*Human-Computer Interaction -- INTERACT'03*
*M. Rauterberg et al. (Eds.)*
*Published by IOS Press, (c) IFIP, 2003, pp.121-128*

# Command Without a Click: Dwell Time Typing by Mouse and Gaze Selections

**John Paulin Hansen[1], Anders Sewerin Johansen[1], Dan Witzner Hansen[1],**

**Kenji Itoh[2] & Satoru Mashino[2]**

[1] The IT University of Copenhagen, Glentevej 67, 2400 Copenhagen, Denmark

[2]Tokyo Institute of Technology, 2-12-1 Oh-okayama Meguro-ku,

Tokyo 152-8552 Japan,

[1]{paulin, dduck, witzner }@it-c.dk, [2]ken@ie.me.titech.ac.jp

**Abstract.** With dwell time activation, completely hands free interaction may be achieved by tracking the user's gaze positions. The first study presented compares typing by mouse click with dwell time typing on Danish on-screen keyboard with 10 large buttons which change according to character prediction. The second study compares mouse and eye-gaze dwell input on a similar Japanese keyboard, but without dynamic changes. In the first study, dwell time selections tend to be a little slower and the overproduction is higher than with click selections. In the second study, mouse and gaze is almost equally fast, but mouse is far more precise than gaze. Consequently, the productivity in terms of characters per minute is 33% higher. The results suggest that users can be productive from the first encounter with dwell time activation, but productivity depends on their familiarity with the input structure and the input mode (i.e. hand or eye).

**Keywords:** interaction techniques, eye movements, keyboards, disability, user interfaces, typing, GOMS analysis

## 1 Introduction

Movements of the hand, torso, head or eye may control pointing in future systems, if they contain suitable tracking capabilities. For instance, an information kiosk behind a shop window may be operated from the outside by hand movements; head tracking may control pointing on head-up displays in vehicles, and eye tracking may control pointing in displays mounted on eyeglasses. In cases like these, the click function either requires an additional input channel (e.g., voice or button) or it will have to be conducted by additional actions (e.g., hand gestures or eye blinks).

Operating ubiquitous and mobile computing devices is often imprecise, error-prone and slow due to external disturbances and interruptions. As an example, it may be difficult to hit small command buttons with a laser pointer while walking around (Myers et. al. 2002). Hitting a button on the touch-screen panel of an in-vehicle information system while driving on a bumpy road is another example. An obvious solution to these problems would be to increase the size of the command input areas, but often the restricted size of mobile displays is a definitive constraint.

Dwell time selection seems to be a promising candidate as a substitute for the traditional mouse click in these scenarios. A dwell time principle makes a more careful selection of troublesome on-screen buttons possible by simply pointing at them for a certain time – continuously or accumulated over several hits. The dwell time period allows the user time to refine, abandon or adjust a selection within the delay time.

We are currently developing a gaze-based typing communication tool, "GazeTalk", designed for people with Amyotrophic Lateral Sclerosis (ALS) who have lost their voice and mobility and may only be able to move their eyes. In order to make the tool widely available, we use standard PCs and of-the-shelf digital camera technology to track eye positions (Hansen, 2000). Therefore we need large buttons on the screen. Moreover, we - like other gaze systems - have to deal with the "Midas Touch" problem: Everywhere you look, a command may get activated without your intention to do so. A dwell time principle has been the preferred solution to this problem since the very first eye-gaze communication systems (Majaranta & Räihä, 2002). Dwell times applied in various systems range from 100 milliseconds (ms) to 3000 ms depending on tracking performance, user skills, type of command and cost of errors.

# 2 Dwell Activation

There are 3 different methods by which dwell activation can be implemented:

*Continuous dwell activation:* A command is executed when a button has been continuously activated for a certain pre-set time. If the activation is terminated before a pre-set time, the time counter is reset. This type of dwell activation is found in most gaze-activated systems.

*Accumulated dwell activation:* A command is executed when a button has been activated for a certain pre-set time independent of the number of actual activations. The time (evidence) counter of the command may be reset by some local or global system event.

*Adaptive dwell activation:* The dwell time period (continuous or accumulated) becomes dependent on user behavior patterns. The patterns can be e.g. frequency of use, frequency of selection errors, number of activations terminated by the user, etc. Lesher et. al. (2000) developed a method for automatic, real-time adjustments of delays in a so-called row-column scanning interface. The delay dropped off quickly from a starting point at 2000 ms stabilizing at 270 milliseconds after approximately 1200 selections.

The remaining time before activation takes place can be indicated within the button as a progress bar or an animation sequence. Feedback can also be integrated by changing the pointer symbol or by auditory signals.

## 2.1 Mouse and Gaze Selection

In an early study, Ware & Mikalian (1987) found the average gaze selection time to be 950 ms, using a 400 ms dwell period compared to 675 ms if button clicks were used for confirmation.

The time difference may be explained in GOMS operator terms by:

$$\mathbf{t}_{(\text{gaze dwell selection})} = \mathbf{t}_{(\text{latency})} + \mathbf{t}_{(\text{eye movement})} + \mathbf{t}_{(\text{dwell time})} \quad (1.1)$$

versus:

$$\mathbf{t}_{(\text{gaze click selection})} = \mathbf{t}_{(\text{latency})} + \mathbf{t}_{(\text{eye movement})} + \mathbf{t}_{(\text{perceive target})} + \mathbf{t}_{(\text{click})} \quad (1.2)$$

Using **t** values from the GOMS literature (e.g. John & Kieras, 1996), the time difference may be illustrated by the following approximate figures:

$$\mathbf{t}_{(\text{gaze dwell selection})} = 250 \text{ ms} + 250 \text{ ms} + 400 \text{ ms} = 900 \text{ ms} \quad (1.3)$$

versus:

$$\mathbf{t}_{(\text{eye gaze click selection})} = 250 \text{ ms} + 250 \text{ ms} + 100 \text{ ms} + 100 \text{ ms} = 700 \text{ ms} \quad (1.4)$$

As suggested in (1.1), $\mathbf{t}_{(\text{dwell time})}$ subsume $\mathbf{t}_{(\text{perceive target})}$ in (1.2).

It can be seen that the time cost of a dwell time compared to click confirmations is:

$$\textbf{Cost} = \mathbf{t}_{(\text{dwell time})} - (t_{\text{perceiving target})} + t_{(\text{click})}) \quad (1.5)$$

The $\mathbf{t}_{(\text{perceive target})}$ may vary from 100 ms if perceiving a simple binary signal to 290 ms if perceiving a complex visual signal, according to John & Kieras (1996). Thus the cost of using dwell time activation can be expected to be relatively high for simple tasks when compared to complex search tasks, where low dwell time settings (e.g. < 300 ms) may be used almost without inducing additional time-consuming delays.

Sibert & Jacob (2000) found gaze selections to be faster than mouse selections (504 ms vs. 932 ms) in a simple-task experiment with a dwell time set at 150 ms. In a more complex task, they found the average gaze selection time to be 1103 ms and mouse selection time to be 1441 ms. In contrast, e.g. Miyoshi & Murata (2001) found gaze selection times to be 1050 ms in average, and mouse selections to be 790 ms using a 400 ms dwell period.

So there is no general agreement in the literature whether gaze dwell selection is indeed faster than mouse dwell selection. In theory, they should be as a major part of the mouse movement time is eliminated from the operator cycle. Once the target is located, the pointer is already there. Some of the disagreement may stem from the fact that studies confirming that gaze based interaction may be faster than mouse based interaction have used relatively simple, repetitive tasks without considering error rates and removing outliers from the data set (e.g. Sibert & Jacob, 2000). Error rates for gaze selections are generally reported to be rather high; (e.g.10% in Ware & Mikalian, (1987)).

*Human-Computer Interaction -- INTERACT'03*
*M. Rauterberg et al. (Eds.)*
*Published by IOS Press, (c) IFIP, 2003, pp.121-128*

In summary, results from previous experiments indicate that the use of dwell time selection may be attractive due to a modest additional time cost. However, the previous experiments have been conducted with very simple, repetitive and highly bounded tasks. It is an open issue how selections by dwell time affect more realistic and involved entry tasks like typing. It also becomes important to investigate the immediate usability of dwell time interaction if this principle is to be used in walk-up-and-use systems (e.g., information kiosks).

# 3 On-Screen Typing

On-screen (soft) keyboards are well-suited for hands free interaction (e.g., by head or eye movements), and they have been used for decades within the Augmentative and Alternative Communication (AAC) community. Character sets can be arranged in a traditional QWERTY order, in alphabetic order or according to frequency-of-use. Acceleration features include predictions of the most likely next characters and words or access to a dictionary of words related to a specific context (such as “dinnertime” or “shopping”).

Most gaze typing systems consist of an eye tracking system in combination with one of the standard AAC on-screen keyboards (e.g., “Point for Windows”, “Wivik” and others). Typing speed is often just a few words per minute (Majaranta & Räihä, 2002).

Salvucci (1999) has demonstrated the potential of fixation tracing by use of hidden Markov Models to predict selections on an on-screen keyboard. Inferring the most likely intended word from a gaze path on a sequence of key candidates compensated for the lack of precision in eye tracking systems and eliminated the need for a dwell time delay. He found that eye typing averaged 822 ms per character, which equals 14.6 word per minute. However, systems like this will be indistinct when confronted with misspellings and unknown words.

Ward & MacKay (2002) invented a novel data entry interface, named “Dasher”, in which selections have become a fully integrated part of a continuous search and navigation process. Users are reported to type by gaze at more than 20 words per minute after one hour of practice. Error rates were found to be less than 5% compared to error rates of approximately 20% for an on-screen keyboard.

In summary, on-screen gaze typing systems have been of great value to people with special needs for more than a decade, even though some of the systems have been slow. Recent experimental systems have demonstrated new ways to speed up typing significantly.

# 4 Experiments

During the development of our on-screen gaze typing system, we have conducted experiments and user evaluations to clarify the initial efficiency of the various designs considered (Johansen & Hansen, 2002). Our focus on initial performance stems from a concern that first impressions of system efficiency may have a major impact on the users’ determination to learn a new system. User motivation is of highest importance to AAC systems as they are often introduced in periods of a life crisis, e.g. during recovery from an accident or during a serious progressing disease such as ALS.

We conducted two experiments. In our first experiment we performed a between-subjects comparison of initial differences between mouse-click typing and mouse-dwell typing on a Danish configuration of the GazeTalk on-screen keyboard with letter and word prediction.

Our second experiment was a within-subjects study of a GazeTalk-based input system for Japanese Hiragana and Katakana (“Kana”) characters, which used the same basic layout and UI-elements as the Danish system. Each subject made dwell-time selections first using a mouse and then using an eye-tracker. The Japanese configuration had no prediction functions. Instead, it displayed the characters in a conventional, static, two-level hierarchical manner. The purpose of the second experiment was to compare mouse-dwell selections with eye-dwell selections at an initial performance level.

**4.1 Experiment 1: Dwell-Time versus Click**

*Subjects:* 25 non-disabled subjects (15 female and 10 male) volunteered to participate in the experiment. Mean subject age was 33.56 years, and mean PC/mouse experience was 10.28 years. All subjects had normal or corrected-to-normal vision.

*Apparatus:* 400MHz Pcs equipped with keyboard, mouse and a 17-inch color monitor (640x480 pixels) were used.

*Subjects’ tasks:* Each subject entered 20 Danish conversation-level sentences, which were dictated to them one at a time. The sentences were projected simultaneously on an overhead projector. The sentences consisted of 800 characters in total. Subjects were instructed to type as quickly and accurately as possible. The sentences were typed in on the system shown in Fig. 1.

The size of each button was approx. 8 cm by 8 cm, and the text field (top left corner) was 16 cm by 8 cm. Text entered into this field was displayed in a 12-point boldface Helvetica font. Two versions of the system were used: One configured for click activation (without the progress bar shown on the highlighted button in Fig. 1) and one configured for dwell time activation (with the progress bar shown). Dwell time was set at 500ms. Butons were highlighted when pointed to with the cursor.

| | | | |
|---|---|---|---|
| This is the text f_ | | A to Z | Backspace |
| [8 most likely words] | A | | O |
| Space | R | L | U |

**Figure 1**: Layout of the on-screen keyboard. The subject is typing "This is the text field". Letter and word predictions are refined continuously as the user types. The progress bar indicates the remaining time before the "I" button is activated by the dwell time selection system.

The primary letter-entry mode featured a dynamic keyboard with six buttons arranged in a three by two matrix, which allowed the user to type the currently most likely six letters directly. The built-in context-sensitive letter prediction algorithm was used to supply the six most likely letters for the dynamic keyboard. They were then placed with the most probable suggestion in the center position ('I' in Fig 1), and the other suggestions placed according to probability in a clock-wise fashion around the center position ('O', 'U', 'L', 'R' and 'A' in Fig. 1). This was done on the assumption that users would quickly learn to anticipate the placement of the desired letter and then – in case the letter prediction did not supply this as the primary candidate –be able to evaluate the other candidates with a minimum of eye movement.

Furthermore, this mode featured buttons for backspace and space as well as buttons for access to word prediction/completion mode and an alphabetical letter entry mode ("A to Z"). The word prediction/completion mode presented the current eight most likely words (the actual words are shown on the "eight most likely words" button) in a four by two matrix and featured buttons for access to alphabetical letter entry mode and the primary letter entry mode. The alphabetical letter entry mode enabled the user to select the desired letter in a two-stage process first by selecting a group of letters (e.g., "ABCDEFG") containing the desired letter, and then by selecting the letter. Learning features of the word prediction system were not enabled. On average, input of a single character required approximately one selection.

*Procedure:* The 25 subjects were randomly divided into two groups of 12 and 13 subjects. The groups were placed in separate classrooms. Each subject was seated in front of a PC with the system turned on and ready for text entry. They adjusted the distance from their head to the screen to 50 cm. The session leader gave a short, oral introduction to the use of the system, and the subjects were allowed five minutes of free-form experimentation to get acquainted with it. Then the experiment began. The session leader read aloud each sentence one at a time, displayed it on the overhead and waited for all of the subjects to type it in before reading the next sentence aloud. The session lasted approx. 1 hour.

*Data analysis:* Only results from completely typed-in sentences typed were included in the analysis. Minor spelling mistakes were ignored. Overall typing time for each sentence measured from the time of the first keystroke to the final punctuation mark was recorded for each subject, and the average time for each selection was determined. A word per minute (WPM) rate was calculated for each subject assuming a standard word-length of five characters (including space). Instead of counting errors, we decided to calculate an overproduction rate since the subjects would often choose a strategy that was not really wrong but just sub-optimal in terms of the number of keystrokes used. For example, some subjects would choose a full word from the list of word predictions even though it had a wrong ending, delete the last characters of the word and then type in the right ending. Overproduction rates were calculated as the number of activations made by the subject minus the minimum number of required activations for the text that was typed divided by the minimum number of required activations. Finally, changes in the overproduction rate were calculated for each subject by comparing the overproduction rate of the first five sentences typed with the overproduction rate of the last five sentences typed. The selection method (click or dwell-time) was treated in an analysis of variance (ANOVA). A Scheffe post hoc test was then conducted.

*Human-Computer Interaction -- INTERACT'03*
*M. Rauterberg et al. (Eds.)*
*Published by IOS Press, (c) IFIP, 2003, pp.121-128*

*Results:* Table 1 summarizes the findings of Experiment 1. There was no significant difference between the two groups on the major variables; the typing rate (WPM) was very close to being significant F(1.23)= 4.09, p = 0.054. No correlation between the variables and age or PC/mouse experience was found.

| | N | Sec. per selec-tion | Word per min. | Over-prod. rate | Change in over-prod. rate |
|---|---|---|---|---|---|
| Mouse click | 12 | 1.87 | 5.51 | 16.9 % | 2.09 % |
| Mouse dwell | 13 | 2.03 | 4.79 | 26.2 % | 17.3 % |
| Difference | | -0.15 | 0.72 | -9.4 % | -15.2 % |
| Std. Error | | 0.11 | 0.36 | 5.17 | 7.81 |
| p-value | | 0.18 | 0.05 | 0.08 | 0.07 |

**Table 1:** The difference between mouse click and mouse dwell (=500ms) selections.

Individual results showed that everybody except for two subjects increased their selection speed during the experiment; in average the click activation group showed a 16% increase in selection speed from the first five sentences to the last five sentences, and the dwell time activation group increased their speed by 24%.

*Discussion of Experiment 1:*

On average the use of dwell time activation added approx. 150 ms to each selection when compared to click activation. This indicates that a short dwell time can be used without major additional cost on response time as predicted in (1.5).

Milliseconds matter when they are part of a repetitive operation such as typing letters and so does the cost of correcting errors. The data on overproduction rates from Experiment 1 suggests that use of the dwell time activation method introduced more erroneous actions and less efficient strategies. This was against expectations: In theory, dwell selections should give the user a certain time to cancel or adjust the selection whereas clicks are definitive actions. Most of the dwell time activation subjects explained that it took them some time to get used to the fact that they could not just leave the mouse pointer anywhere on the screen as they were used to. If they did not “park” it carefully in the text field, it would cause the button below it to become activated – i.e., a mouse-mode version of the “Midas touch” problem. The relatively high change (drop) in overproduction rates for mouse-dwell selections indicates that this is a transient problem. Quite a large group of subjects found the dwell time to be too short especially in the beginning of the experiment. Most likely, many of these initial errors would have been avoided had the dwell time been longer, (e.g., 750 ms) or adaptive instead of continuous.

### 4.2 Experiment 2: Mouse Dwell versus Eye Dwell Selection

*Subjects:* Twelve non-disabled Japanese students (four female and eight male) participated in the experiment. They were paid 1000 Japanese yen (approx. 8 US dollars) per hour for participating. Mean subject age was 21.5 years. All subjects had normal or corrected-to-normal vision and were everyday users of PCs.

*Apparatus:* Two sets of 500MHz PCs equipped with mouse and a 17-inch color monitor (640x480 pixels) were used in the mouse dwell typing sessions. In eye-tracking typing sessions, one of these PCs was equipped with a remote sensing infrared-based eye tracker (a Quick Glance system from EyeTech Digital Systems Inc). The sampling rate of the eye tracker was set at 15 frames per second and regularized over seven frames for noise reduction (blinking, data loss, etc.). This caused an additional delay of approx. 200 ms for the pointer symbol to move. The pointer symbol was visible at all times.

*Subjects' tasks:* Each subject typed 10 Japanese sentences composed by Kana characters in each block dictated by an experimenter one at a time. Kanji (Chinese) ideograms were not used as the system only supported Kana input. Sentences were at a daily conversation level of complexity. In mouse typing sessions, the sentences were visible to the subjects on a piece of paper located close to the monitor. In the eye-tracking session, the experimenter would read the sentence again when a subject needed to get it repeated though it only happened a few times. As in Experiment 1, subjects were instructed to type as quickly and accurately as possible.

The sentences were typed in on an on-screen keyboard with the same display layout as in Experiment 1, but with a Kana character on each button instead of Roman letters (cf. Fig. 2). The Japanese system did not feature a dynamic keyboard or letter/sequence prediction. Dwell time was set at

500 ms for both mouse and eye-tracking inputs for all the subjects. On average, input of a single character required approx. two selections from the hierarchical levels of key menus.

*Procedure:* Experimental sessions were composed of twelve blocks with mouse input and subsequently nine blocks with gaze input. The mouse input experiment was conducted in two sessions, and the subjects participated in six blocks in each session. The eye-tracking trials were divided into three sessions each consisting of three experimental blocks. It took a subject 60 to 90 minutes to complete a session. On the first day, before the experimental session, the subjects were briefed on the experimental procedure and the use of the input system and they got approx. five minutes to get acquainted with the system. The subjects had a five-minute break after every two blocks in the mouse sessions and after each block in the eye-tracking sessions.

| | N | Sec. per selec-tion | Char. per min. | Over-prod. rate | Change in over-prod. rate |
|---|---|---|---|---|---|
| Mouse dwell | 12 | 1.27 | 22.05 | 5,7 % | 3.2 % |
| Eye dwell | 12 | 1.41 | 16.55 | 27.9 % | 20.7 % |
| Difference | | -0.15 | 5.50 | -22.2 % | -17.5 % |
| Std. Error | | 0.04 | 0.84 | 2.96 | 2.59 |
| p-value | | 0.02 | <0.001 | <0.001 | 0.03 |

**Table 2:** The difference between mouse dwell (=500 ms) and eye dwell (=500 ms) selections.

*Data analysis:* Performance measurements and data analysis were identical to the ones used in Experiment 1 except for the fact that we calculated characters per minute (CPM) for the Japanese text input rather than WPM due to the syntax of written Japanese.

*Results:* Results from Experiment 2 are shown in Table 2. Results were only calculated on basis of the first three blocks of each condition to be comparable with the results from Experiment 1. As seen in this table, there are significant differences in the efficiency measures between the two input modes. There was a significant difference on mean selection time ($F(1.22) = 6.24$, $p = 0.02$); on characters per minute ($F(1.22) = 16.66$, $p = 0.0005$); on overproduction rate ($F(1.22) = 27.04$, $p = 0.00003$) and on changes in overproduction rate ($F(1.22) = 5.44$, $p = 0.03$).

Overall, the difference in CPM between mouse and gaze was 5.50. This indicates that mouse dwell interaction is 33% more efficient than gaze dwell interaction. This difference was primarily caused by the fact that mouse-dwell interaction was much more efficient in terms of less overproduction.

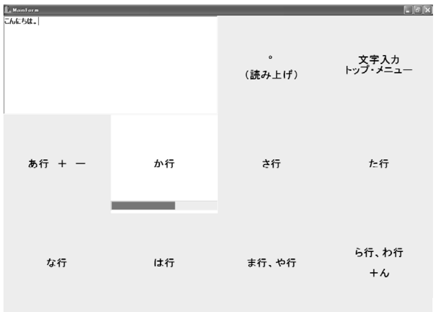


**Figure 2:** Japanese version of GazeTalk

*Discussion of Experiment 2:*
In Experiment 2, average gaze-dwell selection was found to be 150 ms slower than mouse dwell selections. This is in contrast with the findings of Sibert and Jacob (2000) which showed that eye-tracking dwell selections were 428 ms faster than mouse-click selections in a simple task and 338 ms faster in a complex selection task. However, Sibert and Jacob (2000) only included performance on correct trails in their results, while Experiment 2 also included trails with typing corrections, equipment problems, etc., that were far more common under the eye-tracking condition. Evidence of time-consuming interaction problems can be found in the high overproduction rate for gaze-dwell selections, 27.9% compared to the 5.7% overproduction under the mouse-dwell condition.

Subjects explained that they were particularly annoyed when they noticed that the pointer symbol failed to follow their fixation points accurately. All subjects reported that the 500 ms dwell time was too short for gaze-based interaction, and they felt they were not always able to search for the correct button without occasional unintentional activations. Some subjects developed a strategy where they would fixate at a neutral position while they imagined where the target button would be located before they moved their eyes to that position. This will of course slow down the overall typing rate and may also cause additional false activations.

*Human-Computer Interaction -- INTERACT'03*
*M. Rauterberg et al. (Eds.)*
*Published by IOS Press, (c) IFIP, 2003, pp.121-128*

The approx. 200 ms delay in eye tracking system response caused a noticeable lag between movement of eyes and the pointer symbol which may have strengthened the impression that the dwell time period was too short. When the pointer symbol eventually jumped to the fixation point, the countdown for activation had already passed approx. 200 ms.

The progress bar (cf. Fig. 1) may not have been helpful for short (and lagging) dwell times. The bar may induce a time pressure on the user, and because it was placed close to the bottom of the button, it may actually have caused false activations of the button below it if the user followed the movements of the bar when the eye tracker was slightly miss-calibrated.

*Comparison between Experiment 1 & Experiment 2:* Mean selection time for the mouse-dwell conditions were 2.03 seconds in Experiment 1 but only 1.27 in Experiment 2. This difference is significant; $F(1.23) = 68.60$, $p < 0.0001$.

Based on a comparison of semantically identical sentences in Danish and Japanese, we have estimated that it takes two Japanese characters to produce the equivalent of one Danish word. Using this ratio, WPM can be compared to CPM. There was a significant difference between the Danish average production of 4.78 WPM and a Japanese average production of 22.05 CPM/2, $F(1.23) = 186.5$ $p < 0.0001$.

The difference in overproduction (26.2% versus 5.7%) was also significant; $F(1.23) = 22.50$, $p < 0.0001$. These differences indicate that typing on keyboards with character prediction is less productive in the initial phase when compared to static displays with familiar key positions.

# 5 General Discussion

AAC has been pushing the development of hands free typing systems for more than a decade. Mobile text entry may become the next driving force (Johansen & Hansen, 2002). Future user scenarios include doctors interacting with computerized patient records on micro-displays mounted on their eyeglasses while examining the patients or users responding to context-specific information triggered by their current location while they walk down a hallway. In these cases, small-size input displays, motion and noisy tracking conditions will be a challenge to the design of a fast and robust interaction. Similar challenges have been overcome by gaze-operated systems using a dwell time principle. The present experiments suggest that users in general can be productive from the very first encounter with this principle. This confirms the observations made by Glenstrup and Engell-Nielsen (1995) that most visitors could just start using a gaze-dwell operated information kiosk without any further instructions once the system had been calibrated to their eyes.

Dwell time periods for novice users should preferably be longer than the 500 ms used in the present experiments. They should be adjustable and adaptive. Buttons should be given an individual dwell time, reflecting the differences in the time it takes to perceive e.g. individual letters, special characters and full words. The differences in dwell time may also depend on the achieved precision of the tracking system or on the predicted likelihood of a particular selection. Adaptive dwell time settings have been successfully applied in other AAC-systems (Lesher et. al. (2000)) and we intend to make them work for gaze interaction as well.

We suggest that feedback should be provided to the user in a discrete manner that does not interfere with target detection. Reports from users in Experiment 2 indicated that visual feedback on remaining dwell time is counterproductive when the dwell time is low, and subjects also strongly advocated that the pointer symbol should be invisible.

The problems experienced by initial users of highly dynamic displays (as indicated by the markedly lower productivity of Danish users who used the predictive layout when compared to the Japanese users who used a static layout) are most likely caused by confusion with regard to the position of the desired letter. We assumed that the users would appreciate that the letters were positioned according to their current probability, but that turned out not to be the case. Apparently, this placement strategy was wrong and prevented the users from developing an efficient strategy for locating and thus selecting letters as they found themselves constantly scanning the on-screen keyboard in order to locate the desired letter. This is evident from this post-experiment comment from one of the subjects:

*"It's a bit confusing that the letters change place after each selection. You spot the 'T' in one position, and then you have to find it all over again after the next selection"*

Obviously, this scanning is time-consuming, and gives rise to frustration. The design implications are that reduced keyboards should be regular and invariant in the dynamics whenever possible. As a consequence, we decided to assign "home positions" for all letters in the next prototype. Thus, a "T" will now occur on the same button whenever the prediction algorithm determines that it is a likely candidate for the next letter unless another letter that shares the same home position has a higher probability.

Even in laboratory settings, lack of precision and calibration failures are common problems associated with eye-tracking technology. For instance, in Sibert and Jacob's (2000) experiment, the tracking system only worked well for 16 out of 26 subjects tested. This is a most critical issue if gaze interaction is to succeed under demanding mobile conditions. New gaze-typing methods without dwell time (Salvucci (1999); Ward & MacKay (2002)) work well in a stationary setup. For hands free gaze interactions in mobile scenarios, the interface cannot require a high precision in pointing. On basis of our experiments, we believe dwell time activation of large on-screen buttons to be a feasible solution.

# 6 Conclusion

Dwell time activation can be used to improve the robustness of button selections with a modest increase in selection times. Initial users are likely to be less productive with dwell time interaction compared to button clicks, but the results indicate that the overproduction is likely to be transitive. Gaze-based dwell time interactions were found to be significantly less efficient than mouse dwell time interactions, but deficiencies in the design of the interface and tracking failures are primary causes for the problems observed. Finally, novice users typing on a dynamic keyboard seems to be less efficient than novice users typing on a keyboard with a fixed, familiar structure. Dynamic keyboards should thus show regularities and invariants whenever possible.

# References


Glenstrup, Arne. J. & Engell-Nielsen, T. (1995): Eye Controlled Media: Present and Future State. *Bachelor's Thesis, Information Psychology, University of Copenhagen.* Available at: http://www.diku.dk/~panic/eyegaze/article.html

Hansen, D. Witzner, Hansen, John P., Nielsen, Mads, Johansen, Anders S. & Stegmann, Mikkel B. (2002), Eye Typing using Markov and Active Appearance Models, *IEEE Workshop on Applications on Computer Vision*, 132-136.

Johansen, Anders S. & Hansen, John P. (2002), Augmentative and alternative communication: The future of text on the move. *Proceeding at 7th ERCIM Workshop "User Interfaces for all"*, Paris (Chantilly), 367-386.

John, Bonnie E. & Kieras, David E. (1996), The GOMS Family of User Interface Analysis Techniques: Comparison and Contrast, *ACM Transactions on Computer-Human Interaction, Vol. 3, No. 4, December 1996, 320-351*.

Lesher, G.W., Higginbotham, D.J., & Moulton, B.J. (2000), Techniques for automatically updating scanning delays. *Proceedings of the RESNA 2000 Annual Conference*, 85-87.

Majaranta, P. & Räihä, K. J. (2002), Twenty Years of Eye Typing: Systems and Design Issues, *Proceedings of the Symposium on ETRA 2002: Eye Tracking Research & Applications Symposium 2002,* New Orleans, 15– 22.

Miyoshi, T. & Murata, A. (2001), Input Device Using Eye Tracker in Human-Computer Interaction, *IEEE International Workshop on Robot and Human Interactive Communication*, 580-585.

Myers, Brad A., Bhatnagar, Rishi, Nichols, Jeffrey, Peck, Choon Hong, Kong, Dave, Miller, Robert & Long, A. Chris (2002), Interacting at a distance: measuring the performance of laser pointers and other devices. *Pros. of ACM CHI 2002,* 33-40.

Salvucci, D. D. (1999), Inferring intent in eye-movement interfaces: Tracing user actions with process models, *Human Factors in Computing Systems: CH1 99 Conference Proceedings,* Pittsburgh, PA, ACM Press, 254-261.

Sibert, L. E. & Jacob, R. J. K. (2000), Evaluation of Eye Gaze Interaction, *Human Factors in Computing Systems: CH1 2000 Conference Proceedings*, The Hague, Amsterdam., 281 – 288.

Ward, David J. & MacKay, David J.C., (2002), Fast Hands-free Writing by Gaze Direction, Nature 418, p. 838 (22nd August 2002).

Ware, C. & Mikalian, H.T. (1987), An Evaluation of an Eye Tracker as a Device for Computer Input, *Human Factors in Computing Systems: CHI+Gl'87 Conference Proceedings*, Toronto, Canada, 183-188.