Alternatives to Single Character Entry and Dwell Time Selection on Eye Typing

Mario H. Urbina *
Bauhaus-University Weimar
Germany

Anke Huckauf†
Ulm University
Germany

Abstract

Eye typing could provide motor disabled people a reliable method of communication given that the text entry speed of current interfaces can be increased to allow for fluent communication. There are two reasons for the relatively slow text entry: dwell time selection requires waiting a certain time, and single character entry limits the maximum entry speed. We adopted a typing interface based on hierarchical pie menus, pEYEwrite [Urbina and Huckauf 2007] and included bigram text entry with one single pie iteration. Therefore, we introduced three different bigram building strategies. Moreover, we combined dwell time selection with selection by borders, providing an alternative selection method and extra functionality. In a longitudinal study we compared participants performance during character-by-character text entry with bigram entry and with text entry with bigrams derived by word prediction. Data showed large advantages of the new entry methods over single character text entry in speed and accuracy. Participants preferred selecting by borders, which allowed them faster selections than the dwell time method.

CR Categories: H5.2 [Information interfaces and presentation]: User Interfaces—Graphical user interfaces; K.4.2 [Computing Milieux]: Computers and Society—Assistive technologies for persons with disabilities

Keywords: Gaze control, eye tracking, user interfaces, input devices, eye-typing, selection methods, longitudinal study

1 Introduction

Eye-typing describes the process of entering text with eye gaze, which is an indispensable means of communication for motor disabled people who cannot either talk or use a keyboard or a mouse. The most propagated and common way of eye-typing is to select individual characters from an on-screen keyboard by visually fixating on it for a certain time (e.g., [Majaranta and Räihä 2002]). This selection method is called dwell time and its length ranges typically from 400 to 1000 ms ([Jacob 1993; Majaranta et al. 2006]).

One of the main problems concerning eye-typing is the slow text entry speed, which ranges from 6.2 [Hansen et al. 2004] to 8.9 [Miniotas et al. 2003] words per minute (wpm; [MacKenzie and Zhang 1999]). Compared with the average manual typing speed of about 40 wpm [MacKenzie and Zhang 1999], 8.9 words per minute is considerably slower - what is certainly also due to the less frequent usage of gaze input. Therefore, promising tools are not only characterized by a higher typing speed, but also by a steep learning curve. This is scarcely achieved with eye-typing systems based on dwell time selection, since a dwell time restricts the typing performance of the user. Furthermore, it is challenging to find an optimal dwell time for each user: Dwell times which are too short will increase the amount of unintended selections, whereas those that are too long will hamper user's performance, since fixating the gaze at one position slows down the intake of new information as well as the execution of new actions [Huckauf and Urbina 2008b].

Another reason for the poor text entry performance reported with eye-typing is the text entry method. Character-by-character selection is too slow for eye gaze interaction. Alternatives, like word prediction, are required to enhance the input performance. Other than with Dasher [Ward and MacKay 2002], text entry speeds beyond 10 wpm are still uncommon, even when using word prediction algorithms [Hansen et al. 2004; Hansen et al. 2008; MacKenzie and Zhang 2008].

With a desire to provide alternatives for both selection methods and text entry method, we present in this work a longitudinal study of alternative ways of eye-typing based on a hierarchical pie menu [Callahan et al. 1988; Kurtenbach and Buxton 1993], combining two selection methods (dwell time and selection borders [Urbina and Huckauf 2009]) and comparing performance on character-by-character text entry, bigram text entry, and bigram combined with word prediction text entry.

2 Related Work

Since the early days of video based eye tracking, eye-typing has been a topic of active research [Majaranta and Räihä 2002]. Nowadays, text entry research has developed far beyond the classic on-screen QWERTY keyboard selected with the static dwell time method. Bee and André [2008] distinguished three types of writing: typing, gesturing, and continuous writing. We describe the most relevant research done in text entry by gaze and have thereby related the systems and approaches to these three groups.

2.1 Eye-Typing

With eye-typing, letters are selected via dwell time from an on-screen keyboard. A virtual keyboard with QWERTY layout is one of the most common user interfaces for direct eye-typing. Majaranta et al. [2004] presented such a keyboard for their research on (visual and audial) feedback for gaze-based interaction. They employed dwell time selection for typing characters, which was set to 600 ms. They discovered that the combination of click and visual feedback achieved the fastest typing rate, with a mean of 7.55 words per minute.

Contrary to direct eye-typing, multi-tap approaches require more than one selection per character. *GazeTalk* [Hansen et al. 2004] is a multi-tap text entry system that provides three interaction modi:

- *Alphabetical letter-entry mode:* presents groups of letters (e.g. ABCDEFG). Here, letters are ordered into groups. Users need to select first the group which contains the letter and then the letter itself.

*e-mail: mario.urbina@uni-weimar.de
†e-mail: anke.huckauf@uni-ulm.de

ETRA 2010, Austin, TX, March 22 – 24, 2010.
© 2010 ACM 978-1-60558-994-7/10/0003 $10.00

- *Primary letter-entry mode:* this mode supplies a dynamic keyboard that contains six buttons arranged in a three by two matrix that allows the user to type the six more likely letters (generated by a letter prediction algorithm). If the desired letter is not among them, the user needs to get back to the alphabetical letter-entry mode.
- *Word completion/prediction mode:* presents the current eight more likely words in a four by two matrix, as well as two buttons for changing to the other mentioned modes.

With GazeTalk users were able to type up to 6.22 wpm using a dwell time of 500 ms.

Miniotas et al. [2003] developed a text entry system that required less physical space. *Multi-tap* supplies only ten keys, eight of them for text entry, one for space, dot and comma, and the last one for switching between different keyboards (function, letters, numbers and signs). Characters were ordered in groups in the same way as in ordinary mobile phones. To select a letter, the user needs to glance towards the key that contains the letter and remain over the letter for 400 ms, at which point the first letter is enlarged and changes color. To confirm its selection is required to leave the key. If the user continues looking at the key for another 400 ms, the second letter of the group will be highlighted, and so forth. This selection method means that the dwell time for the first letter of the key is 400 ms, 800 ms for the second, 1200 ms for the third, and in the worst case, at least 1600 ms for the forth letter. Nevertheless, Miniotas et al. [2003] reported text entry speeds using Multi-tap of 8.94 wpm.

2.2 Eye Gesturing

Under eye gesturing systems are grouped text entry methods where the user needs to look at different locations to trigger a letter or an action.

With *Eye-S* Porta and Turnia [2008] described nine hot spots on the screen, which the user can dwell on (for 400 ms) to describe a gesture. A specific gesture (selection order) is needed to perform for a desired action. The gestures contain an average three to four hot spots and were inspired by the graffiti gestures used with palms organizers. Experienced users achieved a mean text entry speed of 6.8 wpm. Faster text entry speeds may not be possible with Eye-S.

With a similar approach, Wobbrock et al. [2008] presented *EyeWrite*, which is an adaptation for gaze interaction of their previous work on text entry for palms EdgeWrite. Here, the four corners and center of a separate window (400 x 400 pixel) are used to enter the gaze gestures. For example, for writing a T the user needs to glance at the top-left, top-right and bottom-right corner of the window, finishing the gesture at the center. The points on the corners don't require any dwell time for activation, however the center point is triggered by a short dwell time to confirm the end of each gesture. Even though almost no dwell times are used, users achieved a mean text entry speed of only 4.87 wpm.

Bee and André [2008] adapted *Quickwriting* [Perlin 1998], which is also a text entry system for palm devices, for gaze interaction mapping the selection techniques for natural eye movements. Its interface is divided into eight selection sections. In the center is located a resting area, where eight group of letters are presented. The letters are ordered so, that their position corresponds to their position in the selection sections. In order to select a letter the user needs to glance to the corresponding selection section and go back to the center through the selection section where the letter is located. This text entry system works without any dwell time. In a first study participants achieved 5 wpm.

The the reason for Eye-S and EyeWrite's poor performance is nearly the same, namely the large number of spots that each gesture comprehends. With Eye-S an explicit dwell time restricts the performance, whereas with EyeWrite the physical constrains of eye movements limit the speed of the user's performance. Every saccade (a ballistic eye movement from A to B [Duchowski 2003], which last between 30-120 ms) is followed by a fixation, which lasts on average 200 ms before a new saccade can be started [Jacob 1993]. This means that it takes at least one second to complete the gesture of a letter.

Urbina and Huckauf [2007] developed three dwell time-free text entry systems. The one that performed best, *pEYEwrite*, takes advantage of pie menus with two hierarchical levels. On the first level the pie menu consists of six slices that contains groups of five letters or characters. On the second level each pie slice corresponds a letter or character. To take advantage of the biggest area of the slice, selection is done toward the outer border of the slice, the so-called selection border. To enter a character, the user moves their gaze so that it crosses the selection border of the slice that contains the desired letter. The pie menu on the second level opens immediately. The target letter is selected by glancing again through its respective selection border. The use of selection borders not only allows the user to inspect the pie as long he/she needs to, but also to perform fast multiple selections with saccades that follow each other. In special cases, where the target letter is located in the same direction as the group slice (like in letters "S", "R", "E", "I", "N" and space), the user is able to directly select with one long saccade that crosses the selection border of the pie in the first and second level (see Figure 1). Huckauf and Urbina [2008a] reported novice users achieving a mean text entry speed of 7.85 wpm while an expert achieved 12.33 wpm.

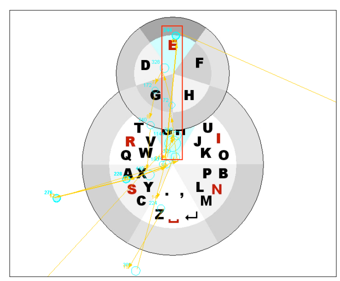

Figure 1: *pEYEwirte. Selection of letter "E" with one saccade.*

2.3 Continuous Writing

With continuous writing there are grouped text entry systems with which the eyes move continuously. Here, moving characters are followed by the user's gaze with pursuit movements, triggering their selection.

In *Dasher* [Ward and MacKay 2002], letters are presented vertically in alphabetical order on the right side of the screen. In order to select a letter, the user points (gazes) to the desired letter. The interface zooms in, and the letter moves towards the centre. The letter is selected when it crosses a vertical line located in the middle of the window. Gazing to the right from the vertical line leads to a selection of the target. Gazing left leads to the deletion of written text, causing the opposite visual effect. The distance between the gazing point and the vertical line controls the writing and deleting speed. This means looking to the middle of the vertical line ceases any action. The cooperation of vertical navigation and horizontal selection leads to a smooth motion of the letters. Considering the difference of text entry speeds of 4.5 wpm without text prediction [Urbina and Huckauf 2007] and around 20 wpm [Tuisku et al. 2008] with word prediction, it is clear that a character or word calculation algorithm is needed to make gaze based text entry more fluent.

StarGazer [Hansen et al. 2008] is like Dasher, a zooming interface. However, the zooming occurs along the z-axis, while the x- and y-axis are used for panning. The letters are ordered alphabetically in circular form. The display pans towards the direction of the gaze while zooming is continuous, comparable with skydiving. Letters are selected when the desired letter has been zoomed enough. After selection, all letters appear again in the circular order. Hansen et al. [2008] reported text entry speeds up to 8.16 wpm.

In both Dasher and StarGazer, the user follows the letters with pursuit eye movements, which makes these interfaces very suitable for gaze interaction. Even though the navigation and zooming for every single character selection takes on average at least as long as a normal dwell time, their zooming technique allows low accuracy eye tracking.

3 Text entry Methods

For the longitudinal study we implemented seven different versions of *pEYEwrite*. A classic version with *single character entry*, three *bigram entry versions*, and three that combined *bigram text entry with word prediction*.

3.1 Single Character Entry, pEYEwrite-C

The single character entry version, *pEYEwrite-C*, is based on the aforementioned pEYEwrite [Urbina and Huckauf 2007]. Compared with [Urbina and Huckauf 2007], letters were re-arranged within the pie menu in order to facilitate the selection of the most frequent letters: The five most frequent letters (according to [Pommerening 2009]), except for "I", which was replaced with "T", and the space-symbol were arranged in a way that they can be typed with one saccade (see Figures 2a and 2b). The other letters were distributed within the slices, so that the upper slice contained only vowels and the remaining letters were grouped with their neighbors in frequent words (like "ch", "br", "gr", "ck" or "nd", see Figure 2a). The letter position in the first pie equaled the letter position in the second pie. This way users know the position of the letter in the second pie in advance and are able to prepare the saccade responsible for the letter selection before the second pie pops up.

In order to type one letter using *pEYEwrite-C*, the respective slice of the first pie has to be selected. Immediately following, a next pie is opened, centered on the border of the corresponding pie (see Figure 2b). The desired letter has to be selected from that pie. Letters can be selected via selection border (crossing the outer border of the slice with the gaze) or via dwell time (600 ms). The users were free to combine both selection methods.

A cancel button is provided on the center of the pie in the second hierarchical level (see Figure 2b) to allow the user to abort the selection process initiated in the first pie. With this, even when a group of letters is already selected, no letter is written. The cancel button is triggered by a dwell time of 750 ms.

One of the main strengths of *pEYEwrite-C* is that novices can easily learn, in an unintended way, the gaze path required to select each letter. Therefore advanced users are able to perform the two-stroke gaze gesture even without looking for the letters, increasing their performance.

3.2 Text Entry with Bigrams

With bigram text entry the user is able to enter, in an optimal case, two letters (or characters) traversing the hierarchical pies only once. To achieve this, the *pEYEwrite-C* interface was extended with a third hierarchical level: the letter group on the first level, the single letters on the second level and up to five following letter candidates on the third level (see Figure 2c). Three different bigram entry approaches were conceived and tested. The first approach provides vowels on the bigram level after each letter, called *pEYEwrite-BiVow*. Presenting vowels as bigram candidates not only provides a high probability for using a vowel (of 40% [Pommerening 2009], but also allows an easy extension of the selection gestures with a further step. Since the vowels are presented in alphabetical order and always have the same position within the pie, users should be able to learn their positions and find them quickly and accurately after little training with the fast three-stroke gaze gesture for selection.

The second approach, called *pEYEwrite-BiSta*, presents the five most probable letters at the bigram level that follows the letter entered in the second level, which is based on a static bigram statistic (available on [Pommerening 2009]). This method increases the probability of a correct bigram candidate up to 70%, but at the same time increases the cognitive load of the user. The user has to learn 25 (letters) * 5 (bigram candidates) = 125 possible gaze paths to be able to enter a bigram with this method, which might require extensive training before the user is able to perform a fast three-stroke gaze gesture.

The third and last approach, *pEYEwrite-BiDyn*, relies on an "dynamic" calculation of the up to five most likely letters to follow, based on a list of 300,000 word-trigrams and their occurrence in a corpus. The corpus was automatically extracted from information available online, like newspapers and e-books. The character prediction algorithm takes in to count the last two written words in the calculation of the letter candidates, increasing the probability of a suitable letter among the candidates, compared with *pEYEwrite-BiVow* and *pEYEwrite-BiSta*. Due to the dynamic calculation, it is impossible for the user to learn which letters will be presented in the bigram level of the pie, increasing the cognitive load of the user even more and restricting a fast gaze gesture usage.

As in *pEYEwrite-C*, letters are selected via selection borders or dwell time. Selecting via selection borders implies that the user can select a letter and continue traversing the pie to the next hierarchical level. Selecting with dwell time implies selecting a letter but not to continue traversing the pie. Once a letter is selected on the last level or by dwell time, the pie menu collapses and the user starts again from the start button (see Figure 3).

3.3 Text Entry with Bigrams and Word Prediction

While enhancing text entry with word prediction seems to be a plausible way to increase the text entry speed and reducing the keystroke per character (kspc) rate, relieving the user, the total number of benefits that word prediction provides is still unclear. The presentation of predicted word candidates draws the attention of the user from the letter entry to the word candidates. He/she then needs to read the predicted words and, if possible, select one of them. Moreover, the more letters that are entered, the better the predicted candidates, drastically reducing the number of letters that do not need to be typed by selecting a word candidate. Considering these two facts makes our a priori assumption about the benefits of word prediction doubtful.

In order to provide empirical evidence and conclude how text entry is affected by word prediction, we included the three already-presented bigram text entry word predictions, calling them *pEYEwrite-WoVow*, *pEYEwrite-WoSta* and *pEYEwrite-WoDyn* respectively. Up to three word candidates (according to the prediction) were presented above the text field, which are selected by

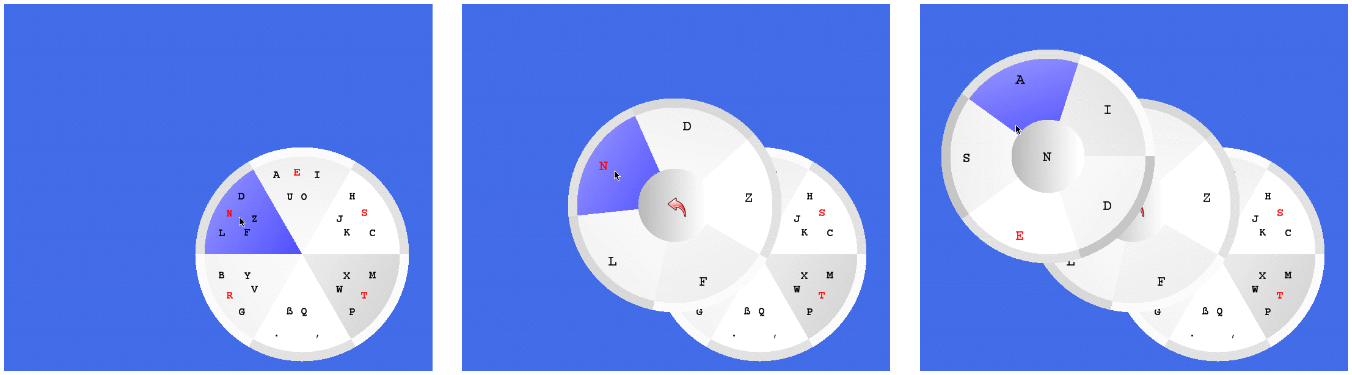

Figure 2: *a) The first level of the hierarchical pie menu contains the group of characters. b) Immediately after selection, the second level pops up providing each letter for selection. On the center of the pie a back arrow can be dwelled for canceling the group selection. c) On the bigram level, five bigram characters are presented. Depending on the bigram entry mode these can be vowels, letters taken from a static bigram-probability table or dynamic calculated.*

dwelling the gaze for one second on them (see Figure 3).

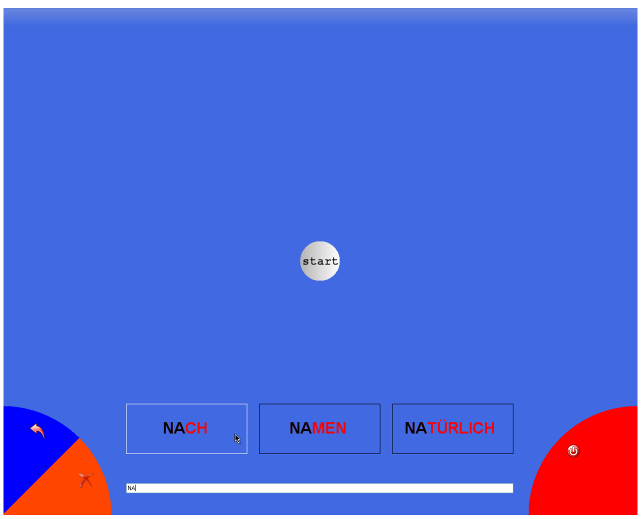

Figure 3: *Word candidates are presented above the text field. The characters already entered are highlighted in black, the characters to enter in red.*

It is important to note that, as expected, the quality of the prediction algorithm as well as of the corpus used has a significant influence on the predicted words and it is not the primary aim of this work to provide a linguistic masterwork. The prediction algorithm is, based on a Markov model of second order, also known as trigrams. The list of trigrams and their frequency in the corpus is the same used in *pEYEwrite-BiDyn*. Here the next word probability is calculated in relation to the two words written before.

4 Longitudinal User Study

The aim of this longitudinal study is to reveal the potential of pie menus for text entry by gaze and to provide and compare alternatives to single character entry. For this reason we measured efficiency and errors. Efficiency was measured in words per minute; wpm. A word is defined as a sequence of five characters, including space. Errors were measured with two variables, corrected errors and remaining errors. Corrected errors present the rate (in %) between deleted and undone sections in relation to all key selections. Remaining errors are evaluated using the Minimum String Distance method (MSD) [Soukoreff and MacKenzie 2001], which looks for the minimum distance of two strings in terms of the editing primitives, insertion, deletion and substitution, and is calculated as follows:

RE = MSD / Max(length correct text, length written text)

Furthermore, we measured the mean number of keystrokes needed to enter a letter. Text entry systems that do not use word completion ideally reach a keystroke per character (kspc) rate of 1.0 indicating that each keystroke produces a character. Therefore, deleting and undoing selections increases the kspc rate. Typing systems using word prediction/completion may reach kspc rates below 1.0, since more than one character may be entered per keystroke.

4.1 Method

4.1.1 Participants

Nine university students participated voluntarily in the experiment. Their ages ranged from 24 to 30 (mean 27.4), they had normal or corrected-to-normal vision and were familiar with computers and with mouse and keyboard usage. Six participants had no prior experience with eye tracking, and the remaining three had similar experience with eye tracking and eye typing. All participants were rewarded. To keep participants motivated during the experiment, the participant who achieved the best typing performance received a prize.

4.1.2 Apparatus

The interfaces were presented on a 21” Sony GMD-F520 CRT display with a resolution of 1280x1024 at a frame rate of 100Hz, and were run on a Dell-PC with a Core2Quad processor at 2,5 GHz and 2 Gb DDR-Ram under Windows XP. The eye tracking device used was a head-mounted Eyelink2 from SR-Research. The tracked gaze coordinates were mapped directly onto the mouse cursor, without using any smoothing algorithms. In order to keep calibration as accurate as possible, a chin rest situated 60 cm in front of the monitor was used.

4.1.3 Task and Procedure

Nine participants were divided into three groups (by three participants). The participants with prior eye tracking experience prior were divided up, with one placed in each of the three groups. The rest of the participants were assigned randomly to each group. Each participant absolved altogether 60 sessions, 20 sessions with *pEYEwrite-C*, 20 sessions with one of the bigram-entry versions of pEYEwrite (*pEYE-write-BiVow*, *pEYEwrite-BiSta* or *pEYEwrite-BiDyn*) and 20 sessions with the corresponding bigram-entry with word prediction (*pEYEwrite-WoVow*, *pEYEwrite-WoSta* or *pEYEwrite-WoDyn*).

Participants sat 60 cm in front of the monitor, and they were helped to mount the eye tracker to their head before the calibration routine was started. On the beginning of each test day and before testing a new interface, the subjects had five minutes training. The participants did not type more than one hour and performed maximal five sessions per day. A session consisted of writing three well-known sayings (which had a mean length of six words per saying). Participants were instructed to enter the saying as fast and accurate as possible. Typographical errors were to be corrected if noticed immediately.

4.2 Results

4.2.1 Single Character Entry - *pEYEwrite-C*

With *pEYEwrite-C* users reached a mean text entry speed over the 20 sessions of 7.34 wpm (standard error se = 0.35) and 8.1 wpm for the last three trials (se = 0.37). Due the significant learning rate and differences over sessions, we will refer the mean values as the average performance of the last three sessions. The maximum entry speed of this device was 10.38 wpm, accomplished by subject number seven on the eighth session. The sessions had, as expected, a significant effect ($F(19,152) = 13.42$, $p < 0.001$) showing a strong learn rate (see Figure 4). Post hoc tests showed that even the difference between the entry speed between the 10th and the 20th session was significant ($F(1,8) = 8.73$, $p < 0.05$).

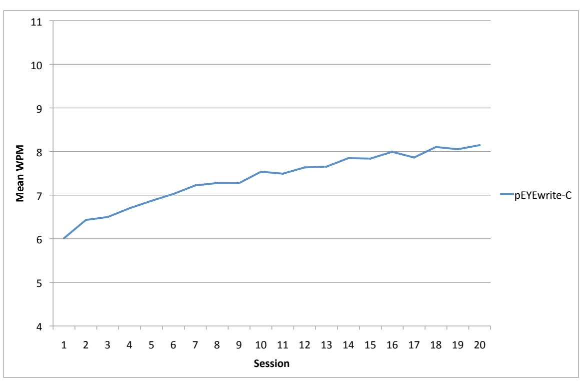

Figure 4: *Word per minutes achieved with pEYEwrite-C on each session.*

Users committed on average 1.41 % errors (se = 0.367) that remained in text. Sessions showed a significant effect over the error rate ($F(19,152) = 3.33$, $p < 0.001$), which ranged form 6.55% (se = 1.95) in the first session to 0.97 % on the last session (see Figure 5). Eight (of nine) subjects performed at least one session without making errors.

The mean corrected error rate produced was of 3.95% (se = 0.72). The sessions showed a significant effect, with $F(19,152) = 2.11$, $p < 0.01$, improving the error rate from 9.85% (se = 1.39) corrected text in the first session to 3.6 % (se = 0.82) in the last session.

The mean kspc rate was 1.18 (se = 0.04). The sessions also showed

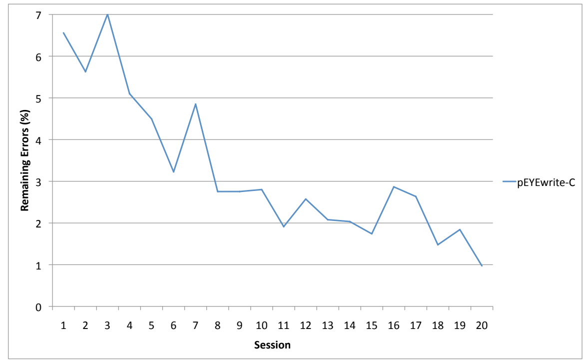

Figure 5: *Rate of errors remaining in text with pEYEwrite-C on each session.*

here a significant effect, with $F(19,114) = 7.192$, $p < 0.001$, reducing the kspc rate from 1.78 (se = 0.205) corrected text on the first session to 1.18 kspc (se = 0.03) on the last session (Figure 6).

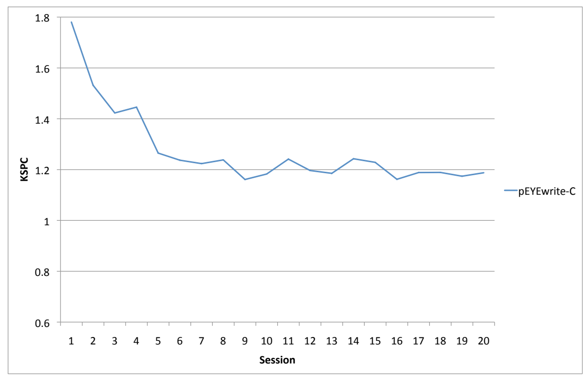

Figure 6: *Mean keystroke per character rate achieved with pEYEwrite-C on each session.*

Furthermore, we analyzed which selection method was preferred by the subjects. 98.57% of the characters were selected with the selection border method and only 1.43% by dwelling on the character (see Figure 8). Throughout the 20 sessions, users improved their text entry performance by 36.6% using selection borders (from 5.9 wpm in the first session to 8.06 wpm in the last session) and only by 4% (from 4.5 wpm to 4.71 wpm) (see Figure 7).

4.2.2 Text entry with Bigrams

We compared the text entry performance and learning rates of three (already mentioned) approaches based on hierarchical bigram entry, *pEYEwrite-BiVow*, *pEYEwrite-BiSta* and *pEYEwrite-BiDyn*. Subjects achieved a mean text entry speed on the last three sessions of 8.05 wpm (se = 0.83, max = 9.42 wpm) with *pEYE-write-BiVow*, 10.08 wpm (se = 0.83, max 12.43 wpm) with *pEYEwrite-BiSta* and 10.64 wpm (se = 0.83, max = 12.85 wpm) with *pEYEwrite-BiDyn*, showing no significant differences ($F(2,6) = 2.67$, $p > 0.05$). The sessions showed a significant effect on performance with $F(9,114) = 14.18$, $p < 0.001$ that can be attributed to learning as well as an interaction between the bigram methods and the sessions ($F(18,114) = 1.57$, $p < 0.05$). In mean, participants improved their text entry performance on 25.11% with *pEYEwrite-BiVow*, 29.87 % with *pEYEwrite-BiSta*, and 18.25 % with *pEYEwrite-BiDyn* (see Figure 9).

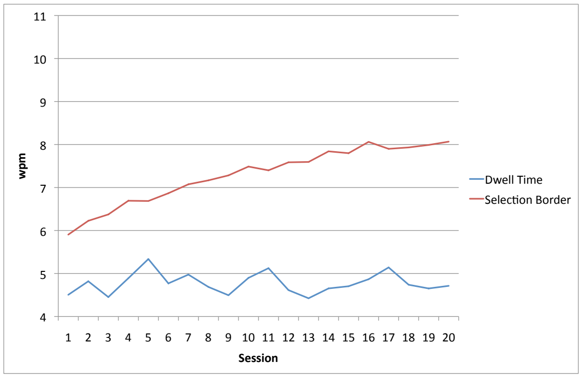

Figure 7: *Word per minutes achieved with the pEYEwrite-C using selection borders and dwell time selection on each session.*

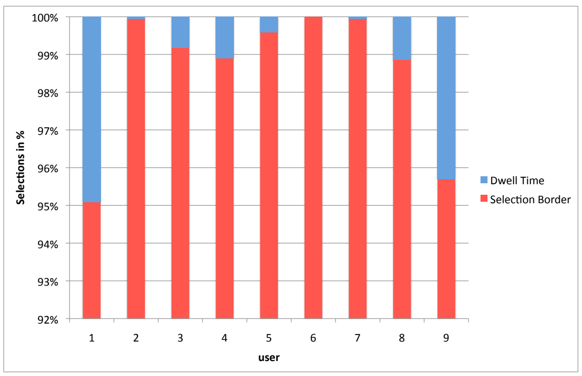

Figure 8: *Rate of selections performed with the selection border and dwell time method.*

The rate of remaining errors was in mean on the last three sessions 1.35% (se = 0.38) with *pEYEwrite-BiVow*, 0.5% (se = 0.38) with *pEYEwrite-BiSta* and 0.96% (se = 0.38) with *pEYE-write-BiDyn*. Neither sessions nor the bigram entry methods produced significant effects. All participants were able to complete at least two error free sessions.

The corrected error rate was on average 2.79% (se = 1.24) with *pEYEwrite-BiVow*, 3.75% (se = 1.24) with *pEYEwrite-BiSta* and 4% (se = 1.24) with *pEYEwrite-BiDyn*, presenting no significant differences. The sessions showed a significant effect on performance with $F(9,114) = 2.115$, $p < 0.01$ which can be attributed to learning. All participants were able to complete at least two error free sessions.

Participants achieved a mean kspc rate of 1.07 (se = 0.33) with *pEYEwrite-BiVow*, 1.09 (se = 0.33) with *pEYEwrite-BiSta* and 1.1 (se = 0.33) with *pEYEwrite-BiDyn*. Here, as well as in the remaining error data, no significant effects of sessions and bigram entry methods were observed. All participants were able to complete at least two error free sessions.

4.2.3 Text entry with Bigrams and Word Completion

We measured text entry speeds combining bigram text entry with word completion. Participants achieved a mean text entry speed of 12.73 wpm (se = 1.21, max = 14.71 wpm) with *pEYEwrite-WoVow*, 12.72 wpm (se = 1.21, max 15.8 wpm) with *pEYEwrite-WoSta* and 13.47 wpm (se = 1.21, max = 17.26 wpm) with *pEYEwrite-WoDyn*. These differences were not of statistically significance and after the

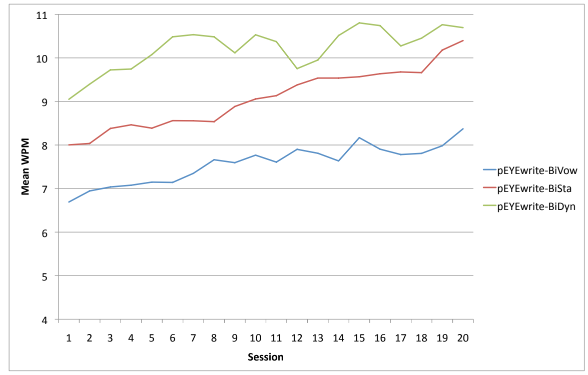

Figure 9: *Word per minutes achieved with the bigram text entry methods on each session.*

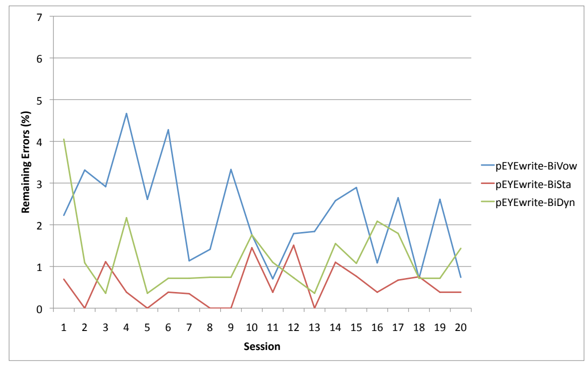

Figure 10: *Rate of errors remaining in text with with the bigram text entry methods on each session.*

20 sessions no significant learning effect was observed.

Participants produced 0.01% (se = 0.147) errors that remained in text using *pEYEwrite-WoVow*, 0.05% (se = 0. 147) with *pEYEwrite-WoSta* and 0.01% (se = 0.147) with *pEYEwrite-WoDyn*. Participants achieved a rate of corrected errors of 3.27% (se = 1.46) with *pEYEwrite-WoVow*, 5.06% (se = 1.46) with *pEYE-write-WoSta* and 2.36% (se = 1.46) with *pEYEwrite-WoDyn*. Here, as well as in the remaining error data, no significant effects of sessions and bigram entry methods were observed. All participants were able to complete at least two error free sessions.

The mean kspc achieved by participants using *pEYEwrite-WoVow* reached 0.711 (se = 0.07), with *pEYEwrite-WoSta* 0.74 (se = 0.07) and 0.705 (se = 0.07) using *pEYEwrite-WoDyn* showing not significant differences between these three text entry modes. Sessions had a significant effect on performance with $F(9,114) = 9.186$, $p < 0.001$.

5 Discussion

The results obtained with the single character entry method support the results presented in the short experiments [Huckauf and Urbina 2008a], showing significant learning effects in all measured aspects using pie menus. Taking the learning data into consideration, it can be assumed that the learning process had not finished even after 20 sessions. This highlights the outstanding qualities of pie menus in gaze interaction.

The selection method data results obtained using with *pEYEwrite-*

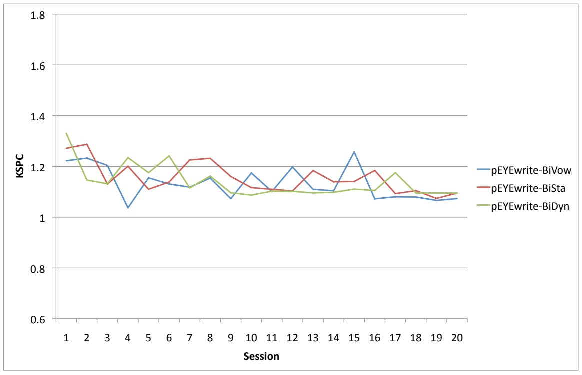

Figure 11: *Mean keystroke per character rate achieved with bigram text entry methods on each session.*

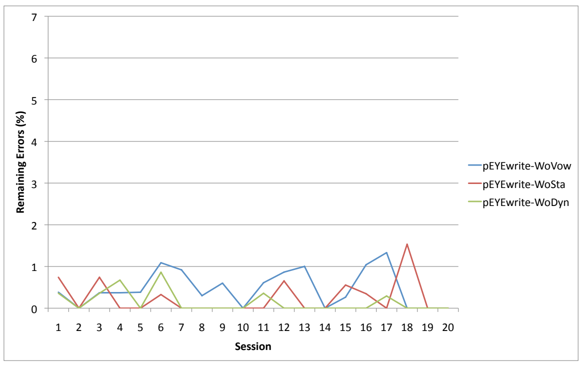

Figure 12: *Rate of errors remaining in text with with the bigram text entry methods combined with word prediction on each session.*

C showed a massive preference for using selection borders instead of dwell time. The text entry speed improvement rate achieved with both selection methods (36% with Selection Borders, 4% with dwell time) and the text entry speed (8.06 wpm with Selection Borders and 4.71 with dwell time) help clarify the favoritism for selection borders. Subjective results reported that 8 of 9 participants reported preferring selecting characters with selection borders. Even though one participant prefered dwell time, no difference was revealed between his/her data and the averages already reported, selecting 94.45% of the characters with selection borders (see Figure 8, user 1).

Results depictured in Figure 9 confirm most of our learning hypothesis on bigram entry. *pEYEwrite-BiDyn* showed on the first sessions better performances (one wpm faster) than *pEYEwrite-BiSta*. This can be attributed to the more probable presentation of a suitable bigram candidate for selection. This gap was closed after the 20 sessions, confirming the better learning qualities of static bigrams (*pEYEwrite-BiSta*) over dynamic bigrams (*pEYEwrite-BiDyn*). Even though no significant effects were found between the bigram text entry methods, post-hoc analysis revealed a significant difference on text entry performance ($p=0.05$) between *pEYEwrite-BiVow* and *pEYEwrite-BiDyn*. This significant difference has mainly two reasons; First, the rate of bigram entries with both methods differed from 22.72% with *pEYEwrite-BiVow* to 35.45% with *pEYEwrite-BiDyn* (31.53% with *pEYEwrite-BiSta*). The second reason was the longer mean selection time achieved on the bigram level (third depth level of the pie menu) with *pEYEwrite-BiVow* (819 ms) compared to *pEYEwrite-BiDyn* (609.98 ms). This is an unexpected result, since the vowels were all presented on the

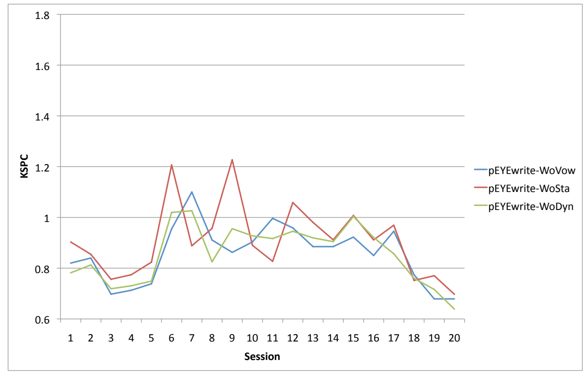

Figure 13: *Mean keystroke per character rate achieved with bigram text entry methods combined with word prediction on each session.*

same position, which should be easy to learn, while the dynamic predicted bigram characters had mostly a different combination and position. Huckauf and Urbina [2008a] reported serial scanning of characters ordered alphabetically, providing poorer performances than arbitrary ordered ones.

The main drawback observed using pEYEwrite combined with bigram entry was the repeated entry of double errors. Some participants selected the wrong first letter and thereby a wrong bigram on the next stage. Even when participants noticed the first error, they preferred to continue selecting the second letter with selection borders and not to abort by dwelling on the center of the pie, causing a double error. Therefore they needed to correct two errors, unnecessarily increasing the kspc and the corrected error rate.

The text entry performance achieved did not differ among the three bigram entry systems with word prediction (*pEYEwrite-WoVow*, *pEYEwrite-WoSta* and *pEYEwrite-WoDyn*), showing a bigger impact of word prediction than each bigram presentation method. These results show that the assumed high cognitive load using word prediction does not hamper the text entry performance, which was much higher as with other text entry systems with word prediction [Hansen et al. 2004; Hansen et al. 2008]. Due order effects we omitted an analysis of variance between single character entry, bigram entry and bigram with word completion. Nevertheless, the typing speed, error and kspc rates presented in this work point out the importance of alternatives to single character entry, like bigram entry and word prediction in eye-typing.

6 Conclusion

We presented three bigram text entry methods, all based on a hierarchical pie menu application. The two methods based on the probability of the next character (calculated statically or dynamically) performed better than presenting vowels as bigram candidates Augmenting the investigated text entry methods with word prediction increases the text entry performance and decreases the cognitive load during typing. The combination of two selection methods was well accepted by the users and enhanced the interaction with pie menus. While participants could not achieve text entry speeds of around 20 wpm as reported by [Tuisku et al. 2008], the mean of about 13 wpm (with a top speed of 17.2 wpm) achieved by combining different selection methods with bigram entry and word prediction provides an alternative to Dasher for fast text entry and encourages the continuation of research on text entry by gaze.

Acknowledgements

This research was partly supported by the Verein zur Förderung von Studenten der Bauhaus-Universität Weimar e.V. The authors would like to thank Michael Pannier and Thomas Knapke for the assistance in implementation and all volunteers who participated in the experiment.

References

BEE, N., AND ANDRÉ, E. 2008. Writing with your eye: A dwell time free writing system adapted to the nature of human eye gaze. In *Proceedings of Workshop on Perception and Interactive Technologies for Speech-Based Systems*.

CALLAHAN, J., HOPKINS, D., WEISER, M., AND SHNEIDERMAN, B. 1988. An empirical comparison of pie vs. linear menus. In *CHI '88: Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM, New York, NY, USA, 95–100.

DUCHOWSKI, A. T. 2003. *Eye Tracking Methodology - Theory and Practice*. Springer-Verlag.

HANSEN, J. P., TORNING, K., JOHANSEN, A. S., ITOH, K., AND AOKI, H. 2004. Gaze typing compared with input by head and hand. In *ETRA '04: Proceedings of the 2004 symposium on Eye tracking research & applications*, ACM Press, New York, NY, USA, 131–138.

HANSEN, D. W., SKOVSGAARD, H. H. T., HANSEN, J. P., AND MØLLENBACH, E. 2008. Noise tolerant selection by gaze-controlled pan and zoom in 3d. In *ETRA '08: Proceedings of the 2008 symposium on Eye tracking research & applications*, ACM, New York, NY, USA, 205–212.

HUCKAUF, A., AND URBINA, M. H. 2008. Gazing with peyes: towards a universal input for various applications. In *ETRA '08: Proceedings of the 2008 symposium on Eye tracking research & applications*, ACM, New York, NY, USA, 51–54.

HUCKAUF, A., AND URBINA, M. H. 2008. On object selection in gaze controlled environments. In *Journal of Eye Movement Research*, vol. 2 of *4*, 1–7.

JACOB, R. J. K. 1993. Interaction techniques: Toward non-command interfaces. *In H.R. Hartson and D. Hix (Eds.) Advances in Human-Computer Interaction 4*, 151–190. Available at http://www.cs.tufts.edu/ jacob/papers/hartson.pdf.

KURTENBACH, G., AND BUXTON, W. 1993. The limits of expert performance using hierarchic marking menus. In *CHI '93: Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM Press, New York, NY, USA, 482–487.

MACKENZIE, I. S., AND ZHANG, S. X. 1999. The design and evaluation of a high-performance soft keyboard. In *CHI '99: Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM Press, New York, NY, USA, 25–31.

MACKENZIE, I. S., AND ZHANG, X. 2008. Eye typing using word and letter prediction and a fixation algorithm. In *ETRA '08: Proceedings of the 2008 symposium on Eye tracking research & applications*, ACM, New York, NY, USA, 55–58.

MAJARANTA, P., AND RÄIHÄ, K.-J. 2002. Twenty years of eye typing: systems and design issues. In *ETRA '02: Proceedings of the 2002 symposium on Eye tracking research & applications*, ACM Press, New York, NY, USA, 15–22.

MAJARANTA, P., AULA, A., AND RÄIHÄ, K.-J. 2004. Effects of feedback on eye typing with a short dwell time. In *ETRA '04: Proceedings of the 2004 symposium on Eye tracking research & applications*, ACM Press, New York, NY, USA, 139–146.

MAJARANTA, P., MACKENZIE, S., AULA, A., AND RÄIHÄ, K.-J. 2006. Effects of feedback and dwell time on eye typing speed and accuracy. *Univers. Access Inf. Soc. 5*, 2, 199–208.

MINIOTAS, D., SPAKOV, O., AND EVREINOV, G. E. 2003. Symbol creator: An alternative eye-based text entry technique with low demand for screen space. In *INTERACT*.

PERLIN, K. 1998. Quikwriting: continuous stylus-based text entry. In *UIST '98: Proceedings of the 11th annual ACM symposium on User interface software and technology*, ACM, New York, NY, USA, 215–216.

POMMERENING, K., 2009. Kryptologie - zeichenhäufigkeiten in deutsch (in german). http://www.staff.uni-mainz.de/pommeren/Kryptologie/Klassisch/1_Monoalph/deutsch.html. Last visited on 31/09/2009.

PORTA, M., AND TURINA, M. 2008. Eye-s: a full-screen input modality for pure eye-based communication. In *ETRA '08: Proceedings of the 2008 symposium on Eye tracking research & applications*, ACM, New York, NY, USA, 27–34.

SOUKOREFF, R. W., AND MACKENZIE, I. S. 2001. Measuring errors in text entry tasks: an application of the levenshtein string distance statistic. In *CHI '01: CHI '01 extended abstracts on Human factors in computing systems*, ACM Press, New York, NY, USA, 319–320.

TUISKU, O., MAJARANTA, P., ISOKOSKI, P., AND RÄIHÄ, K.-J. 2008. Now dasher! dash away!: longitudinal study of fast text entry by eye gaze. In *ETRA '08: Proceedings of the 2008 symposium on Eye tracking research & applications*, ACM, New York, NY, USA, 19–26.

URBINA, M. H., AND HUCKAUF, A. 2007. Dwell-time free eye typing approaches. In *Proceedings of the 3rd Conference on Communication by Gaze Interaction (COGAIN 2007)*, 65–70. Available online at http://www.cogain.org/cogain2007/.

URBINA, M. H., AND HUCKAUF, A. 2009. Selecting with gaze controlled pie menus. In *Proceedings of the 5th International Conference on Communication by Gaze Interaction (COGAIN 2009)*, 25–29. Available online at http://www.cogain.org/cogain2009/.

WARD, D. J., AND MACKAY, D. J. C. 2002. Fast hands-free writing by gaze direction. *Nature 418*, 6900, 838.

WOBBROCK, J. O., RUBINSTEIN, J., SAWYER, M. W., AND DUCHOWSKI, A. T. 2008. Longitudinal evaluation of discrete consecutive gaze gestures for text entry. In *ETRA '08: Proceedings of the 2008 symposium on Eye tracking research & applications*, ACM, New York, NY, USA, 11–18.