The Potential of Dwell-Free Eye-Typing for Fast Assistive Gaze Communication

Per Ola Kristensson*
School of Computer Science
University of St Andrews

Keith Vertanen†
Department of Computer Science
Montana Tech of the University of Montana

Abstract

We propose a new research direction for eye-typing which is potentially much faster: dwell-free eye-typing. Dwell-free eye-typing is in principle possible because we can exploit the high redundancy of natural languages to allow users to simply look at or near their desired letters without stopping to dwell on each letter. As a first step we created a system that simulated a perfect recognizer for dwell-free eye-typing. We used this system to investigate how fast users can potentially write using a dwell-free eye-typing interface. We found that after 40 minutes of practice, users reached a mean entry rate of 46 wpm. This indicates that dwell-free eye-typing may be more than twice as fast as the current state-of-the-art methods for writing by gaze. A human performance model further demonstrates that it is highly unlikely traditional eye-typing systems will ever surpass our dwell-free eye-typing performance estimate.

CR Categories: H.5.2 [Information Interfaces and Presentation]: User Interfaces—Input devices and strategies; K.4.2 [Computers and Society]: Social Issues —Assistive technologies for persons with disabilities;

Keywords: eye-typing, text entry, accessibility, dwell-free eye-typing, augmentative and alternative communication

1 Introduction

Gaze interaction techniques enable users with certain motor disabilities to communicate via an eye-tracker. For users with certain motor disabilities gaze interaction may be the only communication channel available. Given their importance to such users, gaze communication systems have been actively researched for over 30 years [Majaranta and Räihä 2002]. Unfortunately the record-speeds in gaze communication are relatively slow and range from 7–26 wpm [Majaranta and Räihä 2002; Majaranta et al. 2009; Wobbrock et al. 2008; Tuisku et al. 2008; Ward and MacKay 2002].

The primary technique used for gaze communication is eye-typing. To eye-type, the user looks at a letter on an on-screen keyboard. If the user's gaze remains fixed on the same letter for a set time period (the dwell-timeout) the system assumes the user intended to write that letter. Despite significant progress, even the best eye-typing systems are relatively slow with reported entry rates ranging from 7–20 wpm [Majaranta and Räihä 2002; Majaranta et al. 2009]. Majaranta et al. [2009] carried out a longitudinal experiment in which users adjusted the dwell-timeout progressively throughout the experiment. They found that after ten 15-minute sessions users reached a mean entry rate of 20 wpm. Their learning curve showed that participants quickly improved from 5–10 wpm in the first session to 13–20 wpm in the fifth session (Figure 2 in [Majaranta et al. 2009]). Thereafter the learning curve plateaued with only some of the participants making further minor improvements. The best participant improved from 20 wpm in the fifth session to 23 wpm in the last session. It is therefore plausible that these entry rates are close to the limit for traditional dwell-based eye-typing.

Besides eye-typing, the only other fast gaze communication technique is Dasher [Majaranta et al. 2009; Tuisku et al. 2008; Ward and MacKay 2002]. Dasher allows users to write by zooming through a world of boxes. Each box represents an individual letter and the size of a box is proportional to the probability of that letter given the preceding letters. The entry rates for Dasher range between 16–26 wpm [Tuisku et al. 2008; Ward and MacKay 2002] (note that the higher entry rates stem from a single expert user in [Ward and MacKay 2002]). Hence Dasher's performance is similar to eye-typing with a user-adjustable dwell-timeout [Majaranta et al. 2009].

Thus the fastest gaze communication systems available appear to have reached a plateau at around 20 wpm. To break out of this local optimum we propose a paradigm shift in gaze communication: dwell-free eye-typing. In dwell-free eye-typing users do not need to look at each letter for a fixed period. Instead, the system recognizes sequences of words from users' continuous eye-traces. Users need only gaze through the desired letters in their desired phrase or sentence. After looking at a designated area, the system processes the eye-trace and infers the desired word sequence. Since such a system eliminates dwell-timeouts for key selections, this technique could potentially be faster than state-of-the-art dwell-based eye-typing interfaces. This dwell-free interaction technique is also likely to be more natural, fluid, and less frustrating for users. We also hypothesize that dwell-free eye-typing may enable users to learn and thereafter quickly recall the movement patterns for familiar words or phrases from motor memory. This may be plausible given that previous research has shown that users of a stylus interface can reliably recall 15 unfamiliar gesture trajectories per 45 minute practice session [Zhai and Kristensson 2003].

This new dwell-free technique is obviously non-trivial to realize. However, as Shannon [1948] observed in his groundbreaking paper on information theory, natural languages are highly redundant. As a consequence, while a particular eye-trace might be compatible with a vast number of possible letter sequences, the majority of these will be improbable under a language model. By capturing language regularities in such a language model, a system can infer users' intended text from noisy input. A well-known example is speech recognition [Padmanabhan and Picheny 2002]. A more related example is Salvucci's [2000; 1999] work on fixation tracing. Salvucci pioneered the use of hidden Markov models for inferring users' intended individual words from eye-traces over an on-screen keyboard. His system performed isolated word recognition using a small set of 1000 words. Another related example is the gesture keyboard (commercialized as ShapeWriter, T9 Trace and Swype). This technique enables users to write individual words by gesturing on a capacitative on-screen keyboard. Users' intended words are recognized from the finger traces using a pattern recognition algorithm [Kristensson and Zhai 2004; Zhai and Kristensson 2003].

*e-mail: pok@st-andrews.ac.uk
†e-mail: kvertanen@mtech.edu

ETRA 2012, Santa Barbara, CA, March 28 – 30, 2012.
© 2012 ACM 978-1-4503-1225-7/12/0003 $10.00

1.1 Estimating Dwell-Free Eye-Typing Performance

We envision a dwell-free eye-typing system reminiscent of how state-of-the-art continuous speech recognition systems convert spoken utterances into text [Padmanabhan and Picheny 2002]. Analogous to speech recognition, our hypothetical system converts a stream of time-ordered observations into a sequence of words. Both speech recognition and our proposed eye-trace recognition rely heavily on a language model to aid their inference from noisy data. Neither requires the user to explicitly wait for each word or letter to be recognized.

However, engineering such an unproven and complex system is a tremendous undertaking. To put this into perspective, speech recognition and handwriting recognition are separate research fields with dedicated conferences and journals focusing on how to efficiently and accurately recognize users' input. Dwell-free eye-typing recognition may require an undertaking of similar magnitude.

Thus, before championing a major research effort towards creating dwell-free eye-typing systems, we wondered whether users would be able to write substantially faster using this paradigm. Text entry is a complex task, more complex than visual target selection. For instance, eye-typing users must model the desired words, the letter sequences that comprise those words, and the locations of the letters on the keyboard. Therefore, we conducted a text entry study to obtain a human performance estimate of dwell-free eye-typing by simulating a perfect recognizer. As we will see, the possible gains are substantial, with users reaching entry rates that are twice that of traditional state-of-the-art eye-typing interfaces.

We designed our system to factor in how much information a complete dwell-free eye-typing interface would require to accurately infer users' intended words. This enabled us to investigate the plausible human performance of dwell-free eye-typing. Our system knows what the user is intending to write and verifies that the user is gazing at the letter key sequence corresponding to the stimulus. This provides users with an experience similar to what they might expect from a highly accurate dwell-free eye-typing interface.

In our experiment, users had to gaze in the proximity of the desired key but not necessarily directly at it. Our system recognized an intended letter as long as the user gazed within a 1.5 key radii of the center of the key. We also made the keyboard relatively small, measuring 15×6 cm on the screen. This allowed users to move between keys quickly with minimum eye movement. We also decided not to require users to go to the spacebar between words. Both continuous speech recognition and unconstrained handwriting recognition segment users' input into words without explicit demarcation.

2 Method

We recruited eight participants (4 male, 4 female) from the university campus. Their ages ranged from 20 to 29 (mean = 25.5, sd = 4.3). Participants used a Tobii P10 workstation with an integrated gaze-tracker running Windows XP. The physical screen size was 30×23 cm and the resolution was 1024×768 pixels. The on-screen QWERTY keyboard's physical size was 15×6 cm and its screen size was 500×200 pixels. We obtained eye-tracking samples by installing a callback function into the Tobii eye-tracker driver using the Tobii API. This ensured we received the raw and unsmoothed tracker signal. We wanted to use the raw signal because it eliminates lag when users are quickly changing their gaze. The eye-tracker had a sampling rate of 40 Hz and an accuracy of 0.5°. We used the Windows multimedia timer framework to timestamp all eye-tracking events with an accuracy of 3 ± 1 ms.

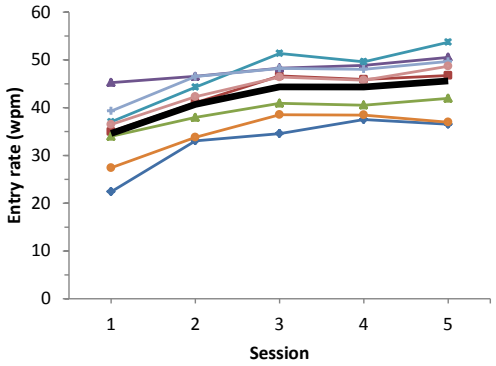

Figure 1: *Entry rate (wpm) for all participants in each session. The bold black line shows the overall mean.*

2.1 Procedure

Each participant was paid £15 to take part in a single 100-minute session. The participant first calibrated the eye-tracker. We used nine evenly distributed circular targets. The participant was asked to gaze at the targets in sequence. Calibration took place in the same screen region used during the typing experiment. After calibration, the participant completed five 10-minute typing sessions. Each session was separated by a 10-minute break. We decided on 10-minute breaks because a pilot study revealed that shorter breaks resulted in decreased performance during later sessions.

The experimental interface presented the user with a memorable phrase from the MacKenzie and Soukoreff [2003] phrase set. Each session used its own set of phrases and all participants wrote the same phrases in the same order. After reading the phrase, the participant pressed the spacebar key on the desktop keyboard. This caused the on-screen keyboard to appear. Similar to Majaranta et al. [2009], the phrase remained at the top of the interface as a reference. The participant was instructed to write the phrase as quickly as possible. When the participant gazed in the vicinity of the first letter key of the phrase, the letter turned red as a visual indication that the system had recognized that letter. Then the participant moved on to the next letter. This was repeated until the participant had completed the entire phrase. To complete the phrase the participant gazed at a result area positioned above the on-screen keyboard. After completion of a phrase, the participant was shown the entry rate in words-per-minute for that phrase.

3 Results

In total we recorded 400 minutes of eye-trace data. Participants entered a total of 2026 phrases. We measured the entry rate in words-per-minute (wpm). We used the standard definition of a word as five consecutive characters. Entry time was measured as the interval from when the user first gazed inside the keyboard to when the user gazed at the result area after having completed the stimulus phrase. While the users did not have to gaze at the spacebar, our entry rate calculation includes the spaces in the stimulus phrase.

The mean entry rate was 36 wpm in the first session and 46 wpm in the fifth session. Participants' entry rate improved in the first few sessions and then reached a plateau (Figure 1). The high entry rates demonstrate that a well-implemented dwell-free interface has the potential to dramatically increase entry rates compared to the current state-of-the-art. In the last 10-minute session, the fastest participant had a mean entry of 54 wpm, and the slowest participant had a mean entry rate of 37 wpm.

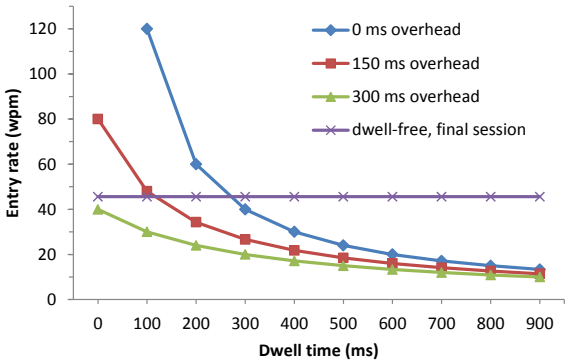

Figure 2: *Theoretical entry rates assuming different dwell-times and overhead times. The horizontal purple line shows the speed of users in the final session (46 wpm). We omit the data point for 0 ms overhead and 0 ms dwell time since its entry rate is infinite.*

We created a human performance model in order to better understand the limitations of standard dwell-based eye-typing. Conceptually, writing a single character using an eye-typing system can be broken down into two components: dwell-time and overhead time. The dwell-time is the duration the user has to gaze at a desired letter key in order to select it. The overhead time consists of the time to transition between keys and the time to perform any necessary error correction. Figure 2 shows theoretical entry rates at different dwell-times and at three different overhead times. Each individual data point on the plot shows the entry rate in wpm as a function of dwell-time plus overhead time. Our dwell-free experimental system is the horizontal purple line intersecting the y-axis at 46 wpm. It is constant in the plot since it is unaffected by dwell-time.

Typical dwell-times in the literature range from 400 to 1000 ms [Majaranta and Räihä 2002]. To get an estimate of a reasonable overhead time we consulted the study by Majaranta et al. [2009]. Using adjustable dwell-timeouts they obtained the fastest entry rates that have been reported for traditional eye-typing interfaces. In their study, participants had an entry rate of 19.8 wpm with a mean dwell-time of 282 ms in the final session. Based on these data points, we calculated an overhead time of 318 ms.

Figure 2 shows that our human performance estimate for dwell-free eye-typing is much faster than the entry rate at any reasonable combination of dwell-time and overhead time for traditional eye-typing. Assuming an overhead time of 300 ms (compared to 318 ms measured in [Majaranta et al. 2009]), even if the participants at maximum performance had reduced their dwell-time to 100 ms (compared to 282 ms measured in [Majaranta et al. 2009]), they would still be far below our human performance estimate of dwell-free eye-typing. Also, very low dwell-timeouts are likely to trigger many false key activations (the so-called "Midas touch" problem [Jacob and Karn 2003]). Assuming no dwell-time for 300 ms overhead (similar to [Majaranta et al. 2009]), our human performance estimate for dwell-free eye-typing performs about the same. This suggests that the overhead times observed by Majaranta et al. [2009] may be similar to what we measured in our experiment.

Figure 3 shows a heat map of where users were looking during the experiment. Unsurprisingly, the most likely letters in English, such as the letter "e", received the most gaze points. We also observed that users gazed in a relatively broad region around the keys. This may have been caused by inaccuracies in the eye-tracker or by users not gazing directly at the center of the keys. Analysis of the trace data showed that 51% of the time participants activated a key by gazing inside it. The remainder of the time participants' gaze passed within the 1.5 key radii threshold but never went inside the key. Figure 4 shows an example eye-trace from one participant. The figure demonstrates how the user serially fixated on each letter in the stimulus phrase.

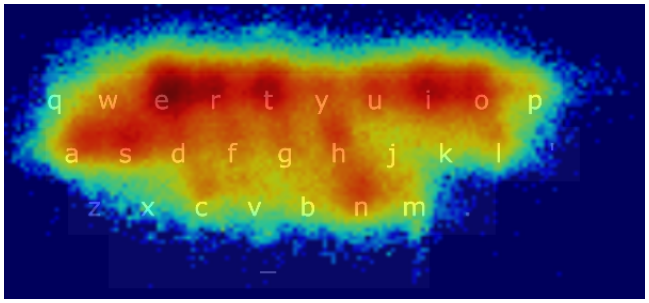

Figure 3: *Heat map of gaze locations during the experiment for all participants and all phrases.*

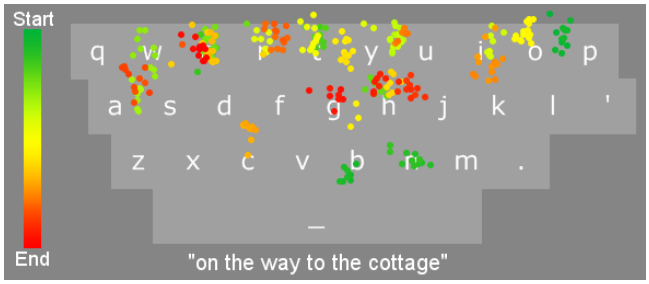

Figure 4: *One participant's sequence of gaze locations for the phrase "on the way to the cottage". Green points are at the start of the trace, red points are at the end of the trace.*

Finally, we collected subjective data from our participants (Figure 5). Participants were asked to rate a set of statements on a 1–7 Likert scale (1 = Strongly Disagree, 7 = Strongly Agree). Encouragingly, users liked the dwell-free eye-typing interface and thought it was fun to use. They also did not find it stressful. However, most users agreed that they needed to concentrate during the task. Participants also perceived they would improve with practice. Open comments were highly positive: "I found it really fun!" and "It feels like playing a computer game!".

4 Discussion

Several factors are likely to affect a human performance estimate of dwell-free eye-typing. First, similar to Majaranta et al. [2009], our stimulus phrase remained visible during the writing task. However, studies have shown that users tend to write faster if they are forced to memorize the phrase beforehand [Kristensson and Vertanen 2012; Soukoreff and MacKenzie 2003]. Second, our participants had to gaze at all letters in the phrases to proceed, which

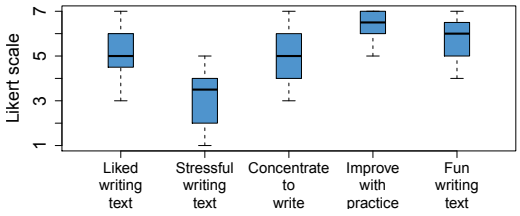

Figure 5: *Box-and-whisker plot of participants' ratings on a 1–7 Likert scale (1 = Strongly Disagree, 7 = Strongly Agree).*

means that their error rate was always zero. However, an actual implementation may be error-tolerant to spelling mistakes and may not necessarily require users to gaze in the vicinity of all the letters in the intended word. This is plausible given that a gesture keyboard can recognize a word without the gesture trace intersecting any of the letter keys for the intended word [Kristensson and Zhai 2004; Zhai and Kristensson 2003]. Third, our study ignored error correction. Error correction will reduce the entry rate. However, the desire to correct errors is likely task dependent. For example, users are probably willing to accept a much higher error rate when engaging in real-time face-to-face communication than when writing emails or essays.

While it may be challenging to create an efficient dwell-free eye-typing interface, prior work on gesture keyboard recognition [Kristensson and Zhai 2004] and speech recognition [Padmanabhan and Picheny 2002] have convincingly demonstrated that it is possible to infer users' intended text from very noisy signals. We believe a promising research direction would use hidden Markov models to model the noisy eye-tracker observations. This would be complemented by a domain-appropriate long-span language model, such as our recently published language model for augmentative and alternative communication [Vertanen and Kristensson 2011].

5 Conclusions

We have argued that existing eye-typing interfaces are close to their maximum speed. To realize significant further gains we need to explore different research directions. We have further argued that one promising direction is dwell-free eye-typing. As a first step we investigated how quickly users could write using a dwell-free eye-typing interface that simulated a perfect recognizer. We found that users reached a mean entry rate of 46 wpm after 50 minutes of practice. This is more than twice as fast as the entry rates observed with an eye-typing interface using adjustable dwell-time [Majaranta et al. 2009] and with the predictive zooming interface Dasher [Ward and MacKay 2002; Tuisku et al. 2008]. Further, by modeling traditional eye-typing performance as a combination of overhead time and dwell time, we demonstrated that traditional eye-typing systems are highly unlikely to ever reach the potential entry rates we observed for dwell-free eye-typing. To surpass our human performance estimate for dwell-free eye-typing a traditional eye-typing interface would either have to involve no overhead time (which is impossible) or use a dwell-timeout of 100 ms or less. However, the record-speeds observed in traditional eye-typing so far have had a mean dwell-timeout of 282 ms [Majaranta et al. 2009].

We believe dwell-free eye-typing is a necessary paradigm shift to enable further progress in gaze communication. This will require a substantial and collective undertaking by the eye-typing community. But this undertaking may well be worth the effort. Our results demonstrate that dwell-free eye-typing offers the potential for substantially faster gaze communication.

Acknowledgements

This work was supported by the Engineering and Physical Sciences Research Council (grant number EP/H027408/1) and the Scottish Informatics and Computer Science Alliance. We thank Tobii Technology AB for loaning us the eye-tracker used in the study.

References

JACOB, R. J., AND KARN, K. S. 2003. Eye-tracking in human-computer interaction and usability research: ready to deliver the promises. In *The Mind's Eye: Cognitive and Applied Aspects of Eye Movement Research*, J. Hyönä, R. Radach, and H. Deubel, Eds. Elsevier, 573–606.

KRISTENSSON, P. O., AND VERTANEN, K. 2012. Performance comparisons of phrase sets and presentation styles for text entry evaluations. In *Proceedings of the 17th ACM International Conference on Intelligent User Interfaces*, ACM Press, in press.

KRISTENSSON, P. O., AND ZHAI, S. 2004. SHARK^2: a large vocabulary shorthand writing system for pen-based computers. In *Proceedings of the 17th Annual ACM Symposium on User Interface Software and Technology*, ACM Press, 43–52.

MACKENZIE, I. S., AND SOUKOREFF, R. W. 2003. Phrase sets for evaluating text entry techniques. In *Extended Abstracts of the 21st ACM Conference on Human Factors in Computing Systems*, ACM Press, 754–755.

MAJARANTA, P., AND RÄIHÄ, K.-J. 2002. Twenty years of eye typing: systems and design issues. In *Proceedings of the 2nd ACM Symposium on Eye-Tracking Research & Applications*, ACM Press, 15–22.

MAJARANTA, P., AHOLA, U.-K., AND ŠPAKOV, O. 2009. Fast gaze typing with an adjustable dwell time. In *Proceedings of the 27th ACM Conference on Human Factors in Computing Systems*, ACM Press, 357–360.

PADMANABHAN, M., AND PICHENY, M. 2002. Large-vocabulary speech recognition algorithms. *IEEE Computer 35*, 4, 42–50.

SALVUCCI, D. D. 1999. Inferring intent in eye-based interfaces: tracing eye movements with process models. In *Proceedings of the 17th ACM Conference on Human Factors in Computing Systems*, ACM Press, 254–261.

SALVUCCI, D. D. 2000. An interactive model-based environment for eye-movement protocol analysis and visualization. In *Proceedings of the 1st ACM Symposium on Eye-Tracking Research & Applications*, ACM Press, 57–63.

SHANNON, C. E. 1948. A mathematical theory of communication. *Bell System Technical Journal 27*, 379–423,623–656.

SOUKOREFF, R. W., AND MACKENZIE, I. S. 2003. Metrics for text entry research: an evaluation of MSD and KSPC, and a new unified error metric. In *Proceedings of the 21st ACM Conference on Human Factors in Computing Systems*, ACM Press, 113–120.

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 *Proceedings of the 5th ACM Symposium on Eye-Tracking Research & Applications*, ACM Press, 19–26.

VERTANEN, K., AND KRISTENSSON, P. O. 2011. The imagination of crowds: conversational AAC language modeling using crowdsourcing and large data sources. In *Proceedings of the ACL Conference on Empirical Methods in Natural Language Processing*, ACL, 700–711.

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 *Proceedings of the 5th ACM Symposium on Eye-Tracking Research & Applications*, ACM Press, 11–18.

ZHAI, S., AND KRISTENSSON, P. O. 2003. Shorthand writing on stylus keyboard. In *Proceedings of the 21st ACM Conference on Human Factors in Computing Systems*, ACM Press, 97–104.