

Speed and Accuracy of Gaze Gestures

Henna Heikkilä

Unit for Computer-Human Interaction
(TAUCHI)
Department of Computer Sciences
University of Tampere, Finland

Kari-Jouko Rähä

Unit for Computer-Human Interaction
(TAUCHI)
Department of Computer Sciences
University of Tampere, Finland

We conducted an experiment where participants carried out six gaze gesture tasks. The gaze paths were analyzed to find out the speed and accuracy of the gaze gestures. As a result, the gaze gestures took more time than we anticipated and only the very fastest participants got close to what was expected. There was not much difference in performance times between small and large gaze gestures, because the participants reached significantly faster speed when making large gestures than small gestures. Curved shapes were found difficult to follow and time-consuming when followed properly. In general, the accuracy in following shapes was sometimes very poor. We believe that to improve the speed and accuracy of gaze gestures, proper feedback must be used more than in our experiment.

Keywords: gaze gestures, gaze interaction, gaze-control, eye tracking

Introduction

Gestures are a familiar concept from mouse and pen-based interaction. In gaze research, however, gestures have only recently started to interest researchers. Gaze gestures are basically short snippets from the user's gaze path and they should be interpreted as commands issued by the user, which makes it challenging to separate gaze gestures from looking around and from other actions.

Some of the biggest challenges in gaze interaction are calibration and accuracy. Calibration problems often occur if the user shifts position during the interaction or if the initial calibration has been done poorly. Even when calibration works for most of the screen, problems may still occur near the edges of the screen, especially at the bottom of the screen. With some eye trackers, the calibration becomes less accurate during the use, and the user has to recalibrate the eye tracker (Hansen et al., 2008). Drewes and Schmidt (2007) claimed that the use of gaze gestures would solve these problems, since gaze gestures can be used without any calibration if designed properly.

Gaze is naturally inaccurate due to the jitter that occurs during fixations. These small movements are undetectable for humans and they impede the accurate detection of fixations (Jacob, 1991). Gaze gestures are based on the shape of the gaze path, not just on the exact point

of the gaze, which alleviates the accuracy problem. Moreover, when using gaze gestures, the eyes do not have to stay still over a certain time period like they have to when using, for example, dwell time as a selection method.

Drewes and Schmidt (2007) introduced gaze gestures also as a solution for the space limitation problem that gaze-controlled applications may suffer from. Gestures are often used in contexts where the screen space is limited, such as pen-based PDAs. By using gestures, the screen space can be freed from menus and toolbars, and the number of commands can be increased without taking extra space from the screen (Drewes & Schmidt, 2007). Nevertheless, the obvious disadvantage is that the user needs to learn the gestures and remember them. This sets limitations on the number and complexity of the gestures.

We present an experiment that was designed to shed light on the characteristics of gaze gestures and to give some guidelines for designing them. Our motivation comes from the desire to design and implement a drawing application that uses gaze gestures. The drawing application itself will be based on objects that can be moved and modified after they are drawn. We believe that this kind of an application is a very suitable platform for gaze gestures for two reasons. First, most of the screen space is needed for the drawing and the use of gaze gestures can

free space from toolbars. Second, gaze gestures can make the interaction more straightforward. For example, the user can delete an object by issuing a certain gaze gesture over that object without needing to first dwell on the object and then to dwell on the toolbar to select the wanted action.

Next, we briefly review related gaze gesture research and discuss the concept of gaze gestures. We then introduce our experiment, discuss our results and conclude with the implications and future directions.

Related Research

For long, dwell time has been the most popular selection method in gaze-only interaction. Now new methods, such as gaze gestures, have increasingly attracted interest among gaze researchers. Next we look at 12 studies on gaze gestures.

Drewes and Schmidt (2007) created a set of scalable gaze gestures that can be used to control applications, such as a music player, in various environments. Their gaze gestures are based on eight directions (horizontal, vertical, and diagonal) and each gesture combines 4-6 directions. For example, the user needs to move their eyes from left to right and from right to left three times to make a gesture. The results from their user study were encouraging as the participants were able to perform most of the given tasks. They noted that the background does not have an effect on the performance of the gestures. They also found that the participants preferred to make large-scale gestures rather than small-scale gestures and the time needed to complete large gestures did not differ significantly from the time of the small gestures.

The technique of Drewes and Schmidt (2007) is unique in that the gestures can be made anywhere on the screen. Gestures are recognized through their characteristic shape that does not occur in normal screen viewing behaviour. In contrast, Møllénbach et al. (2009) introduced gaze gestures where only one stroke is needed. The single-stroke gaze gestures are issued by moving the gaze from one side of the screen to the opposite side. In the implementation of Møllénbach et al., the gaze gesture had to be issued within 1000 milliseconds or the gesture process would reset itself. In other words, if after leaving one of the edge areas the gaze does not arrive at another edge area within the time limit, a gesture is not recog-

nized. The simplicity of the strokes restricts the number of gestures to four (or to eight if diagonal from corner to corner strokes are included).

In recent studies, gaze gestures have often been used for entering text. An early example is the application developed for the VisionKey eye tracker (Kahn et al., 1999), where the user selects characters from an alphabet palette (Figure 1). The selection process has two steps. First, the user gazes at the corner of the palette that corresponds to the position of the target character in one of the squares of the palette. Then, the user fixates on that square of the palette that contains the character. The text entry rate for experienced users is promised to be about one character per second.

←	1	2	3	4	5	6
↓	A	B	C	D	E	7
↖	F	G	H	I	J	8
↗	K	L	M	N	O	9
↔	P	Q	R	S	T	0
↕	U	V	W	X	Y	-
↘	Z	,	.	/	=	

Figure 1. Character palette from VisionKey's writing application (Kahn et al., 1999).

Quikwriting was originally designed for stylus-based interaction (Perlin, 1998), but Bee and André (2008) re-designed it to be used with gaze. Like in VisionKey, the character selection procedure has two steps. First, the user moves eye gaze from the centre area through the group of characters that contains the target character. This move opens the characters from that group in their own sections. Then, the user gazes at the target character and returns back to the centre area. The dotted line in Figure 2 represents the gaze path of a user entering the letter G. The idea of Quikwriting is that expert users can memorise the gaze paths of different characters, and then use them as gaze gestures. In their small-scale experiment, Bee and André (2008) found that the text entry rate of the gaze-based Quikwriting application was 5.0 words per minute, on the average. With a gaze-based on-screen keyboard the same participants achieved an average text entry rate of 7.8 words per minute.

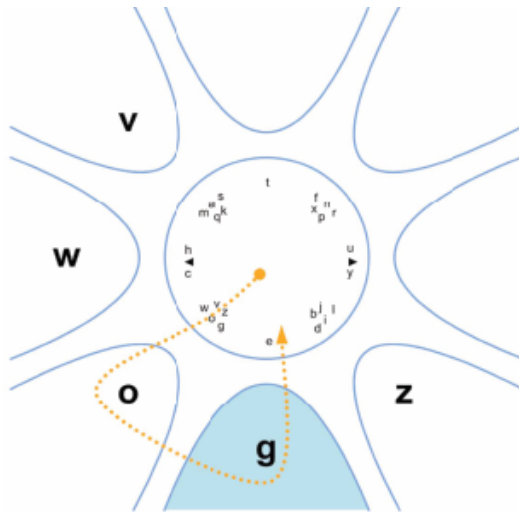


Figure 2. The dotted line represents the user's gaze path when selecting the letter G (Bee & André, 2008).

Huckauf and Urbina (2008) introduced a text entry technique, pEYEWwrite, that is similar to Quikwriting. In pEYEWwrite, two overlapping pie menus (Figure 3) are used to enter a character. When the user looks at the outer frame of one of the slices in the first pie, a second pie containing the characters of that slice opens over the slice. By learning the gaze paths (or the gaze gestures) needed to enter a certain character, an experienced user can write with the application without needing to open the pie menus.

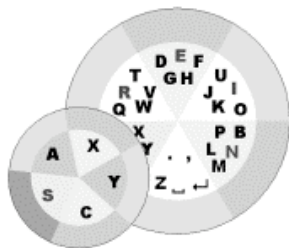


Figure 3. In pEYEWwrite (Huckauf & Urbina, 2008), the user first looks at one of the slices in the pie, which opens another pie from which the user can select the target character.

Huckauf and Urbina (2008) carried out a user study to compare pEYEWwrite with an ABCDE on-screen keyboard. The results indicate that an expert user can write with pEYEWwrite as fast as with the ABCDE on-screen keyboard. However, when comparing the performance of novice users, pEYEWwrite loses to the ABCDE on-screen keyboard in text entry rate.

In their pEYETop application, Huckauf and Urbina (2008) used the pie method for organization of the desktop. The preliminary usability evaluation gave encouraging results: the participants felt that the use of pEYETop was fast and easy.

Huckauf and Urbina implemented two more applications, Iwrite and StarWrite, which both use gesture-like, dwell-free gaze input for entering text (Urbina & Huckauf, 2007). In Iwrite, the user selects a character from the surrounding character frame by moving the gaze from text area through the target character to the dark outer frame. In StarWrite, the user “drags” the target character to the writing area from the character sequence. The action is not actual dragging; instead, the user looks at the target character and then shifts the gaze towards the text field under the character sequence.

Urbina and Huckauf (2007) compared three of their own designs (pEYEWwrite, Iwrite, and StarWrite) against another dwell-time free text entry technique, Dasher (see Ward et al., 2000), and against a QWERTY on-screen keyboard. Their very preliminary results suggest that their own designs beat Dasher in speed. Still, the QWERTY on-screen keyboard was undefeated in speed and also preferred by the participants.

In the techniques presented above, gaze gestures were used to select characters from a character frame or sequence. Another way to use gaze gestures for entering text is to compose the gestures so that they resemble letters of the alphabet. This makes it easier to learn and remember the gestures. One of the applications that use letter-like gaze gestures is EyeWrite implemented by Wobbrock et al. (2008). EyeWrite follows the approach of EdgeWrite (Wobbrock et al., 2005). In EyeWrite, gaze gestures are issued on a writing window (see Figure 4), which includes outlined corner areas and dots to help the gaze gesture issuing process. Each gesture consists of looking at two or more corners of the window in a specified order.

Wobbrock et al. (2008) found that although gaze gestures were judged easier and faster to use and felt less fatiguing than the on-screen keyboard, entering text with gaze gestures was slower than with the dwell-based on-screen keyboard. However, the learning curve lets one expect that with practice an expert can enter text with gaze gestures as fast as the expert with an on-screen keyboard.

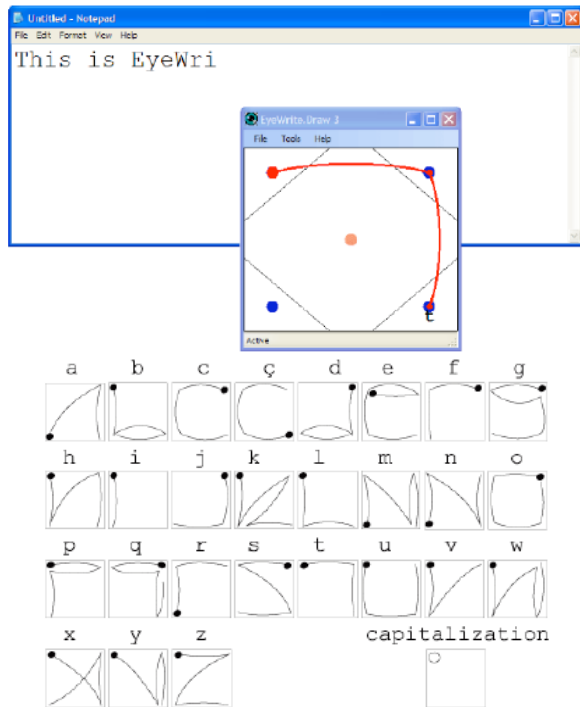


Figure 4. In EyeWrite, the gaze gestures are issued in a separate window. EyeWrite's gaze gestures resemble roman letters. (Wobbrock et al., 2008)

Porta and Turina (2008) developed the Eye-S writing application that also uses letter-like gaze gestures. The gestures in Eye-S are based on looking at some of the nine points on the screen (corners and sides) in a specific order. Gaze gestures can also be used to issue standard commands, such as Ctrl-C. Porta and Turina studied eight novice users and two expert users and found that although entering a full sentence was time-consuming, the users were able to complete all the gestures needed. Compared to EyeWrite, the average writing speed in Eye-S is slower. However, Eye-S has the advantage that no additional window is needed and the gestures can be issued over any window (with commands sent to the active window).

Isokoski (2000) implemented a gaze writing application based on off-screen targets and a text entry technique called Minimal Device Independent Text Input Method (MDITIM). MDITIM is based on four principal cardinal directions (North, East, South, and West). In Isokoski's experiment, these directions were marked on the frame of the monitor. When using MDITIM with eye gaze, the user enters text by looking at the off-screen targets in a certain order. For example, when entering the letter Y,

the user looks at the off-screen targets in the following order: South, East, South, and West.

Istance and his colleagues (Istance et al., 2008; Vickers et al., 2008) also utilised the outside of the screen in their gaze gestures. They created gaze gestures to be used in Second Life, a 3D online community, and in World of Warcraft, a 3D computer game, to switch between modes. The gaze gestures they used were quick glances outside the screen; for example, a quick glance up turned off the gaze control.

Inspired by EyeWrite, Istance and his colleagues implemented a gaze gesture add-on for World of Warcraft (Vickers et al., 2009). The transparent activation zones of the add-on are placed around the game character, so that the user can easily issue the gaze gesture without losing the focus from the game character. With their add-on, 12 different gaze gestures can be used to control the game.

Gaze Gestures

We have seen numerous examples of gaze gestures of different types. Their diversity motivates further analysis of what actually is a gaze gesture. Drewes and Schmidt (2007) defined a gesture as “a sequence of elements, which are performed in a sequential time order”. In gaze interaction, this would mean that a gaze gesture is a sequence of consecutive eye movements. Møllenbach et al. (2009) defined a gaze gesture more precisely as “a controlled saccade or a series of controlled saccades that cause an action when completed”.

Eye movements consist of saccades and fixations. Saccades are time periods when the eyes quickly move from one place to another. For example, Abrams et al. (1989) measured 30–50 milliseconds long saccades when their participants moved the eyes from one fixation point to another. Rayner (1998) reported 30 milliseconds saccades as an average from their experiment concerning reading tasks. Furthermore, the saccade durations depend heavily on the distance the eyes move during the saccades (Rayner, 1998). Fixations are time periods during which the eyes stay relatively still. Fixation durations are reported to be 200–400 milliseconds (Rayner, 1998). However, the durations are dependent on the task at hand. In gaze gestures, fixations can be used for determining the shape of the gaze gesture and for differentiating the saccades from the gaze data.

Gaze gestures are often presented as a dwell time-free interaction method or as a solution for problems that the use of dwell time creates. Nevertheless, most of the gaze gesture studies presented in the previous section use dwell time one way or the other. Often the dwell time is used to determine which fixations should be taken into account when determining the shape of the gaze gesture.

Drewes and Schmidt (2007) specified that a gaze gesture should contain only short fixations. Similarly, Isokoski (2000) defined short fixations to be less than 100 milliseconds. Drewes and Schmidt also defined that long fixations should reset the process of detecting a gaze gesture. In their study, Drewes and Schmidt set the threshold for long fixations at 1000 milliseconds. Porta and Turina (2007) and Møllenbach et al. (2009) used a similar timeout period for resetting the gaze gesture recognition process.

In Eye-S (Porta & Turina, 2008) and in EyeWrite (Wobbrock et al., 2008), short dwell times (400 ms and 250 ms, respectively) are used for determining the path of the gaze gesture. In Eye-S, the user needs to fixate briefly on every hot spot needed for the gaze gesture. A fixation that is too long resets the sequence. In EyeWrite, the user needs to fixate on the centre point when starting the gaze gesture sequence. Then, it is sufficient just to visit a corner of the application window by crossing the line that outlines the corner area. The gaze gesture sequence stops when the user fixates back on the centre point.

Istance et al. (2008) also used very short time periods (50–100 ms) to determine whether the gaze had moved outside the screen or not. A similar time period was used in Isokoski's (2000) study with off-screen targets. These small dwell times were for determining whether the detected eye movements were intentional and the eyes actually stopped on the target or whether the eyes were merely passing by the target.

Most of the techniques presented above require that the user moves their gaze through certain areas during the gaze gesture sequence. However, there are some exceptions. Drewes and Schmidt (2007) used an invisible grid to determine the shape of the gaze gesture. This lets the user produce the gaze gestures in any size and anywhere on the screen. The pie menus for desktop control (pEYETop) suggested by Huckauf and Urbina (2008) can be evoked anywhere on the screen, but the size of the gaze gesture needs to match the size of the pie menu. Also, the gaze gestures used in the application of Istance et al. (2008) are an exception, since they require only a glance outside the screen. As can be seen from Table 1, the variation between the gaze gestures used in previous studies is broad.

In Table 1, we have gathered information about gaze gestures in different studies. We searched answers to following questions: Were the gaze gestures scalable in size? Were the gaze gestures bound to certain location? What was used to determine whether a gaze gesture actually was issued? Were fixations needed to define the gaze gesture? Was dwell time used in the gaze gestures? If dwell time was used, how long was it?

Table 1 shows that only two of the techniques use scalable gaze gestures, where the users themselves can decide on how small or large gestures they want to issue. In addition, most applications use gaze gestures that are location bound. This means that the user needs to issue the gaze gestures in a certain location, for example, in a separate window. These two aspects prevent the users from tuning the gaze gestures according to their own preferences. Table 1 further shows that fixations are often used during the gaze gesture process to determine either the shape of the gaze gesture or the starting and ending points of the gaze gesture. Dwell times for these fixations vary, but they are usually adjustable.

Table 1: Characteristics of gaze gestures in recent studies.

	Scalable gestures	Location bound	What is used to determine which gaze gesture was made?	Fixating needed during the process
Bee and André (2008)	No	Yes	Pie menu (modified)	No
Drewes and Schmidt (2007)	Yes	No	Grid	Yes, short fixations and over 1000 ms fixation resets the process
Huckauf and Urbina (2008)	No	Yes	Pie menu	No
Isokoski (2000)	No	Yes	Off-screen targets	Yes, dwell time less than 100 ms
Istance et al. (2008) Vickers et al. (2008)	Yes	No	Outside the screen	No, but must stay 50-100 ms outside the screen
Kahn et al. (2000)	No	Yes	Alphabet palette	Yes, dwell time adjustable
Möllenbach et al. (2009)	No	Yes	4 target areas	No, but needs to be completed within 1000 ms
Porta and Turina (2008)	No	Yes	9 target areas	Yes, dwell time adjustable
Urbina and Huckauf (2007)	No	Yes	Selection of letters	No
Vickers et al. (2009)	No	Yes	4 target areas	No
Wobbrock et al. (2008)	No	Yes	5 target areas	Yes, to start and to end the process, dwell time adjustable

Our Experiment

The study presented here is our first study on using gaze gestures in a drawing application. The motivation behind this study is to shed more light on the characteristics of gaze gestures and to give us some guidelines for designing them. The experiment was designed with the drawing application in mind, and we were especially interested in learning about how accurately gaze gestures can be performed, and what effects the size of the gesture has on speed and accuracy.

Participants

Sixteen participants, 7 males and 9 females, volunteered for the tests. Their ages ranged from 19 to 33 years (mean 24.8 years). Four of the participants wore eye-glasses during the test. Ten participants had tried an eye tracking system before, but only one of them could be called an expert, since this participant had done research with eye tracking systems.

Apparatus

We used the Tobii 1750 eye tracker (with screen resolution of 1280x1024 pixels) and ClearView 2.6.3 during this experiment to track the participant's eyes and to record the gaze data. The eye tracker was calibrated for each participant and the calibration was done with 16 points.

Tasks

We asked the participants to complete a set of tasks as fast and accurately as they could. Tasks 1, 2, and 3 were done on an empty drawing area and tasks 4, 5, and 6 were done on a drawing area that had a model shape. The tasks as given to the participant were:

1. Draw with your gaze the letter L.
2. Draw with your gaze a triangle.
3. Draw with your gaze a line that goes from left to right and return to the starting point.

4. Draw with your gaze the form of the green rectangle (by looking at each of its corners one by one).
5. Draw with your gaze the form of the green circle (by following the sphere of the circle).
6. Look once at each end of the green line.

The system did not give any feedback from eye movements during the test. Participants did all tasks with both a small (250x250 pixels) and a large (1180x920 pixels) drawing area. All tasks were repeated five times and, in total, each participant did 60 tasks.

Model shapes were used in tasks 4, 5, and 6. In task 4, the size of the rectangle was 146x132 pixels (small) or 655x663 pixels (large). In task 5, the radius of the circle was 60 pixels (small) or 323 pixels (large). In task 6, the line was 161 pixels (small) or 700 pixels (large) long.

The motivation behind designing the tasks was two-fold. The first three tasks were designed to measure the ability to draw an instructed shape. This corresponds to the basic task when using gaze gestures to control an application. The last three tasks were designed to test the ability to follow a given model. These kinds of gaze gestures could be used to select objects in a drawing application.

When planning the experiment, the tasks were divided into four task groups: Task group I containing tasks 1–3 drawn on a small drawing area, Task group II containing tasks 4–6 drawn on a small drawing area, Task group III containing tasks 1–3 drawn on a large drawing area, and Task group IV containing tasks 4–6 drawn on a large drawing area. The order of the four task groups was balanced between participants by using balanced Latin squares. However, the task order within a task group remained the same throughout the tests.

Procedure

In the beginning of the test, the participant was seated in front of the eye tracker, approximately 60 centimetres from the eye tracker. Then, the eye tracker was calibrated. The calibration was checked by asking the participant to look at a star shape on the screen. When the calibration was determined sufficient, the actual test was started and the experimenter left the room. During the test, the participant controlled the advancing of the screens by hitting the enter key.

First, short instructions were shown on the screen to the participant to remind them on the procedure. Then each task screen followed by the screen containing the drawing area was shown to the participant. After each task had been performed five times on both drawing areas, a thank you screen appeared. Last, the participant was thanked for participation in person. Each test session took about 30 minutes.

Results

We were interested in how fast the participants could perform the given tasks and how accurately they could follow the given models (i.e., hit the corners of the rectangles, the ends of the lines and the sphere of the circle).

Although one claimed advantage of gaze gestures is that they do not need to be accurate, we still were especially interested in the accuracy. We wanted to find out whether we could use accurate gaze gestures in our drawing application, for example, to select a rectangle by looking at each of its corners one by one.

Duration

Gestures on an empty drawing area. The durations were measured from the fixation that started the gesture to the fixation that completed the gesture, including the time of the starting and closing fixations. The fastest gestures were small and large L, which were performed in 1648 and in 1756 milliseconds on the average. Only slightly slower was the back and forth line, which took, on the average, 1969 (small) and 2245 (large) milliseconds. The slowest gesture was the triangle. On the average, it took 2559 (small) and 2843 (large) milliseconds. In general, small gestures were 100 to 300 milliseconds faster to make than similar large gestures. See Table 2 for details.

Table 2. Duration of gaze gestures on an empty drawing area.

	Size	Average time	Fastest time	Slowest time
L	small	1648 ms	409 ms	3887 ms
	large	1756 ms	797 ms	3548 ms
triangle	small	2559 ms	1236 ms	5860 ms
	large	2843 ms	1325 ms	6976 ms
back and forth line	small	1969 ms	917 ms	4166 ms
	large	2245 ms	996 ms	5561 ms

Following a given model. The time was measured from the moment the model was made visible by the participant (by pressing a key on the keyboard) to the moment when the participant indicated that the task was complete (again by pressing a key on the keyboard). The fastest task was the line task, where the participant only had to fixate on both ends of the line. On the average, the line task was performed in 2463 (small) and in 2506 (large) milliseconds. The rectangle task took 4338 (small) and 4760 (large) milliseconds on the average to perform. Following the sphere of the circle took somewhat longer than the other shapes. On the average, it took 4694 (small) and 6522 (large) milliseconds to follow the circle properly. Although the times for the rectangle task were similar, especially for the fastest participants, the accuracy was poor: the sphere of the circle was hit only 3 (small) and 6 (large) times during the fastest performances. See Table 3 for details of the durations.

The duration of the gestures in all 12 cases is illustrated in Figure 5. In each case, the median value is shown by a thick horizontal bar, the box indicates the interquartile range, and the whiskers show the minimum and maximum values (with outliers excluded).

Table 3. Duration of gaze gestures for following a given model.

	Size	Average time	Fastest time	Slowest time
rectangle	small	4338 ms	2731 ms	10006 ms
	large	4760 ms	2751 ms	9867 ms
circle	small	4694 ms	2272 ms	10663 ms
	large	6522 ms	3090 ms	13434 ms
line	small	2463 ms	977 ms	6617 ms
	large	2506 ms	1096 ms	6657 ms

Speed

Speed was calculated by dividing distance by duration. The durations were measured from the fixation that started the gesture to the fixation that completed the gesture including these fixations. Before calculating the distances, the corner (or the turning point) fixations of each gesture were determined manually by inspecting the shape of the fixation sequence. The distance was measured from the starting fixation to the ending fixation through these corner fixations. Speed was used as a measure, since it is less sensitive on scale than task duration.

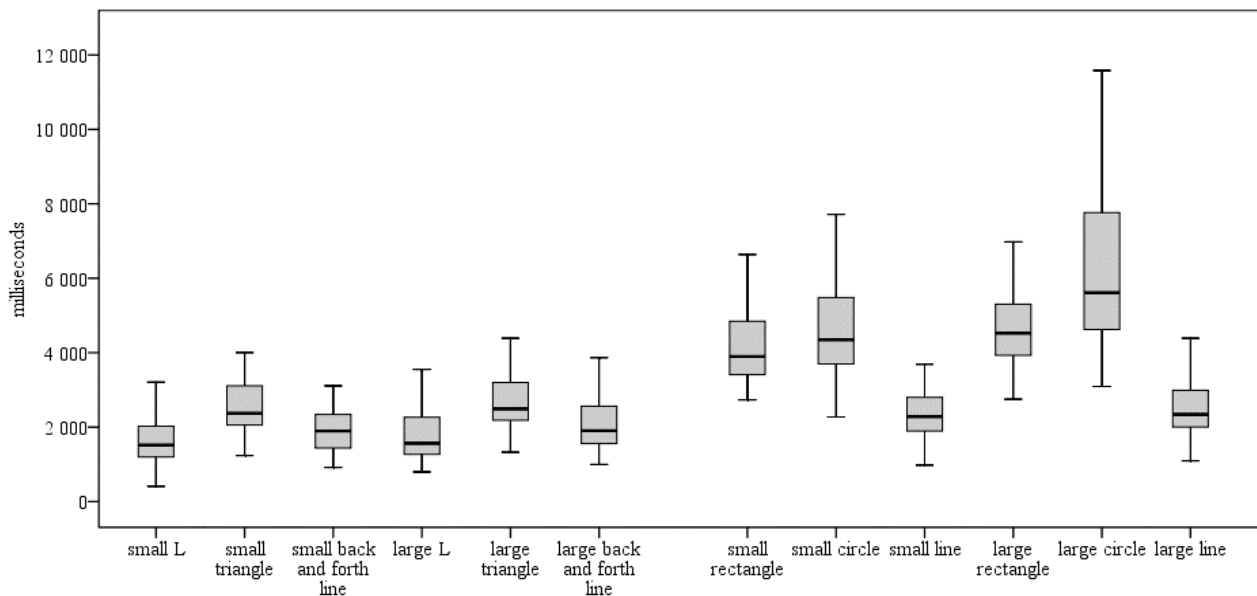


Figure 5. Task durations for making gaze gestures on an empty drawing area and for following a given model.

When speed was measured in pixels per second, faster speeds were obtained in large gestures than in small gestures (see Figure 6). On the average, the speed in large gestures was between 650 and 900 pixels (or 170 and 240 millimetres) per second, whereas in small gestures, the speed was between 150 and 250 pixels (or 40 and 65 millimetres) per second. With our setup, one pixel is equivalent to 0.265 millimetres.

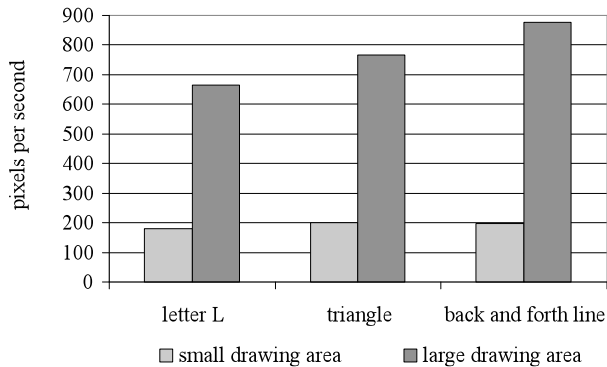


Figure 6. Speed in pixels per second for making gaze gestures on an empty drawing area.

A 3×2 repeated measures ANOVA with gesture type (L, triangle, back and forth line) and gesture size (small, large) as within-subjects factors was performed on average speeds. Greenhouse-Geiser correction was applied when needed. The results showed that the gesture type has a significant effect on speed, $F(1, 21) = 7.67, p < .01$. Also, the gesture size significantly affects speed, $F(1, 15) = 160.97, p < .001$. Moreover, the results showed that there is a significant interaction effect of gesture size and type upon speed as well, $F(2, 30) = 6.73, p < .01$.

As the results showed significant effect of gesture size upon speed, a set of post hoc *t*-tests was done by pairing the small gestures with their large counterparts. All pairwise comparisons of size (i.e. small L vs. large L, small triangle vs. large triangle and small back and forth line vs. large back and forth line) showed that the participants reached significantly faster speeds when making large gestures than making the smaller gestures ($t(15) = 9.98, p < .001, t(15) = 12.61, p < .001, t(15) = 10.00, p < .001$, respectively).

To clarify the effect of gesture type upon speed, another set of post hoc *t*-tests was done by pairing one gesture type with another gesture type (i.e. L vs. triangle, L vs. back and forth line, triangle vs. back and forth line).

The results showed that participants reached significantly faster speeds when making the triangle or the back and forth line gesture than when making the L gesture ($t(15) = 3.17, p < .01, t(15) = 3.80, p < .01$, respectively). When comparing the triangle gesture with the back and forth line gesture, no significant difference was found.

The comparisons between gesture types were also done separately for different gesture sizes and the results revealed that, in large scale, participants reached significantly faster speeds when making the triangle or the back and forth line gesture than when making the L gesture ($t(15) = 2.67, p < .05, t(15) = 3.75, p < .01$, respectively). Other comparisons did not give statistically significant results.

Accuracy

In tasks 4 to 6, the participants were asked to look at the corners of a small and a large rectangle, the ends of a small and a large line and to follow the sphere of a small and a large circle. The distance of gaze from the target was measured in pixels. We used gaze points (not fixations) to determine whether the user hit the target or not. Examples of gaze paths are given in Figures 7–10. For viewing clarity, we have shown as blue circles the fixations (with the radius corresponding to the duration of the fixation), not the raw data points that were used in our analysis. Accuracy distributions are presented in Figure 11 (for rectangle and line tasks) and in Figure 12 (for circle tasks).

Looking at the corners of the rectangle. Only 3 participants hit within 10 pixels on all corners of the small rectangle in every trial and only one of them hit every time closer than 8 pixels. In the large rectangle, only one participant hit closer than 10 pixels in every trial. Most problematic was the lower left corner, where the average distances from the corner were 10.4 pixels (small rectangle) and 11.5 pixels (large rectangle). In other corners, the average distance varied from 4.2 to 8.2 pixels (small rectangle) and from 6.1 to 7.5 pixels (large rectangle).

Looking at the ends of the line. 7 participants hit within 10 pixels on both ends of the small line. Only one of them hit closer than 4 pixels from the end points. 5 participants hit closer than 10 pixels from the ends of the large line. Again, only one participant hit closer than 4 pixels. On the average, the distances from the left and right ends were 5.9 and 6.3 pixels (small line) and 7.7 and 4.2 pixels (large line), respectively.

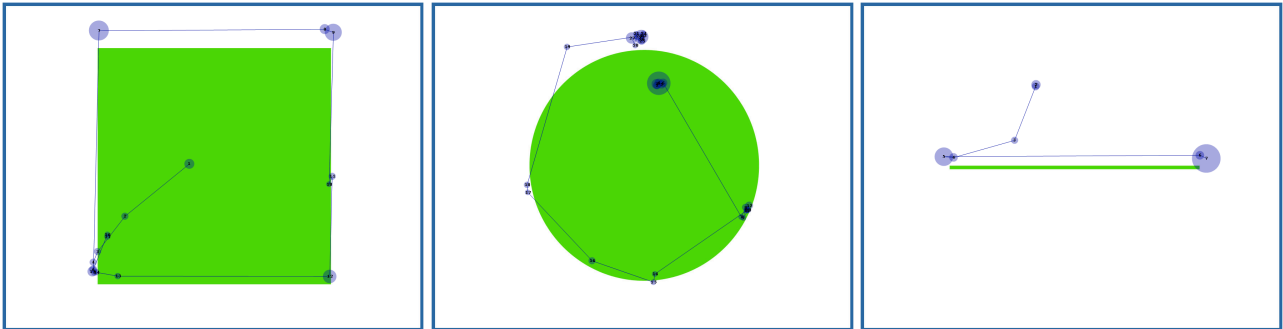


Figure 7. Example gaze paths from very inaccurate participants.

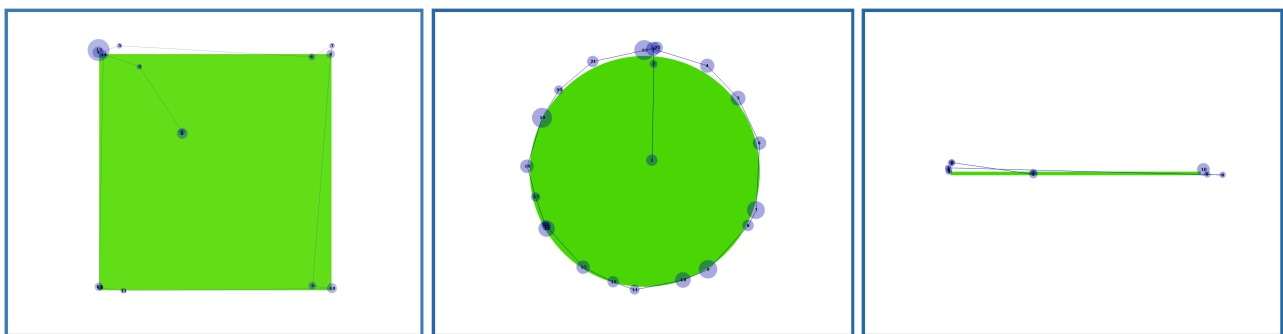


Figure 8. Example gaze paths from very fast and accurate participants.

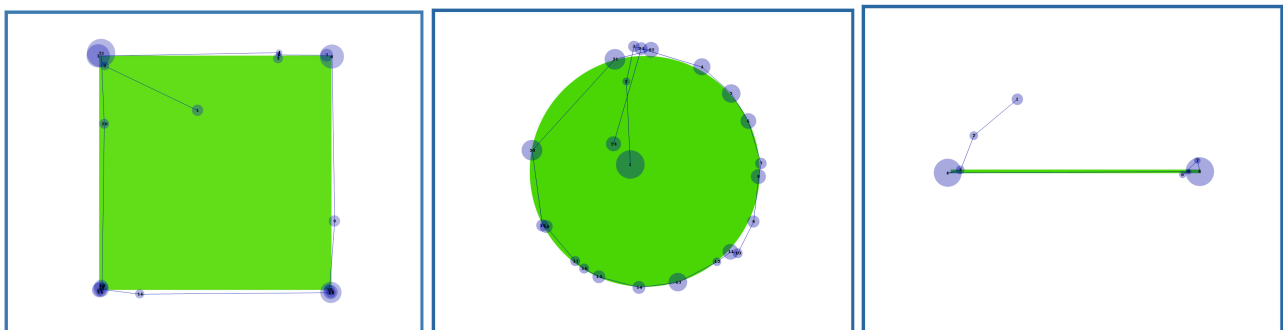


Figure 9. Example gaze paths from accurate but slow participants.

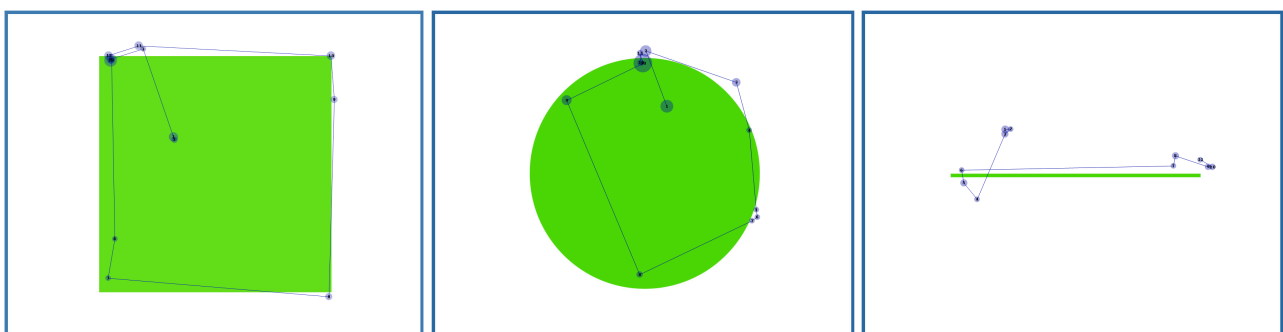


Figure 10. Example gaze paths from fast but inaccurate participants.

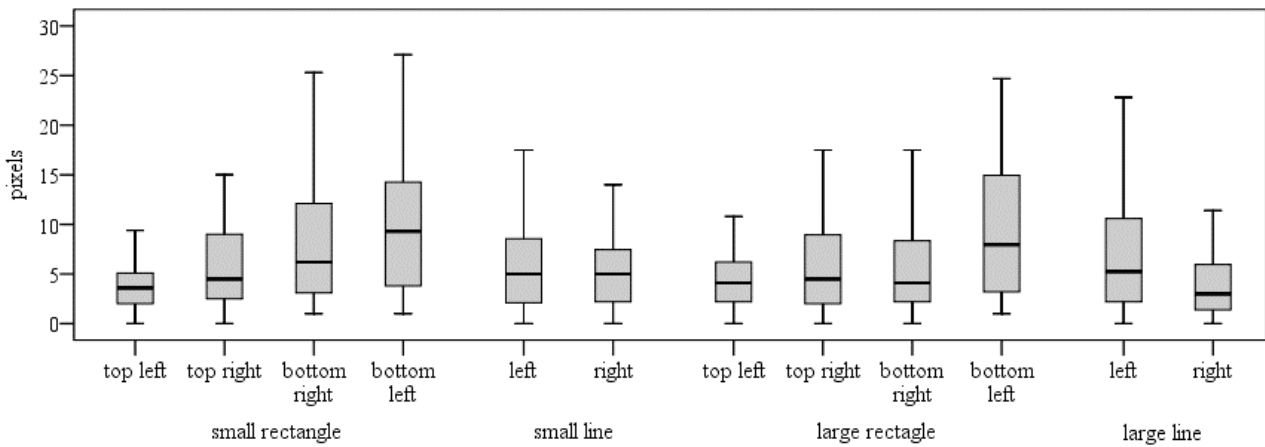


Figure 11. Accuracy in pixels when trying to hit the corners of the rectangles and the ends of the lines.

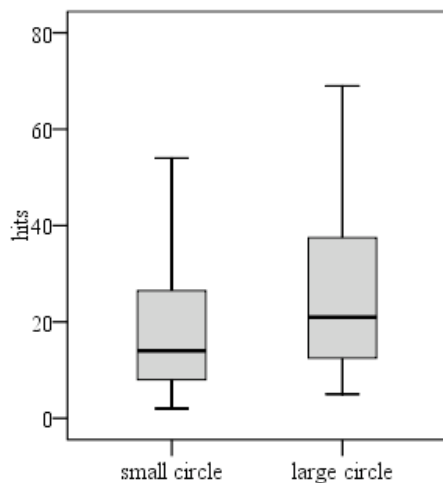


Figure 12. Accuracy in hits showing the hits which were within 2 pixels from the sphere of the circle.

Following the sphere of the circle. Here, a hit means that a gaze point from the participant's data landed within two pixels from the sphere of the circle. The two-pixel interval was used to remove the inaccuracies that result from using the round-shaped drawing. The most precise participant hit the sphere of the small circle 107 times. In the large circle, the number was 95. The most inaccurate participants hit only twice the sphere of the small circle and only 5 times the large circle. On the average, the circles were hit 20 (small circle) and 27 (large circle) times. 3 participants hit at most 20 times the sphere of the large circle. Only 2 participants hit it more than 40 times and one of them hit the sphere over 60 times. The same participant hit

over 30 times the sphere of the small circle. Only three other participants hit the sphere of the small circle more than 10 times.

Discussion

The time needed to make the gestures was a surprise to us. Only the very fastest participants got close to what we expected. We thought that since gaze gestures combine saccades and fixations and since no long fixations are needed, the gaze gestures would be faster than 1000 milliseconds. For example, when a user makes an L gesture, three fixations and two saccades are needed. Then an L gesture should take only about 700 milliseconds, if short fixations are around 200 milliseconds and saccades around 50 milliseconds. The average in our experiment was more than twice as long.

However, we would not yet judge gaze gestures as being slow. Based on the gaze paths we collected from our participants, we believe that our participants did not make the gaze gestures as quickly as they could be done. As an illustration, the fastest way of making an L gesture would be three fixations on the corners of the imaginary L. Our participants, however, did not jump from one fixation to another. Instead, they deliberately moved their eyes along the imaginary line between corners. This way they actually did very short fixations along the way. In other words, they were really drawing the shape with their eyes, instead of simply jump-

ing from one corner to the next. This behaviour was likely caused by our instructions, where we asked the participants to draw certain shapes with their eyes.

Furthermore, in our experiment, the participants did not get any visual feedback from their eye movements. Some participants commented after the test that they would have liked to see the cursor during the experiment. In retrospect, we believe that with visual feedback the participants could perform their tasks faster. Visual feedback could be just a visible cursor. Additionally, a cursor trail showing the last few movements of the cursor could help, since it would show the shape of the gaze gestures while they are made.

We found that the gaze gestures in large scale do not take much longer to perform than the same gaze gestures in small scale. Our statistical analysis showed that the speed becomes significantly faster (that is, longer distance is covered in relatively shorter time) when the gestures are done in large scale. This leads us to the same conclusion that Drewes and Schmidt (2007) found in their study: the time spent on saccades (the time spent when moving from one point to another) does not affect the performance times significantly. Thus, the performance times do not give a reason to favour any size of gaze gestures. The preference may come from the users, since, for example, Drewes and Schmidt (2007) reported that their participants preferred large scale gaze gestures.

Based on the collected gaze paths and the comments from the participants, we assumed that so-called closing gestures (that is, gestures that start and end at the same point, here the triangle and the back and forth line gestures) would be slower to do than the more simple gestures. Nevertheless, when we did statistical analysis on speed, the results showed that the participants actually reached faster speeds when doing these closing gestures. We believe that the difference between observations and statistical analysis originates from the way the data was processed before the statistical analysis. In statistical analysis, we used data that includes only the gaze data from the starting fixation to the ending fixation. The starting fixations were the longer fixations after which the participant started moving towards the next corner of the gesture. The ending fixations were the longer fixations that were assumed to complete the gaze gesture (i.e., in triangle and in back and forth line, the one closest to the start-

ing fixation). Since this process excluded fixations that happened before and after the assumed gaze gesture, in closing gaze gestures this also excluded the cluster of fixations that appeared near the closing point and which can be understood as a search for the starting point, in order to close the gesture at the same point (see Figure 13).

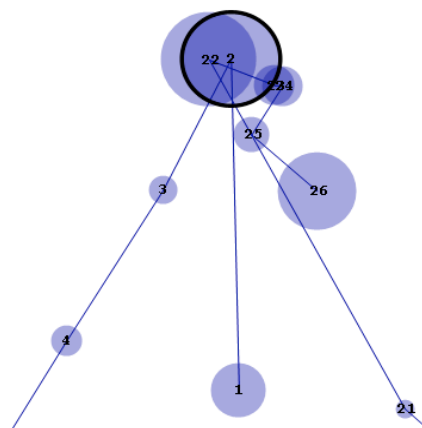


Figure 13. A cluster of fixations (fixation numbers 22-26) has appeared where the participant is trying to end the triangle gesture. The starting fixation (fixation number 2) is highlighted with a bold border.

To conclude the issue of closing gaze gestures, more evidence is needed. A follow-up experiment would show if there is a real difference in performance between open and closing gaze gestures. As an improvement to our experiment, the follow-up experiment should use an algorithm (not manual inspection) to determine whether a gaze gesture has been issued or not and the participants should get some feedback from their eye movements.

Tchalenko (2001) has found that curved shapes are difficult to draw with gaze, because drawing a curve with gaze requires short saccades and fixations near each other. These kind of small adjustments are difficult for eyes. Nevertheless, Tchalenko found that some people are able to follow an existing curve quite well. Similarly, some of our participants managed to follow the sphere of the circle accurately. However, it is very time-consuming and requires a lot of concentration.

When participants were asked to look at the corners of the rectangle and each end of the line, they had trouble to hit the target area. The results were not en-

couraging when, for example, only one participant managed to hit the corners of the large rectangle closer than 10 pixels in every trial. Nevertheless, we believe that the lack of feedback strongly effected on these results. As noted before, the participants did not get any feedback from the whereabouts of their gaze on the screen. A simple cursor showing the gaze point probably would have given us much better results. Another factor that needs to be considered is the effect of the particular application. The density of objects on the screen, or the size of the gesture vocabulary, may allow for less accurate gestures than we aimed at in our experiment.

In addition, we learnt that the users are inconsistent with the drawing direction when issuing a gaze gesture if the direction is not defined. In our experiment, we had three gestures that were possible to issue either clockwise or counterclockwise. The preference between clockwise and counterclockwise varied between participants and also between the participant's repetitions.

Based on the experiences we gained during the experiment, we can draw some guidelines for designing gaze gestures. First, if possible, the gaze gestures should be flexible and scalable, so that the users could issue them in any size and in any place they want. Second, the users need feedback from their eye movements, especially novices. Furthermore, the problems with accuracy and with curved shapes need to be taken into account especially when designing a drawing application.

Conclusions

We presented an experiment that studied gaze gestures. We found that the gaze gestures were not as fast as expected. When comparing small and large gaze gestures, there was only a little difference in performance times. We proposed some improvements that could make the performance faster. One major improvement would be adding feedback. For example, accuracy would improve if the user would see where the application thinks the eyes are. This, however, needs further testing; it is well known that an eye cursor that is off the target because of poor calibration can be misleading, because the user can start to chase the cursor instead of focusing on the target.

Our next step is to design and create a gaze gesture collection suitable for a drawing application. The results from our experiment give us a good starting point. Eventually, the gaze gestures will be tested in their real environment.

Acknowledgements

We thank Saila Ovaska for her help in improving this article and the anonymous reviewers for their useful comments. We also thank Jenni Anttonen for her help with SPSS and the statistical analysis. This research was supported by the University of Tampere.

References

- Abrams, R. A., Meyer, D. E., & Kornblum, S. (1989). Speed and accuracy of saccadic eye movements: Characteristics of impulse variability in the oculomotor system. *Journal of Experimental Psychology: Human Perception and Performance* 15, 3, pp. 529–543.
- Bee, N. & André, E. (2008). Writing with your eye: a dwell time free writing system adapted to the nature of human eye gaze. In *Lecture Notes in Computer Science*, 5078. Springer Berlin/Heidelberg, pp. 111–122.
- Drewes, H. & Schmidt, A. (2007). Interacting with the computer using gaze gestures. In *Proceedings of INTERACT 2007. Lecture Notes in Computer Science* 4663, Springer, pp. 475–488.
- Hansen, D. W., Skovsgaard, H. H., Hansen, J. P., & Møllenbach, E. (2008). Noise tolerant selection by gaze-controlled pan and zoom in 3D. In *Proceedings of the 2008 Symposium on Eye Tracking Research & Applications (ETRA '08)*. ACM Press, New York, NY, pp. 205–212.
- Huckauf, A. & Urbina, M. H. (2008). Gazing with pEYEs: towards a universal input for various applications. In *Proceedings of the 2008 Symposium on Eye Tracking Research & Applications (ETRA '08)*. ACM Press, New York, NY, pp. 51–54.
- Isokoski, P. (2000). Text input methods for eye trackers using off-screen targets. In *Proceedings of the 2000 Symposium on Eye Tracking Research & Applications (ETRA '00)*. ACM Press, New York, NY, pp. 15–21.

- Istance, H., Bates, R., Hyrskykari, A., & Vickers, S. (2008). Snap clutch, a moded approach to solving the Midas touch problem. In *Proceedings of the 2008 Symposium on Eye Tracking Research & Applications (ETRA '08)*. ACM Press, New York, NY, pp. 221–228.
- Jacob, R. J. K. (1991). The use of eye movements in human-computer interaction techniques: What you look at is what you get. *ACM Transactions on Information Systems* 9, 2, pp. 152–169.
- Kahn, D. A., Heynen, J., & Snuggs, G. L. (1999). Eye-controlled computing: The VisionKey experience. In *Proceedings of the Fourteenth International Conference on Technology and Persons with Disabilities (CSUN'99)*, Los Angeles, CA. Available at <http://www.csun.edu/cod/conf/1999/proceedings/session0263.htm>
- Møllenbach, E., Hansen, J. P., Lillholm, M., & Gale, A. G. (2009). Single stroke gaze gestures. In *Proceedings of the 27th international Conference Extended Abstracts on Human Factors in Computing Systems (CHI '09)*. ACM, New York, NY, pp. 4555–4560.
- Perlin, K. (1998). Quikwriting: continuous stylus-based text entry. In *Proceedings of the 11th annual ACM symposium on User Interface Software and Technology (UIST'98)*. ACM Press, New York, NY, pp. 215–216.
- Porta, M. & Turina, M. (2008). Eye-S: a full-screen input modality for pure eye-based communication. In *Proceedings of the 2008 Symposium on Eye Tracking Research & Applications (ETRA '08)*. ACM Press, New York, NY, pp. 27–34.
- Rayner, K. (1998). Eye movements in reading and information processing: 20 years of research. *Psychological Bulletin* 124, 3, pp. 372–422.
- Tchalenko, J. (2001). Free-eye drawing. *Point: Art and Design Research Journal*, 11, pp. 36–41.
- Urbina, M. H. & Huckauf, A. (2007). Dwell-time free eye typing approaches. In *Proceedings of the 3rd Conference on Communication by Gaze Interaction: Gaze-based Creativity and Interacting with Games and On-line Communities (COGAIN 2007)*, Leicester, UK, pp. 65–70.
- Vickers, S., Istance, H., Hyrskykari, A., Ali, N., & Bates, R. (2008). Keeping an eye on the game: eye gaze interaction with massively multiplayer online games and virtual communities for motor impaired users. In *Proceedings of the 7th International Conference on Disability, Virtual Reality and Associated Technologies (ICDVRAT 2008)*. University of Reading, UK (ISBN 07-049-15-00-6).
- Vickers, S., Istance, H., & Hyrskykari, A. (2009). Selecting commands in 3D game environments by gaze gestures. In *Proceedings of the 5th Conference on Communication by Gaze Interaction: Gaze Interaction For Those Who Want It Most (COGAIN 2009)*, Copenhagen, Denmark.
- Ward, D. J., Blackwell, A. F., & MacKay, D. J. (2000). Dasher – a data entry interface using continuous gestures and language models. In *Proceedings of the 13th Annual ACM Symposium on User interface Software and Technology (UIST '00)*. ACM, New York, NY, pp. 129–137.
- Wobbrock, J. O., Myers, B. A., & Kembel, J. A. (2003). EdgeWrite: A stylus-based text entry method designed for high accuracy and stability of motion. In *Proceedings of the ACM Symposium on User Interface Software and Technology (UIST '03)*. ACM, New York, NY, pp. 61–70.
- Wobbrock, J. O., Rubinstein, J., Sawyer, M. W., & Duchowski, A. T. (2008). Longitudinal evaluation of discrete consecutive gaze gestures for text entry. In *Proceedings of the 2008 Symposium on Eye Tracking Research & Applications (ETRA '08)*. ACM Press, New York, pp. 11–18