

Writing with Your Eye: A Dwell Time Free Writing System Adapted to the Nature of Human Eye Gaze

Nikolaus Bee and Elisabeth André

Institute of Computer Science, University of Augsburg,
86135 Augsburg, Germany
{bee, andre}@informatik.uni-augsburg.de

Abstract. We investigate the usability of an eye controlled writing interface that matches the nature of human eye gaze, which always moves and is not immediately able to trigger the selection of a button. Such an interface allows the eye continuously to move and it is not necessary to dwell upon a specific position to trigger a command. We classify writing into three categories (typing, gesturing, and continuous writing) and explain why continuous writing comes closest to the nature of human eye gaze. We propose Quikwriting, which was originally designed for handhelds, as a method for text input that meets the requirements of eye gaze controlled input best. We adapt its design for the usage with eye gaze. Based on the results of a first study, we formulate some guidelines for the design of future Quikwriting-based eye gaze controlled applications.

1 Introduction

As new video-based methods improve contact-free eye tracking and first systems are emerging that can be used with common webcams, the interest in gaze-based interaction increases. Eye tracking systems based on webcams, such as Opengazer¹, have become affordable and thus can be used by the masses in everyday human-computer interaction.

One domain for gaze-based interaction, mainly of interest to physically handicapped people, is gaze-based writing. We will introduce a gaze-controlled input system that enables users to write with their eyes. A human's eye continuously gazes, wanders and normally stops only for a fraction of a second. Common gaze-based writing systems force the users to dwell upon a specific position for a certain time to trigger a command. In contrast to this, we will present a dwell-time free gaze-controlled writing system which matches the nature of human eye gaze in a better manner.

Unfortunately, eye gaze as an input method is more problematic than hands and fingers. The eye is normally used to gain information only and not to trigger any commands or control devices. [Ashmore et al., 2005] summarizes four

¹ More information about this open source eye tracker and a video can be found under: <http://www.inference.phy.cam.ac.uk/opengazer/>

problems which interface developers should consider when using eye gaze for human-computer interaction. (1) The accuracy of eye tracking devices is limited to about $0.5\text{-}1^\circ$ visual angle. 1° corresponds approximately to the size of a thumbnail at arm length [Duchowski, 2007]. This restricts the interaction elements in an interface to a certain size. (2) Eye gaze is recognized with a delay dependent on the frame rate. A 50 Hz system, for instance, incurs delays of 20 ms. When using webcams with 25 Hz, the delay would be 40 ms. (3) Eye gaze is never perfectly still even if one concentrates on a point. It slightly jitters with flicks less than 1° , small drifts of about $0.1^\circ/\text{s}$ and tremors (tiny, high frequency eye vibrations). (4) The Midas touch problem [Jacob, 1991] leads to ambiguities in gaze-controlled interaction. Eyes are used for seeing and gathering information. Naturally, they follow any salient or moving point of interest. Interfaces that use eye gaze should thus be carefully designed and not use too many or intrusive elements that could attract the attention of the user's eye gaze.

We will first give an overview of existing writing systems that are based on eye gaze control. Three kinds of writing approach can be distinguished: typing, gesturing and continuous writing. We will describe our interface that considers the specific characteristics of human eye gaze, i.e. always moving and not able to press something. An experiment to measure the performance and usability of our new input method and its results follow.

2 Related Work

Three types of writing can be distinguished: typing, gesturing and continuous writing.

1. *Typing* – can be performed by pressing keys on a physical keyboard or by selecting letters on an onscreen keyboard using, for example, a game controller.
2. *Gesturing* – is comparable to block lettering. Gesture-based input methods may be found in handhelds with pen-based input devices, such as the Graffiti system from Palm. Each input of a letter is separated by a short interruption (lifting of the pen).
3. *Continuous writing* – reduces the interruptions between letters to a minimum. Cursive handwriting comes close to it.

Taking the humans' eye gaze behavior into consideration, *continuous writing* matches best the requirements of interfaces that utilize eye gaze for input control. Human eye gaze is always moving and always 'on', which can be seen as a pen that always draws. We cannot simply switch off our eye gaze. For handwriting it is necessary to lift the pen to separate single words. As we cannot switch off eye gaze, it would be fascinating to design a gaze-driven interface, where switching is not necessary at all. Before presenting our own system, we will discuss applications that were developed for gaze-controlled writing that fall into the categories introduced above.

2.1 Gaze-Controlled Typing

There are two kinds of common keyboard writing. In the case of direct writing, a keyboard with the keys arranged in alphabetical order or using a typical keyboard layout is displayed. Unlike this, multi-tap writing is based on a hierarchical arrangement of letters. Here the letters are grouped on keys together and users have to repeatedly press buttons to get a single letter. This method is frequently used in mobile phones. To adapt common keyboard writing for eye gaze writing, users have to directly select the keys with their gaze, in a similar way as they would type on a keyboard. They simply type with their eye. Both gaze-controlled direct writing and gaze-controlled multi-tap use dwell time, i.e. users have to fixate a specific point for a specific time, to trigger a key.

[Majaranta et al., 2006] used gaze-controlled interfaces for writing. To write letters, users simply must look at the on-screen button for a certain amount of time. As users write with their eyes in a contact-free manner, there is at first no haptic or acoustic feedback which they might be familiar with from typewriters or keyboards. [Majaranta et al., 2006] investigated several kinds of typing feedback for users: visual, speech and click noise. The visual feedback was implemented with a small highlighted border around the key which is displayed when the user looks at the key. Furthermore, the size of the character on the key shrinks linear to the dwell time. On selection, the background of the key changes its color and the key goes down. When speech is used, the letter is simply spoken out as feedback after its selection. The click noise as feedback is self-explanatory. The authors found in their comparison of speech only, click + visual, speech + visual, and visual only, that click + visual enabled the users to write fastest with their eyes. The maximum writing speed they achieved was about 7.5 wpm (words per minute).

[Hansen et al., 2001] developed a writing system with a hierarchical structure called GazeTalk. They reduced the approximately 30 on-screen keys, common for gaze-controlled keyboard-based systems, to ten in their gaze-based multi-tap system. They applied two different methods. The version for novice users arranges the letters alphabetically. First, letters and special characters are grouped on four buttons. After selecting a button the single characters are shown on single buttons and can be selected for writing. Whereas a gaze-controlled system can select a letter with a single step, this system needs two. The version for advanced users automatically predicts the next letter while the user is writing. In prediction mode only the six most likely letters are shown. If the desired letter is not among them, the user must trigger another button to get back to the alphabetical display.

Among systems without any probabilistic letter prediction or word completion, gaze-controlled keyboard-based systems are the fastest as it takes only one step to enter a letter. But such systems must display all letters at once on the screen. The alphabet has 26 letters adding some command buttons (i.e. space, delete etc.), space for at least 30 buttons is required. Depending on the accuracy of the eye tracker, the buttons need a certain size. And in the end, the writing interface will need a lot of space. With multi-tap systems, the buttons can

be larger and are therefore less vulnerable to inaccuracies of the eye tracking system.

Both the direct writing and the multi-tap approach use dwell time to trigger keys. Dwell time strongly depends on the experience of users and thus has an impact on typing speed and error rate. If the chosen dwell time is too short, users will make more mistakes and if the dwell time is too long, users will strain their eyes. Thus, a reasonable trade-off between typing speed and error rate needs to be found. [Špakov and Miniotas, 2004] developed an algorithm to adjust dwell time in real-time. They found that a dwell time of 700 ms enables the user to type nearly without any wrongly selected keys. [Hansen et al., 2001] used a dwell time of 750 ms for novice users which they decreased after several hours of usage to 500 ms.

2.2 Gaze-Controlled Gesturing

[Isokoski, 2000] developed a system called MDITIM (Minimal Device Independent Text Input Method) for device-independent text input. Originally it was only tested with a touchpad, a trackball, a computer mouse, a game controller and a keyboard. To adopt it for eye gaze control, practically no changes were necessary. Only a modifier key, which was previously controlled by pressing a button, was replaced by an area-of-interest, where users had to look at to trigger it. MDITIM encodes letters in commands of directions, i.e. a = NSW, b = SEW, c = ESW, d = SWE, and so forth. The codes consist of three or four directions. If a user, for instance, wishes to write 'c', her eyes have to go to the right, then down and finally to the left. Writing 'cab' results in ESWNSWSEW, which comes close to a continuous writing system. Nevertheless, MDITIM is not a real continuous writing system. For example, combinations, such as 'dc', are encoded by SWEESW, which includes two equal codes in a row. If the system shall be able to recognize such a combination, there must be an interruption in between.

EyeWrite developed by [Wobbrock et al., 2008] is a pure gesture-based eye writing system similar to Graffiti for Palm handhelds. It is the first system that uses letter-like gestures for eye input, in contrast to MDITIM which encodes the alphabet. The interface – the authors chose a window size of 400×400 – is aware of five areas: the four corners and the middle. To provide some guidance for the eye gaze to write gestures, there are points placed in the corners and in the middle. To write a 't' for example, the gaze must move from the upper left corner to the upper right corner, then to the lower right corner and finally to the middle to indicate that the gesture is terminated. Glancing at the corners suffices to draw the gesture. The system works not completely dwell time free as the eye gaze must stay for a specified time in the middle for segmentation. The authors specified a dwell time of about 250 ms, which corresponds to half of the typical dwell time that systems use or to twice as much as the average fixation time. The usage of this system is rather similar to MDITIM, but with letter-like gestures, it is easier for users to remember the gestures.

A disadvantage of gesture-based typing systems is that users have to learn the gestures by heart or look it up. That makes the systems difficult to use for occasional users. EyeWrite could be easier to use than MDITIM as EyeWrite uses letter-like gestures, which makes the gestures easier to memorize. The authors of MDITIM do not provide a user study with performance measurements. [Wobbrock et al., 2008] conducted a longitudinal study about the performance of their system. Novice users wrote from about 2.5 wpm (words per minute) up to about 5 wpm after 14 sessions.

2.3 Gaze-Controlled Continuous Writing

[Urbina and Huckauf, 2007] introduce three dwell-time free eye writing applications. In Iwrite, keys are arranged alphabetically in a rectangular horseshoe shape. To select a letter users must look at the letter and then look outside the shape. The inner area of the horseshoe displays the currently written text. A similar system called StarWrite arranges the letters on a half-circle in the upper part and a display for the written text in the lower part. Looking at a letter enlarges it and its two neighbors. To select a letter one must then 'drag' it to the lower text field. Again all letters must be displayed at once. Thus, this method is space consuming or vulnerable to inaccuracy of eye tracking.

pEYEWrite [Huckauf and Urbina, 2007] is their third concept of dwell-time free writing. Here letters are arranged hierarchically using pie menus with six sections, where each section groups five letters or special characters. Letters are again arranged alphabetically. The pie is further divided into an inner and an outer part. The letters are displayed in the inner part of the pie and to trigger a selection of a letter, users must gaze at the corresponding section on the outer frame. To write a letter, a user first selects the section that contains the intended letter. After that, a new pie menu pops up that contains one letter in the single sections. After the selection, the pie disappears and the user can continue. This system needs two activations to write a letter.

Maybe the most prominent gaze-controlled text entry system for continuous writing is Dasher [Ward and MacKay, 2002]. It does not use any static elements in its design. Letters move from the right to the left and as soon as a letter crosses a border it is selected. The letters move continuously as long as the users look at the letters. At start the letters are arranged vertically on the very right border of the application. As soon as the user looks at a letter the letter starts to enlarge and moves to the left. Dasher uses probabilistic prediction of letters and word completion. Both concepts are seamlessly integrated in the interface. The probability of a letter is directly depicted by its size which facilitates its selection.

[Isokoski, 2000] describes an adaption of Quikwriting for usage with eye gaze. Quikwriting was developed by [Perlin, 1998] as a new text input interface for stylus-based systems, i.e. handhelds or smartphones. The system is based on two input concepts. First, with Quikwriting users must never lift their stylus from the surface. This approach perfectly matches the nature of human eye gaze. The eye is always gazing at something, e.g. the screen, and we cannot

'lift' our eye gaze from the screen unless we close our eyes. But then we can no longer see anything. Lifting, we better say triggering or switching, is not a natural human eye gaze behavior. Second, the user never has to stop moving the stylus. Of course eyes stop often to fixate something in a scene, but these fixations normally just last around 150-300 ms. This is much shorter than the trigger time in a dwell-time system. As soon as the dwell time is equal to fixation time, everything users look at is selected.

The interface of Quikwriting is divided into eight equally sized sections around a central resting area. To write a letter, the user moves from the center to one of the outer sections, optionally to another adjacent section, and back to the center, which triggers the selection of the letter. Every section is linked to a group of letters. In general, the letters are arranged in such a way that frequent characters can be written faster. Thus training speeds up writing since users familiar with the arrangement would be able to find an intended letter faster than novice users. For instance, one section contains 'h', 'e', and 'c' in this order. To write an 'e', users simply move their stylus to this section and back to the resting area. If they want to write the 'h', they move to this section, after that to the adjacent left section and finally directly back to the central area. [Isokoski, 2000] never seemed to have implemented this system and thus results about its usability are not available.

3 Implementation of Eye Writing Applications

Our objective is to develop a new gaze-controlled dwell-time free system for continuous writing. We hope that such a system would come close to the nature of human eye gaze behavior. Based on earlier research, we will concentrate on an interface design which does not require learning any gestures by heart. Taking the Midas touch problem [Jacob, 1991] into account, our interface shall be comfortable for the eye. Therefore, distracting visual feedback should be handled with care. Taking these requirements into account, we decided to explore the potential of Quikwriting, which was designed for the usage with handhelds in the first place, for a gaze-controlled interface (see Figure 1). In addition, we will provide a comparison with a gaze-controlled keyboard-based system.

3.1 Gaze-Controlled Quikwriting

Two problems occur with the original design of Quikwriting when simply substituting the stylus by eye gaze. When interacting with a stylus, users first search for the section that contains the letter to write and then they move the stylus there to select it. To avoid that the selection of a letter is unintentionally triggered by a visual search process, we decided not to display the letters within the original sections. Instead, we displayed them in the inner resting area, each group of letters close to its linked section. Further, we had to help users memorize which adjacent section is linked to which single letter since users might already forget during the interaction which section they have to gaze at. This would be

fatal for our system. Imagine that the user has selected a group of letters by moving his eye gaze out of the center to an outer section and now has to gaze at an adjacent section to trigger the selection of a letter. Let us assume that the user forgot whether the first or the second adjacent section is the correct one. Thus his gaze would move back again to the resting area. But this process already triggers the selection of a letter. To avoid this problem, we display each single letter within the section the user has to gaze at in order write a letter after having selected a group of letters by moving his eye gaze out of the center (see Figure 1). Then he can look at the section for the letter to write and gaze back to the center.

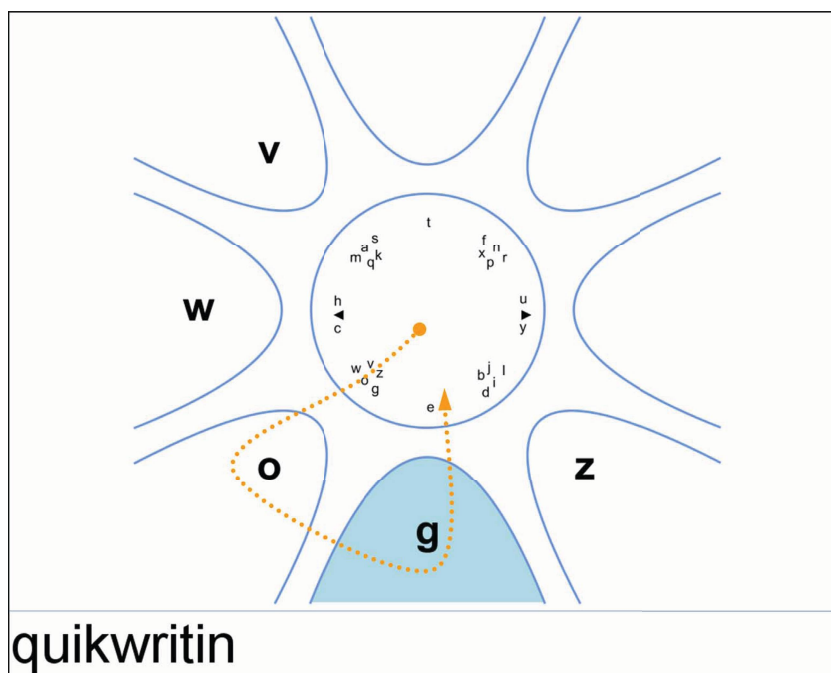


Fig. 1. Adapted interface of Quikwriting for use with eye gaze. The dotted line indicates the gaze path to write the letter 'g'. The user currently looks at 'g'. The light shaded background follows the gaze and indicates which section the user is looking at.

Another problem occurs when a user wants to check what she or he already wrote. For instance, if we place the text area below the writing interface, users may gaze from the center at the text area and pass writing sensitive sections. Looking back to the center would result into the selection of a letter. To avoid this kind of unintended writing actions, we check whether the user is looking at the text field and disable any writing function until the gaze is back in the central area.

3.2 Gaze-Controlled Keyboard-Based System as a Baseline

The performance of writing systems distinguishes a lot in the literature. Some authors measure in characters per minute and others measure in words per minute. Sometimes, the authors do not even describe how many characters make up their words. Most literature considers for European languages a sequence of five characters including spaces and punctuation marks for one word. [MacKenzie, 2003] describe in detail how to measure writing speed in theory and practise for various kinds of interaction device. Not only the measurement methods differ frequently, also the writing speed of gaze-controlled keyboard-based systems, for instance, ranges from about 5 wpm [Wobbrock et al., 2008] to about 11 wpm [Urbina and Huckauf, 2007] for novice users. [Majaranta et al., 2004] report that even among subjects the speed for gaze-controlled writing varies from 7 wpm to over 14 wpm. The huge variance could come from the different eye tracking systems and their accuracy, tracking capabilities and delay. Further, it is important to know if only correctly written letters or all written letters are taken into account. Including wrongly written letters in our analysis, would falsify our results, as they might have been written randomly and unwillingly.

This all makes it difficult for us to compare the performance of our newly developed application with results in the literature. As among systems without word completion, keyboard-based systems are the fastest way to write with eye gaze control, we decided to implement such a system. This gives us a trustier way to compare writing speeds. Our implementation of a keyboard-based system used a dwell time of 750 ms as our study only included novice users that never used an eye tracking system before. The system's response during the writing process was limited to visual feedback. Users were allowed to interrupt looking at a key to trigger it. Every key has its own dwell time buffer. As soon as a dwell time of one key exceeds, all other buffers are reset. This gives users the freedom to look at the already written text while writing a letter. Also for eye tracking systems with lower accuracy, users won't get easily annoyed if the gaze leaves the key for a glance and the dwell time is reset. While looking at one key, the background color changes and slowly fills the button from inside. The letters were arranged in alphabetical order. Additionally we had three command buttons for space, delete and enter.

4 Experiment

To investigate whether the new interface is usable and whether the writing speed can compete with a gaze-controlled keyboard-based interface, we conducted a study with 3 subjects.

We used the iView X RED eye tracking system from SensoMotoric Instruments (SMI), which is contact-free. The eye tracker operates with a sampling rate of 50 Hz and is used in combination with a 19-inch screen with a resolution of 1280×1024 . When a subject is placed in front of the eye tracking system, the system automatically recognises the position of the head and gives hints about

its best position. While the subjects' eye gaze is tracked, they are allowed to move their head freely to a certain extent.

After the subjects were placed in front of the eye tracker, we gave them a short introduction about the eye tracking system. We explained them how they would use their eye gaze to write. Before we started the study, we gave them about 5 minutes to get used to the applications, as our subjects never used an eye tracker before. This was to ensure that the subjects were able to work with our eye tracking system and the tracking accuracy was high enough. We prepared 30 short sentences on small index cards, which were shown and read to them before they started to write a sentence. Every subject had to use the adapted Quikwriting system and the keyboard-based system. Per application they had to write 10 sentences. The sentences were selected randomly per subject. We showed the cards to the subjects to avoid misspelling which would have a side effect on the analyzed error rate. The applications logged the users' writing interactions with a time stamp into a file.

5 Results and Discussion

We analysed the log files and removed all unwillingly and wrongly written letters. This was necessary as sometimes the users wrote letters although they did not intend to do so. And as we were only interested in the writing speed of correctly written letters, the wrong ones were removed. Removing the wrongly written letters normally worsens the writing speed. Writing unwillingly letters often occurs randomly and unexpectedly and therefore the selection of such letters takes a shorter time. We observed (see Table 1) that users were able to write with our adapted Quikwriting 5 wpm. The same subjects achieved a writing speed of about 8 wpm with the keyboard-based interface.

Table 1. Comparison of writing speeds in wpm (words per minute)

	eyeKeyboard	Quikwriting
avg	7.8	5.0
var	0.02	0.3

For all users the keyboard-based system was easier to use, but more exhausting. This was surprising to us since the users were familiar with keyboards, while the adapted Quikwriting was new to them and their usage had to be learned. The reason for the keyboard-based system to be exhausting is the Midas Touch problem. The users always feared that something happened, when they looked somewhere. Some asked if there is a place where nothing happens and they can rest their eyes. In the adapted Quikwriting we have automatically a resting area in the middle of the interface. On the other side the keyboard-based system could be speed up by reducing our current dwell time of 750 ms. For instance,

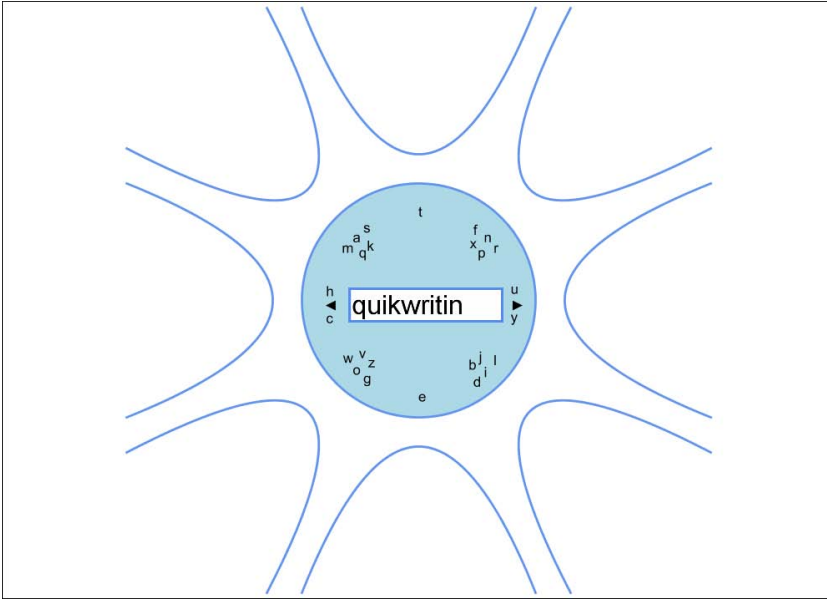


Fig. 2. In a new design of the adapted Quikwriting, we will place the text field in the resting area. This will enable users to check what they have written without moving their eyes to any place outside the control interface.

with a dwell time of 500 ms the writing speed would increase to about 9 wpm. A significant improvement by training is possible since letters are arranged dependent on their probability. We expect that an experienced (i.e. training of 5 days with 15 minutes each day) user could achieve an increased writing speed with Quikwriting up to 9.5 wpm, which comes close to that of a keyboard-based layout.

The error rate of the Quikwriting based system was much higher than the error rate of the keyboard based system. After analysing the log data we found that errors mainly occurred when the user checked what he or she already had already written. We further found that the blocking of any interaction after the users looked at the text field with the already written text did not work in a satisfying manner. Obviously, users did not focus on the written text, as a half glance already sufficed for a human eye to recognise what was written. Therefore, we intend to change the layout of the interface. In particular, we plan to place the text field in a next step to the middle (see Figure 2). This will match the layout of Quikwriting as the eye gaze starts and stops in the middle resting area. Another point is the arrangement of the letters. We kept the layout of Quikwriting as it was originally designed by [Perlin, 1998]. This includes the probability distribution of the characters in the English language. The first six most probable letters in English are E, T, A, O, I, and N, whereas in German they are E, N, I, S, R, and A.

The keyboard-based system could be improved by providing audio feedback to the users after a letter was written. Indeed, subjects already looked at the next letter although the current one was not yet written. Also the dwell time should be adjusted to the users, as for some of them staring for 750 ms was too long.

Comparison With Other Input Methods. The speed of handwriting is about 30 wpm. An average typist writes with 50 to 70 wpm on a computer keyboard. Of course eye gaze controlled writing systems cannot compete with these input methods, but users would need their hands. An input method that needs hands as well uses game controllers. Normally the controller is used to select the letters on a keyboard-based interface displayed on the screen, similar to the one we described in section 2.1. [Költringer et al., 2007] conducted a study with common game controllers. They found that the writing speed of novice users is about 8 wpm using such devices. Experienced users after 15 sessions were able to write 12 wpm. The speed of this input method is comparable to that of eye gaze controlled systems. Since text input becomes more and more important for gaming consoles, eye gaze based input methods will become interesting as soon as webcam-based eye tracking systems become more accurate and reliable.

6 Conclusion

Eye gaze interaction has the potential of becoming a new form of interaction in human-computer interfaces. Currently the interest in such interaction systems mainly comes from physically handicapped people that, for instance, cannot keep their hands calm or move them at all.

We developed a new writing system for eye gaze controlled interaction. Our very first prototype can easily compete with gaze-controlled keyboard-based systems. As we were testing the system for the German language, we expect an improvement after we place the letters according to the occurrence probability of German letters. And with moving the text field from the bottom to the center, a more continuous flow of writing should become possible and increase writing speed.

Based on the results of a first user study, we formulated some guidelines for the design of future Quikwriting-based eye gaze control. Quikwriting was originally designed for the usage with stylus-based input devices. The underlining principles of Quikwriting, (1) always move and (2) never lift the stylus, perfectly match the nature of human's eye gaze and should be considered in future designs for eye gaze interaction. We were able to show that such a system can compete with common writing systems without word completion for eye gaze, such as GazeTalk or pEYEWrite.

References

- [Ashmore et al., 2005] Ashmore, M., Duchowski, A.T., Shoemaker, G.: Efficient eye pointing with a fisheye lens. In: GI 2005: Proceedings of Graphics Interface 2005, pp. 203–210. CHCCS (2005)

- [Duchowski, 2007] Duchowski, A.T.: *Eye Tracking Methodology: Theory and Practice*. Springer, Heidelberg (2007)
- [Hansen et al., 2001] Hansen, J.P., Hansen, D.W., Johansen, A.S.: *Bringing Gaze-based Interaction Back to Basics*. Systems, Social and Internationalization Design Aspects of Human-computer Interaction (2001)
- [Huckauf and Urbina, 2007] Huckauf, A., Urbina, M.: Gazing with pEYE: new concepts in eye typing. In: *APGV 2007: Proceedings of the 4th symposium on Applied perception in graphics and visualization*, pp. 141–141. ACM, New York (2007)
- [Isokoski, 2000] Isokoski, P.: Text input methods for eye trackers using off-screen targets. In: *ETRA 2000: Proceedings of the 2000 symposium on Eye tracking research & applications*, pp. 15–21. ACM, New York (2000)
- [Jacob, 1991] Jacob, R.J.K.: The use of eye movements in human-computer interaction techniques: what you look at is what you get. *ACM Transactions on Information Systems* 9(2), 152–169 (1991)
- [Költringer et al., 2007] Költringer, T., Van, M.N., Grechenig, T.: Game controller text entry with alphabetic and multi-tap selection keyboards. In: *CHI 2007: CHI 2007 extended abstracts on Human factors in computing systems*, pp. 2513–2518. ACM, New York (2007)
- [MacKenzie, 2003] MacKenzie, S.I.: Motor behaviour models for human-computer interaction. In: Carroll, J.M. (ed.) *HCI Models, Theories, and Frameworks: Toward a Multidisciplinary Science* (2003)
- [Majaranta et al., 2004] Majaranta, P., Aula, A., Rähkä, K.-J.: Effects of feedback on eye typing with a short dwell time. In: *ETRA 2004: Proceedings of the 2004 symposium on Eye tracking research & applications*, pp. 139–146. ACM, New York (2004)
- [Majaranta et al., 2006] Majaranta, P., Mackenzie, I., Aula, A., Rähkä, K.-J.: Effects of feedback and dwell time on eye typing speed and accuracy. *Universal Access in the Information Society* 5(2), 199–208 (2006)
- [Perlin, 1998] Perlin, K.: Quikwriting: continuous stylus-based text entry. In: *UIST 1998: Proceedings of the 11th annual ACM symposium on User interface software and technology*, pp. 215–216. ACM Press, New York (1998)
- [Urbina and Huckauf, 2007] Urbina, M.H., Huckauf, A.: Dwell time free eye typing approaches. In: *COGAIN 2007: Gaze-based Creativity and Interacting with Games and On-line Communities* (2007)
- [Špakov and Miniotas, 2004] Špakov, O., Miniotas, D.: On-line adjustment of dwell time for target selection by gaze. In: *NordiCHI 2004: Proceedings of the third Nordic conference on Human-computer interaction*, pp. 203–206. ACM, New York (2004)
- [Ward and MacKay, 2002] Ward, D.J., MacKay, D.J.C.: Fast hands-free writing by gaze direction (2002)
- [Wobbrock et al., 2008] Wobbrock, J.O., Rubinstein, J., Sawyer, M.W., Duchowski, A.T.: Longitudinal evaluation of discrete consecutive gaze gestures for text entry. In: *ETRA 2008: Proceedings of the 2006 symposium on Eye tracking research & applications* (2008)