Master Thesis, Spring 2008
Supervisors: Kenneth Holmqvist & Christian Balkenius
Department of Cognitive Science
Lund University, Sweden.

# NeoVisus

## Gaze interaction Interface components

Martin Tall, m@martintall.com
Video demonstration at http://www.martintall.com/neovisus

***Abstract:***
*This thesis investigates suitable interaction methods for gaze driven computer interfaces. The lack of input devices requires interface components that are especially designed to be driven by gaze input. A set of reusable and configurable interface components were developed to support various interaction styles. The components were then used to build a prototype application containing a game, a photo viewer and a music player. An evaluation of the interface components as well as the prototype application was performed. The use of dynamically appearing target areas for saccadic selection was found to be a suitable interaction method for gaze driven interfaces. The interaction methods helps to alleviate the previously found stress associated with gaze driven interfaces (the midas touch problem). Overall the prototype application received a positive response from the evaluation participants with appreciation for being intuitive to use.*

Keywords: eye tracking, gaze interaction, HCI, user interface, midas touch, target area, saccade selection

## 1. Introduction

Since the introduction of graphical user interfaces for human computer interaction the main input device for the general population has consisted of the keyboard and pointing devices such as the mouse, trackballs, touch-pads, etc. These have evolved to support the common two dimensional interfaces and are used to select/manipulate objects, activate functions, and execute commands. Most users have spent an substantial amount of time to master these devices and there has been few real alternatives available. The computational and graphical processing capabilities of computers today poses few limitations on how interfaces can be visually represented, however the interfaces have not evolved much. Perhaps the cause of this stems from the fact that input devices have remained the same for more than two decades. During recent years novel approaches in interfacing techniques have been incorporated into consumer grade handheld devices with great success. These features touch sensitive displays and have controls capable of motion detection. However, the usage of eye trackers for gaze based interaction has not emerged and made the transfer from the academics to the general public in any wider sense. Several factors contribute to this. The technology is still not fully robust and stable in all environments and for all types of users. Additionally, the equipment comes at a high cost and gaze driven software is hard to come by.

Considering the general direction of technology development over the years it is feasible to imagine a continuous trend where technology will become faster and lighter while providing more capacity at a lower cost. It poses an opportunity for incorporating new technology to enhance the interaction between man and machine. In most cases the connection between where our gaze is directed and what we are interested in is obvious. Being able to track what a person is looking at gives away much of the persons intentions. The rich source of information is invaluable when reaching for novel interfaces and interaction techniques, an opportunity to good to ignore.

In order to develop a novel gaze based interaction interface one cannot rely on traditional Graphical User Interface (GUI) components since they are crafted for a mouse-keyboard based interaction. What is needed are components developed especially for gaze interaction. The over all goal for this thesis is to venture into new interaction methods and to design and implement these in reusable GUI components. My intention is to create an interaction style that relies more on the specific properties of the human visual system, in which movement comes at a more constant and lower cost compared to moving a physical modality. Humans are in general experts at at directing their eye movements by conscious attention. Due to the proximity to natural human behavior this type of interaction have the possibility to be very easy to learn. There is no new physical modality that the user has to map his or her intentions onto. Gaze interaction offers room for novel interaction techniques where objects appear or change when the user looks at them, without necessarily leading to an command execution. The knowledge of the gaze position creates an opportunity to use the display area in a more efficient way. As previous research have indicated that the error rate in selection (by gaze) is higher compared to a mechanical mouse due to the existing noise in the eye trackers (Ohno 1998, Hansen et al, 2003), By creating custom user interface components I hope to alleviated this problem.

## 1.1. Human–Computer Interaction

The term Direct Manipulation was first introduced by introduce by Ben Shneiderman in his keynote address at the NYU Symposium on User Interfaces (Shneiderman 1982) as an interaction style that can be traced back to the-mother-of-all-demos Sutherlands Sketchpad (Sutherland, 1963). The idea is that the objects of interest should be possible to manipulate directly as if they were real physical objects. It requires an interfaces that provides the user with input devices that maps the users intention provide immediate feedback in a suitable (graphical) representation. The Direct Manipulation style was further developed at the UC San Diego Cognitive Science department by Jim Hollan, Ed Hutchins and Donald Norman in 1985. An important aspect is the "sense of directness" between the users intentions and system. This translates into designing interfaces that allows the users to act directly on the graphical representations as if they were real world objects (Hollan et al, 1985) In general there are a set of guidelines or characteristics for Direct Manipulation interfaces. The interfaces should have a clear visibility of the object of interest. Actions upon these objects must be rapid, reversible and incremental. Additionally, complex command language syntax is replaced by graphical representations of objects that can be directly manipulated. The Windows operating system is one example where the user can control a mouse to directly manipulate and observe the results on the screen. However, the distance between the users intention, action upon the mechanical pointing device (mouse) and observation of the result (feedback) could be more direct if other modalities would be considered. Today a range of sensors and modalities exists which enables new interaction methods and styles. Most of the guidelines developed for the Direct Manipulation method are valid when concerning a wider range of interaction techniques and modalities.

### 1.1.1. Human Cognition

Central to developing of novel interaction methods is knowledge about human mind, our brain and its ability for cognition. Using only our hands for interaction with the mouse and keyboard in a silent two dimensional environment leaves a large part of our cognitive capabilities behind. To narrow the gap between man and machine the interface needs to support and understand natural human behavior. We use speech to communicate, hands and arms to manipulate objects in the multi dimensional world we perceive by our senses, i.e., vision, sound, touch, smell etc. Many of these senses are used in conjunction and provide feedback and support to another. Likewise, interfaces should ideally support multi-modal input. The quality of interaction can never be better than the input modality/sensors are at detecting our movements and thus our intentions. However, there are groups of users who are unable or unwilling to use the common input devices such as the mouse and keyboard. Using eye trackers for gaze based interaction is an alternative (or additional) form of input.

### 1.1.2. The human eye

The human eye enables stereoscopic depth vision which is highly flexible to various lightning conditions. It is the most important sense for building an situational awareness, navigating and interacting with the surrounding world. When viewed externally, the organ provides a rich source of information about ones awareness, intentions, and mental state.

The ability to consciously control the direction of our gaze is one of the most valuable feature of the human visual system. It enables us to perform rapid eye movements, know as *saccades*, which brings a specific region of our visual field into view. The high resolution, full color area of our vision covers about the size of a thumbnail on an arms-length distance and is know as the *foveal* region. To *fixate* any object outside of this region a new saccade has to be performed. Additionally, we have the ability to perform *smooth pursuit* of moving objects, it depends on the brains ability to calculate motion paths and then continuously corrected and adjust it (for example watching passing cars) without any conscious effort. The smooth pursuit serves as a good example of how deeper and more autonomous regions in the brain works in conjunction with higher cortical areas which enables conscious control of our gaze position.

The modulation of our attention and hence the direction of our gaze is usually divided into the *top–*

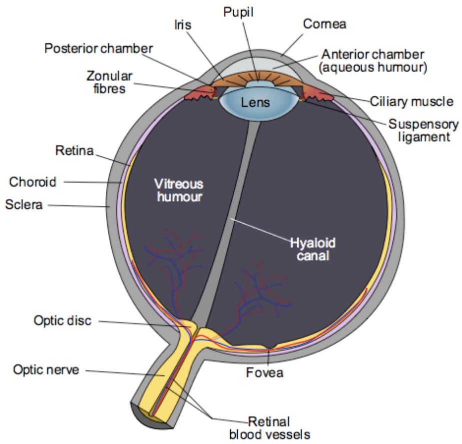


*Fig 1. Schematics of the human eye (Source: Wikimedia Commons)*

*Fig2. Remote based system (left) and high-speed system (right) (Images courtesy of SensoMotoric Instruments)*

*down* and *bottom–up* processes. When we consciously direct our gaze to observe an area the attention is modulated top–down by the cortical regions. When something suddenly appears in our visual field the more autonomous bottom–up processes have the ability to direct our attention to that area (flashing or moving objects, strong contrasting colors etc.) The cortical regions responsible for the top–down control have developed later in the evolution of the human brain have the ability to suppress bottom–up responses. As a result we have the ability to consciously choose to ignore objects. However, the pop out effect for highly contrasting and moving object is a strong modulator for capturing our attention. It is especially important when designing interfaces that are driven by the direction of gaze.

### 1.2. Tracking eye movements

Compared to the state of eye trackers just a few years back much progress has been seen in hardware and image processing algorithms. Several privately held companies now produce eye trackers and associated software although narrowly aiming for users with special needs mainly in the research, marketing analysis and assistive technology for the disabled. As in many other technology sectors things that were once were bulky and expensive high tech creations a few years later can be found in mass produced consumer products. Today it is possible to use of-the-shelf consumers technology to build a low cost eye tracker and several open source initiatives aims at making the technology more accessible (Böhme et al 2005, Corno & Garbo 2005, Hansen & Hammoud, 2007, Li & Parkhurst, 2006) However, the quality and robustness of these systems does not compare to the commercial alternatives just yet. A few years ago most systems used specialized hardware devices for image processing, today the processing power of an average computer is sufficient for the image analysis algorithms used to detects eye movements. Looking towards the horizon the high definition digital video revolution in the consumer market opens up for the further development of low cost eye trackers.

The new generation of remote based eye trackers illustrates how accessible the technology has become. They may not be as precise or fast as the laboratory grade equipment but a wide range of users can calibrate and the use them within seconds. Additionally, the remote systems keep track of the location of the face and allows a limited range of free head movement where as the high speed systems require participants to rest their chin on the apparatus to stabilize the eye image.

The move towards remote systems is an important step in making the technology accessible for the larger population. These systems are unobtrusive, the camera optics are invisible, hidden behind a plastic bezel. They work for approximately 90% of the population including those using contact lenses and some types of glasses. The quality of the eye tracking and gaze position estimation is sufficient for using to drive gaze based interfaces.

The remote based eye trackers usually consists of a camera capable of capturing images in the infra red light spectrum. The camera is typically placed underneath the monitor and is surrounded by a set of infrared light emitting diodes (IR LEDs). The camera, usually in the 1–2 Megapixel range, captures an image of the users face every 30–120 millisecond depending on system. In comparisation the upper range high speed laboratory solutions captures 1000–1500 Megapixel images per second. The obvious benefit of the remote system is that it allows a certain degree of free head movement while the high speed systems rely on a mounted position, typically placing the head on a chin rest. Undoubtedly, the high speed systems have a unmatched accuracy but does not pose a feasible solution for everyday gaze interaction. To achieve the high accuracy eye tracking both these types use industry grade CCD cameras since most consumer grade alternatives record with both lower resolution and lower frame rate (15–30 images per second). However, with the advancement of high-definition consumer appliances, cameras supporting 1920x1080 at 25 frames per second are becoming more accessible. Hypothetically, this could create a situation where mass produced eye tracking devices with adequate performance for gaze interaction could be achieve within a €500–1000 hardware price tag and even lower within a couple of years.

There are several steps in the process of tracking the human eye. Upon capturing the image of the face several image processing steps are carried out. The face is localized by its features such as mouth, nose, eyes etc. A region of interest is then created around the eyes which is the image that undergo further processing. Most eye trackers today rely on the corneal reflection method where infrared light is shined towards the face. The reflections the light creates on the eye is used to calculated the gaze vector in relation of the glint and the position of the pupil. The infrared light spectrum used prevents the light shined towards the user from being visually perceived. One obvious benefit using the remote systems is that the user does not have to wear any specific equipment or place his/her head in a mount.

Most eye trackers relying on the infrared light reflections are sensitive to large amounts of sun light since it has been shown to interfere the corneal reflections (Ruddarraju 2003, Kumar 2007) It causes a issues for using gaze interaction outdoors where large amounts sunlight masks the infrared light emitted from the eye trackers IR LEDs. The cameras in remote based systems cover a specific field of view in front of the eye tracker. The width and depth of the *tracking box* poses a limitation in flexibility in posture which the user can assume. Most remote systems are tolerant to a certain degree of head motion and can continuously track the position of the head and eyes.

Before using an eye tracker it has to be calibrated against the monitor. By displaying a set of points on the screen a correlation between the position of the pupil and the X and Y coordinates of the screen can be achieved. Using a higher number of calibration points gives a higher accuracy in the determination of the gaze position.

There are however a number of factors that over time affects the accuracy of the initial calibration. One issue is the changing properties of the eye in which the eye becomes drier after a prolonged viewing of computer monitors. This affects the corneal reflection (Bunquet et al. 1988, Qvarfordt 2004). Moreover, changes posture and distance from the camera over time reduces the quality of the initial calibration. These factors create an offset in the calibration which becomes most apparent at the edges on the computer screen (Jacob, 1991) The eye trackers ability to compensate for these factors is essential for the over all interaction experience.

A well composed source for more information eye tracking see the COGAIN D5.2 Report on New Approaches to Eye Tracking (2006).

## 2. Previous work

Using eye trackers to gather real time gaze data for the purpose of interacting with a computer interface poses several challenges and requires novel interaction techniques. The human visual system is ideally constructed for surveying and observing the environment while our finger, hands and limbs are used to manipulate objects. Zhai et al (2003) found that overloading the visual channel with motor commands is unnatural and thus undesirable. Designing an interface to be driven only by gaze therefore creates a challenging situation where issuing of a command has to be identified as something that differs compared to normal glancing to view the scene.

Within the domain this is commonly referred to as the "*Midas-touch*" problem (Jacobs et al., 1993) which stems from the old greek tale of the Midas which would turn everything touched into gold. Using gaze direction as the only means of input there is no method of performing activations (such as clicking a button) Somehow the system needs to be able to that distinguish between a user just looking around and gazing with the intent to perform an action. Several methods have been developed to work around this problem. A common solution is to apply *dwell-times* where the user fixates on a point for a prolonged period of time which is interpreted as an intention to activate or execute commands (Hansen 2003, Majaranta 2004). The duration of the dwell time is important and should be adjustable according to personal preference and experience.

The Dwell interaction style poses in general two problems. First, the user is stressed because everywhere he or she looks activation seems to occur, this causes a constant roaming of the eyes., which makes interaction experience stressful, prone to error and fatiguing over time. Second, the interaction is delayed since the user has to sit through the dwell time and fixate on a point for the specified period of time before the command is activated. The dwell-time can be adjusted and tuned but it still poses a delay. Thus, many projects have come to conclusion that dwell-time activation is only preferred when the user cannot use any other mean of activation (buttons, voice etc) Therefore, a majority of the systems utilizing gaze today are multi-modal, incorporating the mouse (Zhai et al, 1999), speech

(Miniotas, 2005) or keyboard hot-keys (Kumar, 2007) to be used in conjunction for perform activation, selection and interacting in general. The Quick Glance Selection Method (Ohno, 1998) introduced a two step method of activation where the user first fixates the command name and then performs a saccade to the target area which activates the command. The selection area is always apparent on the screen which enables experienced users to make a fixation into the target area directly.

The lack of pixel perfect precision created by both the physiological properties of the eyes and the rather noisy eye tracking data have lead several research projects into interaction techniques centered on zooming. As the user fixates on a region of the screen which is then zoomed into, making the objects inside that area larger and easier to discriminate at the point of selection. Examples of such techniques is the ZoomNavigator interface (Skovsgaard, 2008) and the EyePoint system (Kumar, 2007) which works by either automatic and continuous zooming or a two step dwell-based activation. The use of expanding targets have been investigated by Miniotas & Spakov (2004) which caused a 57% reduction in over all error rates but introduces a 10 percent increased activation time. Instead of the zooming in to the targets some projects have instead focused on dynamically resizing the canvas where the users gaze is directed. The EyeWindows (Fono, 2004) interface displays several video clips playing in parallel on the screen. Upon receiving a prolonged fixation the attended video becomes enlarged while surrounding video-tiles dynamically resize to accommodate the change while still playing in the peripheral visual field. Another take on the dynamically resizing canvas is the GazeSpace prototype (Laqua, 2007) which displays seven content panels laid out in a circle, upon receiving a fixation the chosen panel will move into the center of the screen and expand in size, allowing the user to view its full content. When the user looks outside or at another item the viewed item shrinks and returns to the edge to accommodate space for the new item.

Much of the work in the gaze interaction domain has been performed to assist a group of users whom cannot perform movements with their limbs or muscles in general. The gaze based interfaces provides these users with a tool for communication. Gaze interaction has successfully been implemented to improve the quality of life and the ability to communicate for users diagnosed with ALS, Cerebral Pares or similar paralyzing conditions. With gaze driven interfaces these users can go from communicating via blinking to building sentences with eye movements which can be articulated by the computer using text-to-speech synthesizer. The GazeTalk (Hansen, 2007), StarGazer (Skovsgaard, 2008) and Dasher (Ward, 2000) software is today used on a daily basis for text input and communication utilizing gaze alone. The rate of input is ranging from 6-15 words per minute with the GazeTalk and StarGaze while Dasher is capable of 25 words per minute. A normal chat room conversation typically goes at 40 WPM while speech easily reaches above 100 WPM (Hansen et al, 2004). The ongoing research within the Communication by Gaze Interaction (COGAIN) research network enables a more consistent effort and progress within this specific field of gaze interaction.

Additionally, there are some commercial platforms for gaze interaction with Tobii Technologies being one of the more prominent. Their integrated solution is used successfully on a daily basis providing a suit of gaze driven applications for web browsing, email, chat etc. Other companies active in the field consists of Alea Technologies, the EyeTech TM3, the Eye Response Erica System and LC Technologies Eyegaze system. These rely mainly on third party software applications such as the Viking suite, Grid2 and Dynavox which addresses disabled users in general and is not specifically designed for gaze driven interaction.

# 3. Materials and Method

## 3.1. Hardware

The eye tracking equipment consists of a SMI IViewX RED. The remote system is attached below the monitor. According to the manufacturer specifications it provides an accuracy of < 0.5° when the user is positioned within 50-70 cm from the system. With a 50Hz sampling rate it tracks the position of the head and eyes in a rectangular field (i.e., the *trackbox*) of 40 x 40 cm at the maximum 70 cm distance from the screen. The eye tracker provides the coordinates of the gaze position and outputs this as a UDP data stream.

The computer which was used for development and the evaluation experiments consisted of a Intel Core 2 Quad processor running at 2.40 GHz with 2GB RAM and a NVidia GeForce 8500 GT graphics card. The operating system was Windows XP version 2002 with the Service Pack 2. The SMI IView RED eye tracker was connected to the host computer via Firewire 400 and configured and calibrated using the IView X 2.00 build 14.

## 3.2. Software

The interface prototypes were built using a Microsoft based platform using Visual Studio 2008 and Expression Blend (preview 2). All applications were written in C# on the .Net 3.5 platform using the Windows Presentation Foundation. The gaze position data was collected by a custom developed client which connected via the UDP protocol to the SMI IView RED remote eye tracker. The data was broadcasted as a plain text string which was decomposed and used to update an object containing gaze data. The X&Y gaze coordinates was then redirected to replace mouse position by making low level calls to the operating system. The mouse pointer was made invisible. The eye tracker was configured to use filtering and stabilizing algorithms provided by the manufacturer (heuristics level 2) The filtering process introduces a delay ranging from 10-40 ms. During the pre-studies it was shown to provide a beneficial advantage for the interaction experience due to its ability to smoothened data by reducing jitter and noise. A simple custom filtering algorithm was initially developed but abandoned due to the superior performance of the proprietary SMI algorithm.

### 3.3. Motivation

The use of gaze data for interaction with computers is fundamentally different from more traditional computer interaction since there is no input modality (such as the mouse) to be acted upon. The requires specific interaction methods. Due to the physiology properties of the eyes a fixation covers an area of the screen that is larger than a traditional mouse pointer. Eye trackers will never be able to discriminate a gaze position for some of the smaller User Interface (U.I) components used in todays interfaces. Hence, most of the existing applications for mainstream operating systems such as Microsoft Windows to be ill suited for gaze interaction. Additionally, the gaze data provided by the eye tracker is noisy and full of jitter. The eyes are never still when we are fixating or staring at an object, even if we believe them to be (Yardbus, 1967) As a results the fixation point constantly moves. In most cases algorithms are used for smoothing and filtering out noise by fixation detection but they come at the price of latency. The fixation detection algorithms require a larger data sample, it is one of the reasons why most eye trackers use high speed cameras Additionally, most eye tracker creates a degree of added noise due to limitations in image processing algorithms. These factors has to be accounted for when designing gaze driven interfaces.

The commonly used dwell times creates a interaction style that is stressful to use since everywhere the user looks a command seems to be activated. This issue, known as the "*midas touch problem*", enforces a constant roaming of the eyes which interfaces applying only dwell time activation poorly address. For example, the variance in text length displayed on buttons leads to involuntary activation on items that contain longer and thus more time consuming text strings. I seek other means of interaction to remedy the midas touch problem and to create an overall intuitive interface based on highly configurable and reusable GUI components. By further developing the use of target areas (Ohno, 1998) and displaying these dynamically the midas touch problem can be alleviated. This results in components that will display options only when the user is looking at them, providing a direct interaction style based on the contextual position of the users gaze. To handle the noisy and jittery gaze data I intend to use target areas that are larger than the buttons and icons used, this enables the gaze to remain on the target.

### 3.4. Component design

All components are developed to be rely on nothing but gaze or a pointing device to be usable. When working with gaze as the only input, the midas touch problem as described earlier becomes a major issue. The behavior of the components has been shaped to reduce this as much as possible by introducing novel approaches. This includes a dynamically expanding areas which are activated by gaze and creates a layer on top of the other components when activated and "rolled out". Erroneous activation are reduced since the selection icons are not displayed on the interface in its original state, additionally when fixation a button or menu that action does not cause a command to be issued. This lets the user investigate buttons and their icons/labels without activating anything, thus reducing the effects of midas touch. When looking away from the component the activation icons are dynamically hidden from the interface which could reduce the error rate. However, one issue with displaying objects dynamically is that the bottom-up visual processes are attracted by motion. This effect stands in relation to how strong the distracting characteristics for the objects are. I have chosen to make these objects opaque to reduce this effect. The components are designed to be reusable and configurable. Features such as dwell times, icons, sizes etc. on each and every component in the interface can be adjusted for a more dynamic and adaptive interface.

### 3.5. Component: Dwell Button

The first component to be developed was the "GazeButton" which adds support for typical dwell-based activation. The GazeButton supports individual dwell-time which can be used to produce a interface where some functions require a longer fixation and some just a quick glance. It could create a more dynamic and responsive interaction. The component has a set of configurable parameters that specify its layout and operation. Feedback is provided in three stages, 1) to indicate that the button has focus (thin border), 2) to illustrate that the dwell process has started by a growing glow on the icon in the center of the button. It is hypothesized to "lure" the gaze to remain fixated in the middle of the button for the duration of the dwell period 3) Indication of a

*1. Initial state* *2. Gaze locked on component* *3. Dwell process final state*

*Fig. 3 Dwell Button. The visual indication of the dwell progress. Upon gaze entering the component a thin blue border appears around the button. The icon in the center (globe) is surrounded by a glowing white circle, increasing in size as the dwell progress progresses. When the dwell is completed a thicker colored border appears around the button.*

completed dwell process by border is emphasized and enlarged. The component suffers from the *midas touch problem* since a fixation starts the activation timer. Hence, it should be used when the choice does not cause a critical selection and where a selection is easily reversible (navigating between tabs in a web browser, viewing songs on albums etc.)

The use of the surrounding border is optional. It provides an indication of which object that is about to be activated and if the dwell has been completed. At the same it could attract unintentional saccades due to its susceptibility for bottom up cognitive processes.

### 3.6. Component: Binary Choice Button

This component resembles the traditional radio button component where an option can either be selected or deselected, hence the name binary choice of either on / off or yes /no. The component consists of a rectangle which upon fixation expands a second area which acts as a target area. When the user performs a short saccade to the icon in the target area the choice has been performed which is indicated by the changing background of the button. The option can be de-selected using the same method. The component was developed since the placement of text on dwell time activated icons causes involuntary activation (midas touch). The variance in length of the text of various buttons make the dwell time activation highly unstable. In other words, a button containing three words will more often be accidently activated compared to a button with on one word (unless they are configured to have a dwell time that is adjusted for the hypothetical time it takes to read the text) The activation time for the saccade icon in the target area can be configured with optional and individual dwell time.

Upon a fixation of the component the opaque layer containing the activation icon is rolled out. When the user performs a saccade to the activation icon the opacity is reduced and a growing white border around the icon indicates a dwell progress. When the dwell is completed the component changes background to indicate that the item has been selected.

The Binary Choice component has a target area for the activation icon that is larger than the icon. This reduces problems with jitter since the gaze position does not have to be exactly above the icon for the

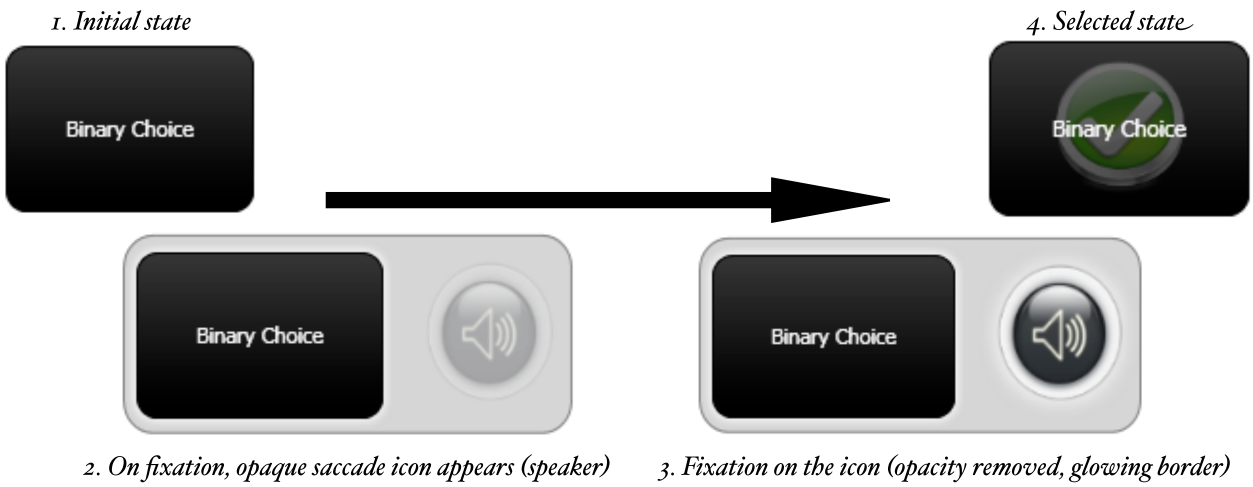


*Figure 4. The Binary Choice component. Upon gaze entering the component a opaque layer expands to the right, reveling the saccade icon (2, shown as a speaker). A growing white border indicates the activation process (3). The changed/selected state is then indicated by the background of the component (4). The speed of the rollout and the activation threshold is configurable.*

*Fig. 5. Binary Choice, the target area (right box) is larger than the actual saccade/selection icon. It raises the tolerance for jitter by reducing the effect of noise from eye tracker*

duration of the dwell. This is shown in the image to the right by the rectangle surrounding the icon.

### 3.7. Component: Radial Saccade Select

The idea behind the component is to make use of dynamic allocation of the display area as well as providing an novel interaction method for activation. Upon fixating the rectangle a animation process is initiated, during this period the icon in the center of the button is highlighted by a glowing border. Next a thin opaque ellipse starts to grow out from underneath the button and expands in size. Upon the completion of the expansion a set of icons laid out at the top, left, right and bottom are made visible. An activation can then be performed by making a short saccade any of the selection icons. Since the second stage icons are displayed within the *parafoveal* field of view and always positioned at the same location (top, bottom, left and right) the user can effortlessly make a saccade to the desired icon. The short but highly specific saccade could reduce the chances of accidently activating a command compared to a one step dwell activation. The activation time for both the expansion and the saccade dwell time can be customized. As the user becomes more aquatinted with the interface the activation times can be reduced or removed, proving a fast and adaptable activation.
To reduce the problem with noisy data and offsets the target area for the icon is expanded to an invisible rectangle on top of the icon. So even if the jittery gaze point is lands outside the icon the option will still be activated.

The number of options and the graphical representations used can be configured. For example only the left and right options could be used, leaving the top and left blank. The component is developed so that it will perform a callback to the originating application upon an activation. The software design supports quick drag-and-drop usage in future development projects, dramatically reducing future implementation times.

### 3.8. Component: Expanding Canvas

A common solution for battling the inaccuracy and jitter of eye trackers is to zoom into the component (Skovsgaard 2008, Kumar, 2007, Miniotas & Spakov 2004) It makes the target area larger and easier to discriminate. Additionally, since the gaze position gives away where your interest lies the display area could be used more dynamically. The Expanding Canvas component enlarged the specific item upon a fixation. This utilized the screen real estate in a more efficient way since items are dynamically enlarged based on what object the user is paying attention to. The magnification rate can be individually specified for each item. When the panel has been magnified an area containing dwell based icons are displayed underneath the main content panel. This solution is to remedy the problems associated with the midas touch problem by providing a secondary target area (as used by Ohno, 1998) which is to be fixate to issue the actual command. This reduces the often experience stress associated with gaze driven interfaces. The component works well with the standard *ListBox* item on the Microsoft .Net platform and can easily be bound to an external data source, for example, to displaying a list of books with an associated cover image.

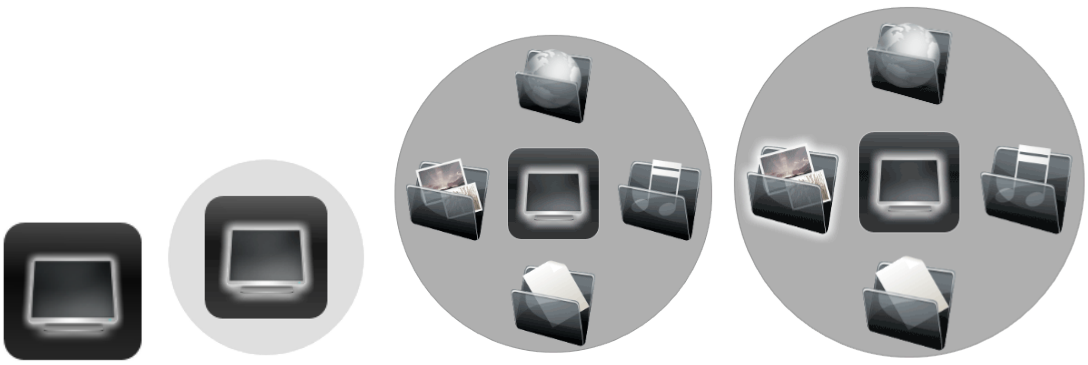

*Fig. 6 The Radial Saccade Pie Menu. Upon gaze entering the component a opaque ellipse expands from underneath the button. Four icons appears on the ellipse. A fixation starts the activation process which is indicated by a glowing border. Both the expansion time and activation time can be configured. The number of icons used is optional between 1-4.*

*Fig. 7 The GazeMemory game. The objective of the game is to find matching pairs of cards. By fixating a card its content is revealed. When a matching pair is found the cards are removed from the "table". The prototype is using the Dwell Button component.*

## 4. Prototype Applications

The general purpose of developing the prototypes are to investigate various interaction techniques utilizing gaze alone. Each prototype uses one of more of the custom developed component and aims at evaluating their performance in tasks that are real world centered, such as playing music or viewing pictures.

### 4.1. Prototype: Memory Game

The first prototype built is a gaze based version of the classic Memory card game. The goal of the game is to memorize the location of cards to find matching pairs. A total of 30 cards facing down are laid on the "table" in a grid 6x5. The cards are turned over by a dwell-time activation meaning that the user has to maintain a fixation on the card for more than 700 milliseconds. Upon glancing over the cards a thin blue border is indicating where the gaze is traced to be. When fixating on a specific card the dwell-timer is activated indicated by a red border. The globe symbol on top of the card will then be highlighted using a glowing white border which expands in size until the dwell is completed and the card symbol (flag) is displayed. The user then continues to the next card. If the two are matching both are removed from the "table", if not they are turned back over.

### 4.2. Prototype: Photo Viewer

The second prototype developed is using the dynamically resizing Expanding Canvas. The purpose of the prototype is to build a gaze based photo gallery. When the user fixates one of the photos the size of the canvas area expands providing a zoom effect. In this mode an additional menu bar is rolled out at the bottom of the panel. This menu houses a dwell icon that, on activation, brings the photo into full viewing mode. By looking outside of the photo or blinking the user can return back to the thumbnail mode. By enlarging a photo which the user is actively looking at

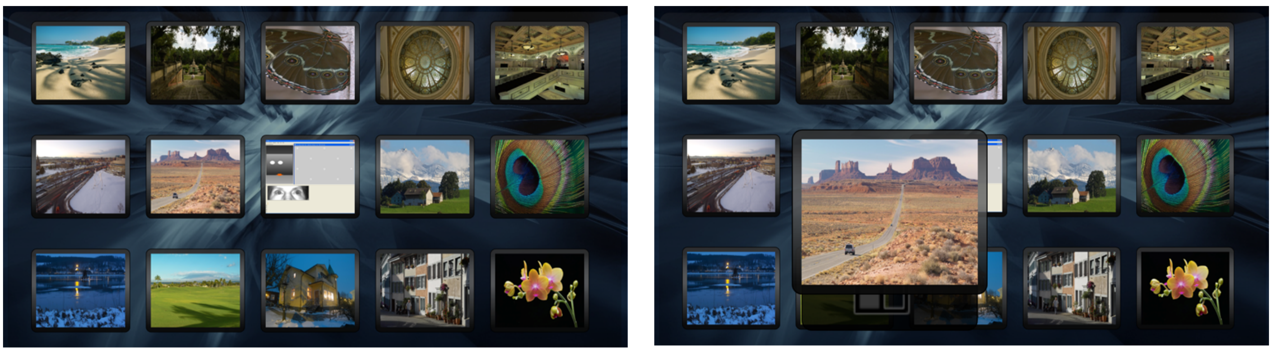

*Fig. 8 The Photo Viewer. By looking at one o the photos it becomes enlarged and reveals a menu bar (right image) at the bottom. By fixating on the expansion icon in the menu bar the photo is brought into a larger view. By looking outside or blinking the interface returns to its original state (left image)*


*Fig 9. The Media Player. Looking at the artists reveals an activation icon underneath the photo. By fixating it the artists albums and songs are displayed. The user can then build a playlist by selecting songs which is listed in the player control area in the upper right corner.*

the screen real-estate can be used in a more effective and dynamic way. Additionally, not only the content of interest are receiving more space but also are the associated options for each object which are revealed. This enables a interface which is clutter free and intuitive since to no static standard menu bars and buttons needs to be displayed.

### 4.3. Prototype: Music Player

The music player prototype utilizes all of the components to create a music library which can be navigated by gaze alone. The user will typically selected an artist, an album and then songs which are added to a playlist. By navigating through the library a playlist featuring one or more songs from multiple artists/albums can be constructed. The playlist can be navigated by the four options on the player control (play, stop, previous or next song) The progression of each song is visually indicated by a bar which fills up as the song plays. Additionally there is a volume controller that increases or decreases the volume by 25% for each selection. The component utilizes the API for the Windows Media Player controls in a multitasking environment which enables the user to continue with other tasks as the play list continues.

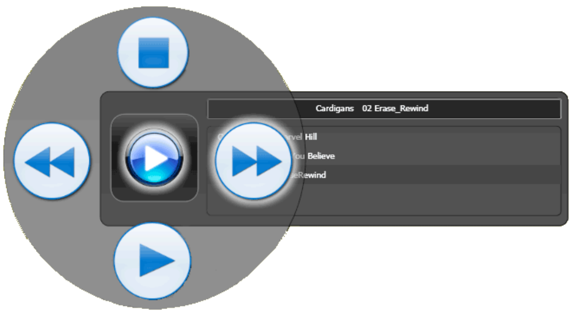


*Fig 10. The Radial Saccade Pie Menu is used to navigate the playlist.*

The Media Player relies on the and Radial Saccade Selection component for controlling the player functions in the prototype. Upon fixating the blue player button it expands on top of the interface and shows the play, stop, forward and backward controls which can be used to navigate the playlist. This is one example where the dwell time can be configured to a low value since the position of the activation icons is easy to learn. The Radial Saccade component is designed to make callbacks to the prototype application when a activation icon has been successfully dwell time activated. This will in its turn active a function to skip to the next song etc. The component remains expanded for the as long as the gaze remains within its borders.

The media player additionally uses the Binary Choice component for selecting songs for inclusion in the playlist. The switching between an artists album is performed by a dwell button which is configured with a non-existing dwell time, hence a quick glance at the cover will make the songs will appear below. When songs are added to the playlist feedback is provided in two ways. First, the background of the song item is changed then the title of the song is added to the playlist (top right).


*Fig 11. The album and song selection view. By using the Binary Choice components songs can be selected and deselected. This lets the user build a playlist (upper right) By looking at the album covers the list of songs instantly changes (no dwell) The button with a note icon is a Dwell Button that returns to the artist view.*

# 5. Evaluation

## 5.1. Measurements

When investigating how a interface performs both the users objective performance as well as their subjective rating is important. The system was assessed in terms of the interfaces effectiveness and efficiency in conjunction with user satisfaction.

- *Effectiveness* was measured between the three variants of the interface configurations in terms of measuring accuracy/error rate. This was defined by the number of actions needed to accomplish the task. This is to be combined with the the activation time per item. Custom developed statistical components triggered by the activation of any U.I components recorded this data.

- *Efficiency* in terms of task completion was measured both by timing and by subjective evaluation. The total time from giving the subjects their task to the completion of it was measured. Each task is to be performed three times. To measure the cognitive load on the subjects as questionnaire relying on the NASA Task Load Index (TLX) was used (see appendix 1). The subjects gave feedback on their experience on physical, mental and temporal effort combined with experienced levels of success and frustration. The questionnaire was presented on-screen between the first two experimental steps/tasks.

- *User satisfaction* was measured by handing out a form at the end requesting subjective opinions on the interface concerning the navigation, design, feedback, ease of use and stability. This was measured at the end of the experiment by two forms. The first is based on the Q.U.I.S interface evaluation (Chin et al, 1997), the second is based upon the IBM Psychometric Evaluation (Lewis, 1995) Question without relevance were removed, for example those concerning help messages (prototype contains none) The questions used can be found under Appendix 2 & 3.

The evaluation was divided into a sequence of tasks that were especially developed for the purpose. All the participants were exposed to the same flow of instructions, practice runs and task sets. The first two steps of the evaluation concerns the performance of the individual components. The configuration of the components in terms of both interaction speed (feedback) and activation threshold (dwell) was configured in three modes. The three configurations had animation times of slow (500 ms.), medium (300 ms.) and fast (10 ms.) which means virtually no delay and causes the selection area to appear as soon as the gaze entered the component. In the same manner the selection time (dwell) for each choice was configured with the same variables, hence the naming of the configurations are *long 500+500, medium 300+300* and *short 10+10*. The idea is to evaluate how the pace of visual feedback and activation speed affects the error rates as well as the total task completion time as a whole.

### 5.2. Procedure

The subjects were given a questionnaire concerning demographics, general computer experience and potential vision issues such as color/glasses etc. After giving the participants a short introduction to the eye tracking apparatus the calibration against the 19" monitor with 1280x1024 resolution was performed. The calibration process used was provided by the SMI IView application. It consisted of a nine calibration points that a randomly activated, subjects then press the space bar when they are fixating on the center of the point. A quick validation of the accuracy of the calibration was performed, if some of the points suffered from a noticeable offset the calibration process was restarted.

The first step in the experiment concerns evaluation of the Binary Choice component. A set of nine buttons were laid out in a grid. The task was to select/turn all the buttons on and then off. This was repeated three times. The task set was then repeated a total of three times, configured with different properties of *slow 500+500*, *medium 300+300* and *fast 10 +10* activation and selection (dwell) times. The data recorded contains the selection time-stamp for each component. Thus, this time includes an additional saccade from the previous component. All the items had to be selected in each set before the next set would be displayed. After performing the minimum of 81 selections required to complete the task a Task Load Index questionnaire was displayed on-screen to catch subject experience which was not spontaneously articulated.

The second step aims at evaluating the Radial Saccade Selection component. The task was to select a number between one and four from the menu. The number to be selected was displayed in a box located in the lower right corner of the screen. When a selection was performed the box would turn red. The subjects then were to perform a saccade back to the box which would then display the next number. The subjects were instructed to perform the switching and selection between the component and the number box as swiftly as possible. The data recorded consists of a timer which was activated upon gaze entering the component. The second time-stamp was issued when a number was selected from the component, additionally it would log each aborted selection (dwell not completed) and the total number of selection. An additional time-stamp was recorded upon gaze leaving the component. Each set contained 20 randomized selection tasks. The task set was repeated three times for a total of 60 selections per subject. Upon completion a second TLX questionnaire was displayed on-screen.

The third and last task in the the evaluation was to use the prototype evaluation. The Media Player, the Memory game and the Photo Browser were combined into a single interface which the participants were free to explore. While not producing any specific measurable data in terms of activation timing it provided a opportunity for observation and spontaneous questions / unstructured interviewing. A conservative approach on giving instructions on how to use the application was taken. The idea was to see how the participants would handle the components in a more real world oriented situation. The session was concluded with two printed standardized evaluation forms were handed out. These consisted of the IBM Psychometric Evaluation as well as the Q.U.I.S questionnaire.

### 5.3. Participants

A group of 19 people participated in the evaluation, seven female and twelve males, ages ranging from 14 to 55 with a mean age of 27. Seven of the participants wore glasses and one had contact lenses, which led to all of them having normal or corrected to normal vision. All participants had normal color vision. There was one case of *nystagmus* which was especially invited to investigate the capability of the eye tracker as well as the interface. Additionally, there was one case of *constant strabismus* causing the participants left eye to be misaligned. The two cases of nystagmus and strabismus caused issues with the calibration of the eye tracker. Additionally, one participant had glasses with anti-reflex coating which made it impossible to get a sufficient corneal reflection. Another participant had glasses with thin round edges which were mistaken for pupils by the eye tracker. These four cases were excluded from the experiment after several unsuccessful attempts to adjust the eye tracker.

The average computer experience was six on a ten point Likert scale, ranging from "none" to "professional IT". The frequency of usage had an average of 8.6 on a ten point scale ranging from "monthly" to "daily". A total of three persons had previous experience with eye tracking and gaze interaction.

## 6. Results

### 6.1. Binary Choice Component

The short temporal configuration (10+10) had a mean completion time per task set of 12 seconds with a standard deviation of 6 seconds compared to medium activation time (300+300) which have a mean time of 16 seconds with a standard deviation of 12 seconds. Finally the long activation time (500+500) produced a mean task completion time at 18 seconds with the standard deviation of 13 seconds.

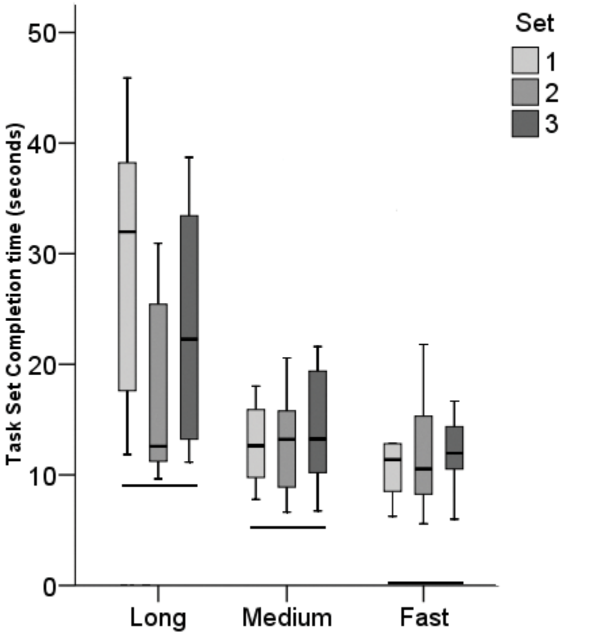


*Fig 12. Task completion times across the different configurations. The horizontal line indicate the theoretical time needed to accomplish the task. The order of sets was the same for everyone with no randomization.Thus, the learning effects for the Long category are clearly noticeable. Three sets with a completion time of more than one minute were excluded from the data due to misinterpretation of the task instructions.*

The short configuration had a mean activation time was 1 second while the medium provided a mean 1.2 seconds. The long configuration displayed activation times well above the 500 ms (animation) + 500 ms dwell time required to perform a selection, when displaying a mean individual activation time on one and a half second.

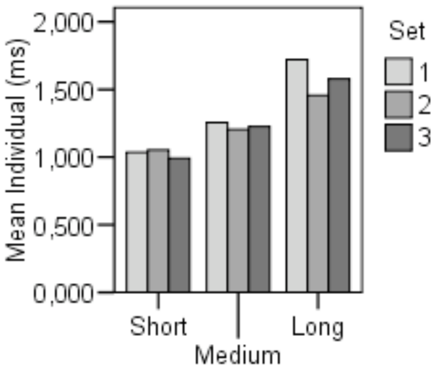


*Fig 13. Binary Choice. Mean individual activation time*

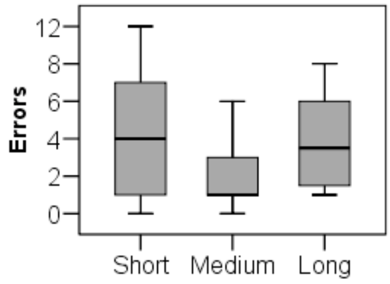


*Fig 13. Error rate for the different configurations. The short bar represents errors for the 10+10 millisecond configuration, medium equals 300+300 ms. and long 500+500 ms.*

Error rates are defined as the number of selections that exceed the nine needed to complete each task set. Two outlining task sets were excluded due to an abnormal error rate stemming from either a misinterpretation of the task or a high offset in the eye tracker gaze position. They contained more than twice number of selections needed to complete the task. The highest error rate was found to be for the short configuration which also had the highest variance. The average mean was *short* 4.03 (SD=3.7), *medium* 1.71 (SD=1.6) and *long* 3.9 (SD=2.6). The bars in figure 13 show the mean average error rate over all sets in the three configurations.

Participants subjective experience of the task set is demonstrated by the TLX questionnaire. The *physical demand* aspect have the widest span from none to very high, following close is the *effort.*

The correlation (Pearson) between subjects response on the two questions related to physical demand and effort was strong (0.88). The correlation between *physical demand* and *frustration* was strongly significant 0.97. The perceived *performance* is clearly modulated by the *frustration* (0.78) and *effort* (0.91).

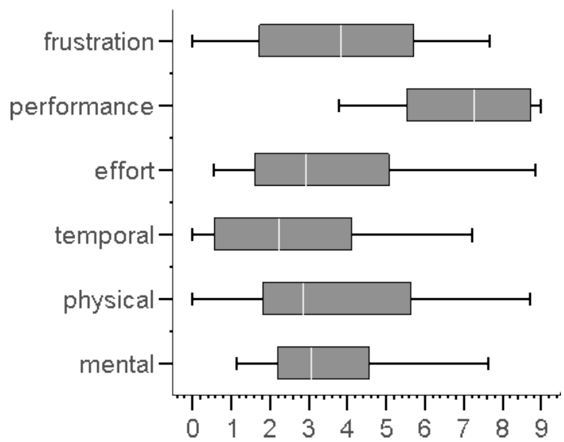


*Fig 14. Task load index for the Binary Choice component. Note: High value on performance equals a positive experience (where as others are aligned opposite, high=bad)*

### 6.2. Radial Saccade Selection Component

The measures of time from gaze entering the component until a selection has been performed. The combined average activation time for all configurations had a mean value of 0.77 seconds with a standard deviation of 0.46.

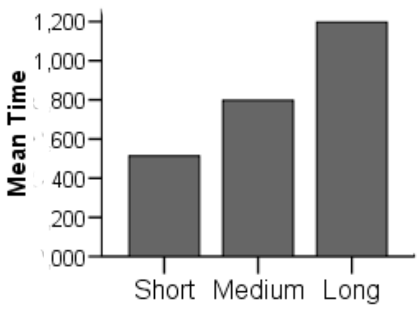


*Fig 15. Mean individual selection time using the Radial Saccade Pie Menu.*

Looking at the different configurations we see that the fast (10+10) had a of mean 0.513 seconds, median 0,406s. with a standard deviation of 0,315s.The medium configuration (300+300) delivers mean of 0.8 second (SD = 0.24) with a median of 0.7 s. (variance 0.06 s.) While the long (500+500) configuration of the component produced a mean of 1.2 seconds (SD = 0.3) with a median on 1.1 second. (variance = 0.11 s.)

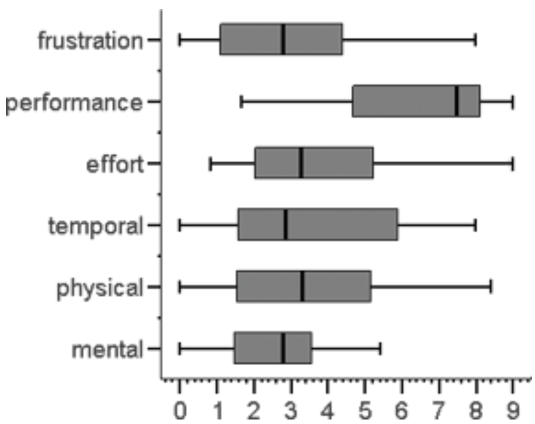


*Fig 16. Task Load Index for the Radial Saccade Pie Menu.*

The subjective experience of the task set testing the binary choice component is demonstrated by the Task Load Index questionnaire. The *physical demand* aspect have the widest span from none to very high, following close is the *effort*.

The correlation (Pearson) between subjects response on the two questions related to *physical demand* and *effort* was very strong (0.94). The *effort* correlates strongly with the *performance* (0.92)

However, the correlation between *physical demand* and *frustration* had a weaker correlation of 0.34 which differs from the Binary Choice component. The same can be seen for the correlation between *performance* and *frustration* (0.42).

### 6.3. Prototype – Q.U.I.S Results

The Q.U.I.S questionnaire was handed out after the participants had used the prototype application. The associated questions appear in order of the questionnaire.

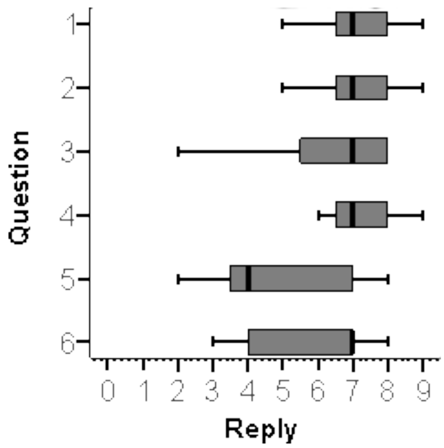


*Fig 16. Q.U.I.S - Overall reactions to the software.*

Questions:
1. Terrible – Wonderful
2. Inadequate power – Adequate power
3. Difficult – Easy
4. Dull – Stimulating
5. Frustrating – Satisfying
6. Rigid – Flexible

The correlation between the *difficult* and *frustration* shown to be non-significant (0.34)

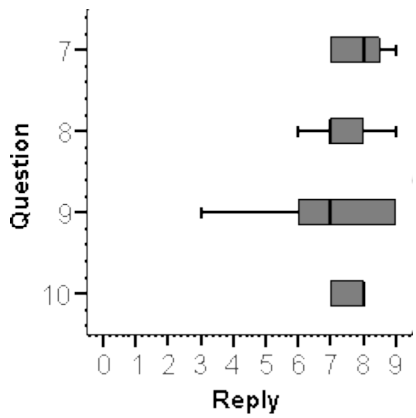


*Fig. 17. Q.U.I.S - Layout.*

7. Characters on the computer screen (hard to read/ easy to read)
8. Sequence of screens (confusing/very clear)
9. Highlighting on the screen simplifies task (not at all/very much)

10. Organization of information on screen (confusing/ very clear)

The low scores on question 9 concerning *highlighting* correlates significantly (0.80) with the *difficult* in the overall reactions.

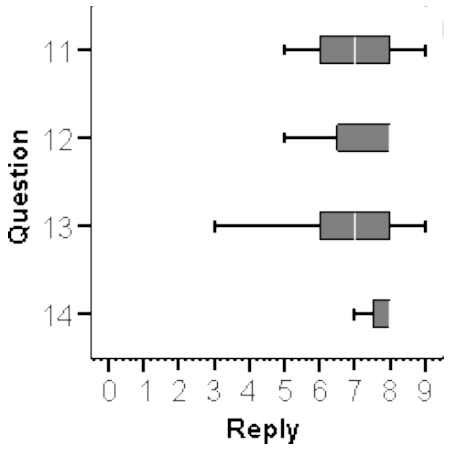


*Fig. 18. Q.U.I.S Learning*

11. Learning to operate the system (difficult/easy)
12. Tasks can be performed in a straight-forward manner. (never/always)
13. Exploring new features by trial and error (difficult/ easy)
14. Remembering navigation / use of commands (difficult/easy)

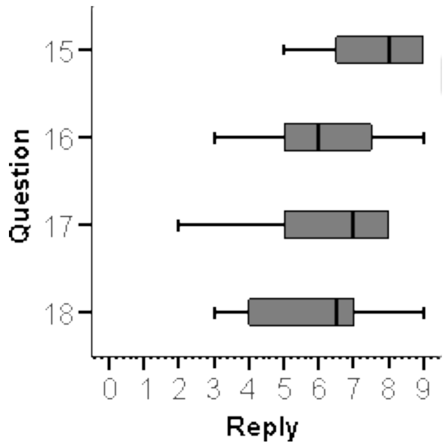


*Fig. 19. Q.U.I.S - Capabilities*

15. System speed (slow/fast enough)
16. Correcting your mistakes (difficult/easy)
17. System reliability (unreliable/reliable)
18. Experienced and inexperienced users needs are take into consideration (never/always)

Question 17 regarding the *reliability* of the system correlates (0.75) with the overall perception of the *ease of use* for the interface.

### 6.4. Prototype – IBM Psychometric Evaluation

The IBM Psychometric questionnaire contains eleven questions which are to be graded on a ten point Likert scale ranging from strongly disagree (0) to strongly agree (9).

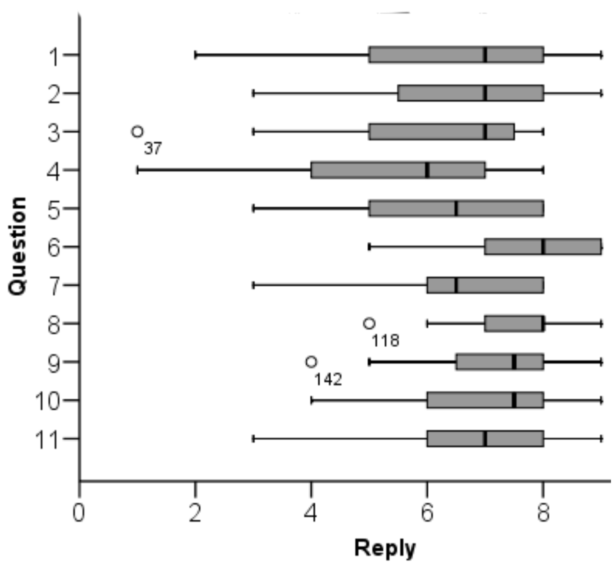


*Fig. 20. IBM Psychometric Evaluation Results.*

1. Overall, I am satisfied with how easy it is to use this system
2. It was simple to use this system.
3. I can effectively complete the tasks using this system.
4. I am able to complete my work quickly using this system.
5. I feel comfortable using this system.
6. It was easy to learn to use this system.
7. Whenever I make a mistake using the system, I recover easily and quickly.
8. The organization of information on the system screens is clear.
9. The interface of this system is pleasant.
10. I like using the interface of this system.
11. Overall, I am satisfied with how easy it is to use this system.

On the IBM Psychometric questionnaire results show that the overall satisfaction was highly correlated (0.97) with the ease of use (Q2) However, the overall satisfaction (Q1) was found to be uncorrelated (0.26) to the perceived swiftness of work completion (Q4)

## 7. Discussion

The majority of the participants found the interface to be stimulating and fun to use. All participants who were successfully calibrated and completed the two first steps in the evaluation were able to use the prototype application with none or very few instructions. The interface was perceived as clear, well structured and a majority were satisfied with how easy it was to use the system. A frequent comment was that the participants shortly forgot about the eye tracker and the specific eye movement they had to perform to navigate and reported that it felt like the system responded more directly to their conscious intention which also has been reported by users in other studies (Jacob et al. 1996) The most prominent source of dislike for the interface came from offsets in the calibration which consistently led to higher error rates, longer task completion times and lower ratings in the questionnaires. The accuracy of the gaze position is essential for a positive experience. Using gaze interaction with a constant offset is cumbersome. As the results from the questionnaires indicate, this factor is represented by the high variance in frustration levels. These indicators correlates with the physical load participants experienced and further with the overall satisfaction of the interface. An offset creates a situation where no activations occur even if the participants reported starting at the components. It is feasible to suggest that these issues are the cause for the variance in ease of use and the of appreciation for the highlight effect (dwell indicator)

In general these issues can be resolved in two ways. Either by making the calibration and image processing algorithms in the eye tracker more robust or by building the interface to accommodate these issues. Making the interface components larger would help to alleviate the problem since the offset would be less noticeable. However, it would make use of a great deal of the screen real estate. In cases of very noisy configurations the zooming interaction style seems promising. One thing observed during the evaluation is that the participants who had an offset in their calibration often tried to handle it by fixating "harder" onto the object. Subjects reported a sense of physical fatigue that leads to frustration within a short period of time.

Gaze interaction differs from traditional input device in terms of feedback. When using a physical modality the user knows if a button has been clicked due to tactile or auditory feedback. If the system does not respond instantly he might click again, if nothing happens something is probably wrong and trouble shooting can begin. With gaze interaction there is no such feedback. The interface relies on visual feedback to indicate the active item and dwell progress by a glowing border. In the cases of offset in the gaze position the lack of feedback made the users feel helpless since they were oblivious to what caused the malfunction. There was simply no response from the interface at all, and viewed from the side of the application everything is just running fine (hard one to troubleshoot). An interesting solution would be an algorithm capable of detecting a constant offset and perform a dynamic re-calibration based on the known position of the GUI components. An alternative would be to activate a calibration sequence if the number of aborted activations becomes to high.

Apart from the frustration associated with calibration offset and inaccuracy/jitter the use of only dwell time activation for the artist selection in the initial version of the Media Player prototype was found to be very frustrating. Participants reported false activations occurring while trying to get an overview of the available artists. Instead the application would switch mode to view albums from a specific artist. The reason is the usage of artists photo placed on the dwell buttons. Extending the dwell time is not the right solution. Dwell time is not suitable for more complex graphical representations or actions which actively performs in-between screens navigation. Having a target area to which users perform an active eye movement is a suitable method for gaze interaction. Displaying them dynamically is a further development on the work by Ohno (1998) It reduces the error rate since the selection is a two step process. On the downside it does require the time of an additional saccade. For the radial saccade menu the the short activation time selections were made on 0.5 seconds.

The use of instant selection (no dwell time) on albums was on the other hand perceived as less frustrating since it only toggled the list of songs without making the application go to another screen or mode. The activation on the albums was easily and effortlessly reversible, thus an activation did not cause a definite function execution.

The experiment employed no randomization in the order which the sets were run. It causes a learning effect seen in the variance between the sets. The practice run before each components and each the task set commenced did not cancel out this effect. Noteworthy is however the fast pace of adaptation, upon the fourth task set (after using the component 27 times) the performance stabilizes between the sets. The learning effect may caused an unfair judgement on a sets configured with the long timing. However, there are two reasons not for using these settings. First, it takes at a bare minimum at least one second to perform the activations. Second, the slow "roll out" animation of the expanding target area causes premature saccades to land outside the component which stops the activation. It is the major source of the high error rate for the slow configuration. The results indicate that a short or instant activation time for displaying the saccade icon is beneficial for performance. The dwell time for the activation icon it self should be around 300 ms. to reduce error rates. Having a virtually non-existing dwell on 10 ms caused involuntary activations when users performed a saccade to the next interface component. There is a trade off between fast activation and error rate. By using the medium (300 ms.) configurations instead of fast (10 ms.) or long (500 ms.) errors can be reduced to half.

In general, the Radial Saccade Menu performs better than the Binary Choice due to the increased size of the component as well as the target areas. The larger are provides a higher tolerance for eye tracking jitter. A larger target area also counterbalance the ballistic nature of eye movements.

The flickering jitter that eye trackers inherently produces creates data which needs to be carefully excluded. For example the item activation time as defined by gaze leaving the component suffers from the phenomena. A saccade across the component will register as both enter and exit producing unfeasible data which has been excluded. Additional data on aborted activations was recorded but due their ambiguous nature it was omitted. It is impossible to discriminate between conscious intended aborted activations and those created by eye tracking noise.

## 8. Future Work

The calibration process should ideally have a higher accessibility and better flexibility. This can be achieved by incorporating the functionality into the application so that a user could access and re-calibrate without restarting the application. A more intuitive process, for example following a bouncing ball, could help making the calibration more intuitive. The option for dynamically updating the calibration points while using the interface deserves to be explored. The advantage of knowing exactly at what position items are placed in the layout could be used to adjust and correct the offset dynamically. It does however requires the assumption that the user is fixating within the icons used.

Recently progress by Zhu and Ji (2007) introduce novel approaches for eye tracking. It minimizes the calibration process to only one time per individual. Furthermore, their approach provides a high tolerance for head movements. These are important steps for making gaze driven interaction a viable alternative.

The interface components have demonstrated that a dynamic usage of the screen real estate to display target areas with selection icons is a feasible solution. Further research into other types of interaction methods suitable for gaze interaction could help to establish a new form of Human-Computer Interaction that differs greatly from the two dimensional desktop analogy of folders, desktops and trash cans. Using gaze to drive Zooming User Interfaces (ZUI) such as Microsoft DeepZoom (SeaDragon) in combination with further developed gaze interaction components deserves further investigation. The combination of gaze to indicate objects of interest and then hand gestures (multitouch) to manipulate the data is an interesting concept.

When the core technology of eye tracking more accessible a rich set of interface components is one important area in making gaze interaction more widespread. Further updates to the NeoVisus library is likely to concern components for range selection, markers, text entry, communication and media functions, etc. The wide range for computer usage today requires flexible building blocks for rapid application development.

## 9. Acknowledgment

I wish to thank my supervisor Kenneth Holmqvist at the Humanities Laboratory for his feedback and guidance. Christian Balkenius at Lund University Cognitive Science department for organizing the masters program. Nils Holmberg for all the feedback and discussions. Martin Pötter at SMI for taking interest in my project. Johan Jernström for fruitful discussions. John Paulin Hansen along with Javier San Agustin and Henrik Skovsgaard at the IT University Copenhagen for inspiration. Björn Samuelsson for his excellent review. My mother and father for taking me out for much needed Sunday nature excursions and picnics.

## 10. References

Bunquet, C. Charliér J. R, Paris V. (1988) "Museum application of an eye tracker" Medical and Biological Engineering and Computing, #26, 277-281.

Böhme, M., Martinetz, T. and Barth, E. (2005). A Gaze-Estimation Algorithm for Single-Camera Remote Eye Tracking. In Proceedings of the BIP Workshop on Bioinspired Information Processing, Lübeck, Germany

Chin, J. P., Diehl, V. A, Norman, K. (Sept. 1987) Development of an instrument measuring user satisfaction of the human-computer interface, Proc. ACM CHI '88 (Washington, DC) 213-218. CS-TR-1926, CAR-TR-328

Corno, F. and Garbo, A. (2005). Multiple Low-cost Cameras for Effective Head and Gaze Tracking. 11th International Conference on Human-Computer Interaction, Las Vegas, USA, July 2005

Duchowski A (2007). Eye Tracking Methodology. Theory and Practice. Springer-Verlag London Ltd.

Duchowski A, Michael Ashmore, Garth Shoemaker (2004) Efficient Eye Pointing with a Fisheye Lens

Fono D, Roel Vertegaal: EyeWindows: evaluation of eye-controlled zooming windows for focus selection. CHI 2005: 151-160

Hansen, D.W and Hammoud, R.I. (2007). An improved likelihood model for eye tracking. Computer Vision and Image Understanding 106 (2007), 220-230

Hansen, J. P. et al. Command Without a Click: Dwell Time Typing by Mouse and Gaze Selections.
In Proceedings of Ninth IFIP TC13 International Conference on Human-Computer Interaction (Interact '03) (Zurich, Switzerland, Sept 1-5, 2003)

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 System, pp. 152-169, 1991

Hutchins, Edwin L.. James D. Hollan, Donald Norman (1985) Direct Manipulation Interfaces.

Kumar, M. et al. EyePoint: Practical Pointing and Selection Using Gaze and Keyboard.
In Proceedings of the SIGCHI conference on Human factors in computing systems (CHI '07) (San Jose, CA, 2007).

Lewis, J. R. (1995) IBM Computer Usability Satisfaction Questionnaires: Psychometric Evaluation and Instructions for Use. International Journal of Human-Computer Interaction, 7:1, 57-78.

Laqua, S. et al. GazeSpace: Eye Gaze Controlled Content Spaces
Published by the British Computer Society Volume 2 Proceedings of the 21st BCS HCI Group Conference HCI 2007, 3-7 September 2007, Lancaster University, UK

Li, D., and Parkhurst, D. J. (2006). Open-source software for real-time visible-spectrum eye tracking. Proceedings of the COGAIN Conference, pgs. 18 - 20.

Majaranta, P. et al. Effects of Feedback on Eye Typing with a Short Dwell Time.
In Proceedings of the Eye Tracking Research & Application Symposium (ETRA 2004) (San Antonio, TX, 2004)

Miniotas, D. et al. Extending the Limits for Gaze Pointing through the Use of Speech.
In Information Technology and Control, 2005, Vol. 34, No. 3, 225-230.

Miniotas D., Špakov O. and Scott MacKenzie (2004). "Eye Gaze Interaction with Expanding Targets"
Late breaking results paper presented at CHI 2004.

Ohno, T. (1998). "Features of Eye Gaze Interface for Selections Tasks". Proceedings of The Third Asia Pacific Computer Human Interaction – APCHI '98. IEEE Computer Society. 1– 6

Qvarfordt, P (2004) "Eyes on Multimodal Interaction" Dissertation No. 893. Department of Computer and Information Science, Linköpings Universitet.

Ruddarraju R, Haro A, Nagel K, Tran Q, Eassa I, Abowd G, Mynatt E (2003) " Perceptual User Interfaces using Vision-based EyeTracking". ICMI' 03, November, 2003, ACM1-58113-621-8/03/001

Shamaie A., Wu Hai and Alistair Sutherland, "Hand Gesture Recognition for Human Computer Interaction," ERCIM NEWS, European Research Consortium for Informatics and Mathematics, No. 46, July 2001. See http://www.ercim.org/publication/Ercim_News/enw46/shamaie.html

Shneiderman, Ben (1982): The Future of Interactive Systems and the Emergence of Direct Manipulation. In Behaviour and Information Technology, 1 (3) pp. 237-256

Shneiderman, Ben (1983): Direct manipulation: A step beyond programming languages. In IEEE Computer, 16 (8) pp. 57-69

Skovsgaard, Henrik (2007): Estimating acceptable noise-levels on gaze and mouse selection by zooming. Paper presented at: AISPC08 - Annual IEEE Student Paper Conference.

Skovsgaard, Henrik (2008): Noise Tolerant Selection by Gaze Controlled Pan and Zoom (TO BE PUBLISHED March 2008)

Sutherland, Ivan E. (1963): Sketchpad: A man-machine graphical communication system. In: Proceedings of the AFIPS Spring Joint Computer Conference 1963. pp. 329-346.

Sturman, D.J. (1992). Whole Hand Input. Ph.D. Thesis. [Available via anonymous ftp at media-lab.mit.edu, ./pub/sturman/WholeHandInput]. Cambridge, MA: Massachusetts Institute of Technology.

Tomfelde, C. M (2007) Dwell-Based Pointing in Applications of Human Computer Interaction. CSIRO ICT Centre, New South Wales, Australia. Published in INTERACT 2007, LNCS 4662, Part I, pp. 560-573, 2007.

Watson, Richard (1993). A Survey of Gesture Recognition Techniques. Technical Report TCD-CS-93-11, Department of Computer Science, University of Dublin, Trinity College, July 1993.

Yarbus, A. L. (1967) Eye Movements and Vision. New York: Plenum Press pp. 1967.

Zhai, S., C. Morimoto, and S. Ihde. (1999) Manual and Gaze Input Cascaded (MAGIC) Pointing. In Proceedings of CHI. Pittsburgh, Pennsylvania, USA: ACM Press. pp. 246-53, 1999.

Zhu, Z., Q. Ji (2007) "Novel Eye Gaze Tracking Techniques Under Natural Head Movements" IEEE Transactions on Biomedical Engineering. Vol. 54. No. 12. December 2007.

## APPENDIX I: TASK LOAD INDEX QUESTIONNAIRE

# Evaluation of Gaze Interaction Interface

## Cognitive Workload / NASA Task Load Index

1. MENTAL DEMAND
How much mental and perceptual activity was required (e.g., thinking, deciding, calculating, remembering, looking, searching, etc.)? Was the task easy or demanding, simple or complex, exacting or forgiving?

1 2 3 4 5 6 7 8 9 10
Low High

2. PHYSICAL DEMAND
How much physical activity was required (e.g., pushing, pulling, turning, controlling, activating, etc.)? Was the task easy or demanding, slow or brisk, slack or strenuous, restful or laborious?

1 2 3 4 5 6 7 8 9 10
Low High

3. TEMPORAL DEMAND
How much time pressure did you feel due to the rate or pace at which the tasks or task elements occurred? Was the pace slow and leisurely or rapid and frantic?

1 2 3 4 5 6 7 8 9 10
Low High

4. EFFORT
How hard did you have to work (mentally and physically) to accomplish your level of performance?

1 2 3 4 5 6 7 8 9 10
Low High

5. PERFORMANCE
How successful do you think you were in accomplishing the goals of the task set by the experimenter (or yourself)? How satisfied were you with your performance in accomplishing these goals?

1 2 3 4 5 6 7 8 9 10
Poor Good

6. FRUSTRATION LEVEL
How insecure, discouraged, irritated,stressed and annoyed versus secure, gratified, content, relaxed and complacent did you feel during the task?

1 2 3 4 5 6 7 8 9 10
Low High

## Appendix II: Q.U.I.S Interface evaluation Questionnaire

| **OVERALL REACTIONS TO THE SOFTWARE** | |
|---|---|
| *terrible* *wonderful*<br>0 1 2 3 4 5 6 7 8 9 | *inadequate power* *adequate power*<br>0 1 2 3 4 5 6 7 8 9 |
| *difficult* *easy*<br>0 1 2 3 4 5 6 7 8 9 | *dull* *stimulating*<br>0 1 2 3 4 5 6 7 8 9 |
| *frustrating* *satisfying*<br>0 1 2 3 4 5 6 7 8 9 | *rigid* *flexible*<br>0 1 2 3 4 5 6 7 8 9 |

| **SCREEN** | |
|---|---|
| *Characters on the computer screen*<br>hard to read easy to read<br>0 1 2 3 4 5 6 7 8 9 | *Sequence of screens*<br>confusing very clear<br>0 1 2 3 4 5 6 7 8 9 |
| *Highlighting on the screen simplifies task*<br>not at all very much<br>0 1 2 3 4 5 6 7 8 9 | *Organization of information on screen*<br>confusing very clear<br>0 1 2 3 4 5 6 7 8 9 |

| **• LEARNING** | |
|---|---|
| *Learning to operate the system*<br>difficult easy<br>0 1 2 3 4 5 6 7 8 9 | *Tasks can be performed in a straight-forward manner*<br>never always<br>0 1 2 3 4 5 6 7 8 9 |
| *Exploring new features by trial and error*<br>difficult easy<br>0 1 2 3 4 5 6 7 8 9 | *Remembering navigation / use of commands*<br>difficult easy<br>0 1 2 3 4 5 6 7 8 9 |

| **• SYSTEM CAPABILITIES** | |
|---|---|
| *System speed*<br>slow fast enough<br>0 1 2 3 4 5 6 7 8 9 | *Correcting your mistakes*<br>difficult easy<br>0 1 2 3 4 5 6 7 8 9 |
| *System reliability*<br>unreliable reliable<br>‘0 1 2 3 4 5 6 7 8 9 | *Experienced and inexperienced users' needs are taken into consideration*<br>never always<br>0 1 2 3 4 5 6 7 8 9 |

# APPENDIX II: Q.U.I.S INTERFACE EVALUATION QUESTIONNAIRE

| | IBM Computer Usability Satisfaction Questionnaire |
|---|---|
| **1** | Overall, I am satisfied with how easy it is to use this system |
| | strongly disagree strongly agree<br>0 1 2 3 4 5 6 7 8 9 |
| **2** | It was simple to use this system |
| | strongly disagree strongly agree<br>0 1 2 3 4 5 6 7 8 9 |
| **3** | I can effectively complete the tasks using this system |
| | strongly disagree strongly agree<br>0 1 2 3 4 5 6 7 8 9 |
| **4** | I am able to complete my work quickly using this system |
| | strongly disagree strongly agree<br>0 1 2 3 4 5 6 7 8 9 |
| **5** | I feel comfortable using this system |
| | strongly disagree strongly agree<br>0 1 2 3 4 5 6 7 8 9 |
| **6** | It was easy to learn to use this system |
| | strongly disagree strongly agree<br>0 1 2 3 4 5 6 7 8 9 |
| **7** | Whenever I make a mistake using the system, I recover easily and quickly |
| | strongly disagree strongly agree<br>0 1 2 3 4 5 6 7 8 9 |
| **8** | The organization of information on the system screens is clear |
| | strongly disagree strongly agree<br>0 1 2 3 4 5 6 7 8 9 |
| **9** | The interface of this system is pleasant |
| | strongly disagree strongly agree<br>0 1 2 3 4 5 6 7 8 9 |
| **10** | I like using the interface of this system |
| | strongly disagree strongly agree<br>0 1 2 3 4 5 6 7 8 9 |
| **11** | Overall, I am satisfied with how easy it is to use this system |
| | strongly disagree strongly agree<br>0 1 2 3 4 5 6 7 8 9 |