Dynamic and Meta-Context Switching for Gaze-Based Interaction

Antonio Diaz-Tula Carlos H. Morimoto
Department of Computer Science
Institute of Mathematics and Statistics
University of São Paulo
{diaztula,hitoshi}@ime.usp.br

***Abstract*—In this paper we investigate the performance of a gaze-based interaction system that combines Dynamic Context Switching and Meta-Context Switching. These methods are extensions of the Context Switching interaction paradigm. The original context switching idea uses fixed-size contexts. Each context carries the same information, so the user can browse freely within a context without worrying about the Midas touch problem. A saccade to the other context triggers the selection of the item under focus. Dynamic context switching dynamically adjusts the size of a context to improve its useful area, where the context that has the user focus is displayed in full size and the other is minimized. Meta-context switching uses meta-keys to allow the user to escape from the current task and select other contexts or change the operation mode. We have designed and conducted two user experiments to evaluate these new gaze interaction techniques and compare them with selection by dwell time in a search task. The task required browsing through several pages using meta-keys. The experimental results show that dynamic context switching improves user performance when compared to fixed-size context switching and do not cause disorientation. The error rate was significantly higher for dwell time due to the Midas touch problem, although the time spent to complete the task was similar for dwell time and dynamic context switching.**

***Index Terms*—gaze-based interaction, dynamic context switching, meta-keys, gaze gesture.**

I. Introduction

Interacting with a computer using eye movements is possible thanks to the development of eye trackers [1]. Eye trackers are devices that track people's eyes and, after a calibration process, estimates the point being observed on a computer monitor. For people with physical disabilities, such as Amyotrophic Lateral Sclerosis and Locked-in Syndrome, gaze interaction represents an opportunity to communicate with the world.

Pointing to an object using gaze can be made naturally by associating the observed point in the monitor with the visual elements. Nonetheless, selecting or "clicking" objects using gaze is still a challenge. Jacob [2] was one of the first to point out the problem of accidental command activation in gaze interaction. This problem, known as the "Midas touch", refers to the selection of any observed object, even if the user did not intend to select it.

To reduce the effect of the Midas touch, researchers have proposed the use of different eye movements for gaze interaction, such as fixations, saccades, gaze gestures, and blinks, as described by Møllenbach *et al.* [3]. The most common way to make selections in gaze interaction is by using fixations, known as *dwell time* [4]. In a dwell time based interface, to select an object (for example, a key in a virtual keyboard) the user must fixate it for a predefined period of time. Examples of interfaces based on dwell time are ERICA [5] and GazeTalk [6].

The fixation time required for key activation is a workaround to reduce the Midas touch problem. Nonetheless, selecting the most appropriate dwell time is not straightforward. Shorter dwell times could improve performance, however they also increase the effect of the Midas touch, making the interaction stressful. On the other hand, longer dwell times reduce accidental selections, but the interaction becomes slower. Longer dwell times could also produce eye fatigue.

Špakov and Miniotas developed an algorithm for on-line adjustment of dwell time, by analyzing the exit time after selection of a key [7]. Though results were good, the system responded slowly to rapid changes in the typing speed and to the involuntary variation in exit time of the users.

Manual adjustment of dwell time using dedicated keys was studied by Majaranta *et al.* [8] and Räihä and Ovaska [9]. The performance obtained with this approach was better than with fixed dwell times. However, users needed to explicitly configure the fixation time to adjust their typing speed. Furthermore, once a dwell time is selected, the user is forced to type at that speed until a new dwell time is explicitly configured.

Discrete and continuous gaze gestures are alternatives to selection by dwell time. Discrete gaze gestures are defined by a sequence of eye movements that defines a unique pattern. An example of an interface based on discrete gaze gestures is EyeWrite [10]. In EyeWrite users type letters by making a sequence of saccades between "hot-spots" constituted by the four corners and the center of the 100 × 100 pixels window. Each letter is associated with a given gesture sequence. Another interface similar to EyeWrite, but with 9 hot-spots instead of 5, is EyeS [11].

Interfaces based on discrete gaze gestures have two main advantages: they eliminate the effect of the Midas touch, and require a smaller screen space than virtual keyboards. Nonetheless, complex gaze gestures could be difficult to learn and remember. Another disadvantage is that, because each gesture is composed by several saccades between the hot-spots, performance is worse than virtual keyboards [10].

Continuous gaze gestures require the user to navigate through a set of items to select the desired one. Examples of such interfaces are Dasher [12] and StarGazer [13]. In Dasher, the user must follow the desired character with her gaze, while the character bar moves from the left to the center of the screen. Selection is completed when the character crosses the horizontal line at the center of the screen. By using a language model, Dasher shows the next most probable characters closer to the selection area, hence improving performance. The main problem with this kind of interaction is that the gaze is always controlling the interface. Any unintentional change in the direction of gaze also results in a response from the interface.

A recent interaction technique, called Pursuits [14], is based on smooth pursuits to select objects. Smooth pursuits are movements where the gaze follows a slow moving object in the scene. In Pursuits, virtual objects are constantly moving following different paths. Selection is made when the user follows the desired object with the gaze for a given period of time. One advantage of this technique is that the eye tracker does not need to be calibrated. However, the number of objects in the screen is limited to about eight.

Context Switching [15] is a selection paradigm for gaze-based interfaces, proposed as an alternative to dwell time and gaze gestures. The method consists of two identical regions called "contexts". To make a selection, the user must focus on the desired object within one of the contexts, and then make a saccade to the other context. In this paradigm the user is free to explore the interface (since selections are triggered by switching contexts), thus reducing the Midas touch problem.

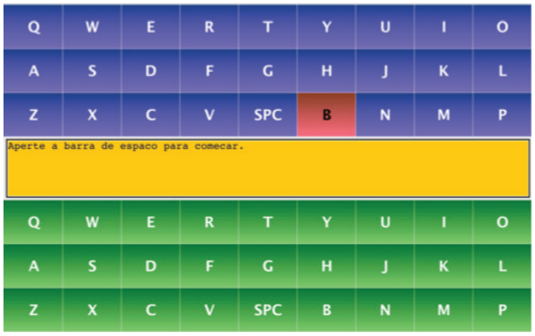
To improve the context switching paradigm, two extensions have been proposed: the use of contexts with dynamic sizes [16] and the use of meta-keys as escape mechanism to extend the functionality of the interfaces [17]. The purpose of dynamic context switching is to increment the useful screen space by dynamically adjusting the size of the contexts. Meta-keys are a further generalization, that makes the paradigm appropriate for much more general interfaces, allowing to switch between different tasks, navigate a collection of elements, or change the operation mode.

In this paper we investigate these two extensions of the context switching and compare them with a dwell-time gaze interface in terms of performance and error rate. The next Section introduces the dynamic context switching and meta-keys extensions.

II. Dynamic and Meta-Context Switching

In context switching [15] only one saccade per selection is needed. Objects are arranged within areas called contexts, that are separated by a "bridge". Objects within a context receive the focus after a short fixation (about 150 ms). Selection is made by saccading to the other context, crossing the bridge entirely. The bridge helps to avoid accidental selections as well as to reduce the effect of the eye tracker noise. It can also be used to show information to the user, like the typed text or the selected items.

Different from the dwell time paradigm, context switching clearly separates focus and selection, associating focus to eye fixations and selection to saccades. As a result, users can naturally adjust their selection speed without the need of configuring any other parameter. Figure 1 shows a virtual keyboard based on this paradigm.



Q	W	E	R	T	Y	U	I	O
A	S	D	F	G	H	J	K	L
Z	X	C	V	SPC	B	N	M	P

Aperte a barra de espaco para comecar.

Q	W	E	R	T	Y	U	I	O
A	S	D	F	G	H	J	K	L
Z	X	C	V	SPC	B	N	M	P

Fig. 1. Virtual keyboard for eye typing based in the context switching paradigm, image reproduced from [15]

A. Dynamic Context Switching

To improve the use of screen space, Tula *et al.* [16] proposed that contexts can be dynamically resized. Thus, the context that receives the user focus is maximized, while the other context is minimized. This would permit, in principle, to display only one context at a time. Because resizing may be disorienting, the contexts do not need to be fully maximized or minimized, but their sizes can be adjusted dynamically. Therefore, the context that has the focus is bigger than the other one, thus having more useful space. As soon as the user switches contexts, the size of the contexts is properly updated.

Using dynamic contexts, applications can present more objects on the screen compared with fixed-size contexts. For example, in a virtual keyboard it would be possible to include punctuation, numeric keypads, and/or symbols. Another advantage of dynamic context switching is that keys could be made bigger, in order the work well with less accurate eye trackers.

B. Meta-Context Switching

No matter how big the screen is, there will be applications that require more space than you have available. Tula and Morimoto [17] proposed the use of meta-keys, or escape keys, to move between different spaces or regions, called meta-contexts.

Consider the "crossing of a bridge" metaphor, where each bridge takes you to a different place. For an eye typing application for example, the main task is to type characters, but we might need to change the keyboard (caps lock, numeric, etc), or do file operations (save, open, close etc). We propose that these secondary tasks can be made available using meta-keys.

Porta and Turina [11] have proposed the use of gaze gestures for general-purpose commands. Complex gestures, like those used in EyeWrite [10] and Eye-S [11], could be difficult to

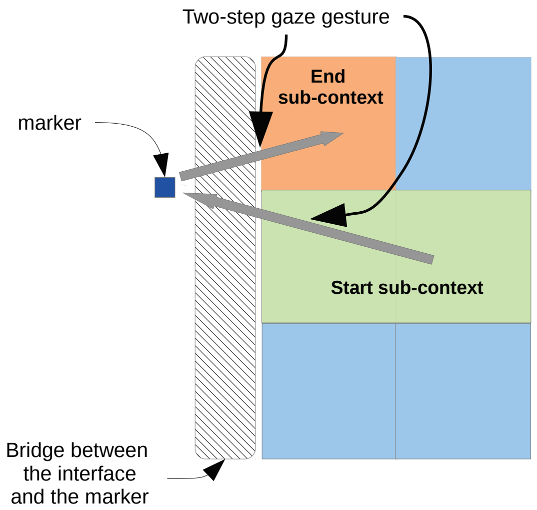


Fig. 2. Graphical representation of the meta-context switching. The two-steps gaze-gesture begins in a sub-context, then goes to the marker, and goes back to an adjacent sub-context, crossing the bridge twice.

learn and cause eye fatigue [11]. On the other hand, single-stroke gaze-gestures [3] could be activated while the user is exploring the interface.

To activate a meta-key, in [17] it was suggested to use more than one saccade for the less frequent operations, such as the secondary tasks described before. Meta-keys were placed near the edges of the screen, and use subregions of a context (such as a row or column) to define "contexts" for the meta-keys, or sub-contexts. The selection of a meta-key requires leaving a sub-context, fixating on a meta-key, and saccading to an adjacent sub-context, like "crossing two bridges" instead of one. One from the sub-context to the meta-key, and another bridge from the meta-key to an adjacent sub-context. Figure 2 shows a graphical representation of the meta-context switching. The two-steps gaze-gesture begins in a sub-context, then goes to the marker, and goes back to an adjacent sub-context, crossing the bridge twice.

Meta-context switching is clearly an eye gesture, but its main feature is that the saccades must be very short, "crossing" between sub-contexts.

We conducted an user experiment to compare the performance of dynamic context switching and dwell time, and to evaluate the combinaton of dwell time with meta-keys. To analyze the data, we adapted the metrics defined in [16] and [17], as described in the next Section.

III. Experiment 1: Comparing fixed-size and dynamic contexts

The objective of Experiment 1, as described in [16], [17], is to compare the performance of dynamic contexts with fixed-size contexts in a multiple selection task, common in real world applications, such as navigating a collection of pictures or multimedia objects. Another objective is to evaluate the use of meta-keys for navigation and command activation, in a task that required browsing through several pages of items.

A. Method

1) Participants

A total of 6 people participated in this experiment. They were all male, able-bodied, with normal or corrected to normal vision. Two of them had never used an eye tracker before, two had already participated in at least one study with eye trackers, and the other were experienced in developing eye trackers and gaze interaction studies. All participants had at least 10 years of experience using computers. They were all students or researchers at the University of São Paulo, aging from 21 to 45 years old.

2) Apparatus

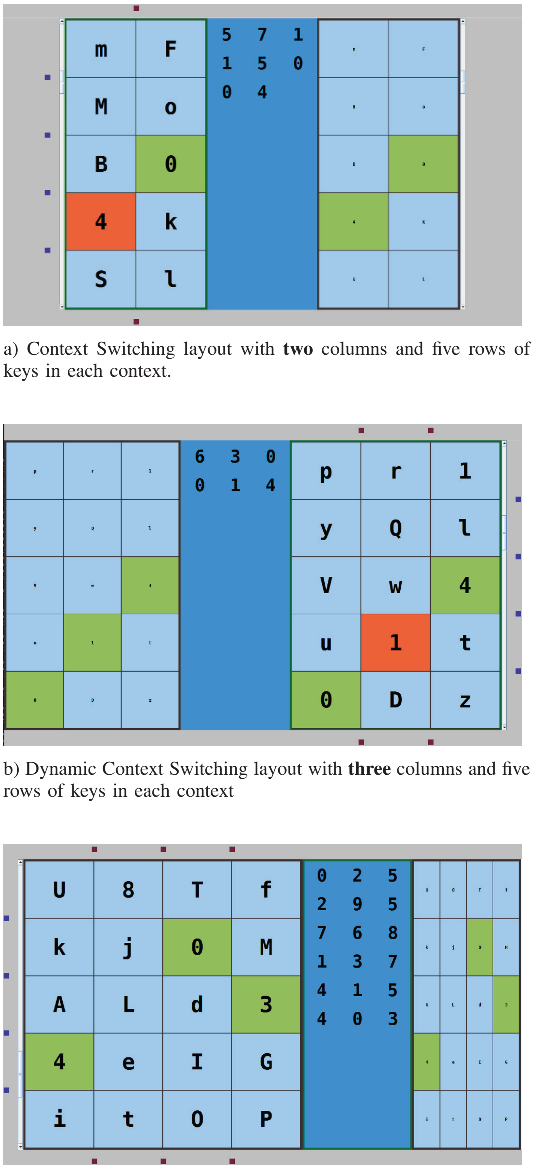
A low-cost, pupil-corneal reflexion eye tracker described by Morimoto et al. [18] was used during the experiment. The eye tracker runs at 30 Hz and has about 1^o in visual angle of spatial accuracy.

Figure 3 shows the three different layouts that were developed for the experiment. All layouts are based on the context switching paradigm, and have two contexts arranged horizontally, with the bridge in between, as shown in Figure 3. The layouts had two columns (2C) of keys in both contexts, shown in Figure 3a, three columns (3C) of keys in both contexts, as can be seen in Figure 3b, and four columns (4C) of keys in both contexts, shown in Figure 3c. All layouts had five rows of keys. Hence, in the 2C layout contexts have 10 keys, in the 3C contexts have 15 keys (50% more than in 2C), and in the 4C layout contexts have 20 keys (100% more than in 2C).

In the 2C configuration the size of the contexts was kept constant. In the 3C and 4C, the context with the focus was displayed in full size, while the other context was smaller. The keys and the bridge were kept with constant size in all configurations. The bridge was used to present the selected items. A short dwell time of 150 ms was used for detecting focus on a virtual key, and the maximum time for selection by context switching (i.e. maximum duration for the saccade) was set to 450 ms for the 2C and 3C configurations, and to 550 ms in the 4C configuration (because on average this layout requires longer saccades to switch contexts).

Selections were made using horizontal saccades, that are faster and more natural than vertical ones [19]. A green border was painted around the context with the user focus (gaze). Unselected keys within contexts were painted light blue. A blue key turned yellow when it received the focus, indicating that it could be selected. After selection, a key was painted green in both contexts. Correcting a wrong selection was possible by (de)selecting an already selected key. A green key turned orange when it received the focus, indicating that it could be deselected.

Meta-keys along the vertical edges of the contexts were used to show up a menu with options to undo the last selection and start/finish each session. An example of the menu is shown



a) Context Switching layout with **two** columns and five rows of keys in each context.

b) Dynamic Context Switching layout with **three** columns and five rows of keys in each context

c) Dynamic Context Switching layout with **four** columns and five rows of keys in each context

Fig. 3. Three different (Dynamic) Context Switching layouts used in Experiment 1. In all layouts the bridge, colored blue, is placed in between the contexts.

in Figure 4. Meta-keys located above and below the contexts were used for paging (navigating between pages). To move to the previous page, for example, starting from any column (start sub-context), the participant had to look at one of the markers on the top (associated to page-up) and then look at an adjacent column (end sub-context), within the same context. To provide proper visual feedback, markers changed their color when the user looked at them.

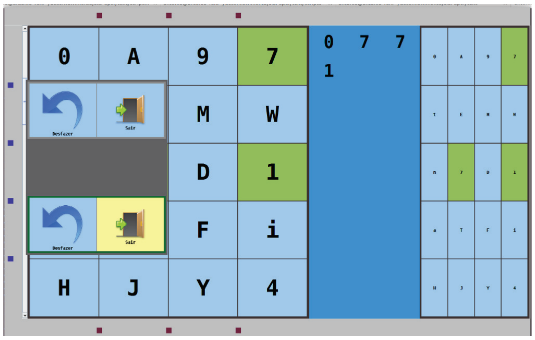


Fig. 4. Menu activated with a meta-key along the vertical edge of the contexts.

3) *Experimental design*

The study was a within-subjects design, where participants used all the three interfaces. The order of presentation of the interfaces followed a latin-square design.

The task was to select all digits from a set of alphanumeric characters (lower and upper case letters from the English alphabet). This task was chosen to reduce the cognitive load of participants during the experiment, so they could focus on the interaction.

The total number of alphanumeric characters per trial was fixed to 120 for all configurations. To force participants browse through all pages using the meta-keys, the number of digits in each trial was picked randomly within the interval [18, 28], uniformly distributed. This corresponds to about 15% to 25% of the total number of alphanumeric characters in the collection.

Before being introduced to the experiment, all participants signed a consent form. After the introduction, participants had a training session of about 10 minutes. Participants were instructed to select all the digits as fast as possible, and to be careful not to leave digits unselected. Each session, including the training session, started with the calibration of the eye tracker. The calibration process was repeated until a reasonable precision was obtained, as evaluated by the experimenter.

After the training session, all volunteers participated in 6 sessions of about 15 minutes each. In each session participants performed 9 trials, 3 for each layout. A session could not be repeated within 30 minutes, so that most volunteers took 2 or 3 days to complete their sessions.

If the eye tracker precision became inadequate due to calibration drift during a session, results of that trial were discarded and the user repeated the trial after recalibration. At the end of the experiment participants were interviewed and answered a questionary. Both the interview and questionary were designed to collect participants' impressions of the interaction using the three layouts and the meta-keys.

4) *Data Analysis*

Precision and **Recall** are metrics used to evaluate how carefully participants execute a search task. In any given trial, let TP be the set of digits actually selected (true positives), FP the set of non-digits (false positives selected incorrectly), and

FN the set of missing digits (false negatives). Precision and Recall are defined as follows:

$$Precision = \frac{|TP|}{|TP| + |FP|} \times 100 \tag{1}$$

$$Recall = \frac{|TP|}{|TP| + |FN|} \times 100 \tag{2}$$

Speed can be compared using the time needed to complete the task, that is computed from the selection of the "Begin" key to the selection of the "Exit" key. In the experiment of Tula *et al.* [16], since each trial could have a different number of digits, it is not fair to use the absolute total time task to compare performance between the different layouts. Therefore, the authors averaged the total time task by the number of selections, including digits and non-digits. This definition has the problem that speed can be improved if the participant selects many non-digits.

In this paper we introduce a modification to this metric, by dividing the total time task by the number of selected digits, excluding the non-digits. After this modification, the **Average time task** (ATT) is here defined as:

$$ATT = \frac{\text{Total time task}}{|TP|} \tag{3}$$

Because users are required to navigate through several pages, it can be assumed that the trial duration is the sum of the time spent selecting objects plus the time needed to switch pages. Therefore, we can separate the selection time from the paging time.

Let's VP be the set of visited pages during a trial. For every page in VP, the **Selection time** is defined from the moment the page was shown to the last selection within that page. Because each page could have a different number of selections, Tula *et al.* computed the **Selection time per digit** ($STPD$) for each page, by dividing the selection time by the number of selections in that page. In this study, we decided to modify this metric by dividing the selection time by the number of selected digits, thus excluding the non-digit selections. The **Average selection time** (AST), i.e., the time needed to make a single selection (independently of the paging time) for each configuration is then computed as follows:

$$AST = \frac{1}{|\text{VP}|} \sum_{p \in \text{ VP}} STPD_p \tag{4}$$

For each page in VP, the **paging time** was computed in [16], [17] from the last selection within that page to the execution of a meta-key to go to the next (or previous) page. The **Average paging time** (APT), i.e., the mean of the paging time for all visited pages for each configuration, is computed as follows:

$$APT = \frac{1}{|\text{VP}|} \sum_{p \in \text{ VP}} \text{paging time}_p \tag{5}$$

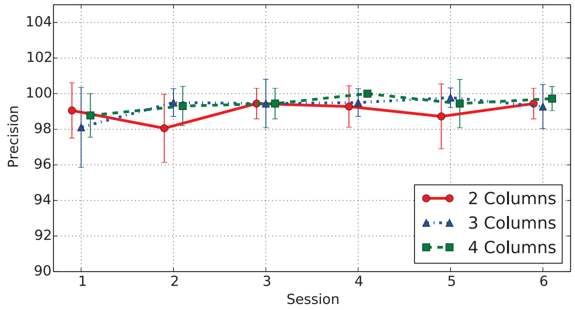


Fig. 5. Precision grand mean for 2C, 3C, and 4C.

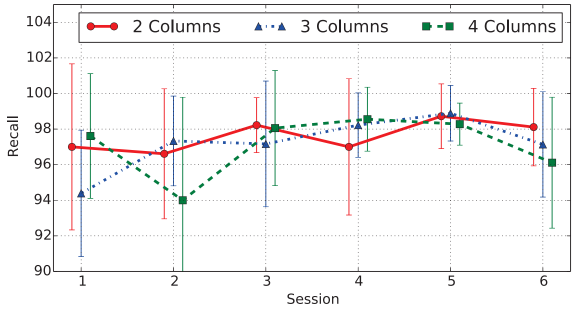


Fig. 6. Recall grand mean for 2C, 3C, and 4C.

B. Results from Experiment 1

The results reported in this Section correspond to the data collected by Tula *et al.* [16] using the modified metrics defined in this paper.

Results of precision and recall are shown in Figures 5 and 6, respectively. Data from the six participants was averaged for each session (horizontal axis) and is shown as a solid line for 2C, dot-dashed line for 3C, and dashed line for 4C. The error bars correspond to one standard deviation. Grand mean of precision was above 98% in all sessions, while the grand mean of recall was above 94%. A one-way, repeated-measures ANOVA showed no significant difference between the three layouts neither for precision, F(1.08, 5.4) = 1.55, p = 0.27, nor for recall, F(2, 10) = 0.39, p = 0.68.

The grand mean of average paging time for the six participants is shown in Figure 7. As can be observed, the 4C layout had a longer APT than the 2C and 3C layouts. A Mauchly's test did not show a violation of sphericity distribution of APT values, W = 0.26, p= 0.07. A one-way repeated measures ANOVA found a significant main effect of layout on APT, F(2, 10) = 18.9, $p < 0.01$. A post-hoc test with Bonferroni correction showed that APT in the 4C layout was significantly longer that in 2C and 3C, p<0.05 in both cases. There was not significant difference in APT between 2C and 3C.

Figure 8 shows the grand mean for the average selection time, computed with data from the six participants for the three layouts. It can be observed that the 4C layout had a longer AST than the other two layouts, while the 2C and

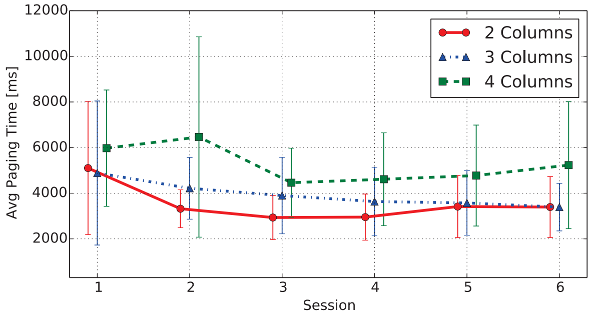


Fig. 7. Average paging time (APT) grand mean for 2C, 3C, and 4C.

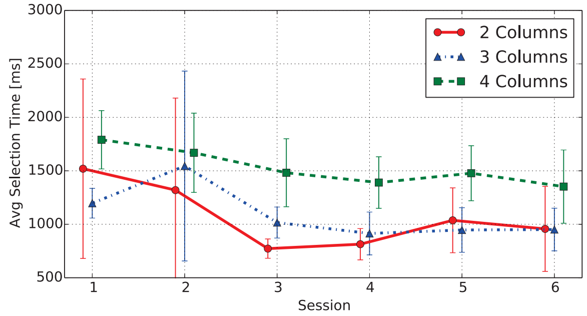


Fig. 8. Average selection time (AST) grand mean for 2C, 3C, and 4C.

3C layouts had similar ASTs. A Mauchly's test showed a slightly violation of sphericity against layout, W = 0.15, p = 0.02, so we used the Greenhouse-Geisser correction method. A one-way repeated measures ANOVA with Greenhouse-Geisser correction (ϵ = 0.54) found a significant main effect of layout on AST, F(1.08, 5.4 = 19.6, p $<$ 0.01. A post-hoc test with Bonferroni correction showed that the AST in the 4C layout was significantly longer than in the 2C and 3C layouts (p<0.05 in both cases). There was not significant difference in AST between 2C and 3C.

Regarding the average task time, the grand mean for the six participants and the three layouts is presented in Figure 9. Interestingly, the 2C layout had the longest ATT in 4 sessions, while the 3C layout had the shortest ATT along the six sessions. A Mauchly's test did not show a violation of sphericity distribution of ATT values against layout, W = 0.73, p = 0.53. A one-way repeated measures ANOVA found a significant main effect of layout on ATT, F(2, 10) = 13.6, p $<$ 0.01. A post-hoc test with Bonferroni correction showed that the 3C layout had a significant lower ATT than the other two layouts (2C and 4C), with p $<$ 0.05 in both cases. There was not significant difference between 2C and 4C, p = 0.64.

C. Discussion

As can be observed in Figure 7, in the 2C and 3C layouts participants needed about 3.5-4 seconds to switch pages, and about 5 seconds in the 4C layout. As pointed out in [17], because the paging time was computed from the last selection to the meta-key activation, this difference was expected since the participants tend to scan the context one last time before switching pages. Hence, in the 2C and 3C layouts, participants learned to scan a single column while relying on their peripheral vision for the adjacent columns, which could not be done in the 4C layout.

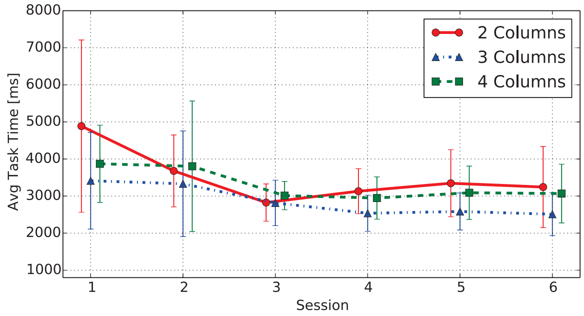


Fig. 9. Average task time (ATT) grand mean for 2C, 3C, and 4C.

TABLE I
SUBJECTIVE IMPRESSIONS OF THE PARTICIPANTS REGARDING SPEED AND COMFORT

	2 columns	3 columns	4 columns
Perceived as faster	3	3	0
More comfortable	4	2	0

When we consider only the time employed to make a single selection with context switching, the ASTs for 2C and 3C were significantly faster than for 4C, as can be seen in Figure 8. Because the average distance between contexts increases with the number of columns, this result was expected, since longer saccades were needed to switch contexts in the 4C layout. This result is also consistent with the participants interviews. As can be observed in Table I, 3 participants rated the 2C and 3C layouts as the faster one. We can also observe in Table I that the 2C layout was considered as the more confortable for 4 participants, and the 3C for 2 participants. The two columns layout was perceived as the simplest to use by all participants.

The ATT metric reflects the overall performance of participants, considering both selection and paging. Figure 9 shows that the 3C layout had a shorter ATT than the other two layouts. Comparing with the 2C and 4C layouts, the 3C layout represents a mid term regarding the number of selectable items on the screen and the number of paging needed to browse the collection entirely. This indicates, as pointed out in [17], that balancing this two factors could result in a better performance for search tasks controlled by the gaze.

1) Subjective evaluation of meta-keys and (dynamic) context switching

Participants were asked about how easy it was to make selections using the interface and to execute the meta-keys. In a Likert scale from 1 (very hard) to 5 (very easy), the average response for selection was 4.7, i.e., the participants found it very easy to make selections using context switching. None of the participants complained about context resizing as disorienting, which could be explained by the saccadic supression mechanism [20]. This supression mechanisms states that during a saccade the eyes are unable to perceive whether a target has been displaced or not.

Activation of meta-keys in the vertical direction received a 2.8 (maximum of 5) to scroll up, and 3.0 to scroll down. Therefore, most participants found it reasonable or good to use. In the horizontal direction, meta-keys made to the right side received a score of 2.4, while meta-keys made to the left were rated as 2.6. The better scores for meta-keys in the vertical direction may just reflect the fact that participants had more time to learn to execute them, since activation of meta-keys in the vertical direction were required more often.

Participants were asked to give their impression about the different interfaces used in the experiment. One participant, with no experience in gaze interaction, said that at times he had to look outside the screen to activate the meta-keys. We believe that by placing meta-keys farther from the contexts, their activation could be facilitated since it will be more robust to gaze tracking errors.

Another participant (with experience in eye tracking and gaze interaction) said that the layouts with 2 and 3 columns where more comfortable than the layout with 4 columns. He mentioned also that searching for digits with 4 columns was less efficient, since with 2 and 3 columns it is possible to use the peripheral vision to quickly explore the content of the keys. Regarding activation of meta-keys, this participant said that though at the beginning he felt some difficulties, after the first sessions he found normal to execute the gestures. He also mentioned that the distance between the meta-keys and the contexts was rather short, hence requiring a more accurate calibration to activate the commands.

IV. Experiment 2: Comparing Fixed-Size and Dynamic Context Switching with Dwell Time

The objective of Experiment 2 is to evaluate the use of selection by dwell time in the multiple selection task. Results from this experiment can be compared with the context switching, in terms of performance and error rate.

A. Method

1) Participants

Five people participated in this experiment, all able-bodied, with normal or corrected to normal vision. Three of them participated in Experiment 1, while the other two had no experience using eye trackers. All participants were students or researchers at the University of São Paulo, aging from 30 to 47 years old, with at least 10 years of experience using computers.

2) Apparatus

A SMI RED500 eye tracker that runs at 500 Hz was used in this experiment. Similar to Experiment 1, a chin rest was used to reduce the participants' head movements.

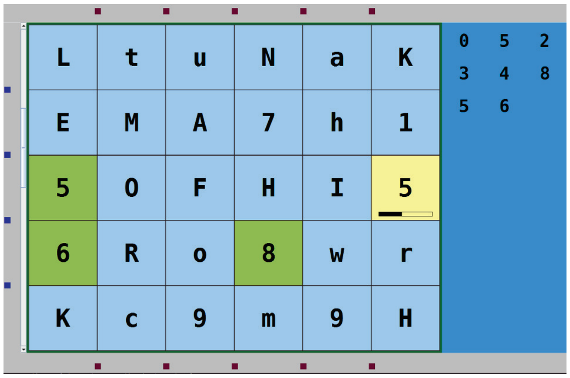


Fig. 10. Dwell time interface used in Experiment 2.

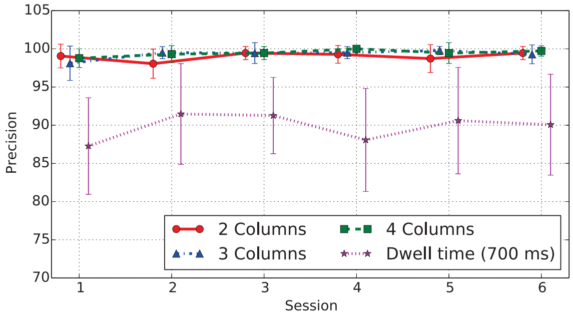


Fig. 11. Precision grand mean for dwell time and for context switching.

There was a single layout, consisting of five rows and six columns of keys, as can be seen in Figure 10. This layout had twice the number of keys than the 3C layout, that was the faster from Experiment 1. The size of the keys was the same as in Experiment 1.

Though it is common to use 500 ms for dwell time [21], some initial tests with the interface revealed that this time was too short to execute the meta-keys without accidentally selecting other keys. Therefore, selections were made by fixating the desired key for 700 ms.

A progress bar at the bottom of the fixated key provided visual feedback about the selection progress. Browsing was made using meta-keys in a similar manner to Experiment 1.

3) Experimental design

The task was similar to Experiment 1, i.e., participants were instructed to select all digits as fast and accurate as possible. With five rows and six columns, the 120 alphanumeric characters fitted in four pages of 30 characters each. Similar to Experiment 1, the number of digits for each trial was picked randomly within the interval [18, 28].

All participants completed six sessions in a single day. In each session, participants executed 3 trials. There was at least 5 minutes rest between sessions.

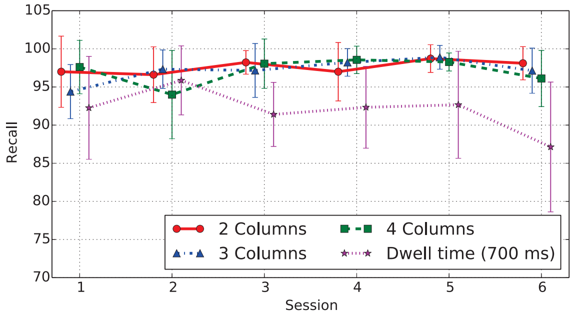


Fig. 12. Recall grand mean for dwell time and for context switching.

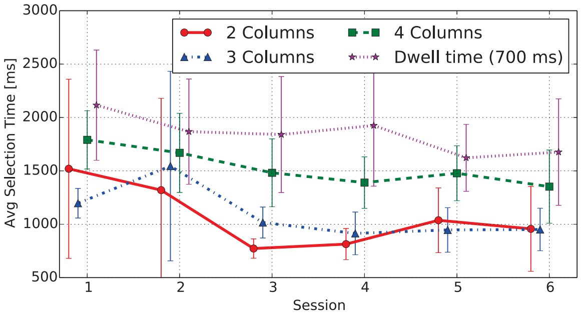


Fig. 14. Average selection time (AST) grand mean for dwell time and 3C.

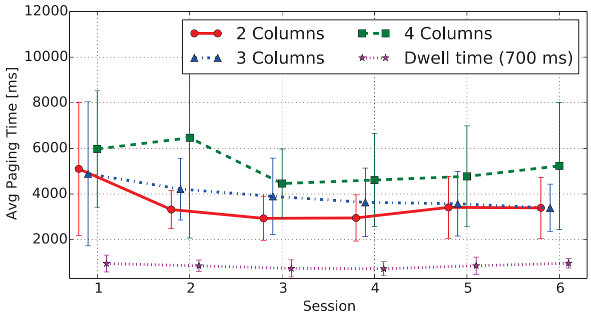


Fig. 13. Average paging time (APT) grand mean for dwell time and 3C.

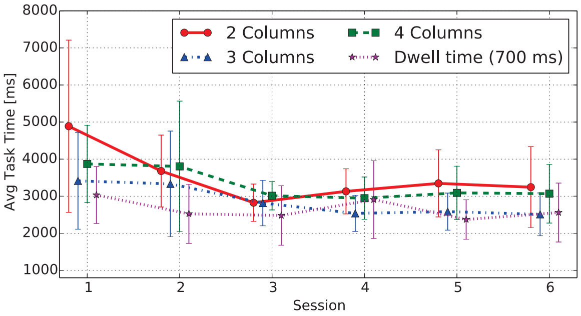


Fig. 15. Average task time (ATT) grand mean for dwell time and 3C.

B. Results from Experiment 2

Figures 11 and 12 show the results of precision and recall, respectively. Data from the five participants was averaged for each session (horizontal axis) and is shown as a dotted line. We also included the results of Experiment 1 to compare. As can be observed, precision for dwell time was consistently lower compared with the all the context switching layouts along the six sessions. A Welch test for unpaired data showed that this difference was significant between dwell time and 2C, t(4.37) = 4.43, $p < 0.01$, between dwell time and 3C, t(4.26) = 4.6, $p < 0.01$, and also between dwell time and 4C, t(4.13) = 4.72, $p < 0.01$.

Results for recall were also lower for dwell time compared with the context switching layouts. This difference was more pronounced in the last four sessions. Using a Welch test for unpaired data we found that recall for dwell time was significantly lower than for 2C, t(5.11) = 2.98, $p < 0.05$, also for 3C, t(4.87) = 2.52, p = 0.05, and also for 4C, t(5.28) = 2.42, p=0.05.

Results for average paging time are shown in Figure 13 as a dotted line. As with precision and recall, we included results of Experiment 1. As can be observed, APT for dwell time was shorter than for the context switching layouts. A Welch test shown that this difference was significant when comparing dwell time with 2C, t(5.37) = 5.57, $p < 0.01$, with 3C, t(5.2) = 4.72, $p < 0.01$, and also with 4C, t(5.12) = 5.23, $p < 0.01$.

Results for average selection time are shown in Figure 14, as a dotted line. As with the previous metrics, we included results of Experiment 1 for comparison. As can be observed, AST for dwell time was longer than for all the context switching layouts along the six sessions. A Welch test for unpaired data showed a significant difference between dwell time and 2C, t(7.07) = 3.19, $p < 0.05$, and also between dwell time and 3C, t(6.14) = 3.26, $p < 0.05$. However, the difference between dwell time and 4C was not significant, t(6) = 1.38, p = 0.22.

Results for average task time are shown in Figure 15 as a dotted line for dwell time. Results for Experiment 1 are also shown for comparison. As can be observed, ATT for dwell time was similar to 3C and a bit shorter compared with 2C and 4C. A Welch test for unpaired data showed no significant difference between dwell time and 2C, t(8.93) = 1.84, p = 0.1, between dwell time and 3C, t(8.55) = 0.55, p = 0.6, and between dwell time and 4C, t(8.8) = 1.56, p = 0.15.

C. Discussion

Results of Experiment 2 have revealed some interesting issues, when compared to the results from Experiment 1.

The significant lower precision and recall for dwell time compared with all the context switching layouts (Figures 11 and 12) can be attributed to the Midas touch problem [2]. During the experiment, we perceived that some participants accidentally selected non-digits (thus reducing precision) or deselected digits that were already selected (thus reducing recall) while exploring the context. Participants also made wrong selections while switching pages or while activating the exit menu, which could explain the lower precision and recall compared to context switching. With context switching the risk of accidental selections is reduced, because to select a key the user must cross entirely the bridge between the two contexts.

The higher average selection time observed in the dwell time layout is expected, since participants had to wait for 700 ms before completing a selection. In the context switching layouts, there is no need to wait, since the saccade to the other context can be executed as soon as the observed key receives the focus (that lasted only 150 ms). To initiate a saccade, there is a latency of about 200-300 ms [22] that could be reduced in some situations [23]. Saccadic movements are very fast, therefore the eye movement last about 50-100 ms. In sum, the theoretical time needed to make a single selection is shorter with context switching than with the dwell time used in our experiment. The use of shorter dwell times can reduce the AST, since the user needs to wait less to complete a selection, but it could also increase the error rate.

Using the dwell time layout, participants exhibited a significant shorter average paging time compared the context switching layouts. A possible explanation is that, with a bigger context, participants have more space to execute the meta-key to navigate. Nonetheless, we believe that the shorter APT could also be a result of accidental selections while executing the meta-key. The paging time was computed from the last selection within a page to the execution of the meta-key. We already mentioned that sometimes participants made wrong selections while executing the meta-key (due to the Midas touch). Therefore, the paging time was shorter because a (possible wrong) selection happened just before the meta-key was completed.

Finally, it is noticeable that the average task time for dwell time was similar to context switching (Figure 15), since no significant difference was found when comparing dwell time to the 2C, 3C, and 4C configurations. It implies that although making a single selection was faster with (dynamic) context switching, the less frequent activation of meta-keys in the dwell time layout (because of the greater number of keys per page) influenced the overall speed. In fact, while the average selection time was shorter for context switching, the average paging time was shorter for dwell time.

V. Conclusion

This paper presented a combination of the concepts of dynamic context switching [16] and meta-context switching as extensions of the traditional context switching paradigm for gaze-based interaction. Like context switching, dynamic context switching has two replicated contexts, but the context that has the focus is displayed in full size, while the other one is reduced. When a saccade that changes contexts is detected, the sizes of the contexts are dynamically adjusted, allowing a better use of screen space and improving the robustness of the system to gaze tracking noise.

Meta-context switching is a way to generalize the paradigm to other applications. It allows the execution of general-purpose commands like navigating, switching between different applications, and changing the mode of operation. Meta-context switching requires the use of meta-keys [17] that can be activated by crossing more than one bridge, i.e., using more than one saccade.

We have conducted two user experiments to compare the performance of a 2-columns, fixed context switching layout, 3 columns and 4 columns dynamic context switching layouts, and a dwell time layout with 6 columns. All interfaces used meta-context switching to navigate between several pages of items. Results showed that, among the context switching interfaces, the best performance was obtained for the 3C layout. Participants did not feel disoriented by the context resizing, actually, some of them did not even notice it. This can be explained by the “saccadic masking phenomenon” which suppresses our visual perception during saccades [20]. Results also showed that participants learned the meta-keys easily and were able to use them to complete the tasks successfully.

Comparing the context switching layouts with dwell time, we found that participants commited more errors using dwell time than (dynamic) context switching. Though the dwell time layout had almost twice the keys of the context switching layouts, there was no significant difference in the speed to complete the task. Our results suggest that dwell time should not be used together with meta-keys activated by gaze gestures. Future experiments could evaluate the use of escape mechanisms activated by dwell time, instead of meta-keys. One possibility is to include additional buttons in the dwell time interface for browsing and activating other commands, and compare the performance and error rate with context switching and meta-context switching.

Acknowledgment

This research was supported by the São Paulo Research Foundation (FAPESP), grants 2010/14075-4, 2011/00267-1, and 2012/04426-0.

References

[1] C. H. Morimoto and M. Mimica, “Eye gaze tracking techniques for interactive applications,” *Computer Vision and Image Understanding*, vol. 98, pp. 4–24, 2005.

[2] R. J. K. Jacob, “What you look at is what you get: eye movement-based interaction techniques,” in *Proceedings of the SIGCHI conference on Human factors in computing systems: Empowering people*, ser. CHI ’90. New York, NY, USA: ACM, 1990, pp. 11–18. [Online]. Available: http://doi.acm.org/10.1145/97243.97246

[3] E. Møllenbach, J. P. Hansen, and M. Lillholm, “Eye Movements in Gaze Interaction,” *Journal of Eye Movement Research*, vol. 6, pp. 1–15, 2013.

[4] P. Majaranta and K.-J. Räihä, “Twenty years of eye typing: systems and design issues,” in *Proceedings of the 2002 symposium on Eye tracking research & applications*, ser. ETRA ’02, New York, NY, USA, 2002, pp. 15–22.

[5] C. Lankford, “Effective eye-gaze input into windows,” in *Proceedings of the 2000 symposium on Eye tracking research & applications*, ser. ETRA ’00. New York, NY, USA: ACM, 2000, pp. 23–27. [Online]. Available: http://doi.acm.org/10.1145/355017.355021

[6] J. P. Hansen, K. Tørning, A. S. Johansen, K. Itoh, and H. Aoki, “Gaze typing compared with input by head and hand,” in *Proceedings of the 2004 symposium on Eye tracking research & applications*, ser. ETRA ’04. New York, NY, USA: ACM, 2004, pp. 131–138.

[7] O. Špakov and D. Miniotas, “On-line adjustment of dwell time for target selection by gaze,” in *Proceedings of the third Nordic conference on Human-computer interaction*, ser. NordiCHI ’04. New York, NY, USA: ACM, 2004, pp. 203–206. [Online]. Available: http://doi.acm.org/10.1145/1028014.1028045

[8] P. Majaranta, U.-K. Ahola, and O. Špakov, “Fast gaze typing with an adjustable dwell time,” in *Proceedings of the 27th international conference on Human factors in computing systems*, ser. CHI ’09. New York, NY, USA: ACM, 2009, pp. 357–360. [Online]. Available: http://doi.acm.org/10.1145/1518701.1518758

[9] K.-J. Räihä and S. Ovaska, “An exploratory study of eye typing fundamentals: dwell time, text entry rate, errors, and workload,” in *Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems*, ser. CHI ’12. New York, NY, USA: ACM, 2012, pp. 3001–3010. [Online]. Available: http://doi.acm.org/10.1145/2208636.2208711

[10] J. O. Wobbrock, J. Rubinstein, M. W. Sawyer, and A. T. Duchowski, “Longitudinal evaluation of discrete consecutive gaze gestures for text entry,” in *Proceedings of the 2008 symposium on Eye Tracking Research & Applications*, ser. ETRA ’08. New York, NY, USA: ACM, 2008, pp. 11–18. [Online]. Available: http://doi.acm.org/10.1145/1344471.1344475

[11] M. Porta and M. Turina, “Eye-s: a full-screen input modality for pure eye-based communication,” in *Proceedings of the 2008 symposium on Eye Tracking Research & Applications*, ser. ETRA ’08. New York, NY, USA: ACM, 2008, pp. 27–34. [Online]. Available: http://doi.acm.org/10.1145/1344471.1344477

[12] D. J. Ward and D. J. C. MacKay, “Fast hands-free writing by gaze direction,” *Nature*, vol. 418, no. 6900, p. 838, 2002.

[13] D. W. Hansen, H. H. Skovsgaard, J. P. Hansen, and E. Mollenbach, “Noise tolerant selection by gaze-controlled pan and zoom in 3d,” in *Proceedings of Eye Tracking Research & Applications, ETRA 2008*. Savannah,GA: ACM Press, 2008, pp. 205–212.

[14] M. Vidal, A. Bulling, and H. Gellersen, “Pursuits: Spontaneous interaction with displays based on smooth pursuit eye movement and moving targets,” in *Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, ser. UbiComp ’13. New York, NY, USA: ACM, 2013, pp. 439–448. [Online]. Available: http://doi.acm.org/10.1145/2493432.2493477

[15] C. H. Morimoto and A. Amir, “Context switching for fast key selection in text entry applications,” in *Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications*, ser. ETRA ’10. New York, NY, USA: ACM, 2010, pp. 271–274.

[16] A. D. Tula, F. M. S. de Campos, and C. H. Morimoto, “Dynamic context switching for gaze based interaction,” in *Proceedings of the Symposium on Eye Tracking Research and Applications*, ser. ETRA ’12. New York, NY, USA: ACM, 2012, pp. 353–356. [Online]. Available: http://doi.acm.org/10.1145/2168556.2168635

[17] A. D. Tula and C. H. Morimoto, “Meta-keys: Extending the functionality of gaze-based interaction,” in *Proceedings of the 13th Brazilian Symposium on Human Factors in Computing Systems*, ser. IHC ’14. Porto Alegre, Brazil, Brazil: Sociedade Brasileira de Computação, 2014, pp. 285–292. [Online]. Available: http://dl.acm.org/citation.cfm?id=2738055.2738101

[18] C. Morimoto, D. Koons, A. Amit, M. Flickner, and S. Zhai, “Keeping an eye for hci,” in *Computer Graphics and Image Processing, 1999. Proceedings. XII Brazilian Symposium on*, 1999, pp. 171–176.

[19] H. Collewijn, C. J. Erkelens, and R. M. Steinman, “Binocular co-ordination of human vertical saccadic eye movements.” *The Journal of physiology*, vol. 404, pp. 183–97, Oct. 1988.

[20] B. Bridgeman, D. Hendry, and L. Stark, “Failure to detect displacement of the visual world during saccadic eye movements.” *Vision Research*, vol. 15, no. 6, pp. 719–722, 1975. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/1138489

[21] P. Majaranta and K. J. Räihä, “Text entry by gaze: Utilizing eye tracking,” in *Text entry systems: Mobility, accessibility, universality*, I. MacKenzie and K. Ishii, Eds. Morgan Kaufmann, 2007, pp. 175–187.

[22] S. Liversedge, I. Gilchrist, and S. Everling, *The Oxford Handbook of Eye Movements*, ser. Oxford Library of Psychology. New York: OUP Oxford, 2011.

[23] M. G. Saslow, “Effects of components of displacement-step stimuli upon latency for saccadic eye movement,” *J. Opt. Soc. Am.*, vol. 57, no. 8, pp. 1024–1029, Aug 1967.