effect of primary direction of movement for 2-legged gestures (block 3 only)

| Time (ms) per gesture | Oblique: 2 legs (2 gestures) | H/V: 2 legs (2 gestures) |
|---|---|---|
| **n** | 48 | 48 |
| **Mean** | 507 | 480 |
| **Stddev** | 125 | 150 |

Paired t-test p (1 tail) ≈ 0.05

**4b** 3-legged gestures (block 3 only)

| Time (ms) per Gesture | Oblique: 3 legs (2 Gestures) | H/V: 3 legs (2 Gestures) |
|---|---|---|
| **n** | 48 | 48 |
| **Mean** | 872 | 888 |
| **Stddev** | 278 | 269 |

Paired t-test p (1 tail) ≈ 0.25

There is small, but significant, difference between the times to complete 2-legged gestures where the primary direction of eye movement was horizontal/vertical compared with those with the primary direction being oblique. (Table 4a). There was no significant difference between the primary directions of movement for 3-legged gestures (Table 4b). These had either 2 oblique movements and 1 horizontal or vertical movement, or vice-versa. Adding a movement in the non-primary direction may have masked any small differences between the 2 primary directions.

Here there is a significant main effect, and gestures that begin with a leftward move are completed more quickly than those that begin with a rightward move (Table 5).

**Table 5** Main effect of direction of first eye movement (block 3 only)

| Time (ms) per Gesture | Left First (4 Gestures) | Right First (4 Gestures) |
|---|---|---|
| **n** | 96 | 96 |
| **Mean** | 669 | 704 |
| **Stddev** | 272 | 307 |

Paired t-test p (2 tail) < 0.02

Rather surprisingly, the source of the effect lies within the 3-legged gesture, and there is no difference in gesture completion for 2-legged gestures between those that begin with a leftward movement compared with a rightward movement, as shown in Tables 6a and 6b.

**Table 6a** Simple effect of direction of first eye movement for 2-legged gestures (block 3 only)

| Time (ms) per Gesture | Left First (4 Gestures) | Right First (4 Gestures) |
|---|---|---|
| **n** | 48 | 48 |
| **Mean** | 493 | 494 |
| **Stddev** | 144 | 133 |

Paired t-test p (2 tail) ≈0.96

**6b** 3-legged gestures (block 3 only)

| Time (ms) per gesture | Left First (4 Gestures) | Right First (4 Gestures) |
|---|---|---|
| **n** | 48 | 48 |
| **Mean** | 846 | 914 |
| **Stddev** | 255 | 287 |

Paired t-test p (2 tail) < 0.01

## 4.6 Analysis of errors

Errors were categorized as either being a valid gesture in terms of an allowable sequence of zones, but taking longer than the 2 second timeout period, or being a recognizable gesture but not the one that the participant was being asked to make at the time. In most cases, the latter category applied to 3-legged gestures where one of the regions was missed, so that it was recognized as a 2-legged gesture instead. In the introduction to section 4, it was noted that participants frequently referred to the gesture diagram during the first gesture. Consequently, all errors that were made before the first of the 5 repeated gestures in each set were ignored.

The frequency of errors in each category for block 3 only are show in Table 7. The current error analysis does not adequately detect attempts to make 2 legged gestures where the major region was missed out. The error data is therefore more reliable for 3 legged gestures. The total number of errors for these was 55, summing across both categories. These arose from 480 correct gestures (24 participants x 4 3 legged gestures x 5 gestures in the block). This represents an total error rate of 11%.

The result in section 4.5.4 was that gestures that began with a rightward movement first were significantly slower than gestures that began with a leftward movement. One reason could be that the former were perceived as being more difficult to make, which could be reflected in a greater number of errors made in rightward first gestures. Table 8 shows the errors made in block 3 for left first and right first gestures respectively.

The probability of this occurring by chance is p = 0.62 (chi square = 0.23, 1 df) and thus we cannot conclude that the number of made errors explains why rightward first gestures are slower.

**Table 7** Frequency of time out (A) and wrong gesture (B) errors for block 3 for each of the gesture parameter combinations

| | Primary Oblique | | | | Primary H-V | | | | Total | |
|---|---|---|---|---|---|---|---|---|---|---|
| | First Left | | First Right | | First left | | First Right | | | |
| **Error Type** | A | B | A | B | A | B | A | B | A | B |
| **2 Legs** | 1 | 1 | 3 | 0 | 1 | 0 | 0 | 0 | 5 | 1 |
| **3 Legs** | 1 | 6 | 4 | 11 | 4 | 10 | 6 | 13 | 15 | 40 |
| **Total** | 2 | 7 | 7 | 11 | 5 | 10 | 6 | 13 | 20 | 41 |

**Table 8** Frequency of time out (A) and wrong gesture (B) errors in block 3 separated by direction of the first eye movement in the gesture.

| Error Frequency | | Left first | Right First | Total |
|---|---|---|---|---|
| Error Type | A | 7 | 13 | 20 |
| | B | 17 | 24 | 41 |
| Total | | 24 | 37 | 61 |

## 5 Evaluating gestures during free game play

### 5.1 User experiences

We built a gaze gesture driven interface to support locomotion and fighting, and tested this in World of Warcraft with 12 able-bodied participants, all of whom were experienced gamers. We used the diamond shaped gesture regions and mapped the locomotion controls to the 4 2-legged gestures. Just like in most 3-D environments, the 'W','A', 'S' and 'D' keys can be used to control the movements of the players avatar in the game. A top region gesture switched a stream of 'W' key events on, and another top gesture switched the stream off. This caused the character to move forward. A bottom region gesture did the same for 'S' key events, causing backward movement. A left region gesture sent one 'A' event causing a turn to the the left and a right region gesture sent one 'D' key event causing a turn to the right.

The eight 3-legged gestures were assigned to commands for targeting other characters (in order to attack them), for launching attack spells and for launching healing spells. Similar commands were grouped into the same gesture region for ease of learning. The configuration of the gesture interface and the circular icons interface is shown in Figure 9.

The players were asked to freely locate and attack monster characters for about 5 minutes. This came at the end of an experiment where participants used gestures and other interaction techniques for locomotion and spell casting in a series of structured tasks. These took in total about 20 minutes to complete. In this study we actually compared different interfaces in real playing situations, but due to lack of space the study will be reported in detail elsewhere.

The outcome was very positive. All players were able to use the gesture driven interface to successfully move around in the game and to target and attack monster characters, after very little practice. Control over locomotion using gestures was experienced to be difficult, particularly during fighting. Turning was achieved by nudging their own character around in a series of discrete steps and this was effortful and time consuming. Although this was not a problem during locomotion over long distances that

**Figure 9** Using Gaze Gestures to control the game. The four triangle areas (one highlighted just to make it visible in this figure) are displayed as transparent layers, each one having the small round "hole" in it to help the player's focus in the triangle.

require occasional changes in direction. Gestures were considered to be very effective however for issuing discrete commands such as spell casting.

### 5.2 Frequency of accidental gestures during game play

There is a danger of course that unintentional gestures will result from normal eye movements. For the game playing described in section 5.1, the maximum time for the gestures was reduced from 2 seconds to 800 ms for 2-legged gestures, and 1600 ms for 3-legged gestures. This was done to try to reduce the likelihood of unintentional gestures. In a separate small study we asked 2 of the 12 gamers to play World of Warcraft for 15 minutes using keyboard and mouse while their eye movements were recorded.

We examined the frequency of gestures detected with both the diamond and the square shaped gesture zone configurations. We expected that the diamond configuration of regions would lead to a higher frequency of unintended gestures as we expected a greater frequency of vertical and horizontal eye movements, then oblique movements. The results are shown in Table 9.

**Table 9** Accidental gestures during 18 minutes of free play.

| | Gesture | player 1 | player 2 |
|---|---|---|---|
| **Diamond** | 2 leg: up | 3 | 12 |
| | 2 leg: right | 5 | 2 |
| | 2 leg: bottom | | 2 |
| | 2 leg: left | 1 | |
| | **Total** | 9 | 16 |
| **Square** | 2 leg: upper left | 1 | |
| | 2 leg: upper right | 1 | 2 |
| | 2 leg: lower left | 1 | 1 |
| | **Total** | 3 | 3 |

The observed data matched the expected data with far fewer unintended gestures where the primary direction was oblique compared with being horizontal and vertical (an average of 3 in the former case compared with 12.5 in the latter in 18 minutes of continuous play). It is noteworthy, but not surprising that no unintended 3-legged gestures were detected. Reliability of the chosen gesture scheme evidenced by few unintentional gestures is an important factor in the design of a gaze gesture based interface.

## 6 Discussion and Conclusions

We have investigated some of the design space of gaze gesture schemes intended for interacting with Massively Multiplayer Online Role Playing Games (MMORPGs). People can learn to make 2 and 3-legged gaze gestures fairly reliably after only a short amount of practice. The average times for completion of 2-legged and 3-legged gestures are 490 and 880 ms respectively, which compares favourably with dwell periods that are used in desktop applications for command selection by gaze. This means of interaction comes with the advantage of not having to fixate accurately on small targets in order to select them.

We were able to predict fairly well the ratio of completion times between 2 and 3-legged gestures on the basis of a simple comparison of the number of fixations and saccades in each. We were also able to predict the difference between 2-legged gestures where the primary direction of eye movement was, in one case, horizontal and vertical, and in the other, oblique. We did find a difference between gestures where the first movement was leftward, and where it was rightward, although this was confined to 3-legged gestures. We expected a possible effect in the other direction, so this was an unexpected effect. We were not able to find support for that in the literature. Becker (1991) states that there is tentative evidence that upward directed movements reach higher velocities than downward ones, but does not state anything about left and right bound movements. In fact, Abrams, Meyer and Kornblum (1989) found that there is no speed difference between left and right saccades. As we are not able to offer an explanation for this, we do not offer it as a significant finding. We are how-ever encouraged to look further into the area of modelling user performance when making different kinds of gaze gesture in order to be able to predict user performance with different schemes and variations on these.

In previous studies the time measured for a leg in a gaze gesture has varied a lot, e.g. Drewer and Schmidt: 557 ms, Heikkilä and Räihä 824-1190 ms and Mollenbach et al. 334 ms (see Section 2). Our figures vary from 247 ms (= 493/2) to 293 ms (= 880/3) for a 3 and 2-legged gestures. Why do the times vary so much and why are our times less than others? When we are comparing these small times and difference between them, we are working with single saccades and often single fixations and accuracy and consistency in time measurement is important. Using readymade fixation detecting algorithms provided by the manufacturer without reporting the exact parameters used for fixation detection is problematic. The 'eye-mouse' approach is also unreliable as the operating system needs to recognise gesture regions; update the cursor position; callback any mouse over events; and so on. How fast this sequence happens will partly depend on what other processes are doing at the same time on the same machine. We think that to report times for gestures reliably it is necessary to work at the sub-fixation level with times of gaze points that end and begin the starting and terminating fixations respectively.

We believe that gaze gestures are an effective means of interacting with MMORPGs, particularly for tasks which involve selection of commands, rather than continuous control of locomotion. 12 experienced gamers were able to use a gaze gesture only interface for free game play with World of Warcraft after very little training. We have shown that the rate of unintentional gestures during game play is much lower with the square configuration of regions (mostly oblique eye movements) compared with the diamond shaped configuration mostly (horizontal and vertical movements). If the directions of the gestures are not important (as they are in the case of locomotion tasks) then the square configuration is better to use in the games interface. It is likely however that the strength of gestures as an interaction technique for gaze–based gaming lies in its combination with other gaze-based interaction techniques, rather than trying to use it exclusively for all tasks..

## References


ABRAMS, R. A., MEYER, D. E., AND KORNBLUM, S. 1989. Speed and accuracy of saccadic eye movements: Characteristics of impulse variability in the oculomotor system. *Journal of Experimental Psychology: Human Perception and Performance 15*, 3, 529-543.

BECKER. W. 1991. Saccades. In Carpenter, R. H. S. (ed) *Eye movements. Vision and visual dysfunction; vol.8,* Basingstoke: Macmillan, 95-133.

BEE, N., AND ANDRE, E. 2008. Writing with Your Eye: A Dwell Time Free Writing System Adapted to the Nature of Human Eye Gaze, *Lecture Notes in Computer Science 5078*, Springer Berlin / Heidelberg, 111-122

DREWES, H. & SCHMIDT, A. 2007. Interacting with the Computer using Gaze Gestures. In *Proceedings of INTERACT 2007. Lecture Notes in Computer Science 4663,* Springer, 475–488.

HEIKKILÄ, H., AND RÄIHÄ, K.-J. 2009. Speed and Accuracy of Gaze Gestures. In *Journal of Eye Movement Research 3*, 2:1, 1-14.

HUCKAUF, A. & URBINA, M. H. 2008. Gazing with pEYEs: towards a universal input for various applications. In Proc. of the 2008 Symposium on Eye Tracking Research & Applications-ETRA '08. ACM Press, New York, NY, pp. 51–54.

ISTANCE, H., BATES, R., HYRSKYKARI, A., AND VICKERS, S. 2008. Snap clutch, a moded approach to solving the Midas touch problem. *Proc. of 2008 Symposium on Eye Tracking Research & Applications (ETRA '08).* ACM, 221-228.

ISTANCE, H., VICKERS, S., & HYRSKYKARI, A. 2009. Gaze-Based Interaction with Massively Multiplayer On-line Games. *In Proceedings of the 27th international Conference Extended Abstracts on Human Factors in Computing Systems,CHI '09. Conference.* 4381-4386

KAHN, D. A., HEYNEN, J., & SNUGGS, G. L. 1999. Eyecontrolled computing: The VisionKey experience. In *Proceedings of the Fourteenth International Conference on Technology and Persons with Disabilities. CSUN'99*, Los Angeles, CA.

MOLLENBACH, E., HANSEN, J. P., LILLHOLM, M., & GALE, A. G. (2009). Single stroke gaze gestures. *In Proceedings of the 27th international Conference Extended Abstracts on Human Factors in Computing Systems,CHI'09.* ACM, New York, NY, pp. 4555–4560.

PERLIN, K. 1998. Quikwriting: continuous stylus-based text entry. *Proc. of 11th Annual ACM Symposium on User interface Software and Technology (UIST '98),* 215-216, a demo available at http://mrl.nyu.edu/~perlin/demos/Quikwrite2.html

RAYNER, K. 1995. Eye movements and cognitive processes in reading, visual search, and scene perception. In Findlay, J. M., Walker, R., & Kentridge, R. W. (Eds.) *Eye Movement Research: Mechanisms, Processes and Application.* New York: Elsevier, 3-21.

WOBBROCK, J.O., RUBINSTEIN, J., SAWYER, M., AND DUCHOWSKI, A.T. 2008. Longitudinal evaluation of discrete consecutive gaze gestures for text entry. In *Proc. 2008 Symposium on Eye Tracking Research & Application.ETRA '08.* ACM, 11-18.

# Appendix H

# Research Output 7

H. Istance, S. Vickers, A. Hyrskykari
*The Validity of Using Non-Representative Users in Gaze Communication Research*
Proceedings of the ACM symposium on Eye tracking research & applications ETRA '12 ACM Press, New York, NY March 2012

*Copyright 2012 ACM, Included here by permission*

# The validity of using non-representative users in gaze communication research

Howell Istance*, Stephen Vickers* and Aulikki Hyrskykari**
*De Montfort University,** University of Tampere
hoi@dmu.ac.uk, svickers@dmu.ac.uk, ah@uta.fi

## Abstract

Gaze-based interaction techniques have been investigated for the last two decades, and in many cases the evaluation of these has been based on trials with able-bodied users and conventional usability criteria, mainly speed and accuracy. The target user group of many of the gaze-based techniques investigated is, however, people with different types of physical disabilities. We present the outcomes of two studies that compare the performance of two groups of participants with a type of physical disability (one being cerebral palsy and the other muscular dystrophy) with that of a control group of able-bodied participants doing a task using a particular gaze interaction technique. One study used a task based on dwell-time selection, and the other used a task based on gaze gestures. In both studies, the groups of participants with physical disabilities performed significantly worse than the able-bodied control participants. We question the ecological validity of research into gaze interaction intended for people with physical disabilities that only uses able-bodied participants in evaluation studies without any testing using members of the target user population.

**CR Categories:** H.5.2 [Information Interfaces and Presentation]: User Interfaces – Evaluation/methodology; Input devices and strategies.

**Keywords:** eye tracking, representative users, gaze communication, assistive input devices, physically disabled user groups.

## 1 Introduction

Gaze-based interaction techniques have been investigated for the last two decades for various purposes, often with the intention of benefiting people with physical disabilities. Text entry has received much attention in this regard. Much effort has been devoted to reducing the impact of using dwell-times as a means of signaling a selection of an on-screen object. Other interaction techniques, such as those based on saccades, have been investigated, motivated often by the desire to remove the dwell time element in order to gain performance improvements. Gaze gestures too have been investigated as means of activating a command, motivated again by a need to overcome some of the disadvantages of dwelling on an often small area of the screen until the dwell period has expired.

In this body of work, there are a number of assumptions commonly made. First, it is acceptable to base evaluation studies on the performance of able-bodied participants, even though the intended beneficiaries are people with disabilities of various kinds. It is rare that studies that rely on able-bodied participants go on to verify the findings in some way with members of the actual target user group, and report these. Second, the usability criteria important for able-bodied users of, say, text entry systems are just as appropriate for users with physical disabilities. Increasing the rate at which text, for example, can be entered is often the paramount concern, rather than, say, the ease with which the gaze-based technique can be used, or adapted for use, by a particular individual. Third, the emphasis on gaze-only interaction (with all of the attendant problems this causes) is justified, even if a large number of people with physical disabilities have varying degrees of motor abilities that might be used together with gaze. The whole 'Midas Touch' issue is a consequence of trying to use gaze for everything. Of course, there are groups of people with physical disabilities for whom gaze-only interaction will be the best choice of input modality.

There are indeed many applications of gaze-based interaction which are intended for an able-bodied user community, and which quite correctly rely on empirical evaluation with able-bodied participants and main stream usability considerations. The above concerns do not apply to these applications. It is however difficult to envisage a situation where an able-bodied user would choose to enter text by means of gaze-only interaction techniques, or to control a game without any hand operated input devices.

These concerns are echoed by Sears and Hanson (2011) in their discussion of how users are represented in accessibility research in general. They point to the problems that arise of evaluation of accessibility with users who are not representative of the target user group. They acknowledge that studying representative users is often problematic. However they argue that the literature contains numerous studies which have missed insights or given rise to inaccurate conclusions as a consequence of studying non-representative users. An extreme case of this would be where an interaction technique was shown to have particular advantages over other candidate techniques in studies with able-bodied users, only to be found to be unusable by most of the target user group.

Copyright © 2012 by the Association for Computing Machinery, Inc.
Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions Dept, ACM Inc., fax +1 (212) 869-0481 or e-mail permissions@acm.org.
ETRA 2012, Santa Barbara, CA, March 28 – 30, 2012.
© 2012 ACM 978-1-4503-1225-7/12/0003 $10.00

***Figure 1:*** *A participant from Group CP performing the middle-right-middle gesture sequence.*

## 2 Background and related work

Users with adequate cognitive ability and eye control but impaired muscle control can have considerable benefit from gaze-based communication. There are many different types of conditions that may affect physical abilities in different ways. The origins of these conditions are diverse. They may be hereditary or genetic, a problem encountered during birth, an illness affecting the brain, nerves or muscles or an accidental spinal or brain injury.

Some conditions, such as muscular dystrophy (MD), multiple sclerosis (MS), or amyotrophic lateral sclerosis (ALS) are neuromuscular and progressive [Bushby and Anderson, 2001], which means that over time muscle fibers will gradually weaken reducing muscular control. Other conditions, such as cerebral palsy (CP) are neurological and non-progressive [Rosenbaum et al. 2007]. People with CP constitute the largest group of physically disabled users which are thought to potentially benefit from gaze control technology [Donegan et al. 2011]. As with progressive neuromuscular diseases, there are many variants of CP but these are all chronic motor conditions that affect body posture, control and movement. In mild cases a person may have a limp or discomfort when walking, whereas in more severe cases a person may have no voluntary control over their arms, legs or even their tongue. In some neurological conditions such as locked-in syndromes or conditions caused by strokes and traumatic brain injuries the person affected may be completely paralysed whilst retaining almost all cognitive function.

Hornoff and Cavender [2005] evaluated their EyeDraw system using four users who had cerebral palsy. The four people already used an eye tracking system. They used the EyeDraw system remotely without ever meeting the researchers in person. Before and after this they answered a set of questions designed by the researchers. A new version of the system was designed by using the results from both the remote users and from a test with able bodied users. The researchers acknowledged that without the remote participants they would not have understood that the drawing skills of the disabled user group were considerably less developed than those of the able bodied participants in the laboratory study. GazeTalk is an gaze-based text entry system, which has been developed with continuous reference to and input from people with ALS, who are one of the main target user groups of the system [GazeTalk, 2006]. A high level of successful use by individuals with ALS has been reported.

## 3 Performance of participants with physical disabilities versus able-bodied participants

Two studies were conducted to establish to which the performance of able-bodied participants could be considered to be representative of people with physical disabilities. The studies were carried out in collaboration with a special needs school in UK, which has about 110 pupils, all of whom have motor impairments, most use wheelchairs, and about a third of the pupils have little or no verbal communication. One group of participants had some type of cerebral palsy (Group CP), while the other group had some type of muscular dystrophy (Group MD). The control group consisted of able bodied university students (Group AB). One of the studies compared the abilities and performance of the groups to make a series of defined gaze gestures between the two groups and the control group, while the other study investigated abilities and performance when making dwell-time selections of on screen buttons.

It is important to note that the conditions under which the data was collected were generally looser than would be normally be expected from an experimental comparison of performance. There were differences in the way in which instructions were given to each participant. These depended on individual factors, such as the level of verbal communication the young person had. Individual amounts of encouragement and help were given depending on the progress made the participant. Also the number of trials was limited so as not to compromise the participant's motivation to take part. This meant that testing all combinations of conditions was not possible, and a random selection of trials was used instead. Consequently the conditions under which the data were collected in both studies were broadly similar between the three groups but not exactly the same.

### 3.1 Study 1: Investigation of Gaze Gestures

The screen was divided into 9 zones (see Figure 1). In addition, 4 off screen zones were added, one along each edge of the screen. Small circles were visible in the centre of each of the zones as a fixation target. Gestures made into these off-screen zones have been previously used to change gaze interaction mode (Istance et. al. 2008).

Participants were seated in a way that was most comfortable for them in front of the screen under which was mounted a Tobii X120 eye tracker. All gestures started in the centre zone and required at least one fixation in one of the 8 other zones on screen, or one of the 4 off-screen zones, and then finished with one fixation in the centre zone (all were 2-legged gestures). Each gesture pattern was first demonstrated by the zones highlighting in sequence with accompanying audio tones. The participant was asked to then make the same sequence. This is similar to the 'Simon Says' type of game (see Fig.1). The patterns with the off-screen zones were indicated by an arrow in the direction of the required zone appearing instead of an on-screen zone being highlighted. The times recorded for a valid gesture begun from the time of the last gaze point in the last fixation in the centre zone, until the first gaze point of the first fixation back in the centre after the sequence had been completed. Any number of fixations in the target on-screen or off-screen zone were permitted in a valid gesture, and there was no limit on the time to make a valid gesture as long as the fixations were made in the appropriate zones. Any fixation in any other zone rendered the sequence invalid. Visual and verbal feedback was given on the successful completion of a gesture. Verbal encouragement was given throughout the trial. Data was collected in a similar way from participants in Group AB. Participants were asked to make a total of twelve 2-legged gestures, generated at random.

#### *3.1.1 Outcomes of the user trials*

Given the wide variability between the abilities of people with the same physical disability, we expected the variation in performance in Groups CP and MD to be considerable and to be much greater than the variability within Group AB. The most significant metric of performance is the number of gestures successfully completed. As can be seen this is very variable within the two groups of participants with physical disabilities. These range from completing all 12 sequences to completing none at all. A contingency table showing the frequencies of successful and unsuccessful gestures within each group is shown in Table 1. It is clear from inspection of the data that there is a highly significant group effect. Fisher's Exact Probability test for 2x2 contingency tables can be used here. We can examine first the difference between the CP and the AB group. The null hypothesis is that successes and (dis)ability are independent. There is overwhelming evidence to reject this, as $p < 0.0001$.Similarly, we can test the same hypothesis considering now the MD and AB groups only. Again, there is overwhelming evidence to reject this ($p < 0.00001$) There is also a significant difference between CP and the MD group ($p = 0.0007$), where the MD participants exhibit significantly better performance than the CP group.

| | CP | MD | AB | total |
|---|---|---|---|---|
| **success** | 23 | 48 | 68 | 139 |
| **fail** | 36 | 23 | 1 | 60 |
| **total trials** | 59 | 71 | 69 | 199 |

***Table 1:*** *Number of successes and fails in gesture completions summed across all participants*

There were several reasons why participants in CP and MD Groups failed to complete all 12 gestures. One was that involuntary head movements made during a trial caused the eye tracker to lose the eyes such that data was unobtainable. Another was that some participants had difficulty fixating inside the target zone initially, or back inside the centre zone. In some cases, participants had more difficulties fixating in zones in some parts of the screen than others. The 'gesture time (median)' row in Table 2 shows the median of times taken to complete the valid gesture sequences by each participant. The group medians do not include data from the participants who were unable to complete any gestures. The significance of the observed differences between the groups was calculated by resampling. The sum of the absolute differences between each pair of medians was used as the test statistic. The observed value is 562 ($|515 – 375| + |656 – 375| + |515 – 656|$) from the data shown in Table 2. The probability of the sum being as large as this by chance is $p= 0.0013$. The difference between the groups in terms of the times taken to complete successful gestures is highly significant. Examining the differences just between the CP and MD groups, the probability of obtaining an absolute difference between the medians as large as, or larger than, 141ms is $p = 0.26$. Consequently there is no significant difference between these groups in the time taken to complete the gestures.

### 3.2 Study 2: Investigation of dwell times

A similar study with dwell time selection of targets was carried out on a separate occasion with 3 similar groups of participants. Several target sizes were included, although only data from trials with the largest (300px, $7.5^o$ visual angle) are shown here. The screen was divided into a 3 x 3 grid, and the target appeared in one of the cells. As soon as it appeared, the participant selected it using a dwell time set to 800ms. If the target had not been selected within 10 seconds, then that trial was abandoned and considered a fail. The frequencies of successes and fails are shown in Table 3. Again, we consider pairs of groups and use Fishers Exact Probability for 2x2 contingency tables. Comparing only CP and MD groups, there is overwhelming evidence to reject the null hypothesis that (dis)ability and successes are independent ($p < 0.00001$). However, now there is no effect this time between the MD participants and the AB participants. Considering the times taken to make successful dwell time selections, the data shown in Table 4 includes the 800 ms dwell time. Using resampling the sum of the absolute differences between group means, the probability of the value of the statistic as being as large or larger than the observed value of 1203ms is $p= 0.002$. There is a highly significant overall group effect. Considering only MD and the AB groups, the probability of obtaining a difference as large as, or larger than, the observed difference of 93ms is $p = 0.056$. This difference is not significant, but close to being so.

| | CP | MD | AB | total |
|---|---|---|---|---|
| **success** | 14 | 49 | 44 | 107 |
| **fail** | 15 | 1 | 0 | 16 |
| **total trials** | 29 | 50 | 44 | 123 |

***Table 3:*** *Number of successes and fails in dwell selection completions summed across all participants*

| | Cerebral palsy group | | | | | | Muscular dystrophy group | | | | | | | Able bodied group | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **CP1** | **CP2** | **CP3** | **CP4** | **CP5** | **median** | **MD1** | **MD2** | **MD3** | **MD4** | **MD5** | **MD6** | **median** | **AB1** | **AB2** | **AB3** | **AB4** | **AB5** | **AB6** | **median** |
| age/gender | 14/f | 17/m | 8/f | 26/m | 13/m | | 16/m | 16/m | 13/m | 17/m | 16/m | 16/m | | 27/m | 28/m | 26/m | 22/m | 32/m | 26/f | |
| sequences attempted | 11 | 12 | 12 | 12 | 12 | | 12 | 11 | 12 | 12 | 12 | 12 | | 11 | 12 | 12 | 10 | 12 | 12 | |
| sequences completed | 7 | 3 | 2 | 11 | 0 | **4.6** | 9 | 10 | 12 | 8 | 9 | 0 | **8.0** | 11 | 12 | 12 | 10 | 11 | 12 | **11.5** |
| gesture time (median) | 453 | 422 | 235 | 578 | - | **515** | 609 | 742 | 399 | 789 | 656 | - | **656** | 422 | 375 | 368 | 360 | 359 | 383 | **375** |
| gesture time (IQR) | 430 | 172 | 16 | 274 | - | | 657 | 386 | 359 | 371 | 375 | - | | 180 | 51 | 141 | 70 | 109 | 150 | |

***Table 2*** *Results from trials with gestures*

| | Cerebral palsy group | | | | Muscular dystrophy group | | | | | | | | Able bodied group | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **CP1** | **CP2** | **CP3** | **median** | **MD1** | **MD2** | **MD3** | **MD4** | **MD7** | **MD8** | **MD9** | **median** | **AB1** | **AB2** | **AB3** | **AB4** | **AB5** | **AB6** | **median** |
| age/gender | 14/f | 17/m | 8/f | | 16/m | 16/m | 13/m | 17/m | 16/m | 16/m | 17/m | | 27/m | 26/m | 22/m | 30/f | 32/m | 26/f | |
| sequences attempted | 7 | 13 | 9 | | 8 | 7 | 8 | 6 | 7 | 7 | 7 | | 7 | 7 | 8 | 7 | 7 | 8 | |
| sequences completed | 7 | 7 | 0 | | 8 | 7 | 8 | 6 | 7 | 7 | 6 | | 7 | 7 | 8 | 7 | 7 | 8 | |
| dwell select time (median) | 1203 | 1703 | - | **1680** | 1094 | 1078 | 1109 | 1313 | 1344 | 1563 | 1664 | **1172** | 1125 | 1297 | 1062 | 1110 | 1031 | 1062 | **1079** |
| dwell select time (IQR) | 1219 | 1227 | - | | 43 | 140 | 583 | 1056 | 658 | 508 | 918 | | 55 | 40 | 12 | 274 | 24 | 11 | |

***Table 4*** *Results from trials with dwell time selection.*

## 4 Discussion

This work has been motivated by the need to verify that the results obtained from earlier investigations of gaze gestures with able-bodied participants (Istance et.al. 2010) are applicable to groups of users with physical disabilities. If the able-bodied participants in the current study are representative of the users with physical disabilities in terms of their eye movements, then we would expect to find no significant differences between the respective performances of the 3 groups. Considering gaze gestures first, then there is a big gulf between the performance of the able-bodied group on one hand and both of the groups with physical disabilities on the other. There is also a significant difference between CP group and the MD group in terms of successful gesture completions. It is well known that people with physical disabilities are far from a homogenous group. People's abilities and limitations vary widely, even within the same nominal category of disability, and furthermore, even within the same individual [Donegan et al. 2009]. A person's abilities may change gradually with time, and these may also vary over the day, as a consequence of fatigue or of illness related to the disability. This variability between participants in the same group is evident in our data. Performance ranges between nearly all successes to no success at all. Even when we consider the actual durations of successful gestures (from first fixation to last fixation) the differences between the 3 groups is significant and considerable. However, there is no significant difference in gesture times between the CP and MD groups. Considering gaze gestures as an interaction technique, then there is no reason on the basis of this study to consider able-bodied people's performance to be representative of the performance of groups of people with muscular dystrophy or cerebral palsy.

Considering dwell-time selection, the difference between successes and fails in completing dwell selections between the 3 groups is significant, but now there is no difference between the MD group and AB group. It is the performance of the CP group that accounts for the overall effect. However, in terms of the time taken to complete dwell selection tasks, there is a difference between the MD and AB groups, which is close to being significant. The data set here is very small, but it gives reason to suspect that able-bodied people's performance may not be representative of the performance of these types of disabilities with dwell selection either.

## 5 Conclusions

Many researchers take the position that able-bodied participants constitute a practical and realistic alternative to evaluation studies with participants who actually have physical disabilities. There is probably little or no alternative when investigating the design space surrounding novel gaze interaction techniques than to use able bodied participants, at least for initial exploratory studies, and to obtain base-line data for comparative purposes. However, the most fundamental tenet of user-centred design requires us to put the characteristics of the target user group or groups at the centre of the design process. This requires at least some verification that the interaction techniques evaluated by able-bodied users can be used by people with various types of disabilities. Sears and Hanson (2011) argue that expectations of how accessibility research is conducted and reported must be raised if this research is to have proper impact. The work is paper shows that there is a real danger that research into gaze-based communication will lack ecological validity if it cannot be demonstrated that it is to be applicable to the target user group.

## References


BUSHBY, K. M. D. AND ANDERSON, L. V. B. (2001), Muscular dystrophy: methods and protocols, Human Genetics. 109(4), 466-467.

DONEGAN. M., MAJARANTA, P., HANSEN, J. P., HYRSKYKARI, A., HIROTAKA A., HANSEN, D. W., AND RÄIHÄ, K.-J. (2011) Conclusion and a Look to the Future. in *Gaze Interaction and Applications of Eye Tracking: Advances in Assistive Technologies* (Majaranta et al., eds). IGI Global, 365-376.

DONEGAN, M., MORRIS, J. M., CORNO, F., SIGNORILE, I., CHIÓ, A., PASIAN, V., VIGNOLA, A., BUCHHOLZ, M. AND HOLMQVIST, E. (2009) Understanding users and their needs. *Universal Access in the Information Society, **8(4)***, 259-275.

Gazetalk,(2006) GazeTalk 5 eye communication system. http://www.gazegroup.org/research/15 (13.10.2011)

HORNOF, A. J., CAVENDER, A., (2005) EyeDraw: enabling children with severe motor impairments to draw with their eyes. *In Proceedings of the SIGCHI conference on Human factors in computing systems* (CHI '05). ACM, New York, NY, 161-170.

ISTANCE, H., HYRSKYKARI, A., VICKERS, S., AND CHAVES, T. (2009) For Your Eyes Only: Controlling 3D Online Games by Eye-Gaze. *In Proceedings of the 12th IFIP TC 13 International Conference on Human-Computer Interaction: Part I* (INTERACT '09), Springer-Verlag, Berlin, Heidelberg, 314-327.

ISTANCE, H., HYRSKYKARI, A., IMMONEN, L., MANSIKKAMAA, S., AND VICKERS, S. (2010) Designing Gaze Gestures for Gaming: an Investigation of Performance. *Proc. of 2010 Symposium on Eye Tracking Research & Applications (ETRA '10)*. ACM, 323-330.

ROSENBAUM, P., PANETH, N., LEVITON, A., GOLDSTEIN, M., BAX, M., DAMIANO, D., DAN, B. AND JACOBSSON, B. (2007), A report: the definition and classification of cerebral palsy*, Developmental medicine and child neurology*, 49, Issue Supp. s 109, 8–14.

SEARS, A., HANSON, V., (2011) Representing Users in Accessibility Research. *In Proceedings of the SIGCHI conference on Human factors in computing systems* (CHI 2011). ACM, New York, NY, 2235-2238.

# Appendix I

# Research Output 8

S. Vickers, H. Istance, A. Hyrskykari
*Performing Locomotion Tasks in Immersive Computer Games with an Adapted Eye-Tracking Interface* ACM Transactions on Accessible Computing, Volume 5 Issue 1, September 2013

*Copyright 2013 ACM, Included here by permission*

# Performing Locomotion Tasks in Immersive Computer Games with an Adapted Eye Tracking Interface

Stephen Vickers, De Montfort University
Howell Istance, De Montfort University
Aulikki Hyrskykari, University of Tampere

Young people with severe physical disabilities may benefit greatly from participating in immersive computer games. In-game tasks can be fun, engaging, educational and socially interactive. But for those who are unable to use traditional methods of computer input such as a mouse and keyboard, there is a barrier of interaction that they must first overcome. Eye-gaze interaction is one method of input that can potentially achieve the levels of interaction required for these games. How we use eye-gaze (or the gaze interaction technique) depends upon the task being performed, the individual performing it and the equipment available. To fully realize the impact of participation in these environments, techniques need to be adapted to the person's abilities. We describe an approach to designing and adapting a gaze interaction technique to support locomotion, a task central to immersive game playing. This is evaluated by a group of young people with cerebral palsy and muscular dystrophy. The results show that by adapting the interaction technique, participants are able to significantly improve their in-game character control.

Categories and Subject Descriptors: K.4.2 [**Social Issues**]: Assistive technologies for persons with disabilities

Additional Key Words and Phrases: accessibility, accessible gaming, eye-gaze, eye tracking, adaptive interface

## 1. INTRODUCTION

Online first and third person computer games are engaging, entertaining and immersive worlds. They can be social places where new acquaintances are made or existing friends met with. People are represented as avatars, a virtual projection of themselves where they choose to reveal as much or little about their appearance as they wish. What they do in these worlds is dependent upon the particular genre and target audience. Some are purely for social and entertainment purposes and may be designed for young children, teens or adults. Others have a more serious gaming component and are aimed at simulation, learning and commerce. Despite the actual genre or game, a collection of common game tasks (activities) exist: locomotion, camera control, object manipulation, communication and application control [Hand 1997; Bates et al. 2009]. Sometimes tasks are performed on their own, other times in parallel with one another. The method of performing these tasks is also common and typically requires various combinations of mouse and keyboard input. However, if a person is not able to use a mouse or keyboard to the extent required by the game tasks then they face a barrier of interaction. Different games may require a different level of interaction to be per-

Author's addresses: S. Vickers and H. Istance, Centre for Computational Intelligence, De Montfort University, The Gateway, Leicester, LE1 9BH, United Kingdom. A. Hyrskykari, TAUCHI-Unit, Department of Computer Sciences, University of Tampere, FIN-33014, Finland.
Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies show this notice on the first page or initial screen of a display along with the full citation. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, to redistribute to lists, or to use any component of this work in other works requires prior specific permission and/or a fee. Permissions may be requested from Publications Dept., ACM, Inc., 2 Penn Plaza, Suite 701, New York, NY 10121-0701 USA, fax +1 (212) 869-0481, or permissions@acm.org.
© 0 ACM 1936-7228/0/-ART0 $15.00
DOI 10.1145/0000000.0000000 http://doi.acm.org/10.1145/0000000.0000000

formed. Some of this interaction being used to perform tasks in real-time. Although, many augmentative and alternative communication devices exist as replacements to mouse, keyboard and game-pads, they are often too slow and cumbersome to use effectively in the context of immersive computer games.

Eye-gaze tracking systems can offer a high-bandwidth method of computer input with the potential of meeting the interaction requirements of these games. Video based, non-invasive, remote units exist that can estimate where a person is looking on a computer screen to within 0.5 degrees of visual accuracy (based on a distance of 60cm between user and screen). There has been much research on how best to use this 'gaze point' with much of it aimed at performing 2D based computer interaction tasks (e.g. [Zhai et al. 1999; Skovsgaard et al. 2010]) and text entry for communication (e.g. [Majaranta and Räihä 2002; Johansen et al. 2003]). There has been some work on using gaze input as an additional input device for computer gaming (e.g. [Smith and Graham 2006; Isokoski et al. 2009]) but very little on using gaze as a single input device for physically disabled users.

An example to illustrate the impact that game playing can have is that of a 14 year old girl who attends a state school in the UK that specializes in educating young people with physical disabilities. She has little verbal communication and attempts at physical movements can result in uncontrollable body movements. She uses a powered wheelchair that is controlled by her carers. During the game playing sessions at the school which are presented in this paper, she was able to control the movement of her own character using eye movements (see Figure 1). She was able to decide independently of others where her character would go and her parents and teachers commented on how engaged and motivated she appeared whilst doing so. After the sessions, she started to learn how to drive her own wheelchair using her head switch. The teachers reported, that in their opinion, this was as a direct consequence of her success in being able to control the game character.

Fig. 1. Controlling an in-game character for the first time using only eye movements.

There is often a phenomenalist approach to gaze interaction configuration [Donegan et al. 2005] as each person's abilities are unique. This means it is difficult to make any generalizations about which interaction technique suit different types of disability

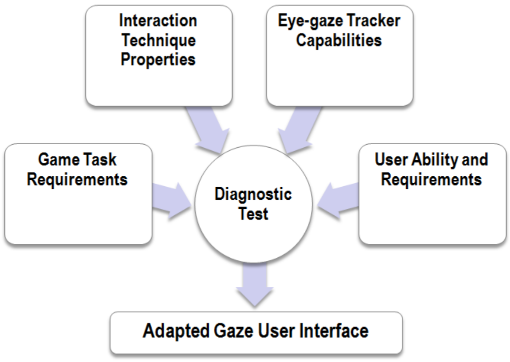


Fig. 2. The approach used in generating an adapted gaze user interface.

best. This results in human expert configuration being required to suit an individual. The aim of this work is to provide the basis for adaptive gaze driven interfaces for individuals with physical disabilities wishing to participate in immersive games and environments. These games have the potential to improve the quality-of-life as a means of leisure and entertainment in addition to providing support for game-based education. It is not possible to cover all game genres and the individual tasks within them so the focus in this paper is on performing locomotion based tasks. From an entertainment point-of-view, this is the task that makes the game immersive and engaging. From a serious games perspective, this task is central for the use of immersive games in education.

The approach used involves a diagnostic test that requires knowledge about the interaction technique, the task, the capability of the eye tracker and the requirements of the user, Figure 2. This paper first reviews previous literature on gaze interaction and adaptive user interfaces. Following, we present our implementation of performing gaze based locomotion and a method of adapting to suit an individual. This leads into our evaluations with physically disabled children and young adults with cerebral palsy and muscular dystrophy.

## 2. BACKGROUND

### 2.1. Eye-gaze Interaction

Eye movements are both fast as a means of input and are natural as a means of pointing when compared to other input devices. Users will typically look at the area of the screen where they wish to move to before they physically operate a mouse [Ware and Mikaelian 1987]. People do not necessarily think about the eye movements that they are making, so they are often performed subconsciously. But with practice, it is possible for a person to control their gaze such that it can be used as an effective computer input pointing device.

*2.1.1. Gaze Pointing.* One approach to using gaze interaction for computer control is by gaze pointing to emulate a mouse. Here the system cursor is placed at the gaze position with the underlying application being unaware that the cursor movement, button and keystroke events are originating from an eye tracker. Stampe and Reingold [1995] found that users who are already familiar with the desktop mouse found gaze pointing easy, intuitive and required virtually no training. In contrast, a person with severe physical disabilities who is not familiar may take much longer [Donegan et al. 2006]. Gaze pointing allows us to place the system cursor on objects that we wish to interact with. When it comes to interacting with the object such as initiating a mouse click, one of two methods can be used. The first is with multi-modal selection in which the user is physically capable of using a second input device such as a switch, sensor or voice (e.g. see, [Kumar et al. 2007; San Agustin et al. 2008]). The second is with mono-modal selection in which the user is unable to use a second input device and so selection must be achieved using only gaze input.

*Multi-modal selection* is achieved by the user moving the system cursor over the object that they wish to interact with and then activating their second input device. If the operation of the second input device is likely to have an affect on detecting the user's gaze position then filtering of the gaze data is required in order to stabilize the cursor [Jacob 1990; Stampe and Reingold 1995].

*Mono-modal selection* is achieved by the user performing either a natural or deliberate eye movement in order to initiate a command. In this context, deliberate eye movements are those where the user must use their eyes in an unnatural way, such as staring at an object for a prolonged period of time or perhaps using the eyes like a pen to draw specific patterns.

*2.1.2. Dwell.* Dwell selection is the deliberate fixating or dwelling on the object of interest for a set minimum time period; a dwell click [Jacob 1990]. This is usually longer that a typical fixation with times ranging from 600ms to 1500ms [Majaranta and Räihä 2002]. Most eye tracking systems allow for some kind of adjustment of what constitutes a dwell. In addition to being able to vary the length of a dwell it can also be made up of several smaller fixations and so the eye tracker needs to have a position tolerance of which consecutive fixations must fall within. This maybe useful as it can be difficult for a person to maintain their gaze within a small specified point for the length of a required dwell. One of the disadvantages with dwell selection is the inadvertent selection of objects by looking at them for too long. This is known as the Midas Touch problem [Jacob 1990]. It is possible to set an overly long dwell time (several seconds) to partially overcome the problem but this means that task times become longer and users can become frustrated and tired [Majaranta et al. 2006]. In general, there is a balance and trade off between long dwell times (longer task times, frustration and tiredness) and short dwell times (shorter task times but inadvertent selections).

### 2.2. Fitting Assistive Technology to the User

Setting an eye-gaze system for a user often requires expert knowledge of the system and the individual. Configurations such as, changing the dwell select time; changing the size of the interface buttons and their location on screen; creating custom dwell keyboard configurations for communication; the amount and speed of a zooming interface; all require expert knowledge. This is often configured on a time expensive, individual user and trial-and-error basis [Randolph and Moore Jackson 2010; Dawe 2006].

Previous work, on user interface adaptation for physically disabled users include, adaptation of web pages (e.g. see, [Mankoff et al. 2002; Bigham et al. 2006]); automatic generation of GUIs [Gajos et al. 2008] and dynamic adaptation of GUIs [Carter et al. 2006]. No specific research into interfaces automatically generated to suit an individual using eye-gaze could be found; this has previously been a manual process.

Of particular interest to this work is the research by Gajos et al. [2008] and Wobbrock et al. [2011]. The authors developed two systems for automatically generating user interfaces. The first system SUPPLE, uses a preference elicitation engine in order to model a preferred user interface configuration. The elicitation engine is a computer guided process with which the user must select their preference out of a pair of user interface fragments presented on screen. The fragments differ in presentation but

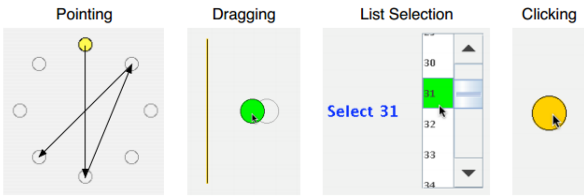


Fig. 3. Four task types used to measure participant's motor capabilities, SUPPLE++. Included here by permission [Gajos et al. 2008].


Fig. 4. Two different user interfaces based on SUPPLE preference elicitation for two different users (AB03) and (MI09). The first interface is the baseline interface. Included here by permission [Gajos et al. 2008].

functionally would be the same. The second system SUPPLE++, models user's motor abilities from a set of one time motor performance tests. Here, four task types are used (also see Figure 3):

— Pointing : A set of tasks based on ISO9241-9 standard [ISO 2000], where the user is required to move the pointer to targets at various sizes and distances
— Dragging : A task designed to be similar to the action of using an interface scroll bar component
— List selection : A task where the user is required to make multiple selections from a given list of menu items or scroll bar components
— Multiple clicking : A task where the user is required to click within circular targets of varying sizes.

The type of input device used by participants in their studies was dependent upon the individual and included: mouse operated by fingers, one hand or two hands; trackball operated by chin, backs of the fingers, back of the hand and bottom of the wrist. The range of conditions by participants included cerebral palsy, muscular dystrophy, spinal cord injury and Parkinson's. Based on the outcomes of the models two interfaces will be generated, one based on preference and one based on ability, see Figure 4. They compared the two interfaces along with a traditional dialog interface and measured the individuals ability (in time and subjectively) to complete a series of tasks. They found that participants were able to perform the tasks 10% faster using the preference interface and 28% faster using the ability interface than when using the base-line. Additionally, users made 73% fewer errors when using the ability interface and subjectively, strongly preferred (in terms of efficiency and ease of use) both the preference and ability interfaces.

### 2.3. Eye-gaze and Gaming

Isokoski and Martin [2006] used a First Person Shooter game in order to assess the performance of eye gaze as an extra input modality to mouse and keyboard. The control system of such games uses W,A,S,D/cursor keys for movement and mouse for camera control and shooting. The trials required subjects to walk around a bespoke game environment and shoot at moving targets as they appear. Eye-gaze was used for aiming of the weapon, with an XBox 360 controller to move the avatar. Their first findings showed that eye gaze will not always improve the performance of the players when compared to using the hand controller for aiming instead of gaze, but they found that the number of hits from gaze is comparable to the game pad controller used alone, and that gaze was often more entertaining.

Smith and Graham [2006] performed an experiment using a similar control system (gaze to aim, mouse to move the camera, keyboard to move avatar) on an open source port of the game Quake 2 called Jake2. Similar to Isokoski et al., Smith and Graham did not find any advantage in performance. However, their subjective user results showed that using eye gaze offered a more immersive experience than using a mouse and keyboard. Jönsson [Jönsson 2005] also found that a combination of mouse and eye gaze was more immersive than mouse and keyboard when performing trials using the first person shooter Half Life.

Smith and Graham also performed trials using a version of the 80s arcade game Missile Command. In this game the user is required to defend several cities at the bottom of the screen from missiles falling from above. The missiles are to be destroyed by the user intercepting their flight path with a gunshot. Trials were conducted using a mouse to aim and shoot, and then by using gaze to aim and a button to shoot. The trials showed that there is a need to fire far ahead of the missile path and this is easily achieved using a mouse. However, it is difficult when using eye gaze to fire ahead as there is a constant distraction of the missile itself (the users looked at the missile rather than where they wished the missile to go). Thus, the majority of eye gaze shots missed and fell behind the missile. A similar style of game: Breakout, was tested by Dorr et al. [2007] and they demonstrated in their implementation that eye-gaze can outperform a mouse. Breakout involves hitting and destroying bricks by bouncing a ball off a paddle that the user controls at the bottom of the screen. The paddle is moved in a horizontal direction only and so is simply controlled with left and right movements of a mouse or cursor keys and these can be easily translated for use with gaze. The trials required users to compete on a one-on-one basis: one user with gaze and the other with mouse. The results showed that two-thirds of all rounds were won by the gaze player. The game play using gaze was so natural and effective that one user did not realize she was even playing the game and after 2 minutes asked when the experiment would actually start. This was due to her constantly watching the ball, meaning that the paddle would follow her gaze and so ball and paddle would always intersect, provided she watched the ball.

### 2.4. Avatar Locomotion in Immersive Games

There has been much work on general avatar control and locomotion in 3D immersive and virtual environments (see [Hand 1997; Tan et al. 2001; Burigat and Chittaro 2007]) but little research on avatar control for people with physical disabilities. Yuan et al. [2011] provide a comprehensive review on accessible gaming in general. The authors examine several genres of gaming and describe several strategies that could make the games more accessible.. This ranges from reduction or automation of the interaction required to support physically disabled users, to the replacement and enhancement of visuals to support visually impaired users. The authors review several examples of avatar control but consider immersive games and virtual environments to be different in terms of the inherent tasks found within. Bates et. al [2009] have a different approach with the consideration that both virtual environments and immersive games both require real-time interaction. The authors discuss a barrier of control and a Turing test of interaction, in which players are unable to determine whether another player is disabled by how their avatar responds in game. Players found in both environments expect almost immediate responses from other players in terms of movement and communication. If a disabled player is unable to respond to another player within an acceptable time period, then that player may misinterpret this as meaning the disabled player is ignoring them.

Trewin et al. [2009] developed their own virtual environment (PowerUp) and implemented a range of accessibility features. This allowed users to configure input controls and avatar response times as well as portions of the in-game feedback such as text and speech. The authors evaluated several methods of avatar control (both looking around and walking) that included: key press, toggle/latch key press, mouse move, mouse click (with on-screen buttons) and assisted control with auto-look and auto-walk. One of their findings was that a simple key press was more effective for walking tasks and mouse/track pad/track ball movement preferred for looking tasks.

Folmer et al. [2011] developed a technique for single switch navigation in an immersive environment. Traditional switch scanning systems are typically slow and not necessary task aware and so the authors developed a hold-and-release system. This method allows the users to latch control inputs on and off. Although this requires the user to perform an additional switch press to disengage a movement, it can be more suited to tasks such as forward movement which require prolonged key presses.

## 3. DESIGNING AND ADAPTING A GAZE ONLY INTERACTION TECHNIQUE FOR USE IN IMMERSIVE GAMING ENVIRONMENTS

To better understand the interaction requirements of the tasks being performed in these games and environments a task analysis was conducted. This involves observing expert players performing typical game tasks. The rationale behind using expert players is that in becoming experts these players have found the most efficient ways of playing and interacting with a particular game. This helps us in separating the most important components of the interaction from the less important. During this process the following data is collected:

— Gaze position
— Think-aloud and retrospective think-aloud
— Video capture of game window
— Low level input event capture

The gaze position provides us with information where the user's attention is during a particular task. This is crucial as in order to design or select an interaction technique that feels natural to use then it is important to allow the user to focus their attention on what is happening on screen. The think-aloud protocols identify what the user was thinking and intending to do during the task. The low level data identifies what the user was actually doing by capture of mouse and keyboard events generated.

Different genres will have different task sets but here we are focusing on MMORPG (Massively Multiplayer Online Role Playing Games) and MUVE (Multi User Virtual Environments). Five expert players in World of Warcraft took part in the task analysis. An expert in this case is defined as a player who has reached the highest character level in the game and has been successful in end-of-game content. All were male and students at De Montfort University with a mean age of 22. None of them had previously used an eye tracker. A task was created that required the player to speak to a non-player character, search for a series of monsters and attack them all before returning back to the character. The task took approximately five minutes to complete.

The data was segmented into game tasks and then aggregated for all participants. All were analyzed but only locomotion type tasks are discussed here. Two were identified: standing still but searching and actual locomotion. During instances of the locomotion task much of the participants gaze is focused on and around the avatar. This is due to their gaze scan path moving from the avatar in the centre - out to the left or right - back to the avatar - and then to the opposite side. This produces a horizontal scanning eye gesture that develops into a ‘butterfly’ visualization, see Figure 5. Interestingly, it was found that avatar rotation was often preceded by the gaze point being at the associated left/right butterfly visualization edge.


Fig. 5. Heatmap showing eye-gaze fixations when performing locomotion tasks using keyboard and mouse.

The input event data, as expected, showed W, A, S and D keyboard events for the avatar movement in addition to left mouse drag for some camera manipulation. Although, the S key (backward movement) was used the least frequently, it was used on occasions in positioning of the character and also when character locomotion was used in combination with other tasks, such as fighting.

### 3.1. Designing a Novel Method of Gaze Only Locomotion

The gaze data indicated that a large portion of fixations are on the avatar, and during this time there is a reduced amount of rotation compared to when the gaze point moves to the edges. So whatever technique is implemented, the user should be able to move the avatar forward whilst keeping their visual attention on the avatar. Further, when the user scans horizontally there is a possibility that they are wanting to turn in that direction. One interaction technique that would allow this task requirement to be fulfilled is the transparent overlay [Vickers et al. 2008].

*3.1.1. Transparent Overlays.* In our instance the transparent overlays operate as follows. They are gaze sensitive zones that may or may not be visible to the user. When the user’s gaze enters one of the zones for at least 80 ms then a series of mouse or key events is sent to the game. It is important to note that 80 ms is simply 6 samples[1] of eye tracking data recorded at a rate of 1 sample per 16 ms (60 Hz). This is in contrast to a dwell which, as discussed in Section 2 ranges from 600 ms to 1500 ms [Majaranta and Räihä 2002]. The filtering of eye tracking data to 5 samples is required to avoid accidental activation of other zones as the user looks from one zone to another. This number may be optimized further but during testing this value gave the most responsive performance and so offers a useful starting point. With all input devices there is latency in their operation, both from a physical and cognitive perspective [Card et al. 1983] and although no comparative study could be found it would serve as useful future work. The zones can be positioned around the screen based upon the gaze attention data. The necessary input events can then be assigned to each zone. This results in a layout as found in Figure 6 (right). The zone configuration is based on screen segmentation rather than placing buttons around the screen. The use of spherical zones was considered but this would have resulted in the screen areas outside the rounded corners of the zones being unusable. Further fine tuning can occur within each zone if necessary such as dynamic rotation and movement speed variation but here the focus is only on directional control. This allows a user to perform the locomotion part of the task but not the standing still and looking around part. To achieve this the movement regions from the screen are removed resulting in Figure 6 (left). These two interaction modes: 'locomotion mode' and 'look around mode' offer an interaction technique starting point. Although, it cannot be assumed that the technique in this form will be fully accessible, so possible variations need to be considered.


Fig. 6. Transparent overlay positions for 'looking around' (left) and 'locomotion' (right).

We propose that gaze interaction adaptation can be applied using the following taxonomy:

— Complexity: both in regards to the demand on the user and the number of available options to the user
— Accuracy: the size of the interaction elements
— Spatial position: the positioning of interaction elements
— Temporal: the length of time required to initiate a single step of the interaction technique
— Intelligent: learning user behavior in order to adapt or part automate techniques

In this first instance, we want to consider the possibility of applying spatial positioning and complexity as a means of adapting the locomotion interaction technique.

[1] 80 ms consists of 6 sample points, which encompass 5 x 16 ms intervals.

### 3.2. Spatial and Complexity Adaptation of the Locomotion Interaction Technique

The task analysis has indicated the input events that are required for this task. In applying a priority control system to the avatar movement controls and the interaction zones, different technique configurations can be created. Each configuration can be considered as being one of the varying levels of avatar control proposed in Figure 7. As the range of controls reduce, so do the levels with the simplest being Level 1;

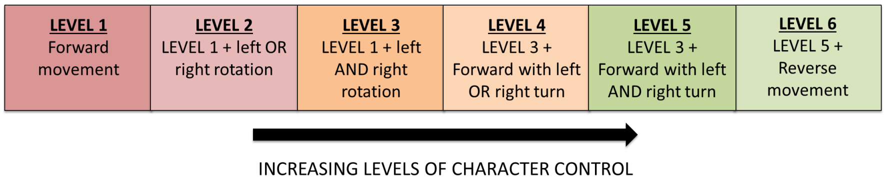


Fig. 7. Proposed priority levels of avatar control.

here there is only moving the character in a forward motion. Configurations that are classed as Level 6 fulfill the whole range of avatar control requirements. Based upon this, avatar controls and interaction zone preferred positions can be organized as in Table I. Although, the task analysis showed that moving the avatar backward was performed the least frequently it is still included as part of the levels of control. This is because it is anticipated that users of the eye-gaze only interaction technique may become stuck in the virtual environment and would need a method to manipulate their character backward. Essentially, this method is using spatial adaptation (moving the

Table I. Overview of the priority system to which functions/controls are assigned to zones. See Figure 8 for zone positions.

| | **Preferred Avatar Control Zone** | | | |
|---|---|---|---|---|
| *Avatar Control List* | *1st* | *2nd* | *3rd* | *4th* |
| Forward | $Z_5$ | $Z_8$ | $Z_2$ | $Z_4$ or $Z_6$ |
| Left | $Z_7$ | $Z_4$ | $Z_1$ | $Z_8$ or $Z_2$ |
| Right | $Z_9$ | $Z_6$ | $Z_3$ | $Z_8$ or $Z_2$ |
| Forward/left | $Z_4$ | $Z_1$ | x | x |
| Forward/right | $Z_6$ | $Z_3$ | x | x |
| Backward | $Z_8$ | x | x | x |

overlay zones) and complexity adaptation (changing the number of available overlay zones). Gajos et al. based their user interface adaptation on the use of two methods: user preference (elicitation) and assessment. There may be occasions when users are unable to communicate preference of this detail to us, so here the adaptation is based on assessment; a diagnostic test. The format and function of the test being derived from the interaction technique that it represents. The metric used within the test must be aimed at what the user can achieve with the hardware available to them, the current environmental conditions and their own current needs and requirements. Considering this it is possible to determine a series of influences that are likely to affect gaze interaction performance.

— Hardware: This includes the eye-gaze tracker hardware; its tolerance to head movement; latency and software.

— Environment: This includes lighting and any distractions that are surrounding the user.
— User: This is all of the factors relating to the user in consideration of using an eye-gaze tracker. This includes head control; head positioning; eye construction and eye control; understanding; stress and emotional health; physical health.

The level of these attributes have a direct influence on the amount of noisy data produced that our adaptation should consider. A higher quality eye-gaze tracker may be more tolerant to many of these influencing factors than a lower quality device; additionally, an individual may perform differently depending upon the time of day, level of health, fatigue and so on. It is important to collect all data and that the diagnostic test does not simply record passing or failing a specific task. So, it is a measure of interaction reliability whilst performing the test rather than success in completing it.

### 3.3. Diagnostic Test and Reliability Metric

The locomotion interaction technique uses several on-screen zones that are operated when the user gazes within them for at least 80ms. The diagnostic test is designed to operate in a similar manner and is described in a series of steps in Table II. Figure 8 shows a test in progress. Based upon the test design, influencing factors and data

Table II. The steps of the diagnostic test.

| *Step* | *Event* |
|---|---|
| 1 | The participant is presented with a black, blank screen and a countdown timer to indicate the start of the diagnostic test |
| 2 | Once the timer has completed an on-screen target will appear within a random time interval between 1-5 seconds |
| | The on-screen target will be in one of the positions and of the same size as one of the interaction zones |
| 3 | The participant must then perform a single fixation anywhere within the bounds of the on-screen target |
| 4 | If successful the on-screen target turns green and disappears |
| 5 | A new on-screen target will then appear at a random time interval between 1-5 seconds |
| | There are nine possible on-screen target locations (see Figure 8) and each position appears twice during the test; thus a total of 18 test sequences |
| | The order that the on-screen targets are presented to the participant are random |

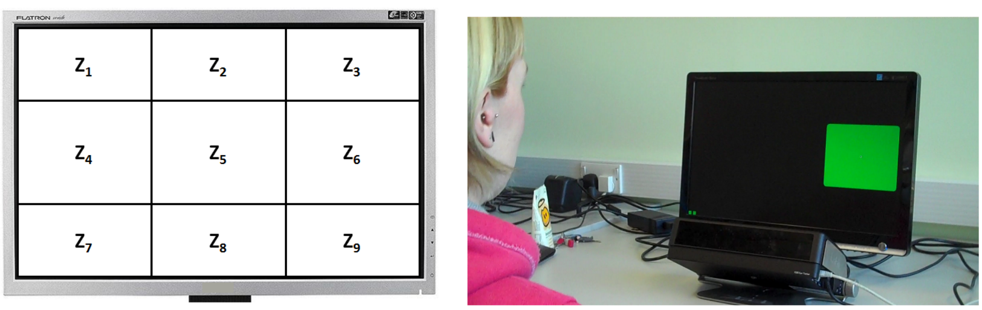


Fig. 8. Diagnostic test zone positions, based upon the transparent overlay 'locomotion' interaction technique (left) and diagnostic test in progress (right).

collected the following assumptions are made:

(1) The longer the time taken to complete the test sequence then the poorer the reliability.
(2) The more fixations made by an individual when performing a test sequence then the longer the time taken to complete the test sequence; and subsequently the poorer the reliability.
(3) The longer the length of time that at least one eye is not being tracked then the longer the time taken to complete the test sequence; and subsequently the poorer the reliability.

One important consideration is for the number of fixations made and the percentage of time that the eyes are being tracked. Here, there is a situation of one not existing without the other, for example, if the eyes are not being tracked then it is impossible to measure and determine the number of fixations being made; likewise, if fixations are being measured then the eyes are being tracked. Thus, only one of these data measures can be used at any one time. Considering this, the choice of which data measure to use must default to the lower of the two. The metric of reliability is defined as follows:

$$R_{interaction} = \begin{cases} \frac{(R_{time}+R_{fixation})}{2} & \text{if } R_{fixation} < R_{eyes} \\ \frac{(R_{time}+R_{eyes})}{2} & \text{if } R_{eyes} < R_{fixation} \end{cases} \tag{1}$$

Where $R_{time}$ is a measure of reliability relative to the time taken to complete the test sequence; $R_{fixation}$ is a measure of reliability relative to the number of fixations made; and $R_{eyes}$ is the percentage of time that the eyes were visible by the eye tracker during the test sequence. Re-factoring as a percentage $R_{time}$ is defined as:

$$R_{time} = \frac{(T_{max} - T_{user})}{(T_{max} - T_{min})} \tag{2}$$

Re-factoring as a percentage $R_{fixation}$ is defined as:

$$R_{fixation} = \frac{(F_{max} - F_{user})}{(F_{max} - F_{min})} \tag{3}$$

Where $F_{user}$ is the number of fixations made and $F_{max}$ is the maximum number of fixations allowed by the test. $R_{eyes}$ is defined as:

$$R_{eyes} = \frac{T_{eyes}}{T_{user}} \tag{4}$$

Where $T_{eyes}$ is the length of time that the eyes were visible by the eye tracker during the test sequence and $T_{user}$ is the length of time to complete the test sequence. Note that parameters are given an equal weighting to serve as a starting point.

### 3.4. Setting Parameters to Determine $T_{max}$, $T_{min}$, $F_{max}$, $F_{min}$

$T_{min}$ and $F_{min}$ can be set by the interaction technique requirement; that is, the minimum time to initiate an interaction zone is a fixation of at least 80 ms. In order to determine the values for $T_{max}$, $F_{max}$ we performed an initial evaluation of the diagnostic test.

To provide a base-line measure, five able-bodied participants were recruited from De Montfort University to perform the test. All were students with four being male and one female. The mean age was 23. All had taken part in previous eye-gaze studies and had used the locomotion interaction technique once previously. All five completed the diagnostic test with no difficulties. The results are as follows. The mean $F_{user}$ was

**ALGORITHM 1:** Adapting the locomotion interaction technique based upon interaction reliability

**Input**: A set $C_{list}$ (Control List) = {*see Table I of Controls*}
**Input**: A set $P_{list}$ (Preference List) = {*see Table I*} of $Z_i$ (Zone Positions)
**Input**: $R_{zone}$ (Interaction reliability for each zone)
**Input**: $R_{interaction}$ (Mean interaction reliability based on all zones)
**Output**: Matrix of preferred control positions:

$$Z = \begin{vmatrix} Z_1 & Z_2 & Z_3 \\ Z_4 & Z_5 & Z_6 \\ Z_7 & Z_8 & Z_9 \end{vmatrix}$$

```
if R_interaction < 70% then
    foreach Control in C_list do
        foreach Z_i in P_list do
            if Z_i is available then
                if R_zone > 30% then
                    Z_i = Control
                    break;
                end
            end
        end
    end
end
```

2.6 fixations and the mean $T_{user}$ was 496 ms. It is necessary to set a maximum time and fixation limit to avoid prolonged test sequences. This was decided by the use of an arbitrary safety factor that was set during development of the diagnostic test. By rounding the figures and considering a safety factor of 5 for the maximum number of allowable fixations and a factor of 10 for the maximum allowable time the parameter values can be summarized as follows:

— $T_{min}$ = 80 ms
— $T_{max}$ = 5000 ms
— $F_{min}$ = 1
— $F_{max}$ = 10

### 3.5. Diagnostic Test Algorithm

The priority function system operates as described in Algorithm 1. It operates as follows: if the user's overall $R_{interaction} >= 70\%$ then it can be assumed that they are capable of using the transparent overlay with no modification. If it is less than 70% but greater than 30% then avatar movement controls should be assigned to zones based on the preferred positions shown in Table I. There is a balance between the user attempting to use a zone and the building of frustration for failed attempts and so a bottom value is required. These values are arbitrary and to exist as a starting point for evaluation. Zone positions can be seen in Figure 8. An example of how an adapted interface may look to the user can be seen in Figure 9.

### 3.6. Risk of Non-conventional Design

Gestalt[2] psychology suggests that the brain operates with a tendency to self-organize in a holistic, parallel way. The use of 'gestalt laws' [Sternberg 2003] such as closure,

[2]Gestalt (German) - 'an organized whole that is perceived as more than the sum of its parts'

Fig. 9. A modified version of the locomotion interaction technique. During this training phase, the transparent overlays are visible and can be switched on and off as required.

similarity, proximity, symmetry and continuity are encouraged in user interface design. The unmodified interaction technique follows these laws, in that the controls are presented symmetrical to one another. By applying the same laws to the modification of this interaction technique, the amount of modification possible would be limited. Therefore, in this context the gestalt laws are considered lower in priority than the providing of function to the user. This may result in a user interface that appears unnatural in function and appearance but is customized to suit the individual.

### 3.7. Implementation

Our previous work has focused on gaze-only control of existing immersive games and virtual environments [Istance et al. 2009]. This has resulted in us developing a middleware software platform called 'Snap Clutch'. The original concept was a method of temporarily disengaging eye control by glancing off-screen, or 'de-clutching' eye control, as a means to overcome the Midas Touch problem [Istance et al. 2008]. Today, the software exists as a means of interacting with virtual environments and computer games. It takes input from an eye-gaze tracking system and converts that into key and mouse system events through different interaction techniques [Vickers et al. 2008]. An advantage of it being a middleware solution is that it works independently of the target game and so theoretically will work with any game that accepts events from the operating system message queue. It was developed using the Tobii SDK but is also compatible with the Eye Tracking Universal Driver[3]. The architecture of Snap Clutch is such that new gaze interaction techniques can quickly be created and added for use by the application. There are four interaction techniques (or modes) available to a user at any one time. In order to change between each mode the user need simply glance gesture off one of the four edges of the screen. The user is notified by a mode change with speech (if activated) and a green bar that sits on the screen edge that they have just glanced off. During this previous work several different gaze-only interaction techniques and their

[3]ETU Driver: http://www.sis.uta.fi/ csolsp/projects.php

ability in fulfilling the range of tasks found in MMORPG/MUVE environments were evaluated. These tasks included: character locomotion, camera control, object manipulation, application control, communication and fighting. Participants would perform a series of tasks using gaze only interaction and also keyboard and mouse. Within these studies it was found that although gaze-only was not as quick as the keyboard and mouse, a beginner level of gameplay was possible using only eye-gaze. The limitation with this previous work however, is that only able-bodied participants took part in the study.

## 4. INITIAL EVALUATION

Two different participant groups took part in the initial evaluation of the locomotion interaction technique.

— Group A: This group is formed of individuals that would greatly benefit from eye-gaze and could be used as their primary means of communication. In this study all participants have cerebral palsy (CP). Using the Gross Motor Function Classification System (GMFCS) all are level 5 and the Manual Ability Classification System (MACS) are level 5.
— Group B: This group is formed of individuals who do not necessarily need to use eye-gaze as their primary means of communication at this moment in time, but due to their degenerative condition it may aid or benefit them in the future. In this study all participants have muscular dystrophy (MD).

### 4.1. Participants

*4.1.1. Group A Participants: .* Participant GPA-1 is male, 13 years old and has athetoid cerebral palsy. His condition is such that he is always sitting in a divergent position, see Figure 10. Most of his body movements are the result of involuntary actions and any deliberate movements appear erratic. He is unable to speak but able to vocalize and use facial expressions for yes and no answers. His primary communication is through an Etran frame and an electronic communication aid operated using a switch activated through kicking with his left foot. This is with great effort and results in long switch activation times.

Participant GPA-2 is female, 14 years old and has cerebral palsy. Most of her body movements are the result of involuntary actions and any attempt at deliberate movements are erratic. She has difficulty in speaking but is able to communicate slowly with some single words which requires much effort. Her primary communication is through an electronic communication aid operated via a head switch mounted on her wheel chair.

Participant GPA-3 is female, 8 years old and has cerebral palsy. She has minimal muscle tone and struggles with all types of body movement. When at full health she is able to use a touch sensitive switch but fatigues after a few minutes. She is unable to speak but can vocalize and smile when she is motivated. She is able to use an Etran frame for classroom activities for short periods.

Participant GPA-4 is male, 17 years old and has cerebral palsy. Most of his body movements are the result of involuntary actions and any attempts at deliberate movements are erratic. He has virtually no spoken language but can vocalize to make yes and no answers. His primary means of communication is via two hand operated switches to operate an electronic communication device. This requires much effort and often requires several attempts for a successful switch press.

All group A participants have had brief sessions on using an eye tracker for communication during the year previous to the sessions starting.

Fig. 10. Participant GPA-1 sitting in his natural divergent position.

*4.1.2. Group B Participants: .* Participants GPB-1 through to GPB-8 are all male and aged from 15 to 18 years; all have muscular dystrophy. Their current condition means that they require a wheel chair for mobility that is controlled via a joystick. They can all communicate verbally and are able to use a mouse, keyboard and gamepad however, it is at a slow pace with poor accuracy and often only for short periods of time. Low screen resolutions are typically used to allow for inaccuracies when using a mouse as many of the individuals are restricted with their hand and arm movements.

### 4.2. Procedure

The purpose of the initial evaluation was to investigate how well each participant could use the unmodified interaction technique. The evaluation took place over two sessions. Each session lasted 15-30 minutes and the time in between these sessions was approximately two weeks. To assist with the trials a senior support teacher familiar with the children taking part was present throughout. World of Warcraft was used as the virtual environment. It was chosen because in addition to being an online social computer game, it is also an interesting, colorful and engaging virtual environment that can be freely explored. The game was configured as part of a private server so that there was full control over the environment. No other players were present or enemies spawned during the study.

World of Warcraft and Snap Clutch were run from a Dell Latitude D830 laptop with 4GB of RAM and 1GB of video RAM. Output from the laptop was sent to a 20inch wide Samsung 206BW monitor set at a resolution of 1680x1050 px. A Tobii X120 eye tracker was positioned below the monitor. The monitor and eye tracker were placed on a rise-and-fall table. The table height and eye tracker positioning was adjusted to suit each participant. The Snap Clutch configuration used only allowed an individual to control locomotion or stop and look around:

— Gesture off top of screen: ‘Locomotion’ mode
— Gesture off bottom of screen: ‘Look around’ mode

Each individual was given a simple explanation of what the eye tracker is and how it was going to allow them to play a computer game by only looking at the screen. It was presented in a manner that was age-appropriate and as informally as possible so as to not apply any stress. Next, Snap Clutch was started within the game and it was explained how the interaction technique for walking around in the world works. The participant was then free to explore the world as they wished. The success of the sessions was determined by the participants ability to fulfill three objectives:

(1) Can the participant start and stop their avatar moving?
(2) Can the participant follow simple instructions? a. E.g. ‘Look at that house over there. Why don’t you go and have a closer look.’
(3) Can the participant move their avatar in all directions as required?

#### 4.3. Results and Observations

GPA-1: The individual arrived at both sessions in positive emotional health but not physically. That is, he appeared excited and motivated to take part but would quickly fatigue. Eye tracker positioning and calibration was difficult due to his divergent positioning but was achieved after approximately 10 minutes. Due to the position of his head restraint it was only possible to track his left eye. Once in the game the participant had difficulties starting and stopping his avatar. This was due to involuntary head movements with the eye tracker subsequently failing to track his eye. On both sessions the individual became tired after approximately 5 minutes of gameplay.

GPA-2: Prior to both sessions the individual had been responding well during class and was feeling physically well. Her carer commented that she was feeling a little anxious for the first session but was excited for the second. Eye tracker positioning and calibration was achieved after approximately 5 minutes. This was due to involuntary head movement. Upon entering the game the participant struggled to move her avatar in the world. This was due to head movement and the eye tracker being unable to locate her eyes. She was able to start and stop her avatar but movement was limited and there was not any real control. After several adjustments to the individuals wheel chair head support, it was necessary for us to offer hand-head support (using our hands to hold the individuals head in place to prevent it from falling forward). As she relaxed into the session, her head was slowly released until she was able to support herself. During the session her involuntary head movement reduced. In the second session, she was given navigational instructions for exploring points-of-interest: a farm; an old fort and a camp site. She was able to explore these areas with few problems.

GPA-3: This individual arrived at both sessions in good physical and emotional health. She appeared very relaxed and not particularly anxious or excited for the first session but seemed excited for the second session. As this individual often tilts her head back at 45 degrees, it was necessary to adjust her head support to raise her head enough so that the eye tracker could see her eyes. For the first session hand-head support was provided but this was not required for the second session. Calibration was completed easily and quickly. When in the game, she was able to start and stop avatar movement but not able to turn left and right. When realizing that it was herself that was making the avatar start and stop walking, she began to vocalize loudly in an excited manner. During both sessions, she experienced a lot of involuntary head movement causing the eye tracker to often lose track of her eyes.

GPA-4: Prior to both sessions this individual was in good physical and emotional health. He appeared excited to be taking part in the study and also playing a computer game like World of Warcraft for the first time. During both sessions, the participant found the calibration process to be quite difficult. Once the game began, he was able to start and stop his avatar moving but only able to turn left during both sessions.

Occasionally, it was necessary to provide hand-head support as his head would sometimes fall forward. The amount of involuntary head and arm movement reduced from the start of the session to the end. Toward the end of the second session he walked into a river in-game and stopped his avatar, leaving him floating in the water. At this point, the individual appeared to relax his body in his chair and made virtually no body movement, see Figure 11.

Fig. 11. Participant GPA-4 appeared to relax his body into his chair when he navigated his avatar into the water.

GPB-1 to GPB-8: All participants arrived in good physical and emotional health for both sessions. Only one individual (GPB-2) had difficulty with calibration due to his natural body positioning resulting in his head tilting back at an angle. Once the game began, he also had some difficulty in controlling his avatar. He was able to start and stop it moving but had little control over turning his avatar left and right. The remaining participants completed the calibration process with no major difficulties. When in the game, each one could manipulate and control their avatar as they wished. They were able to fulfill all three objectives with only minor difficulties. After each session participants were asked to comment on how they found using the interaction technique. Each one enjoyed the experience and they all thought that this method of avatar control was easier and more fun than a traditional gamepad. Possible improvements were also identified in regards to decreasing the speed of avatar rotation; increasing (and decreasing) the size of the central zone around the avatar and the ability to change avatar speed.

### 4.4. Discussion

This evaluation demonstrated to us that locomotion based activities in immersive, virtual, game-like environments is a motivating and engaging experience for both groups of participants. An interesting observation was how the environment in which the game is set (land, water, forest etc) and the state of the avatar (walking, running, swimming etc) has an effect of the individual's ability to relax. Anxiety and stress has an effect on avatar control and GPA-2 showed that as her confidence increased, her anxiety reduced, lowering her stress levels resulting in more successful avatar control. This resulted in greater relaxation and subsequently a reduction in involuntary head movement. The success greatly increased her motivation for the second session and she was able to follow navigational instructions. An increase in motivation and engagement was also observed with GPA-3 whom is normally quiet and non-verbal in class, sometimes appearing to lack motivation. It was observed in both sessions that she found the experience highly engaging and motivating despite having little control over her avatar.

These observations are summarized in Table III. The performance column is split into the three task objectives and is intended to be a simple evaluation on the individual's ability to control their avatar. There was no obvious learning effect observed in terms of performance over the two sessions but the participants did relax into the second session quicker than the first.

Table III. Summary of the performance objective results for Group A and Group B from the initial evaluation.

| | *Session* | *Objective 1* | *Objective 2* | *Objective 3* |
|---|---|---|---|---|
| *GPA-1* | *1* | Yes | No | No |
| | *2* | No | No | No |
| *GPA-2* | *1* | Yes | Yes | Yes |
| | *2* | Yes | Yes | Yes |
| *GPA-3* | *1* | Yes | No | No |
| | *2* | Yes | No | No |
| *GPA-4* | *1* | Yes | No | No |
| | *2* | Yes | No | No |
| *GPB-1* | *1* | Yes | Yes | Yes |
| | *2* | Yes | Yes | Yes |
| *GPB-2* | *1* | Yes | No | No |
| | *2* | Yes | No | No |
| *GPB-3... GPB-8* | *1* | Yes | Yes | Yes |
| | *2* | Yes | Yes | Yes |

*Notes*
*Objective 1: Start and stop avatar moving?*
*Objective 2: Follow navigation instructions?*
*Objective 3: Move avatar in all directions?*

## 5. DIAGNOSTIC TEST AND INTERACTION TECHNIQUE ADAPTATION

In order to adapt the interaction technique a diagnostic test was built as follows. The diagnostic test was written in C# using the Tobii SDK. The same hardware configuration was used as previously. All four individuals in Group A and all eight in Group B took part in this study. Twenty able-bodied (AB) participants also took part to provide a base-line measure to compare against. These participants will be referred to as Group C.

An explanation of how the tests worked was given and each individual was allowed to run through part of the test to facilitate understanding. The test then began as described in Table II. Upon completion, an adapted gaze interface for locomotion is proposed based on Algorithm 1.

### 5.1. Results and Analysis

Before discussing the individual results of those participants that require adaptation to their interaction technique we will first look at the results holistically. The median $R_{interaction}$ result for CP participants in Group A was 46%; for the MD participants was 81% and for AB participants, 86%, see Table IV and Figure 12. To achieve 100%

overall reliability is extremely difficult as this would mean that the participant performed an optimal test sequence every time. This includes being able to perform each sequence without any cognitive or action delay. As such, the base-line result from the AB participants of 86% is as expected. This suggests too that the reliability of the MD participants is close (5%) to the AB participants.

Table IV. Reliability ($R_{interaction}$) results from the diagnostic test for all three participant groups

| | $n$ | *Median* | *Mean* | $\sigma$ |
|---|---|---|---|---|
| Group A | 4 | 46% | 49% | 15% |
| Group B | 8 | 81% | 78% | 12% |
| Group C | 20 | 86% | 87% | 4% |

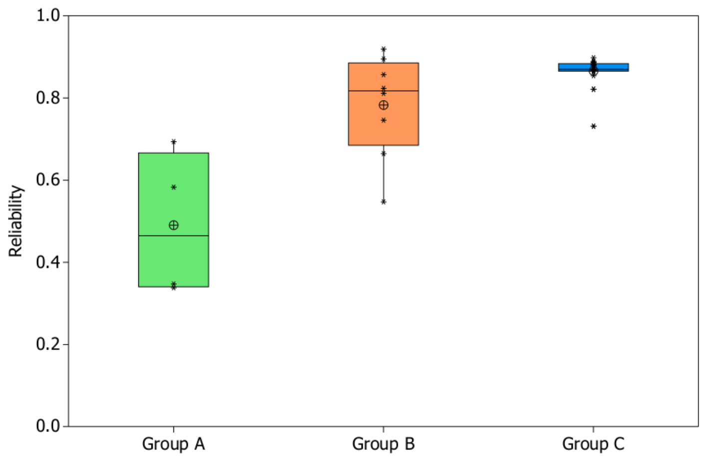


Fig. 12. Reliability by user group.

By examining the interquartile range of the AB group, it can be seen that there is a tight distribution with a high level of $R_{interaction}$ and only few outliers. Whereas, there is a wider distribution with the CP and MD groups. This suggests that there is a significant difference between participants in Group A and Group C and participants in Group A and Group B. Re-sampling[4] methods were used to verify the significance:

— CP participants and MD participants = 75 out of 10,000 ($p$ = *0.0075*) of the re-sampled statistics are more or less extreme than our observed result so we can reject the null hypothesis
— CP participants and AB participants = 0 out of 10,000 of the re-sampled statistics are more or less extreme than our observed result so we can reject the null hypothesis
— MD participants and AB participants = 61 out of 10,000 ($p$ = *0.0061*) of the re-sampled statistics are more or less extreme than our observed result so we can reject the null hypothesis

[4]Data was re-sampled into groups 10,000 times using median without replacement.

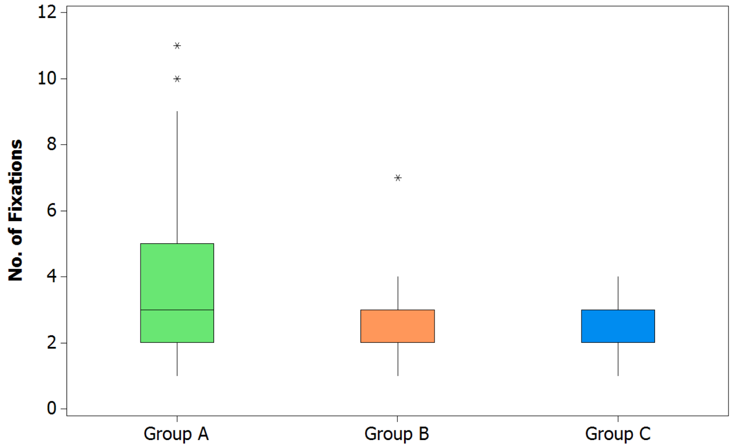


Fig. 13. Distribution of fixations made

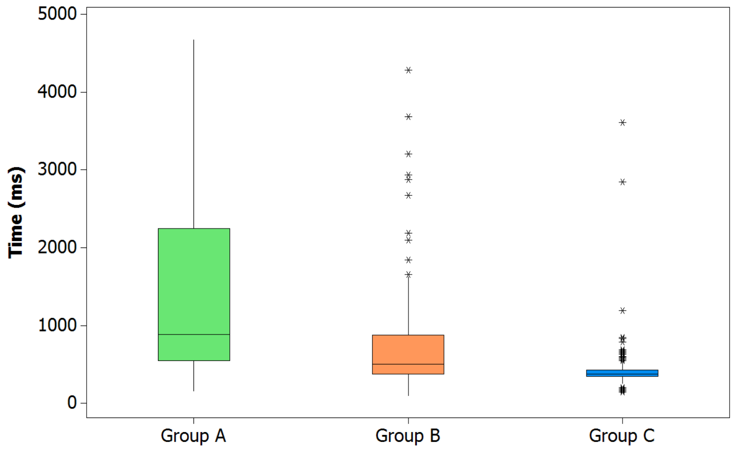


Fig. 14. Distribution of completion time

Figures 13, 14 and 15 show the distribution of fixations, completion time and length of time eyes are tracked for all three groups. In all of these attributes, the CP participants have a much wider distribution then the AB and MD participants. Examining only fixations (Figure 13), the AB and MD participants have similar distributions. The completion time (Figure 14) is much tighter for AB participants than MD participants. The difference in distribution between groups appears much greater when examining the length of time the eyes are not tracked (Figure 15) with AB participants being tracked almost all of the time.

There is a general assumption that those with severe physical disabilities often retain good control over their eye movement however, the observed data shows that this is not necessarily the case for those with CP.

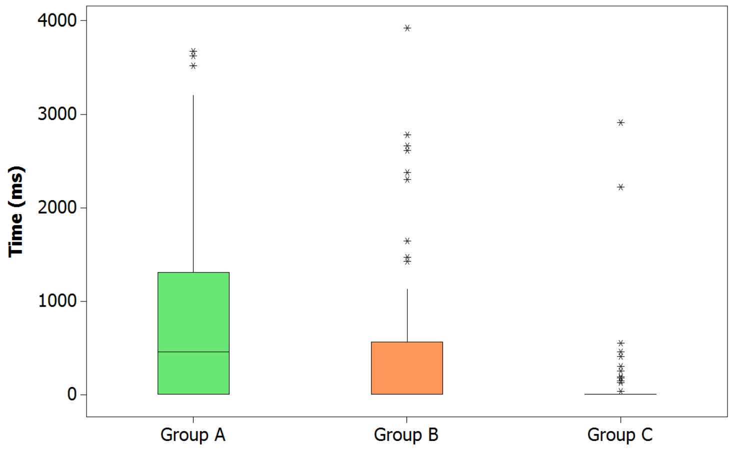


Fig. 15. Distribution of time eyes are not being tracked

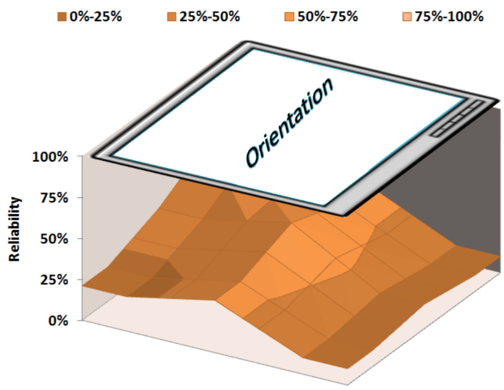


Fig. 16. This shows an example surface map indicating mean $R_{interaction}$ within the areas of the interaction technique elements. The overlaying image of the monitor indicates the orientation of the surface.

### 5.2. Adapted Interfaces and 3D Surface Map Visualization

A 3D surface map of the screen can be created using the mean $R_{interaction}$ values to each zone. This shows a visual representation of the parts of the screen a user can reliably gaze as required by this specific interaction technique. In the example found in Figure 16, an image of a monitor has been overlay to indicate the orientation of the surface map.

Following the diagnostic test, the controls for GPA-1 are assigned as Figure 17; for GPA-3 in Figure 18; and for GPA-4 in Figure 19. GPA-2 was able to control her avatar with the original interaction technique during the initial evaluation and was only 1% away from 70% with her mean $R_{interaction}$, therefore, it was considered not necessary to modify the technique. For reference, her surface is shown in Figure 20.

The surface maps produced indicate that there are certain areas of the screen that the participants had difficulty in accessing. More can be understood on the problems encountered by looking closer at the $R_{fixation}$ and $R_{eyes}$ values as well as the gaze data collected. GPA-1, scored low $R_{interaction}$ values for zone 8 (9%) and zone 9 (6%). The failure to complete the zone 8 test sequences can be explained to the individuals

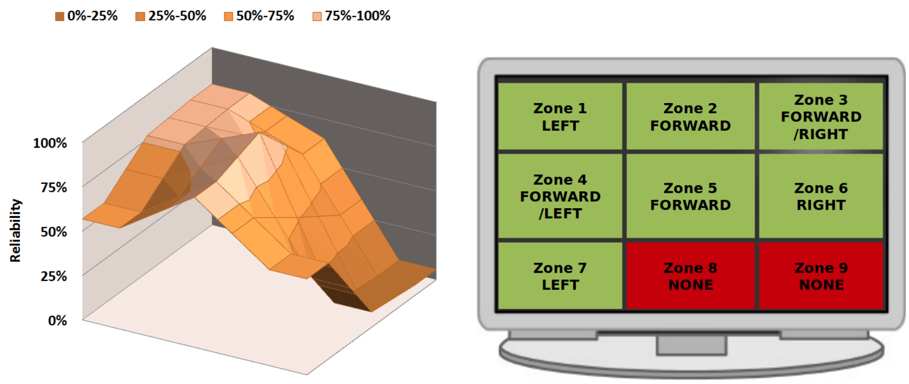


Fig. 17. Mean $R_{interaction}$ surface map for GPA-1 (left) and their adapted interaction technique (right).

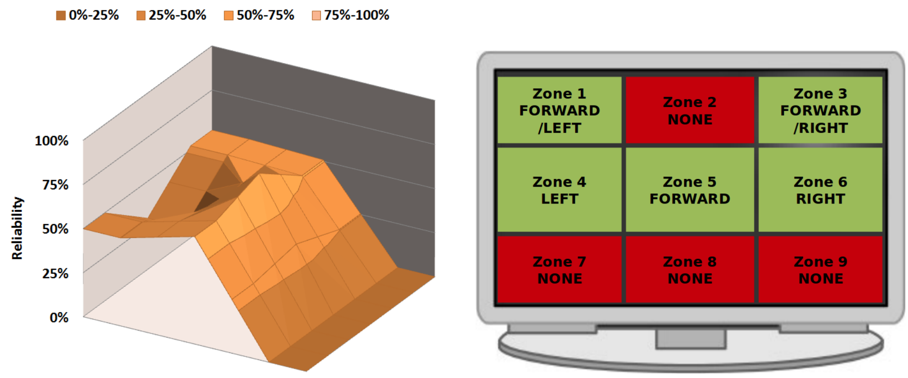


Fig. 18. Mean $R_{interaction}$ surface map for GPA-3 (left) and their adapted interaction technique (right).

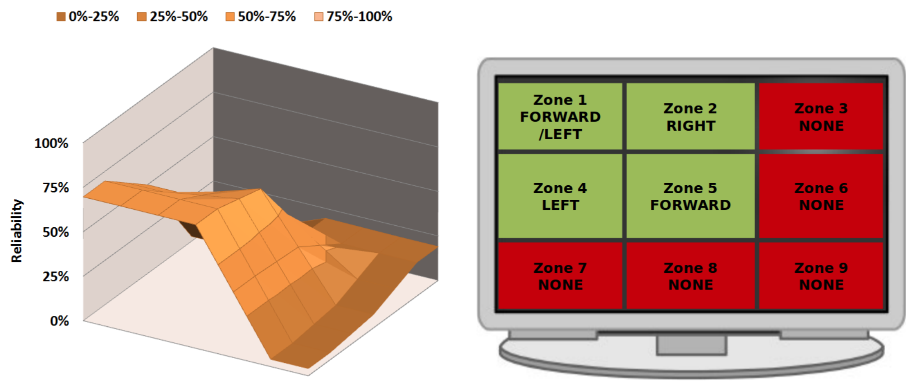


Fig. 19. Mean $R_{interaction}$ surface map for GPA-4 (left) and their adapted interaction technique (right).

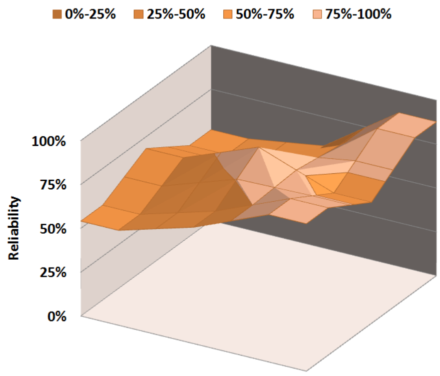


Fig. 20. Mean $R_{interaction}$ surface map for GPA-2

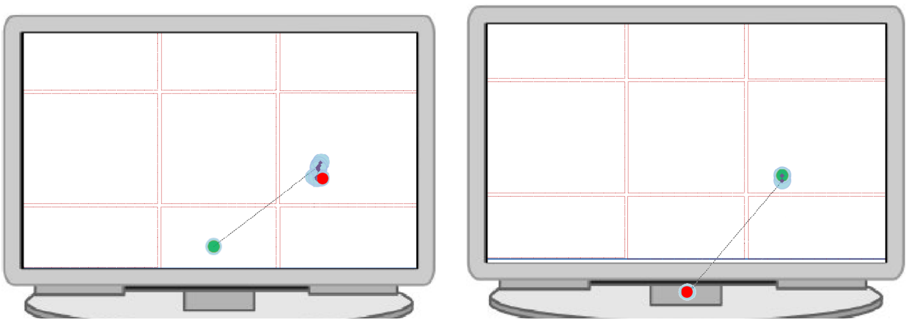

Fig. 21. GPA-1: Fixation eye gaze scan path for zone 9 attempt 1 (left) and attempt 2 (right). The green circle indicates the first fixation and the red circle the last fixation made.

eyes being lost for 98% of the time, in contrast to zone 9 in which they were lost for only 28% of the time. The large number of fixations made and the gaze scan paths, see Figure 21, suggest that the individual was attempting to fixate within zone 9 but the clustering indicates that the hardware influencing factors, such as eye tracker calibration or accuracy for this region of the screen may be poor. GPA-3, scored low $R_{interaction}$ values for zone 2 (29%), 7 (0%), 8 (0%) and 9 (0%). The failure to complete zone 2 can be explained by the individuals eyes being lost for 79% of the time. However for zones 7, 8 and 9 the gaze scan path suggest that the user was distracted and did not attempt to look within the correct zone, this was also noted in the study (see Figure 22). GPA-4, scored low $R_{interaction}$ values for zone 3 (3%), 6 (19%), 7 (4%), 8 (8%) and 9 (19%). The failure to complete zones 3, 7 and 8 can be explained by the individuals eyes being lost (respectively) for 94%, 92% and 85% of the time. One possible explanation for both users GPA-3 and GPA-4 struggling with the bottom three zones could be due to them having difficulties in holding their head up right without support.

## 6. VALIDATION OF THE ADAPTED INTERACTION TECHNIQUE

The purpose of this part of the study is to validate that the adapted technique is more suited to each individual than the original version.

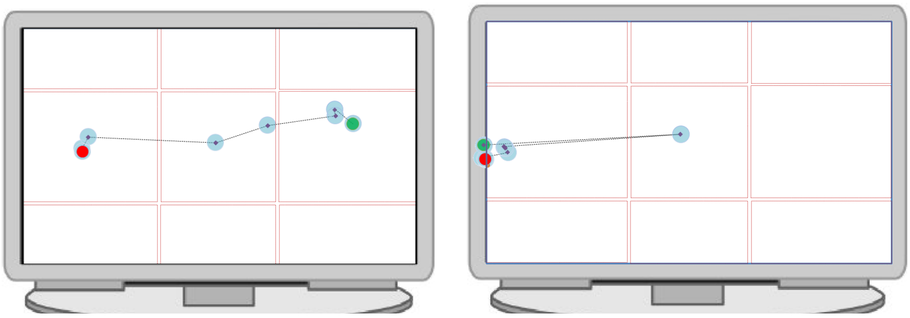

Fig. 22. GPA-3: Fixation eye gaze scan path for zone 7 attempt 1 (left) and zone 8 attempt 2 (right). The green circle indicates the first fixation and the red circle the last fixation made.

### 6.1. Procedure

The study was carried out with all participants from Group A and B. The twenty able-bodied (AB) participants from the diagnostic test assessment in the previous Section also took part to provide a base-line measure to compare against. These participants are referred to as Group C. This part of the study followed the diagnostic test after a short break of several minutes. Snap Clutch was started within the game and it was explained how the modified interaction technique for walking about in the world works. To aid the participants, the transparent overlays were visible at a low opacity level (15%). Each person was then free to walk around to familiarize themselves with the new configuration. The visible overlays were toggled on and off during this training phase until such time that each user was familiar. Once familiar, the participants were to complete an orienteering task within the game. In the sessions previous to this participants used the unmodified version of the interaction technique.

### 6.2. Orienteering Task

To make the task more structured than within the initial evaluation in Section 4 an orienteering challenge was devised. The challenge being that each participant follow a navigational instruction to some landmark within the game. Upon arriving at the landmark a new navigational instruction is given. A map that had a combination of open spaces, buildings, trees and water was chosen for the task and three way points were created. A typical GPS arrow based navigation system was used to direct to the three way points with descriptions given as follows:

(1) 1st Waypoint: 'Find the pond and swim to the center'
(2) 2nd Waypoint: 'Find the fort'
(3) 3rd Waypoint: 'Find the center of the village'

The way points were read to the participants as they appeared and they were free to take any route that they wished in order to reach them, see Figure 23. None of the participants had prior knowledge to the way points or their locations. The time to reach each waypoint and the routes taken were recorded. The same qualitative measure to that found in the initial evaluation in Section 4 was also taken to show any improvement in avatar control.

### 6.3. Analysis and Results - Group A

GPA-2 used the unmodified version of the interaction technique and was able to navigate to all three way points. Previously, she was able to navigate and control her character with little problem during the initial evaluation. GPA-1, GPA-3 and GPA-4 were


Fig. 23. Start position of the orienteering task. The arrow moves dynamically directing the participant to each waypoint.

all able to control their character in the world and follow navigation instructions to at least the first waypoint. This is a great improvement when comparing to the observations made during the initial study in which, all three individuals were only able to, at best, start and stop their character moving. However, despite having better control over their character they were unable to navigate to all way points. GPA-1 navigated to two way points with GPA-3 and GPA-4 only navigating to one. The difficulties observed during the initial evaluation were still present with the modified technique, that is for instance, the eye tracker losing their eyes; but there were now added difficulties that were previously experienced with able-bodied participants found in our previous work. For instance, on many occasions participants would turn too sharply and then try to re-correct their turning resulting in a turn-overshoot error. Also, individuals would get their character stuck against a fence or in a corner; this problem was particularly challenging to overcome with the modified technique as there was now no method for walking backward. However, it is positive that these difficulties are being experienced, as they demonstrate that the modified interaction technique has allowed them a comparable level of control to those using the unmodified technique. During navigation to the second waypoint, participant GPA-3 appeared to ignore the instruction given and explore the areas of the world that she wanted to. Although, this has an affect on the remainder of the data collected for this part of the task for this participant, it is seen as a positive outcome as she was able to independently go where she wanted. In terms of task completion time, GPA-2 was the only participant with a complete time: 128 seconds. Therefore, the time to reach only the first waypoint is the only time comparison between the four participants; of which GPA-2 was the quicker, Table V.

*6.3.1. Comparison with Initial Evaluation.* Using the same qualitative measure found in the initial evaluation, three individuals from Group A (CP) made improvements with their avatar control, Table VI.

Using the levels of character control found in Figure 7, GPA-1 has improved from level 1 to level 5; GPA-3 has improved from level 2 to level 5 and GPA-4 has improved from level 2 to level 4.

Table V. Summary of task completion times from the orienteering task and associated results from the diagnostic test (* indicates that the participant used the original technique and not an adapted version).

| | Diagnostic Test | Time to Reach Way points (s) | | | |
|---|---|---|---|---|---|
| *Participant* | $\overline{R_{reliability}}$ | *1st* | *2nd* | *3rd* | *Total* |
| GPA-1 | 58% | 86s | 65s | DNF | DNF |
| *GPA-2 | 69% | 43s | 52s | 33s | 128s |
| GPA-3 | 35% | 75s | DNF | DNF | DNF |
| GPA-4 | 34% | 99s | DNF | DNF | DNF |

Table VI. Observation evaluation comparison between the initial evaluation and the orienteering task using the control level scale in Figure 7. The original interaction technique was used in the initial evaluation and an adapted version used for the orienteering task.

| | Initial Evaluation (Original) | | | | Orienteering Task (Adapted) | | | |
|---|---|---|---|---|---|---|---|---|
| *Participant* | *Obj. 1* | *Obj. 2* | *Obj. 3* | *Control Level* | *Obj. 1* | *Obj. 2* | *Obj. 3* | *Control Level* |
| GPA-1 | Yes | No | No | 1 | Yes | Yes | Yes | 5 |
| GPA-2 | Yes | Yes | Yes | 6 | N/A | N/A | N/A | N/A |
| GPA-3 | Yes | No | No | 2 | Yes | Yes | Yes | 5 |
| GPA-4 | Yes | No | No | 2 | Yes | Yes | Yes | 4 |
| GPB-2 | Yes | No | No | 2 | Yes | Yes | Yes | 4 |
| GPB-3 | Yes | Yes | Yes | 5 | Yes | Yes | Yes | 4 |

*Notes*
*Obj. 1: Start and stop avatar moving?*
*Obj. 2: Follow navigation instructions?*
*Obj. 3: Move avatar in all directions?*

A possible limitation with this method of direct comparison between the two studies is that the tasks participants were asked to perform are different. In the initial evaluation, participants were free to explore the world as they wanted. As such, the above qualitative assessment was used to assess performance based upon observation of each participant. With this second study a more structured task was devised although participants were still given some freedom on how they would complete the task.

### 6.4. Further Analysis

The results of the diagnostic test suggest that six MD participants from Group B do not require any modification to the transparent overlay interaction technique. Only two MD participants in Group B achieved low $R_{interaction}$ requiring some modification; GPB-2 and GPB-3. All participants were able to reach the first waypoint with 89% of the AB participants and 88% of the MD participants reaching all three way points. One MD participant (GPB-2) was only able to reach the first waypoint. GPB-3 also used a modified version of the interaction technique but was able to reach all three way points, albeit within a much longer time than other participants in Group B. Using the levels of character control found in Figure 7, GPB-2 has improved from level 2 to level 4 and GPB-3 has reduced from level 5 to level 4, see Table VI. During this trial, there were difficulties in calibrating the eye tracker for participant GPB-3 and so a poor calibration was used. As a result, the diagnostic test made its adaptations with the interaction technique offering a reduced level of avatar control.

As all AB, MD and CP participants from the three groups were able to reach at least the first waypoint, it is more appropriate to use only the first waypoint time data for comparison of all groups. When comparing the results to Group A; the median time for Group B was 35 seconds, for Group C was 31 seconds, with Group A being 81 seconds, Table VII and Figure 24. The order of group task time performance is the same as

Table VII. Time for all three participant groups to reach the first waypoint.

| | $n$ | *Median* | *Mean* | $\sigma$ |
|---|---|---|---|---|
| Group A | 4 | 80.8 sec | 76 sec | 20.7 sec |
| Group B | 8 | 35.2 sec | 35.8 sec | 7.5 sec |
| Group C | 20 | 30.5 sec | 31 sec | 9.2 sec |

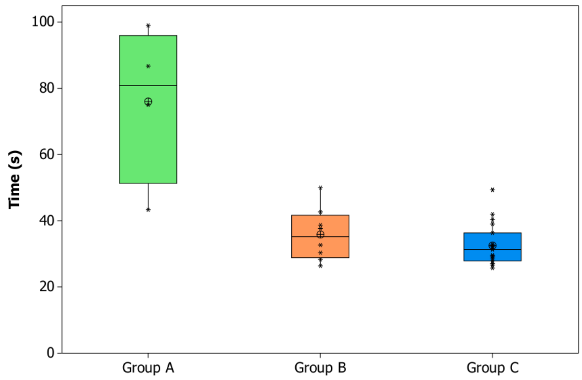


Fig. 24. Time to reach the first waypoint by user group

with the reliability results: CP Group A < MD Group B < AB Group C. However, the difference between the MD group and the AB group appears much smaller than with the reliability results, with the distribution for the MD group similar to that of AB but with the AB group being slightly tighter and with a quicker median time. There still remains a large difference when comparing both AB and MD groups with the CP group, with the CP group having a much wider distribution and a longer median task time. This suggests that there is a significant difference between CP participants in Group A and AB participants in Group C; and CP participants in Group A and MD participants in Group B. Re-sampling[5] methods were used to verify the significance.

— CP participants and MD participants = 94 out of 10,000 ($p$ = *0.0094*) of the re-sampled statistics are more or less extreme than our result so we can reject the null hypothesis
— CP participants and AB participants = 8 out of 10,000 ($p$ = *0.0008*) of the re-sampled statistics are more or less extreme than our result so we can reject the null hypothesis
— MD participants and AB participants = 1760 out of 10,000 ($p$ = *0.176*) of the re-sampled statistics are more or less extreme than our result so we cannot reject the null hypothesis

## 7. DISCUSSION AND FUTURE WORK

There are several perspectives from which the results from this work can be viewed. One of these is the usability of the technique by participants with different types of physical disability, and the extent of the variation between individuals, even within the same nominal category of disability. Another perspective is the extent to which performance can be improved by adapting the interaction technique to an individual person's abilities on the basis of the outcomes of a diagnostic test, and importantly

[5]Data was re-sampled into groups 10,000 times using median without replacement.

the prospects of doing so automatically in the future. Another perspective is the impact that being able to control an avatar independently has had on the self-esteem of some of the participants, and the implications for using immersive environments for educational purposes.

Considering first the differences between participants, the diagnostic evaluation highlighted the differences between the groups of CP, MD and AB participants. The overall pattern of results obtained from the test showed that the CP participants significantly performed less well than both the MD group and the AB group. The performance of the MD group was closer to that of the AB group. However in terms of the overall reliability metric (see Figure 12), the MD group performed significantly less well than the AB group. This demonstrates that in this context at least (gaze-based control of locomotion within immersive environments) that able-bodied participants cannot be considered to be representative of motor-impaired users. Consequently, the validity of experimental investigations that use able-bodied participants has to be called into question. The question of experimental validity and representation of users is discussed further by [Sears and Hanson 2011] and [Istance et al. 2012]. The results also show the considerable variation within groups described by the same nominal disability. In particular the CP participants in which there was a large distribution in the results even though they are all the same GMFCS and MACS level (GMFCS 5 and MACS 5). This too needs to be borne in mind when discussing the suitability of gaze and other interaction techniques for people with physical disabilities as though they belong to a homogeneous group. This work demonstrates that this is clearly not the case.

The main research question that we have addressed in this paper, is the extent to which it is possible to modify a general interaction technique based on the abilities of the user. This is directly analogous to the ability-based modification in the Supple system of Gajos et al., although the task in this investigation is quite different. We have developed a metric to express the reliability of controlling gaze in different regions of the screen. We have expressed control of avatar locomotion in immersive games in an order series of levels. We have devised an algorithmic approach to re-allocating gaze responsive zones used for different levels of control based on the reliability metric. Most of the MD participants and all of the AB did not need any individual modification to the interaction technique, while most of the CP participants did. The performance of all participants using the modified interface improved compared with the unmodified interface. There were however some other differences in the before and after tasks which may have influenced performance. Specifically, the task in the after condition was more directed toward specific goals than the before task.

The next question is how this configuration process could be done automatically. Two options present themselves. The diagnostic test could be run by a teacher, parent or helper on an occasional basis; saving the results would cause the interaction techniques to be modified automatically. The test itself can be improved considerably and animated characters on a colored background could replace the static blocks of color. It could also be turned into a 'Splat'[6] style target acquisition game. The other option is to continually monitor the gaze position of the user and together with other performance indicators make the interaction system attentive. This means maintaining a model of the current state of the user and adapting the interaction and the gameplay accordingly. If we pursue this option, it is possible to also include emotion monitoring. The question is whether the system can detect if the child is enjoying themselves, or is bored, frustrated, or tired. This could potentially generate automatic warnings that

[6]'Splat' style games are inspired by the traditional Whac-A-Mole games: http://en.wikipedia.org/wiki/Whac-A-Mole

can be sent to the parent or teacher, together with a choice for them to initiate some appropriate remedial action. Instead of just stopping the activity if the child appears to be disinterested, an automatic modification of the current activity could compensate for this.

All participants were able to use the locomotion interaction technique but three of the cerebral palsy and one of the muscular dystrophy participants struggled greatly, even after the interaction technique had been modified. The question is then why did they struggle? There are probably three main causes. It could be that the interaction technique was still not optimally configured to the abilities of the individual, and that a different configuration would have been better. It could also be that the choice of eye tracking equipment did not allow the participant's gaze to be tracked reliably over the whole screen. This means that the eye tracker used may not have been appropriate for the participant in question, particularly as some needed additional head support during the trials. We used a desk mounted, high quality eye tracker. Lightweight glasses-mounted eye trackers have recently been brought to market intended primarily for market research applications. However these systems are currently, prohibitively expensive. It may well be that these offer a much better eye tracking solution for people with posture and stability of posture issues. This will be the subject of future work. If the reliability measured by the diagnostic test can be improved, then the adaptation needed will be different and less. Finally the source of difficulty these participants had could be the consequence of their disability, rather than the equipment or the interaction technique. Steering a character continuously in an immersive world for someone not used to this level of independent action may have been both exciting and tiring.

We acknowledge as a limitation of this work that we don't know how these factors influenced the extent of the improvement in performance, although this undoubtedly existed. A better means of eye tracking may lead to a different set of outcomes. Current developments in near-to-eye displays may accelerate the time when lightweight reliable head mounted eye trackers become economically viable for users with physical disabilities. These systems are not economically attractive solutions at this time.

The final perspective is that of impact. Regardless of the difficulties experienced, all CP participants showed encouraging signs of engagement and motivation through emotive and cognitive expression. This engagement and enjoyment was also echoed in the responses from the MD participants. One individual (GPA-3) showed a level of emotive expression that her mother had never seen before. Her mother was present for half of the sessions and stated how she had never seen such expression in such a short space of time. A second (GPA-4) was able to relax his body completely by mirroring his avatar's in-game actions when he came to rest in some water. Following adaptation, these participants became more engaged as they were able to manipulate their avatars with much greater confidence and control. One individual (GPA-2), who did not require a modified technique, is now able to control and navigate her own powered wheelchair using a head switch. The school staff and her parents believe that this has been due to her feeling empowered by taking control of her own in-game avatar. In demonstrating these levels of cognition and interaction capabilities, two of the participants have now secured local authority funding for Tobii CEye communication devices. These two individuals are now using these on a daily basis within some of their classes. This augurs well for the role of immersive games in education for students with physical disabilities to realize their true academic potential. Enabling greater engagement in education by means of completing tasks and quests that have an educational purpose is an important area for future investigation. This is not simply to allow children to play games, important as that may be, but to enable children to undertake activities in a way that would not be possible in the real world because of their limited abilities.

A further limitation of the work we have reported is that it is restricted to control of locomotion. Completing education-related tasks and quests will require additional techniques to interact with in-world objects and to communicate with others in the same virtual space. It is not envisaged that gaze will be used for everything. Additional input devices and modern interaction techniques including gesture recognition may be used. In some cases, automating part of the control action by using artificial intelligence techniques found in commercial computer games will be the most appropriate way forward. This forms part of the future research program.

## 8. CONCLUSIONS

There are enormous possibilities for enabling young (and older) people with physical and cognitive disabilities to interact with immersive games through the use of modern interaction techniques. These may use eye tracking, gesture recognition and touch tablets as input, as well as more usual input devices. These devices may be used either singularly or in combination, according to the needs and abilities of the individual concerned. Computer games, and particularly immersive games, offer many opportunities for the education of students with special needs. They provide a means of interacting in a virtual space free from the limitations imposed by the persons disabilities. For example, a person with cerebral palsy may have no verbal communication, limited control of coordinated muscle movements and mobility issues that require a wheelchair. In an immersive games environment, the same person can move freely and communicate with others in the same virtual space. Multi-user online environments can reduce the social isolation that is often a problem during out-of-school hours for children with severe mobility problems. The environment can support activities that not only have a direct educational purpose, but activities that can enhance life skills in general and build self-esteem. An essential requirement is being able to match and adapt the interaction techniques to the needs and abilities of the individual concerned, and to the game-related activity. It is important that there is high-level automatic support for this adaptation, so that teaching staff are not expected to acquire a range of new technical skills in order for these benefits to be realized.

This research has shown that we can design interaction techniques that can be used successfully by people with severe levels of physical disability to interact with immersive games worlds. We have described a method of knowledge elicitation and task analysis that can be used to design general techniques. Moreover, we can adapt these techniques to suit an individual via a simple diagnostic performance test. This adaptation can be done automatically so that manual configuration or even minor adjustments are not needed.

There are a number of research questions to be addressed to fully realize the potential that these technical opportunities offer for special needs education. One is how best to use the adaptation process, either by occasionally running the test and storing the adapted technique, or by monitoring the performance of the user continually. There are further opportunities to incorporate the emotional state of the user and adjusting either the interaction technique or the games activities themselves as a consequence. Another question is the extent to which interaction is helped or hindered by the measuring device. Wearable devices, such as eye trackers mounted in glasses frames may enable a more reliable means of interaction over the whole display surface. These devices may also accommodate for different postures used within the interaction. A final question is how best to use immersive games for different educational purposes, for both specific curriculum-related tasks, as well as broader skill-building activities.

The work so far has met an enthusiastic reception from our colleagues who are education specialists. We thank them for their expertise and input and sharing our vision

of how modern games and interaction technologies can facilitate inclusive education in the future.

## ACKNOWLEDGMENTS

The authors would like to thank all of the staff, pupils and parents at Ash Field Academy in Leicester, UK. A special thank you to Jane Lynch and Helen Newton for their expertise, assistance and support with this project. Further, a thank you to the reviewers and editorial team for their valuable comments and feedback.

## REFERENCES

BATES, R., VICKERS, S., AND ISTANCE, H. SpringerLink, 2009. Gaze interaction with virtual on-line communities: Levelling the playing field for disabled users. *Universal Access in the Information Society 9,* 3, 261–272.

BIGHAM, J. P., KAMINSKY, R. S., LADNER, R. E., DANIELSSON, O. M., AND HEMPTON, G. L. 2006. Webinsight:: making web images accessible. In *Proceedings of the 8th international ACM SIGACCESS conference on Computers and accessibility*. Assets '06. ACM, New York, NY, USA, 181–188.

BURIGAT, S. AND CHITTARO, L. 2007. Navigation in 3d virtual environments: Effects of user experience and location-pointing navigation aids. *International Journal of Human Computer Studies 65,* 11, 945–958.

CARD, S., MORAN, T., AND NEWELL, A. 1983. *The psychology of human-computer interaction*. Hillsdale, N.J. : L. Erlbaum Associates.

CARTER, S., HURST, A., MANKOFF, J., AND LI, J. 2006. Dynamically adapting guis to diverse input devices. In *Proceedings of the 8th international ACM SIGACCESS conference on Computers and accessibility*. Assets '06. ACM, New York, NY, USA, 63–70.

DAWE, M. 2006. Desperately seeking simplicity: how young adults with cognitive disabilities and their families adopt assistive technologies. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '06. ACM, New York, NY, USA, 1143–1152.

DONEGAN, M., OOSTHUIZEN, L., BATES, R., DAUNYS, G., HANSEN, J., JOOS, M., MAJARANTA, P., AND SIGNORILE, I. 2005. D3.1 user requirements report with observations of difficulties users are experiencing. *Communication by Gaze Interaction: COGAIN, IST-2003-511598: Deliverable 3.1*.

DONEGAN, M., OOSTHUIZEN, L., BATES, R., ISTANCE, H., HOLMQVIST, E., AND LUNDLV, M. 2006. D3.3 report of user trials and usability studies. *Communication by Gaze Interaction: COGAIN, IST-2003-511598: Deliverable 3.3*.

DORR, M., BOHME, M., MARTINETZ, T., AND BARTH, E. 2007. Gaze beats mouse: A case study. In *Proceedings of the 3rd Conference on Communication by Gaze Interaction: COGAIN '07*.

FOLMER, E., LIU, F., AND ELLIS, B. 2011. Navigating a 3d avatar using a single switch. In *Proceedings of the 6th International Conference on Foundations of Digital Games*. FDG '11. ACM, New York, NY, USA, 154–160.

GAJOS, K. Z., WOBBROCK, J. O., AND WELD, D. S. 2008. Improving the performance of motor-impaired users with automatically-generated, ability-based interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '08. ACM, New York, NY, USA, 1257–1266.

HAND, C. 1997. A survey of 3d interaction techniques. *Computer Graphics Forum 16,* 5, 269–282.

ISO. 2000. *ISO 9241-9:2000. Ergonomic requirements for office work with visual display terminals (VDTs) – Part 9: Requirements for non-keyboard input devices*. International Organization for Standardization.

ISOKOSKI, P., JOOS, M., MARTIN, M., AND SPAKOV, O. 2009. Gaze controlled games. *Universal Access in the Information Society 8*, 323–337.

ISOKOSKI, P. AND MARTIN, B. 2006. Eye tracker input in first person shooter games. In *Proceedings of the 3rd Conference on Communication by Gaze Interaction: COGAIN '06*.

ISTANCE, H., BATES, R., HYRSKYKARI, A., AND VICKERS, S. 2008. Snap clutch, a moded approach to solving the midas touch problem. In *Proceedings of the 2008 symposium on Eye tracking research & applications*. ETRA '08. ACM, New York, NY, USA, 221–228.

ISTANCE, H., VICKERS, S., AND HYRSKYKARI, A. 2009. Gaze-based interaction with massively multiplayer on-line games. In *CHI '09 Extended Abstracts on Human Factors in Computing Systems*. CHI EA '09. ACM, New York, NY, USA, 4381–4386.

ISTANCE, H., VICKERS, S., AND HYRSKYKARI, A. 2012. The validity of using non-representative users in gaze communication research. In *Proceedings of the Symposium on Eye Tracking Research and Applications*. ETRA '12. ACM, New York, NY, USA, 233–236.

JACOB, R. J. K. 1990. What you look at is what you get: eye movement-based interaction techniques. In *Proceedings of the SIGCHI conference on Human factors in computing systems: CHI '90*. 11–18.

JOHANSEN, A. S., HANSEN, J. P., HANSEN, D. W., ITOH, K., AND MASHINO, S. 2003. Language technology in a predictive, restricted on-screen keyboard with dynamic layout for severely disabled people. In *Proceedings of the 2003 EACL Workshop on Language Modeling for Text Entry Methods: EACL '03*. 59–66.

JÖNSSON, E. 2005. If looks could kill - an evaluation of eye tracking in computer games. M.S. thesis, Royal Institute of Technology, Stockholm.

KUMAR, M., PAEPCKE, A., AND WINOGRAD, T. 2007. Eyepoint: practical pointing and selection using gaze and keyboard. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '07. ACM, New York, NY, USA, 421–430.

MAJARANTA, P., MACKENZIE, I. S., AULA, A., AND RAIHA, K. J. 2006. Effects of feedback and dwell time on eye typing speed and accuracy. *Universal Access in the Information Society 5,* 2, 199–208.

MAJARANTA, P. AND RÄIHÄ, K.-J. 2002. Twenty years of eye typing: systems and design issues. In *Proceedings of the 2002 symposium on Eye tracking research & applications*. ETRA '02. ACM, New York, NY, USA, 15–22.

MANKOFF, J., DEY, A., BATRA, U., AND MOORE, M. 2002. Web accessibility for low bandwidth input. In *Proceedings of the fifth international ACM conference on Assistive technologies*. Assets '02. ACM, New York, NY, USA, 17–24.

RANDOLPH, A. B. AND MOORE JACKSON, M. M. 2010. Assessing fit of nontraditional assistive technologies. *ACM Trans. Access. Comput. 2,* 4, 16:1–16:31.

SAN AGUSTIN, J., HANSEN, J. P., AND MATEO, J. 2008. Hands-free selection by combining gaze pointing and emg selection. In *Scandinavian Workshop on Applied Eye Tracking: SWAET '08*.

SEARS, A. AND HANSON, V. 2011. Representing users in accessibility research. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '11. ACM, New York, NY, USA, 2235–2238.

SKOVSGAARD, H., MATEO, J. C., AND FLACH, M. F. H. J. P. 2010. Small-target selection with gaze alone. In *2010 Symposium on Eye Tracking Research and Applications: ETRA '10*.

SMITH, J. D. AND GRAHAM, T. C. N. 2006. Use of eye movements for video game control. In *Proceedings of the 2006 ACM SIGCHI international conference on Advances in computer entertainment technology*. ACE '06. ACM, New York, NY, USA.

STAMPE, D. M. AND REINGOLD, E. M. 1995. Selection by looking: A novel computer interface and its application to psychological research. *Eye movement research: Mechanisms, processes and applications*, 467–478.

STERNBERG, R. 2003. *Cognitive Psychology* 3rd Ed. Thomson Wadsworth.

TAN, D. S., ROBERTSON, G. G., AND CZERWINSKI, M. 2001. Exploring 3d navigation: Combining speed-coupled flying with orbiting. In *Conference on human factors in computing systems; CHI 2001 anyone anywhere*, J. A. Jacko, Ed. ACM, 418–425.

TREWIN, S., LAFF, M., HANSON, V., AND CAVENDER, A. 2009. Exploring visual and motor accessibility in navigating a virtual world. *ACM Trans. Access. Comput. 2,* 2, 11:1–11:35.

VICKERS, S., ISTANCE, H., HYRSKYKARI, A., ALI, N., AND BATES, R. 2008. Keeping an eye on the game: Eye gaze interaction with massively multiplayer online games and virtual communities for motor impaired users. In *Proceedings of the 7th International Conference on Disability, Virtual Reality and Associated Technologies: ICDVRAT '08*. 159–166.

WARE, C. AND MIKAELIAN, H. H. 1987. An evaluation of an eye tracker as a device for computer input. In *Proceedings of the SIGCHI/GI Conference on Human Factors in Computing Systems and Graphics Interface*. CHI '87. ACM, New York, NY, USA, 183–188.

WOBBROCK, J. O., KANE, S. K., GAJOS, K. Z., HARADA, S., AND FROEHLICH, J. 2011. Ability-based design: Concept, principles and examples. *ACM Trans. Access. Comput. 3,* 3, 9:1–9:27.

YUAN, B., FOLMER, E., AND HARRIS, JR., F. C. 2011. Game accessibility: a survey. *Univers. Access Inf. Soc. 10,* 1, 81–100.

ZHAI, S., MORIMOTO, C., AND IHDE, S. 1999. Manual and gaze input cascaded (magic) pointing. In *The CHI is the limit, human factors in computing systems: CHI '99*. 246–253.

Received October 2012; revised April 2013; accepted June 2013