EyePoint: Practical Pointing and Selection Using Gaze and Keyboard

Manu Kumar, Andreas Paepcke, Terry Winograd
Stanford University, HCI Group
353 Serra Mall, Stanford, CA 94305-9035
{sneaker, paepcke, winograd}@cs.stanford.edu

ABSTRACT

We present a practical technique for pointing and selection using a combination of eye gaze and keyboard triggers. EyePoint uses a two-step progressive refinement process fluidly stitched together in a look-press-look-release action, which makes it possible to compensate for the accuracy limitations of the current state-of-the-art eye gaze trackers. While research in gaze-based pointing has traditionally focused on disabled users, EyePoint makes gaze-based pointing effective and simple enough for even able-bodied users to use for their everyday computing tasks. As the cost of eye gaze tracking devices decreases, it will become possible for such gaze-based techniques to be used as a viable alternative for users who choose not to use a mouse depending on their abilities, tasks and preferences.

Author Keywords

Pointing and Selection, Eye Pointing, Eye Tracking, Gaze-enhanced User Interface Design.

ACM Classification Keywords

H5.2. User Interfaces: Input devices and strategies.

INTRODUCTION

The keyboard and mouse have been the dominant forms of input on computer systems. Eye gaze tracking as a form of input was primarily developed for disabled users who are unable to make normal use of a keyboard and pointing device. However, with the increasing accuracy and decreasing cost of eye gaze tracking systems [1, 2, 9, 13] it will soon be practical for able-bodied users to use gaze as a form of input in addition to keyboard and mouse – provided the resulting interaction is an improvement over current techniques. The GUIDe (Gaze-enhanced User Interface Design) project [12] in the HCI Group at Stanford University explores how gaze information can be effectively used as an augmented input in addition to keyboard and mouse. In this paper we focus on using eye gaze for the purpose of pointing and selection.

We begin by analyzing common mouse actions and characterizing their uses. We then present some background on eye tracking and related work in the field of gaze-based pointing, followed by a description and discussion of the evolution of our design. Next, we present an evaluation of EyePoint compared to pointing with a standard mouse. Our studies show that users strongly preferred the experience of using gaze-based pointing over the mouse even though they had years of experience with the mouse. The performance of EyePoint is similar to the performance of a mouse, though with higher error rates. In the discussion section we present an analysis of the results and implications for future improvements.

MOTIVATION

Human beings look with their eyes and typically, when they want to point to something (either on the computer or in real life) they look before they point [11]. Therefore, using eye gaze as a way of pointing on a computer seems like a natural extension of our human abilities. However, to date, research has suggested that using eye gaze for any active control task is not a good idea. In his paper on MAGIC pointing [28] Zhai states that "*to load the visual perception channel with a motor control task seems fundamentally at odds with users' natural mental model in which the eye searches for and takes in information and the hand produces output that manipulates external objects. Other than for disabled users, who have no alternative, using eye gaze for practical pointing does not appear to be very promising.*"

Researchers have tried numerous approaches to eye based pointing [3, 10, 14, 16, 18, 20, 25, 26, 28]. However, its use outside of research circles has been limited to disabled users who are otherwise unable to use a keyboard and mouse. Disabled users are forced to tolerate customized interfaces, which provide large targets for gaze based pointing. This necessitates specialized applications for disabled users limiting their options. The lack of accuracy in eye-based pointing and the performance issues of using dwell-based activation have created a high enough threshold that able-bodied users have preferred to use the keyboard and mouse over gaze based pointing [11].

In his 1990 paper Jacob [10] states that: "*what is needed is appropriate interaction techniques that incorporate eye movements into the user-computer dialogue in a convenient and natural way.*" In a later paper in 2000, Sibert and Jacob [22] conclude that: "*Eye gaze interaction is a useful source of additional input and should be considered when designing interfaces in the future.*"

For our research we chose to investigate how gaze-based pointing can be made *simple*, *accurate* and *fast enough* to not only allow disabled users to use it for standard computing applications, but also make the threshold of use low enough that able-bodied users will actually prefer to use gaze-based pointing and selection.

POINTING AND SELECTION

We began our research by conducting a contextual inquiry into how able-bodied users use the mouse for pointing and selection in everyday computing tasks. We observed users while they worked and noted when they would use the keyboard and when they would use the mouse.

While there are large individual differences in how people interact with the computer, we noted there were some things in common. For instance, nearly everyone used the mouse rather than the keyboard to click on links while surfing the Web. Other tasks for which people used the mouse were: launching applications either from the desktop or the start menu, navigating through folders, minimizing, maximizing and closing applications, moving windows, positioning the cursor when editing text, opening context-sensitive menus and hovering over buttons/regions to activate tooltips.

These functions can be decomposed into their underlying mouse actions. The basic operations being performed to accomplish the above actions are the well-known single click, double click, right click, mouse-over, and click-and-drag. For a gaze-based pointing technique to be truly useful, it should support all of the above fundamental pointing operations.

It is important to note that our aim is not to replace or beat the mouse. Our intent is to design an effective gaze-based pointing technique, which can be a viable alternative for users who choose not to use a mouse depending on their abilities, tasks or preferences. Such a technique need not necessarily outperform the mouse but must perform well enough to merit consideration (such as other alternatives like the trackball, trackpad or trackpoint).

EYETRACKING BACKGROUND

Jacob et al. [11] and Ashmore et al. [3] provide a good summary of the issues for gaze-based pointing in their paper: eye tracker accuracy, sensor lag, fixation jitter and the "Midas Touch" problem [10].

Current state-of-the art eye trackers are accurate to about 0.5-1^o of visual angle, meaning accuracy is limited to 16-33 pixels when viewing a 17" display set to 1280x1024 (96dpi) resolution at a distance of 50cm [3, 24]. However, jitter in eye movement makes it difficult to maintain a steady gaze at a single point. Eye tracking data is therefore often noisy, which adds to the accuracy problem.

Mouse and keyboard actions are deliberate acts which do not require disambiguation. The eyes, however, are an always-on device [11] and it is therefore necessary to distinguish between involuntary or visual search/scanning eye movements and eye movements for performing actions such as pointing or selection. This effect is commonly referred to as the "Midas Touch" problem [10].

In addition, current eye trackers require calibration (though some require only a one-time calibration). The accuracy of the eye-tracking data usually deteriorates over time due to a drift effect caused by changes in eye characteristics over time [23]. Users' posture also changes over time as they begin to slouch or lean after some minutes of sitting. This results in the position/angle of their head changing. While most eye trackers claim to work with eye glasses, we have observed a noticeable deterioration in tracking ability when the lenses are extra thick or reflective.

RELATED WORK

Jacob [10] introduces gaze-based interaction techniques for *object selection*, *continuous attribute display*, *moving an object*, *eye-controlled scrolling text*, *menu commands* and *listener window*. This work laid the foundation for eye-based interaction techniques. It introduced key-based and dwell-based activation, gaze-based hot-spots, and gaze-based context-awareness for the first time. Issues of eye-tracker accuracy were overcome by having sufficiently large targets in custom applications.

Zhai et al. [28] presented the first gaze-enhanced pointing technique that used gaze as an augmented input. In MAGIC pointing, the cursor is automatically warped to the vicinity of the region in which the user is looking at. The MAGIC approach leverages Fitt's Law by reducing the distance that the cursor needs to travel. Though MAGIC uses gaze as an augmented input, pointing is still accomplished using the mouse.

Salvucci and Anderson [20] also use gaze as an augmented input in their work and emphasize that all normal input device functionality is maintained. Their system incorporates a probabilistic model of user behavior to overcome the issues of eye tracker accuracy and to assist in determining user intent. Furthermore, Salvucci and Anderson prefer the use of *gaze button* based activation as opposed to dwell-based activation. The probabilistic model relies on the use of semantic information provided by the underlying operating system or application and hence is not conducive to general use on commercially available operating systems and applications.

Figure 1. Using EyePoint - progressive refinement of target using look-press-look-release action. The user first looks at the desired target. Pressing and holding down a hotkey brings up a magnified view of the region the user was looking in. The user then looks again at the target in the magnified view and releases the hotkey to perform the mouse action.

Yamato et al. [26] also propose an augmented approach, in which gaze is used to position the cursor, but clicking is still performed using the mouse button. Their approach used automatic and manual adjustment modes. However, the paper claims that manual adjustment with the mouse was the only viable approach, rendering their technique similar to MAGIC, with no additional advantages.

Lankford [14] proposes a dwell-based technique for pointing and selection. The target provides visual feedback when the user's gaze is directed at it. The user has the ability to abort activation by looking away before the dwell period expires. Lankford also uses zooming to overcome eye tracker accuracy measures. The approach requires one dwell to activate the zoom (which always appears in the center of the screen) and an additional dwell to select the target region and bring up a palette with different mouse action options. A third dwell on the desired action is required to perform the action. This approach does implement all the standard mouse actions and while it is closest to our technique (described below), the number of discrete steps required to achieve a single selection and the delays due to dwell-based activation make it unappealing to able-bodied users. By contrast, our approach innovates on the interaction techniques to make the interaction fluid and simple for all users.

Follow-on work to MAGIC at IBM [7] proposes a technique that addresses the other dimension of Fitt's Law, namely target size. In this approach the region surrounding the target is expanded based on the user's gaze point to make it easier to acquire with the mouse. In another system [4], semantic information is used to predictively select the most likely target with error-correction and refinement done using cursor keys.

Ashmore and Duchowski et al. [3] present an approach using a fish-eye lens to magnify the region the user is looking at to facilitate gaze based target selection by making the target bigger. They compare approaches in which the fish-eye lens is either non-existent, slaved to the eye movements, or dynamically appearing. The use of a fish-eye lens for magnification is debatable. As stated in their paper, the visual distortion introduced by a fish-eye view is not only confusing to users but also creates an apparent motion of objects within the lens' field of view in a direction opposite to that of the lens' motion.

Fono and Vertegaal [8] also use eye input with key activation. They show that key activation was preferred by users over automatic activation.

Finally, Miniotas et al. [18] present a speech-augmented eye-gaze interaction technique in which target refinement after dwell based activation is performed by the user verbally announcing the color of the correct target. This again requires semantic information and creates an unnatural interaction in which the user is correcting selection errors using speech as a modality.

EYEPOINT

Our system, EyePoint, uses a two-step progressive refinement process fluidly stitched together in a look-press-look-release action. This makes it possible to compensate for the accuracy limitations of current state-of-the-art eye gaze trackers. Our approach allows users to achieve accurate pointing and selection without having to rely on a mouse.

EyePoint requires a one-time calibration. In our case, the calibration is performed using the APIs provided in the Software Development Kit for the Tobii 1750 Eye Tracker [24]. The calibration is saved for each user and re-calibration is only required in case there are extreme variations in lighting conditions or the user's position in front of the eye-tracker.

Once the system is calibrated, the user is ready to use EyePoint. The user simply looks at the desired point on the screen and presses a hotkey for the desired action - single click, double click, right click, mouse over, or start click-and-drag. EyePoint then brings up a magnified view of the region the user was looking at. The user looks at the target again in the magnified view and releases the hotkey. This results in the appropriate action being performed on the target (Figure 1).

To abort an action the user can look away or anywhere outside of the zoomed region and release the hotkey, or press the *Esc* key on the keyboard.

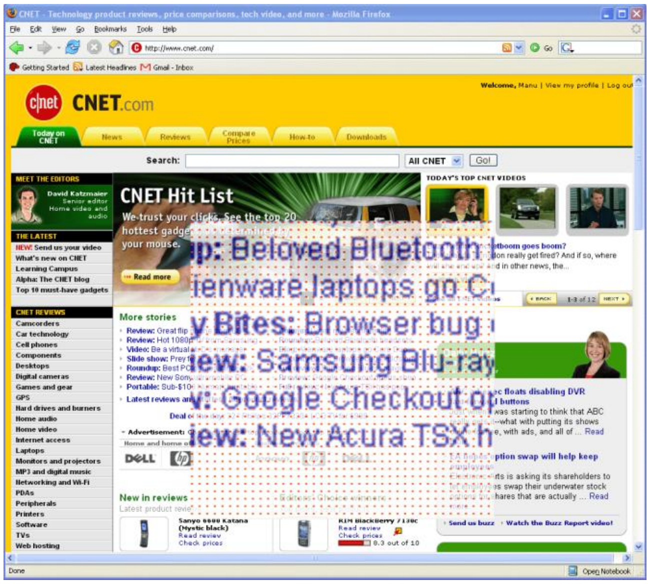


Figure 2. Focus points - a grid of orange dots overlaid on the magnified view helps users focus their gaze.

The region around the user's initial gaze point is presented in the magnified view with a grid of orange dots overlaid (Figure 2). These orange dots are called *focus points* and may aid in focusing the user's gaze at a point within the target. This mechanism helps with more fine-grained selections. Further detail on focus points is provided in the following section.

Single click, double click and right click actions are performed as soon as the user releases the key. Click and drag, however, is a two-step interaction. The user first selects the starting point for the click and drag with one hotkey and then the destination with another hotkey. While this does not provide the same interactive feedback as click-and-drag with a mouse, we preferred this approach over slaving movement to the user's eye-gaze, based on the design principles discussed below.

DESIGN EVOLUTION

In this section, we will describe the design process we followed for the development of EyePoint.

Design Principles

We agreed with Zhai [25] that overloading the visual channel for a motor control task is undesirable. We therefore resolved to push the envelope on the interaction design to determine if there was a way to use eye gaze for practical pointing *without* overloading the visual channel for motor control.

Another basic realization was from Fitt's law - that providing larger targets improves the speed and accuracy of pointing. Therefore, to use eye gaze for pointing it would be ideal if all the targets were large enough to not be affected by the accuracy limitations of eye trackers and the jitter inherent in eye gaze tracking. A similar rationale was adopted in [7].

As recognized in prior work [3, 8, 14, 17, 27] zooming and magnification help to increase accuracy in pointing and selection. We sought ways in which zooming and magnification could be used in a unobtrusive way that would seem natural to users and unlike [3], would not cause any visual distortion of their context.

As previously stated, our goal was to devise an interaction technique that would be universally applicable – for disabled users as well as able-bodied users.

We concluded that it is important to a) avoid slaving any of the interaction directly to eye movements (i.e. not overload the visual channel for pointing), b) use zooming/ magnification in order to overcome eye-tracker accuracy issues c) use a fixation detection and smoothing algorithm in order to reduce tracking jitter and d) provide a fluid activation mechanism that is fast enough to make it appealing for able-bodied users and simple enough for disabled users.

Initial Prototype

Our resulting design used a two-step progressive refinement of the target. The eye tracker constantly tracks the user's eye- movements[1]. A modified version of Salvucci's Dispersion Threshold Identification fixation detection algorithm [21] is used along with our own smoothing algorithm to help filter the gaze data. When the user presses and holds one of four action specific hotkeys on the keyboard, the system uses the key press as a trigger to perform a screen capture in a *confidence interval* around the user's current eye-gaze. The default settings use a confidence interval of 120 pixels square (60 pixels in all four directions from the estimated gaze point). The system then applies a *magnification factor* (default 4x) to the captured region of the screen. The resulting image is shown to the user at a location centered at the previously estimated gaze point.

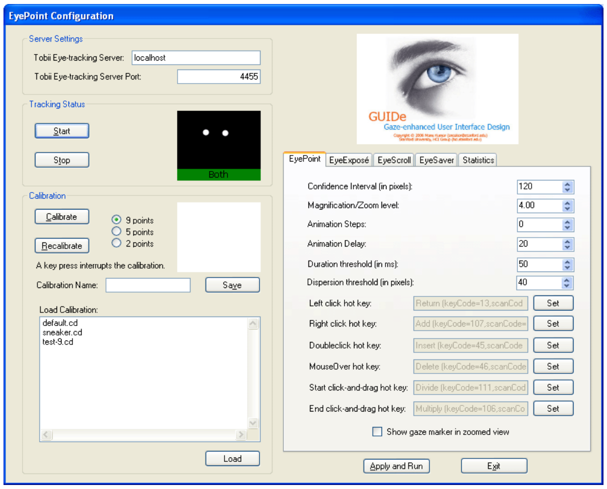
The user then looks at the desired target in the magnified view and releases the hotkey. The user's eye gaze is recorded when the hotkey is released. Since the view has been magnified, the resulting eye-gaze is more accurate by a factor equal to the magnification. A transform is applied to determine the location of the desired target in screen coordinates. The cursor is then moved to this location and the action corresponding to the hotkey (single click, double click, right click etc.) is executed. EyePoint therefore uses a secondary gaze point in the magnified view to refine the location of the target.

Iterative Refinements

The first issue we observed was with the placement of the magnified view. When the user would look at a corner of the screen (for instance to click on the close button of an

---

[1] If the eye tracker were fast enough, it would be possible to begin tracking only when the hotkey is pressed, alleviating long-term use concerns for exposure to infra-red illumination.



Tobii Eye-tracking Server: localhost
Tobii Eye-tracking Server Port: 4455
● 9 points
○ 5 points
○ 2 points
A key press interrupts the calibration.
Calibration Name:
Load Calibration:
default.cd
sneaker.cd
test-9.cd
Confidence Interval (in pixels): 120
Magnification/Zoom level: 4.00
Animation Steps: 0
Animation Delay: 20
Duration threshold (in ms): 50
Dispersion threshold (in pixels): 40
Left click hot key: Return (keyCode=13,scanCod
Right click hot key: Add (keyCode=107,scanCode=
Doubleclick hot key: Insert (keyCode=45,scanCode
MouseOver hot key: Delete (keyCode=46,scanCode
Start click-and-drag hot key: Divide (keyCode=111,scanCod
End click-and-drag hot key: Multiply (keyCode=106,scanCo
☐ Show gaze marker in zoomed view

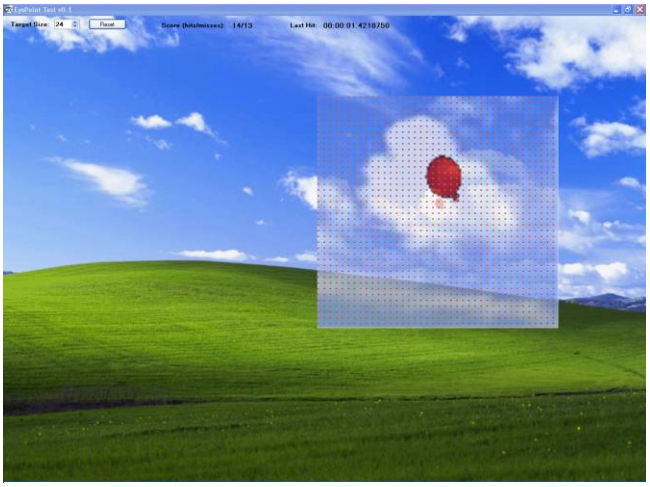
Figure 3. EyePoint configuration screen.

application), our initial prototype's algorithm would center the magnified view at the estimated gaze-point. This resulted in part of the magnified view being displayed beyond the screen boundary, moving the desired target off-screen. To remedy this, we modified the placement algorithm to account for screen boundaries so that the magnified view would always be centered at the estimated gaze-point but offset by the appropriate amount to remain within screen boundaries.

To refocus on the target in the magnified view the user has to perform a secondary visual search. Although this secondary search is always within the same area in which the user was already looking, the user must make one or more saccades in order to locate the target in the magnified view. To facilitate the secondary visual search we added animation to the magnified view such that it appears to emerge from the initially estimated gaze point.

Our pilot studies also showed that though some users would be looking at the target in the magnified view, the gaze data from their fixation was still noisy. We isolated this to whether the user was looking at the target as a whole (a gestalt view) or focusing at a point within the target. Focusing at a point reduced the jitter and improved the accuracy of the system. This led to the introduction of *focus points* in the design – a grid pattern of dots overlaid on the magnified view. Focus points assist the user in making more fine grained selections by focusing the user's gaze. In most cases, the focus points may be ignored by the user, however, they may be useful when the user wants to select a small target (Figure 2).

Some users in our pilot study wondered whether it would be useful to give feedback on what the system thought they were looking at. While this went strongly against our primary design principle of not slaving any visual feedback to eye movements, we implemented an option (Gaze Marker) to show the current gaze point as a blue dot in the magnified view. When the same users tried the system with the gaze marker turned on, they quickly concluded that it was distracting. The time to acquire targets increased since they were now trying to get the gaze marker in precisely the right position before releasing the hotkey (which is unnecessary since the magnification allows for some room for error). As a result, we turned off the gaze marker by default, but decided to test it further in our evaluation.



Target Size: 24
Score (hits/misses): 14/13
Last Hit: 00:00:01.4218750

Figure 4. EyePoint training/test application (used for Balloon Study). This screenshot shows the magnified view with focus points.

Our initial prototype used Ctrl-Space as the hotkey for activating EyePoint. However, our pilot users pointed out that Ctrl-Space was used by some applications. We reconsidered our choice of hotkeys and tried to optimize the default settings to follow Fitt's Law for the motor movements required by users to move their fingers to the appropriate key. We chose to use the keys on the numeric keypad of an extended keyboard as the default hotkeys for EyePoint (Figure 1 Press) since they are not frequently used, are on the right hand side of the keyboard (close to the typical location for a mouse), and provide much bigger keys. As a design constraint, we felt it was better to have a single key press-hold-release action than a multi-key combination. The ideal placement for EyePoint hotkeys would allow the user's hands to always remain on the keyboard (by having dedicated buttons directly below the spacebar, for example).

Final Prototype

The final prototype for EyePoint used our two-step refinement approach from the initial prototype with the modifications described above. In addition, it included several options to allow users to customize the selection of hotkeys, change the default settings for the confidence interval, the magnification factor, the number of animation steps and the animation delay. The EyePoint configuration screen is shown in Figure 3.

DISABLED & ABLE-BODIED USERS

EyePoint works equally well for disabled users and able-bodied users. The hotkey-based triggering mechanism makes it simple for able-bodied users to keep their hands on the keyboard to perform most pointing and selection

Figure 5. EyePoint real-world web-surfing task. The music link in the navigation column on the left has been highlighted in orange.

operations. For laptop users we have considered using gestures on a trackpad where touching different parts of the trackpad would activate different mouse actions.

For disabled users the EyePoint hotkeys could be mapped to alternative triggering devices such as foot-pedals, speech/ gestures or even mouth-tube based (breathe in to activate, breathe out to release) triggers. We hypothesize that these will be more effective than dwell-based activation, but have not performed tests. Dwell based activation is also possible in cases where the user does not have the ability to use any alternative approaches. In this case we would propose an approach similar to [14], but with off-screen targets to first select the action/mode, followed by dwell based activation (with audio feedback [15]) of the magnified view.

EVALUATION

To evaluate EyePoint, we conducted user studies with 20 able-bodied subjects. Subjects were graduate students and professionals and were therefore experienced computer users. Our subject pool had 13 males and 7 females with an average age of 28 years. Fourteen subjects did not require any vision correction, 4 subjects used contact lenses and 2 wore eyeglasses. None of the subjects were colorblind. Sixteen subjects reported that they were touch-typists. Subjects had an average of 15 years of experience using the mouse. None of the subjects had prior experience using an eye-tracker.

We conducted both a quantitative and qualitative evaluation. The quantitative task compared the speed and accuracy of three variations of EyePoint with that of a standard mouse. The three variations of EyePoint were: a) EyePoint with focus points b) EyePoint with Gaze Marker and c) EyePoint without focus points. Since the affect generated by an interaction and the subjective user experience is also a key measure of the success and impacts adoption, our qualitative evaluation included the user's subjective feedback on using gaze-based pointing. Consistent with Norman's views in Emotional Design [19],

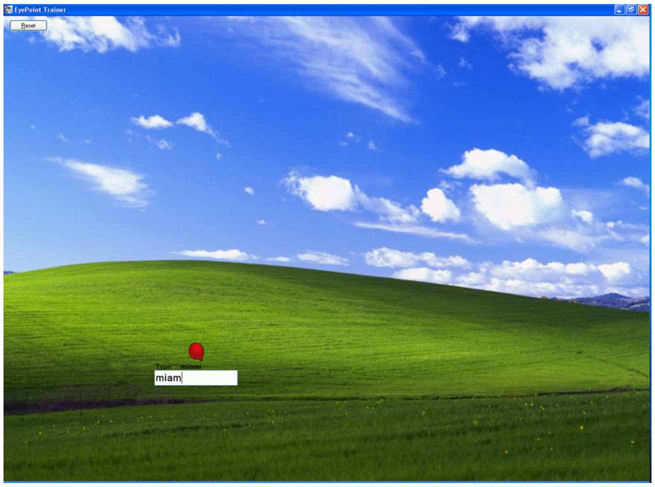


Figure 6. Mixed task study for pointing and typing. When the user clicks on the red balloon, a textbox appears below it. The user must type in the word shown above the textbox.

we believe that speed and accuracy must meet certain thresholds. Once that threshold is met, user preference may be dictated by other factors such as the subjective experience or alternative utility of the technique.

Quantitative Evaluation

We tested performance and accuracy using three independent experiments: a) a real-world web browsing task b) a synthetic pointing only task and c) a mixed typing and pointing task. The orders of both the tasks and the techniques were varied to counterbalance and minimize any learning effects. Subjects were first calibrated on the eye-tracker and then underwent a 5-10 minute training phase where they were taught how to use EyePoint. Subjects practiced by clicking on links in a web browser and also performed 60 clicks in the EyePoint training application (Figure 4). Studies lasted a total of 1 hour and included one additional task reported in a separate paper. The *spacebar* key was used as the trigger key for all three EyePoint variations. Animation of the magnified view was disabled as it would introduce an additional delay (user configurable, but generally about 60-100ms).

Study #1 - Web Study

For a real-world pointing and selection task we asked users to navigate through a series of web pages. The pages were taken from popular websites such as Yahoo, Google, MSN, Amazon, etc. To normalize effects of time for visual search and distance from the target, we disabled all links on the page and highlighted exactly one link on each page with a conspicuous orange highlight (Figure 5).

Users were instructed to ignore the content of the page and simply click on the highlighted link. Each time they clicked on the link a new web page appeared with another highlighted link. The amount of time between presentation of a page and the click was measured. A misplaced click was recorded as an error. Trials were repeated in case of an error. Each subject was shown 30 web pages. The task was

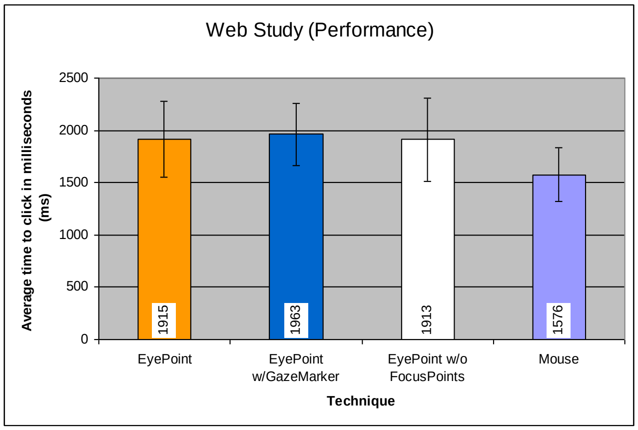


Figure 7. Study #1 - Web Study performance results.

repeated with the same set of pages for all 4 pointing techniques, with ordering counterbalanced.

Study #2 – Balloon Study

For a synthetic task that tested raw pointing speed, we built a custom application that displayed a red balloon on the screen. The user's task was to click on the balloon. Each time the balloon was clicked, it moved to a new location (Figure 4). If the user clicked, but did not hit the balloon, this was recorded as an error and the trial was repeated. Users were instructed to click on the balloon as quickly as they could. The application gathered timing data on how long users took to perform the click. The size of the balloon was varied between 22px (the size of a toolbar button), 30px and 40px. The resulting study is a 4x3 within subjects study (4 techniques, 3 sizes).

Study #3 - Mixed Study

We devised a mixed typing and pointing task in which subjects would have to move their hands between the keyboard and the mouse. In this study subjects first clicked on the target (a red balloon of constant size) and then typed a word in the text box that appeared after they clicked (Figure 6). We measured the amount of time from the click to the first key pressed on the keyboard and the time from the last character typed to clicking on the next balloon. Subjects did not have to press Enter (unlike [6]). As soon as they had typed in the correct word, the system would show the next balloon. The amount of time to correctly type the word shown was not considered since we were only interested in the subject's ability to point and not how well they could type. If the subject clicked, but did not hit the balloon, this was recorded as an error and the trial was repeated.

The sum of the two times measured is the round-trip time to move the hands from the keyboard to the mouse, click on a target and then return back to the keyboard. The mixed study compared only EyePoint with the mouse (i.e. no variations of EyePoint).

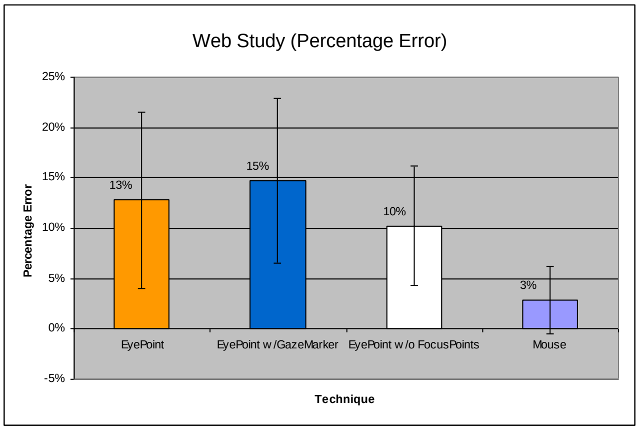


Figure 8. Study #1 - Web Study accuracy results.

Qualitative Evaluation

For the qualitative evaluation users were asked to fill out a questionnaire to provide their comments and opinions on the interaction techniques. They were asked to pick between gaze-based pointing and the mouse for speed, accuracy, ease of use and user preference. In addition, subjects were also asked about which of the EyePoint variations (with focus points, with gaze marker or without focus points) they liked best.

RESULTS

Study #1 – Web Study Results

Figure 7 shows the performance results from the Web Study. A repeated measures ANOVA for technique shows that the results are significant ($F(3,57)=11.9$, $p<.01$). Contrast analyses showed a significant difference for each eye-based technique when compared to the mouse. Results for the gaze marker condition were also significant. However, there was no significant difference between the focus points and no focus point conditions. The average time to click with the mouse was 1576 milliseconds as compared to 1915 milliseconds with EyePoint.

Figure 8 shows the accuracy results for Study #1 – Web Study. A repeated measures ANOVA shows that the error results are significant ($F(3,57)=14.9$, $p<.01$). Contrast analyses showed a significant difference for each eye-based technique when compared to the mouse. Errors amongst the three eye-based variations were not significant. The mouse had an average error rate of 3%, while the EyePoint error rate was 13%. The no focus points condition had an average error rate of 10%.

Qualitative results for the web study showed that subjects' opinions were evenly split on which technique was faster (EyePoint or mouse) and which was easier to use. Although all subjects felt that the mouse was more accurate, three quarters of the subjects said they would choose to use EyePoint for this task over the mouse since they felt it was faster, easier or just cooler. A majority of the subjects preferred having focus points and felt that the focus points gave them something to "hold" on to.

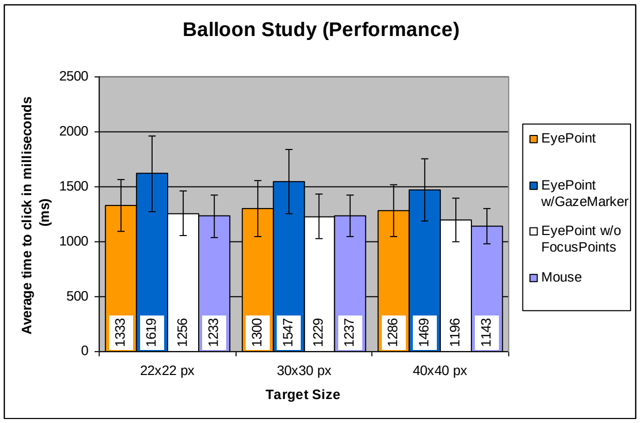


Figure 9. Study #2 - Balloon Study performance results.

Study #2 – Balloon Study Results

Figure 9 shows the performance results for Study #2 – the Balloon Study. EyePoint performs on average about 100ms slower than the mouse. A repeated measures ANOVA for size and technique showed a significant effect for size (F(2,38) = 26.8; p < .01), and for technique (F(3, 57) = 14.8; p <.01). We found no interaction effect between size and technique. Contrast analyses showed that significant differences existed for all pairs of sizes. For technique, contrasts showed a significant difference between all pairs of techniques except EyePoint with no focus points vs. mouse.

Figure 10 shows the error rates for Study #2 – Balloon Study. In accordance with theory, the size of the target did have an appreciable impact on the error rates. Contrast analyses showed that the difference in error rates between the gaze-based techniques was not significant. The differences between each of the gaze based techniques and the mouse were significant. It should be noted that the error rates for gaze-based pointing techniques were considerably higher than in the Web study. We will discuss these results in the next section.

Qualitative results for the balloon study showed that subjects found the mouse to be faster and more accurate. However, the gaze-based techniques were easier to use and again three quarters of the subjects said they would prefer to use the gaze-based technique for this task. Subjects felt that moving the mouse was fatiguing over time and that it was easier to click using the gaze-based methods despite the speed disadvantage.

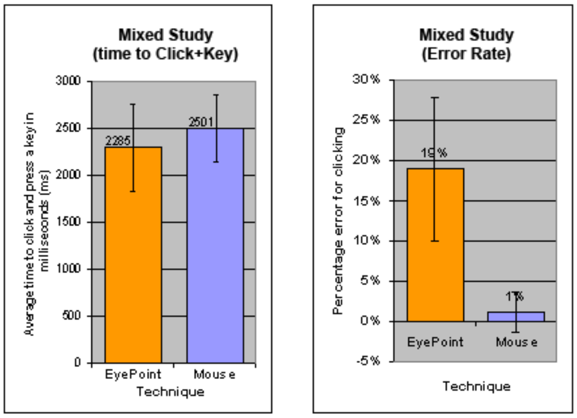


Figure 11. Study #3 - Mixed Study performance/error results.

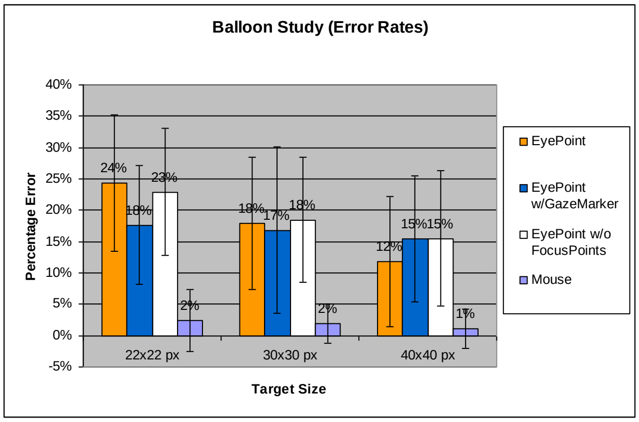


Figure 10. Study #2 - Balloon Study accuracy results.

Study #3 – Mixed Study Results

Figure 11 shows the performance results for the total round trip time to point to a target and return to the keyboard. EyePoint is faster than the mouse in this task. A paired sample two-tailed *t*-Test showed that the results are statistically significant with $p < .05$.

Figure 11 also shows the accuracy results from Study #3 – Mixed Study. It should be noted that while the gaze-based technique had better performance, it lacked in accuracy when compared to the mouse. A paired sample two-tailed *t*-Test showed that the error results are statistically significant with $p < .01$.

Qualitative results for the mixed study showed a strong preference (>90%) for EyePoint on the speed, ease of use and user preference dimensions - primarily since users didn't have to move their hands off the keyboard. The mouse was preferred only on the accuracy dimension.

ANALYSIS AND DISCUSSION

The above results present an incomplete picture without a deeper analysis. If we isolate the actions the user must perform to point and click on a target with the mouse, the total time would be:

$$T_{mouse} = t_{acquire\ target} + t_{acquire\ mouse} + t_{acquire\ cursor} + t_{move\ mouse} + t_{click\ mouse}$$

By contrast, the total time for selection using EyePoint would be:

$$T_{eyepoint} = t_{acquire\ target} + t_{acquire\ hotkey} + t_{press\ hotkey} + t_{reacquire\ target} + t_{release\ hotkey}$$

It can be reasonably expected that the time to acquire the target, i.e. perform a visual search for the target is the same in both cases. The time to acquire the mouse vs. the hotkey would depend on Fitt's Law [5, 6]. In our studies we found that having a large hotkey such as the space bar, reduced the acquisition time for the hotkey (Study #3). The key performance difference between using the mouse and using the eye arises from the second visual search to re-acquire the target in the magnified view. We observed that subjects were able to parallelize tasks when using the mouse. For instance, they would already have their hand on the mouse and begin moving it even before they had performed the visual search. This may be the result of years of practice with using the mouse. Due to the concurrent nature of the sub-tasks for pointing with the mouse, the amount of time it takes the user to move the mouse and the amount of time it takes the user to perform a secondary visual search when using gaze are similar (assuming the time to click the mouse and release the key are similar).

Based on the empirical results and the model proposed above, we find that the performance of EyePoint is similar to the performance of the mouse and can actually be faster than the mouse in some cases.

The analysis of error rates is a little more complex. While the results shown in the previous section suggest that the error rates when using gaze-based pointing are considerably higher than those when using the mouse, the graphs do not tell the complete story. A deeper analysis of the error data showed that the error rates varied significantly across subjects. The eye-gaze tracker works better for some subjects than others. The accuracy of the eye-tracking depends not only on the individual, but on the quality of the calibration and the posture of the subject over the course of the experiments.

If we partition the error data into subjects for whom the eye-tracker worked well and subjects for whom the eye-tracker didn’t work as well, the error rates for the first group are closer to 10% while those for the second group are closer to about 33%.

In the case of the balloon studies, we observed that since the task required subjects to click on the ballons in rapid succession, some subjects would press the EyePoint hotkey prematurely, in anticipation of the next balloon, before they even actually looked at it. This resulted in a significantly higher error rate. In practice, we can reasonably expect that subjects will look at the target before activating the hotkey.

The implementation of EyePoint uses a fixation detection algorithm that expects the subject’s gaze to be within a certain region for at least 25-50 ms before it updates the current gaze coordinate. This resulted in timing issues in the balloon studies. Subjects would see the balloon in their peripheral vision and press the hotkey before their foveal vision fixated on the target. To reduce such errors we propose a change to base the initial fixation on a window of time that extends slightly beyond the hotkey activation time, thereby giving the subject the ability to focus on the target before the gaze-point is determined.

Our observation of the subjects while they performed the study also revealed other interesting details. One subject, for instance, laughed and smiled a lot, which caused the subject’s eyes to squint and resulted in a loss of eye-tracking accuracy (sometimes no data at all). Our pilot studies included a subject with astigmatism and weighted contact lenses which reduced the accuracy of the eye-tracker, possibly due to the differential movement of the weighted contact lenses. For subjects with glasses we found that large frames with thin lenses work better than narrow frames and thick lenses; the former because the rim of the frame doesn’t occlude the vision of the camera and the latter since it reduces the visual distortion of the eyes.

In the qualitative evaluation subjects also reported that they found that the gaze-based techniques required more “focus” and more “concentration” and therefore found the studies to be fatiguing over time. Each subject participated in the study for one hour during which they had to click on approximately 500 targets with their eyes and about 100 targets with the mouse. In standard use, we do not expect such intense usage.

While our studies randomized trials in order to compensate for learning effects, we did observe learning effects as subjects adapted to the system. For real-life usage one might therefore expect improvement over time as users adapt to using gaze-based pointing. This would also reduce the cognitive load of pressing the right hotkey for the desired action.

Subjects strongly preferred EyePoint over using the mouse. They felt that it was more natural since they were already looking at the target when they wanted to point. It allowed them to keep their hands on the keyboard and was therefore much faster for mixed tasks that involved typing and pointing. They also felt that EyePoint reduced the risk of repetitive stress injury from using the mouse.

CONCLUSION

In keeping with the guidelines from Zhai and Jacob, we believe that it is indeed possible to devise appropriate interaction techniques that use gaze without overloading the visual channel. Interaction techniques that use gaze as an augmented input are more compelling for able-bodied users than gaze-only approaches. The user’s eye gaze provides context and information about the user’s attention and intention. A well designed multi-modal application can use this information to devise a more intelligent interaction with the user.

EyePoint presents a practical and innovative interaction technique that combined the use of gaze and key based activation into a single look-press-look release action. This transforms a two-step refinement process into a single fluid action and prevents overloading the visual channel while still using gaze-based target refinement. EyePoint makes gaze-based pointing equally compelling for use by both disabled and able-bodied users.

ACKNOWLEDGMENTS

The authors would like to thank Shumin Zhai and David Beymer for several interesting discussions and their feedback on this research. The feedback from the anonymous CHI reviewers helped to substantially improve the presentation of this work. The authors would also like to acknowledge Stanford Media X and the Stanford School of Engineering matching funds grant for providing the funding for the GUIDe (Gaze-enhanced User Interface Design) project.

REFERENCES

1. *IPRIZE: a $1,000,000 Grand Challenge designed to spark advances in eye-tracking technology through competition*, 2006. http://hcvl.hci.iastate.edu/IPRIZE/
2. Amir, A., L. Zimet, A. Sangiovanni-Vincentelli, and S. Kao. An Embedded System for an Eye-Detection Sensor. *Computer Vision and Image Understanding, CVIU Special Issue on Eye Detection and Tracking* 98(1). pp. 104-23, 2005.
3. Ashmore, M., A. T. Duchowski, and G. Shoemaker. Efficient Eye Pointing with a FishEye Lens. In Proceedings of *Graphics Interface*. pp. 203-10, 2005.
4. Beymer, D., S. P. Farrell, and S. Zhai. 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
5. Card, S. K., W. K. English, and B. J. Burr. Evaluation of mouse, rate-controlled isometric joystick, step keys, and text keys, for text selection on a CRT. *Ergonomics* 21(8). pp. 601-13, 1978.
6. Douglas, S. A. and A. K. Mithal. The effect of reducing homing time on the speed of a finger-controlled isometric pointing device. In Proceedings of *CHI*: ACM Press. pp. 411-16, 1994.
7. Farrell, S. P. and S. Zhai. System and method for selectively expanding or contracting a portion of a display using eye-gaze tracking. USA Patent 2005, International Business Machines Corporation
8. Fono, D. and R. Vertegaal. EyeWindows: Evaluation of Eye-Controlled Zooming Windows for Focus Selection. In Proceedings of *CHI*. Portland, Oregon, USA: ACM Press. pp. 151-60, 2005.
9. Hansen, D. W., D. MacKay, and J. P. Hansen. Eye Tracking off the Shelf. In Proceedings of *ETRA: Eye Tracking Research & Applications Symposium*. San Antonio, Texas, USA: ACM Press. pp. 58, 2004.
10. Jacob, R. J. K. The Use of Eye Movements in Human-Computer Interaction Techniques: What You Look At is What You Get. In Proceedings of *ACM Transactions in Information Systems*. pp. 152-69, 1991.
11. Jacob, R. J. K. and K. S. Karn, Eye Tracking in Human-Computer Interaction and Usability Research: Ready to Deliver the Promises, in *The Mind's eye: Cognitive and Applied Aspects of Eye Movement Research*, J. Hyona, R. Radach, and H. Deubel, Editors. Elsevier Science: Amsterdam. pp. 573-605, 2003.
12. Kumar, M., *GUIDe: Gaze-enhanced User Interface Design*, 2006. Stanford. http://hci.stanford.edu/research/GUIDe
13. Kumar, M., *Reducing the Cost of Eye Tracking Systems*. Technical Report CSTR 2006-08, Stanford University, Stanford, April 2006. http://hci.stanford.edu/cstr/reports/2006-08.pdf
14. Lankford, C. Effective Eye-Gaze Input into Windows. In Proceedings of *ETRA: Eye Tracking Research & Applications Symposium*. Palm Beach Gardens, Florida, USA: ACM Press. pp. 23-27, 2000.
15. Majaranta, P., I. S. MacKenzie, A. Aula, and K.-J. Räihä. Auditory and Visual Feedback During Eye Typing. In Proceedings of *CHI*. Ft. Lauderdale, Florida, USA: ACM Press. pp. 766-67, 2003.
16. Majaranta, P. and K.-J. Räihä. Twenty Years of Eye Typing: Systems and Design Issues. In Proceedings of *ETRA: Eye Tracking Research & Applications Symposium*. New Orleans, Louisiana, USA: ACM Press. pp. 15-22, 2002.
17. McGuffin, M. and R. Balakrishnan. Acquisition of Expanding Targets. In Proceedings of *CHI*. Minneapolis, Minnesota, USA: ACM Press. pp. 57-64, 2002.
18. Miniotas, D., O. Špakov, I. Tugoy, and I. S. MacKenzie. Speech-Augmented Eye Gaze Interaction with Small Closely Spaced Targets. In Proceedings of *ETRA: Eye Tracking Research & Applications Symposium*. San Diego, California, USA: ACM Press. pp. 67-72, 2006.
19. Norman, D. A., *Emotional Design: Why we love (or hate) everyday things*. New York: Basic Books. 256 pp. 2004.
20. Salvucci, D. D. Intelligent Gaze-Added Interfaces. In Proceedings of *CHI*. The Hague, Amsterdam: ACM Press. pp. 273-80, 2000.
21. Salvucci, D. D. and J. H. Goldberg. Identifying Fixations and Saccades in Eye-Tracking Protocols. In Proceedings of *ETRA: Eye Tracking Research & Applications Symposium*. Palm Bech Gardens, Florida, USA: ACM Press. pp. 71-78, 2000.
22. Sibert, L. E. and R. J. K. Jacob. Evaluation of Eye Gaze Interaction. In Proceedings of *CHI*. The Hague, Amsterdam: ACM Pres, 2000.
23. Tobii Technology, AB, Drift Effects, in *User Manual: Tobii Eye Tracker and ClearView Analysis Software*. Tobii Technology AB. p. 15, 2006.
24. Tobii Technology, AB, *Tobii 1750 Eye Tracker*, 2006. Sweden. http://www.tobii.com
25. Wang, J., S. Zhai, and H. Su. Chinese Input with Keyboard and Eye-Tracking - An Anatomical Study. In Proceedings of *CHI*. Seattle, Washington, USA: ACM Press. pp. 349-56, 2001.
26. Yamato, M., A. Monden, K.-i. Matsumoto, K. Inoue, and K. Torii. Button Selection for General GUIs Using Eye and Hand Together. In Proceedings of *AVI*. Palermo, Italy: ACM Press. pp. 270-73, 2000.
27. Zhai, S., S. Conversy, M. Beaudouin-Lafon, and Y. Guiard. Human On-line Response to Target Expansion. In Proceedings of *CHI*. Ft. Lauderdale, florida, USA: ACM Press. pp. 177-84, 2003.
28. Zhai, S., C. Morimoto, and S. Ihde. Manual and Gaze Input Cascaded (MAGIC) Pointing. In Proceedings of *CHI*. Pittsburgh, Pennsylvania, USA: ACM Press. pp. 246-53, 1999.