

Calibration-free Text Entry using Smooth Pursuit Eye Movements

Yasmeen Abdrabou
German University in Cairo
Bundeswehr University Munich
yasmeen.abdrabou@guc.edu.eg

Mohamed Khamis
University of Glasgow
mohamed.Khamis@glasgow.ac.uk

Mariam Mostafa
German University in Cairo
mariam.mostafa@student.guc.edu.eg

Amr Elmougy
University of Canada in Egypt
amr.elmougy@uofcanada.edu.eg

ABSTRACT

In this paper, we propose a calibration-free gaze-based text entry system that uses smooth pursuit eye movements. We report on our implementation, which improves over prior work on smooth pursuit text entry by 1) eliminating the need of calibration using motion correlation, 2) increasing input rate from 3.34 to 3.41 words per minute, 3) featuring text suggestions that were trained on 10,000 lexicon sentences recommended in the literature. We report on a user study (N=26) which shows that users are able to eye type at 3.41 words per minutes without calibration and without user training. Qualitative feedback also indicates that users positively perceive the system. Our work is of particular benefit for disabled users and for situations when voice and tactile input are not feasible (e.g., in noisy environments or when the hands are occupied).

CCS CONCEPTS

• **Human-centered computing** → **Human computer interaction (HCI); Text input.**

KEYWORDS

Calibration-Free, Text Entry, Gaze Interaction, Smooth Pursuits

ACM Reference Format:

Yasmeen Abdrabou, Mariam Mostafa, Mohamed Khamis, and Amr Elmougy. 2019. Calibration-free Text Entry using Smooth Pursuit Eye Movements. In *2019 Symposium on Eye Tracking Research and Applications (ETRA '19)*, June 25–28, 2019, Denver, CO, USA. ACM, New York, NY, USA, 5 pages. <https://doi.org/10.1145/3314111.3319838>

1 INTRODUCTION AND RELATED WORK

New technologies are explored to enhance the user's experience and to maximize the interaction of the user, especially in public displays. One of the most popular interaction technologies is eye trackers. Using eye tracking makes it easier to see through a user's eyes, to know what catches the user's attention and to know more

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
ETRA '19, June 25–28, 2019, Denver, CO, USA

© 2019 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 978-1-4503-6709-7/19/06...\$15.00
<https://doi.org/10.1145/3314111.3319838>

about the user and his/her behavior. With the new advancements in eye tracking techniques, it became more affordable. However, it still suffers from the calibration problem. Calibration process is considered to be tedious, time-consuming for the user and memory space consuming for the application/system. The consumed time and the memory space during calibration is bearable (but not desirable) for an application with a small number of users or for a desktop application. But for a public displays' system which mainly depends on immediate usability by many users with short interaction times, it is not acceptable. A better use is to have a calibration-free system deployed on public displays [Khamis et al. 2015, 2016]. It also can be used where tactile or voice interaction is not possible.

Since the human eyes and gaze behavior reflects cognitive processes and can give hints of our thinking and intentions as they reveal a lot about the user's interests and behavior. It is also said that that humans often look at things before taking action [Land and Furneaux 1997]. This means that using human eyes in interaction will be fast and reliable as the eyes always look at what they want or what seems interesting for the user. Therefore, there is a wide range of application areas which make use of the eye-tracker. Eye-trackers is used in marketing [Wedel and Pieters 2008], usability in human-computer interaction [Poole and Ball 2006], psychology [Schwarzkopf 2015], gaming, etc and text entry is not an exception. Research has been done for using gaze as a text entry methodology [Benson 2011; Kurauchi et al. 2016; Majaranta et al. 2009; Morimoto and Amir 2010; Sengupta et al. 2017; Tuisku et al. 2008; Tula et al. 2012]. It was also used by removing the calibration process by doing a one-point calibration technique [Lutz et al. 2015]. However, no calibration free applications have been implemented yet.

Accordingly, in this paper, we introduce a calibration-free gaze based text entry using smooth pursuits. This application does not take extra time for calibration, hence reduce the wasted time. It uses motion correlation for this purpose where the user traces the moving letters by gaze. We achieved 3.41 WPM unlike the literature using the same layout with 3.34 WPM.

2 DESIGN AND IMPLEMENTATION

In this paper, we describe a calibration-free gaze-based text entry using smooth pursuits. The system determines which object the user is looking at by correlating eye movements and object movements. When the system detects eyes on the screen the objects start rotating, when the eyes follow the rotation path, a correlating is being calculated using the dynamic objects on the display with the

smooth pursuit eye movement performed when the eyes follow a moving object. By using an eye-tracking device, a stream of the user's gaze points (XY-coordinates) will be collected, as well as the positions of the application's objects on the screen (XY-coordinates). Having these stream of XY-coordinates and by computing the correlation between each of the objects' positions with the gaze points, in the X-axis and in the Y-axis separately. If an object has a correlation with the gaze points > 0.9 in the X-axis and in the Y-axis, then this object will be selected. Otherwise, no object will be selected and the system will collect more points and correlate again till one of the object's correlation values pass the threshold. We used math.net library to calculate the correlation [Math.NET 2018]. It is important to note that the eye tracker was calibrated at the beginning until the Tobii software reported accurate tracking¹.

2.1 User Interface

This eye-controlled text entry system is an approach towards a calibration-free eye-based text entry which is suitable for public displays. The system combines a two-stage interface concept [Bee and André 2008; Lutz et al. 2015] with interaction designed specifically for a calibration-free approach using smooth pursuit movements. The detection mechanism used is related to the approach used in the study made by Khamis et al [Khamis et al. 2016]. The layout of the system will follow the layout proposed by Lutz, Venjakob, and Ruff in their text entry system using eye pursuits [Lutz et al. 2015]. The system's interaction is designed to provide a calibration-free application which will make use of the smooth pursuit eye movements and its characteristics [Khamis et al. 2016].

2.2 Layout Design

The layout used is similar to the SMOOVS text entry system [Lutz et al. 2015], which is implemented similarly to the layout of the EEG3-based Hex-O-Spell, which is a mental text entry system [Blankertz et al. 2007]. The EEG3-based Hex-O-Spell layout is a hexagonal layout with hexagonal tiles on it, the layout is used in this system will be typical to SMOOVS (as shown in figure 1a).

The user interface used in the application is listing all of the English alphabets, backspace, space and some special characters which are full-stop, comma, exclamation mark, and question mark. The reason behind the division of the layout into six objects is that this division approach which is supported by a smooth pursuit-based interaction [Vidal et al. 2013] where the detection rate from four to six objects was high and similar to each other. But when the number of presented objects exceeds six, the detection rate drops. This system is using six interactive clusters, a cluster is a group of six hexagon where each hexagon holds a character. Each cluster holds up six neighboring letters from the alphabets, and the last cluster of hexagons will contain the Y, Z and some special characters which are (, ! ?). At the center of the screen, an empty white text box is placed to hold the selected characters by the user, and it is placed at the center not to distract the user. The user can type and at the same time track what is written so far. At the bottom of the screen the sentence written so far by the user is presented, and at the top of the screen, the target sentence to be written by the user is viewed. The bottom cluster consists of 2 hexagons.

¹The calibration was done by a user who did not take part in the experiment

The first hexagon is '<', which represents the backspace, deletes the last character from the text box at the center. The second one is '>' represents the space it also can be used to confirm the written word to be appended to the sentence written (previously confirmed). Within each cluster, the first and last character of the cluster appears closest to the center of the screen. By looking at these two closest hexagons, the user can determine the range of characters covered by the cluster. The interactive clusters are arranged in a circular layout to help to reach the best differentiation between each of the interactive objects cf. figure 1a. The cluster and the hexagon (character) selection are done via calibration-free selection.

2.3 Display Dynamics

The user can interact with the system's six interactive clusters through two interaction stages (two selections) [Blankertz et al. 2007]. The first interaction stage is to select a cluster and the second one is to select the desired character from the six characters which belong to the selected cluster or a word suggestion. These interactions are done via three phases of the layout cf. figure 1.

The objects' movement consists of five animations. Before the user detection gaze, the layout is stationary (phase 0). If a valid pursuit movement is detected, the clusters start rotation (first animation) to select a cluster (as shown in figures 1a & 1b), then if a cluster is selected, the rotation of the clusters will stop. The selected cluster move outwards (second animation), phase 2, as shown in figure 1c. Phase 3 starts when the hexagons of the selected cluster start to move in distinct directions (third animation) from their directions within a cluster and also arranged in a circular layout as in figure 1d. This minimizes the variability in difficulty to follow objects' trajectories as all of the orientation of the objects is changed due to the animation. Also, these animations are like leading the user's eyes to know where to interact next. When the hexagons reach their new positions, they start rotation (fourth animation) to select a hexagon/character. In addition, another layer is added with the word suggestions which moves anti-clockwise "reverse animation than the hexagons "phase 4" (fifth animation), figure 1e". After phase 4, phase 0 is initiated again in which the layout returns to its initial state and the word/character is written cf. figure 1f.

3 EVALUATION AND RESULTS

The experiment consists of one session. The participants start the experiment with one word training "Hello" and then the experimental task was to type 5 given phrases, 4 of the 5 phrases were proposed by MacKenzie and Soukoreff [MacKenzie and Zhang 1999]² and the fifth phrase was a popular English pangram³. The order of the phrases was randomized for each participant. Before the text entry user-interface is loaded, the phrases are displayed to the user in a message box. Then the session starts when the participant hits the start button of the shown message box. In addition, the required phrase is shown at the top of the screen so the participants can take a look at. The participants were told that if they made typos while writing, they are free to modify it or to leave it as it is. For the

²My mother makes good cookies ., The force is with you !, Yes , this is a very good idea ., Did you have a good time ?

³a sentence containing every letter of a given alphabet at least once "The quick brown fox jumps over the lazy dog."

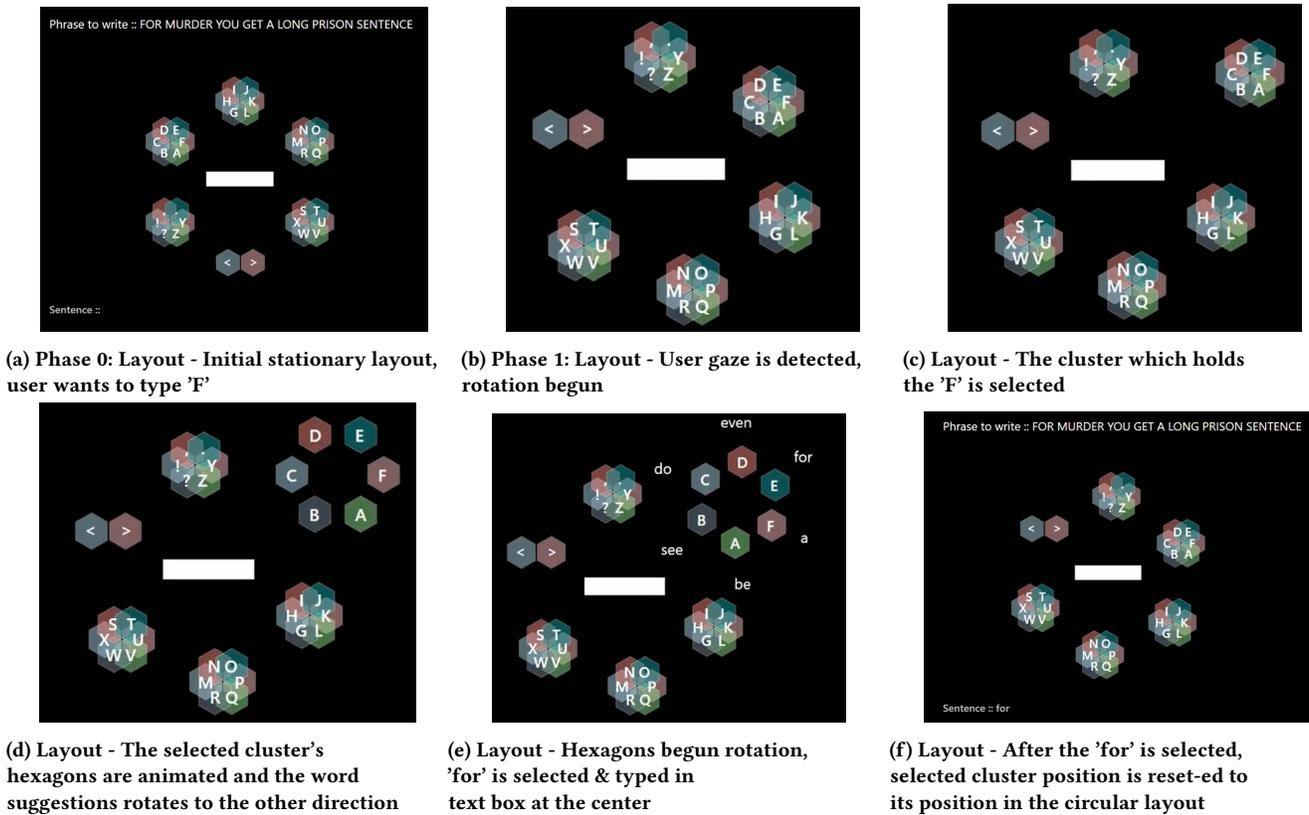


Figure 1: System Phases

language model, we used a lexicon dictionary with 10,000 sentences [COCA 2018] for the training Presage, the word prediction library [Presage 2018]. At the end, the word with the highest probability is then added to the interface.

3.1 Apparatus and Participants

An HP laptop core i7 with 2.60 GHz CPU and 16GB RAM⁴ used in the experiment with a screen resolution of 1024x768 pixels “17 in”. A Tobii 4C Eye Tracker⁵ is used to track the user’s gaze on the display screen. The implementations was done using C#. All of the participants were sitting steady and the distance between their eyes and the laptop screen was approximately 40 cm.

We invited 26 participants (10 male, 16 female), aged between 16 and 29 (M = 21.6; SD = 2.5). Three participants had experience with gaze-interaction. From the participant, 8 wear glasses. Participants were recruited by a word of mouth and had different backgrounds (middle-high school, pharmacy, engineering, and computer science).

3.2 Experiment Procedure

Before beginning the experiment, each of the participants was told to sit steady and to keep their distance between their eyes and the laptop screen unchanged as possible. Additionally, they were told that they can move their heads normally. The experiment aim was

explained and the participants filled the consent and demographic forms. At the beginning of the experiment, the experimenter explained how the objects of the user interface act, move and illustrated the dynamics of the display. Also, the detailed steps to select a character or to select one of the suggested words are told to the participants. Then the participants started their one-word trial. After the experiment, a semi-structured interview took place.

3.3 Metrics Used for Evaluation

To evaluate the system’s usability and accuracy, we used a set of performance metrics. We considered the errors left in the sentences and the corrected errors as explained in the following points:

- Typing Speed: measured in words per minute (WPM), where a word is any sequence of five characters, including letters, spaces, punctuation, etc.
- Minimum String Distance "MSD": the minimum distance between two strings defined in terms of editing primitives. The primitives are insertion, deletion, and substitution.
- Error Rate: calculated by comparing the transcribed text (text written by user) with the presented text, using the Minimum String Distance (MSD) method [MacKenzie and Zhang 1999]. This method does not take into account corrected errors.
- Keystrokes Per Character "KSPC": a measure of the average number of keystrokes used to enter each character of text. Ideally, KSPC = 1.00, indicating that each key press produces a character. If participants correct mistakes during entry, the KSPC value is

⁴<https://store.hp.com/us/en/mdp/laptops/envy-15-204072--1>

⁵<https://help.tobii.com/hc/en-us/articles/213414285-Specifications-for-the-Tobii-Eye-Tracker-4C>

greater than 1. For example, if "hello" is entered as h e l x (delete) l o, the final result is correct (0% error rate), but the KSPC value is $7/5 = 1.4$ (7 keystrokes to enter 5 characters). KSPC is an accuracy measurement reflecting the overhead incurred in correcting mistakes as described by Majaranta [Majaranta 2009].

3.4 Pilot Test

A pilot test took place on 9 participants (4 male, 5 female) to test the system speed and the radius of moving hexagon's path. The participants aged between 18 and 25 (Mean = 21, SD = 2). Only two participants had an experience with the eye-controlled interfaces. All participants had normal or corrected-to-normal vision.

The pilot test took place on 12 sentences randomly ordered for each participant from Soukoreff and MacKenzie's proposed dataset [MacKenzie and Zhang 1999]. Each sentence was tested with different radius and speed. As a result, we choose 155 "pixels from the center" to be the radius and 3.8 "px/sec" to be the animation speed. The test used 150, 155 and 160 as radius and matched it with 4 different speeds 3.6, 3.8, 4, 4.2. The bigger radius made it harder for the eyes to select the letters and the smaller ones were not accurate for the correlation to differentiate between them. For the speeds, the faster ones were harder for the eyes to follow and had a high error rate and the slower ones had lower typing speed and no error rate. For the small hexagons, we chose 65 as a radius.

3.5 Results and Discussion

The average typing speed of the system achieved is 3.41 WPM, the minimum typing speed achieved is 2.93 WPM and the maximum typing speed is 3.86 WPM without training. Also, a typing of 4.7 WPM is reached as seen in figure 2. The average MSD error rate, which is the error rate of the uncorrected characters, is 3.04% "figure 3", and the average KSPC that calculates the error rates of the corrected words is 1.04 "figure 4". The error rates of the system are very low which means that the system is reliable and accurate in selecting characters, that most of the participants achieved KSPC = 1 and MSD error rate nearly 0% which is ideal. For the interviews, showed positive feedback from all participants and they assured that their usage to the system. They also reported that the system was easy to use and they learned to use it quickly. They also mentioned that by frequent usage they will be faster in typing.

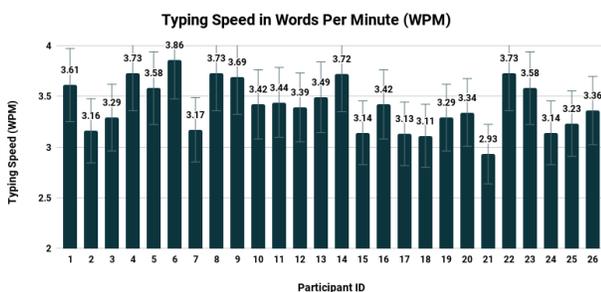


Figure 2: Overall Words Per Minute achieved per participant

3.5.1 Reflecting on the Literature. After described in the system with the results, now it is time to reflect on the literature found. It

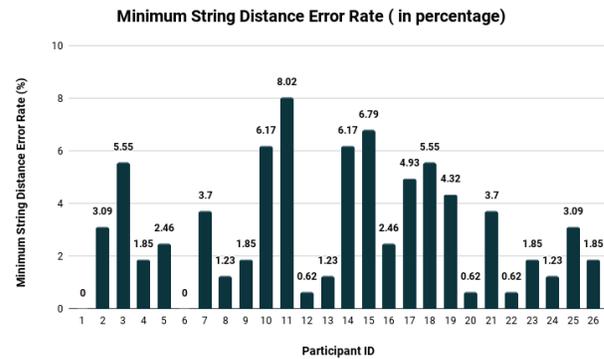


Figure 3: Minimum String Distance per participant

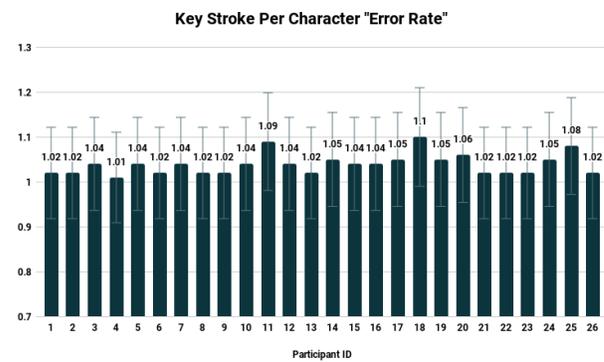


Figure 4: Keystroke Per Character for each participant

is clear that the main idea of this systems is taken from SMOOVs as mentioned previously, and hence, the similarities between both systems can be summarized into the following points. First, we accommodated the same layout and hexagon distributions. Second, we replicated the same movements for the big cluster along with the animations on how the hexagons separate from each other. We further added another level for word predictions along with cluster movements for the characters. In addition to the reverse movement for the prediction cluster. These are points regarding the layout. For the implementation, we are proposing a calibration free interaction with the system which will make it more usable on public displays and consumes shorter time for interaction with more WPM.

4 CONCLUSION AND FUTURE WORK

In this paper, we introduced a calibration-free gaze-based text entry based on smooth pursuits eye movements. Users were able to write with a 3.41 WPM as an enhancement over existing systems. We explained the system implementation and evaluation methodology along with the study. Qualitative feedback showed that the participants are willing to use it and the system perceived good usability. Finally, the system can be used when tactile or voice inputs are not feasible and it can also be in a good benefit to disabled users. In the future, we will enhance the motion correlation technique to reduce the error rate and reevaluate the system with a bigger number of participants to calculate the learning curve.

REFERENCES

- Nikolaus Bee and Elisabeth André. 2008. Writing with your eye: A dwell time free writing system adapted to the nature of human eye gaze. In *International Tutorial and Research Workshop on Perception and Interactive Technologies for Speech-Based Systems*. Springer, 111–122.
- George S Benson. 2011. Popular and influential management books. *Useful research: Advancing theory and practice* (2011), 289–308.
- Benjamin Blankertz, Matthias Krauledat, Guido Dornhege, John Williamson, Roderick Murray-Smith, and Klaus-Robert Müller. 2007. A note on brain actuated spelling with the Berlin brain-computer interface. In *International Conference on Universal Access in Human-Computer Interaction*. Springer, 759–768.
- COCA. Featured on April 2018. Corpus of Contemporary American English.
- Mohamed Khamis, Florian Alt, and Andreas Bulling. 2015. A field study on spontaneous gaze-based interaction with a public display using pursuits. In *Adjunct Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2015 ACM International Symposium on Wearable Computers*. ACM, 863–872.
- Mohamed Khamis, Ozan Saltuk, Alina Hang, Katharina Stolz, Andreas Bulling, and Florian Alt. 2016. TextPursuits: using text for pursuits-based interaction and calibration on public displays. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. ACM, 274–285.
- Andrew Kurauchi, Wenxin Feng, Ajjen Joshi, Carlos Morimoto, and Margrit Betke. 2016. EyeSwipe: Dwell-free text entry using gaze paths. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. ACM, 1952–1956.
- Michael F Land and Sophie Furneaux. 1997. The knowledge base of the oculomotor system. *Philosophical Transactions of the Royal Society of London B: Biological Sciences* 352, 1358 (1997), 1231–1239.
- Otto Hans-Martin Lutz, Antje Christine Venjakob, and Stefan Ruff. 2015. SMOOVs: Towards calibration-free text entry by gaze using smooth pursuit movements. *Journal of Eye Movement Research* 8, 1 (2015).
- I. Scott MacKenzie and Shawn X. Zhang. 1999. The Design and Evaluation of a High-performance Soft Keyboard. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '99)*. ACM, New York, NY, USA, 25–31. <https://doi.org/10.1145/302979.302983>
- Päivi Majaranta. 2009. *Text entry by eye gaze*. Tampere University Press.
- Päivi Majaranta, Ulla-Kaija Ahola, and Oleg Špakov. 2009. Fast gaze typing with an adjustable dwell time. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 357–360.
- Math.NET. June, 2018. Math.NET. <https://numerics.mathdotnet.com>
- Carlos H Morimoto and Arnon Amir. 2010. Context switching for fast key selection in text entry applications. In *Proceedings of the 2010 symposium on eye-tracking research & applications*. ACM, 271–274.
- Alex Poole and Linden J Ball. 2006. Eye tracking in HCI and usability research. *Encyclopedia of human computer interaction* 1 (2006), 211–219.
- Presage. June, 2018. Presage. <https://presage.sourceforge.io>
- Stefan Schwarzkopf. 2015. Measurement devices and the psychophysiology of consumer behaviour: A posthuman genealogy of neuromarketing. *BioSocieties* 10, 4 (2015), 465–482.
- Korok Sengupta, Raphael Menges, Chandan Kumar, and Steffen Staab. 2017. Gazethekey: Interactive keys to integrate word predictions for gaze-based text entry. In *Proceedings of the 22nd International Conference on Intelligent User Interfaces Companion*. ACM, 121–124.
- Outi Tuisku, Päivi Majaranta, Poika Isokoski, and Kari-Jouko Riihää. 2008. Now Dasher! Dash away!: longitudinal study of fast text entry by Eye Gaze. In *Proceedings of the 2008 symposium on Eye tracking research & applications*. ACM, 19–26.
- Antonio Diaz Tula, Filipe de Campos, and Carlos H Morimoto. 2012. Dynamic context switching for gaze based interaction. In *Proceedings of the Symposium on Eye Tracking Research and Applications*. ACM, 353–356.
- Mélodie Vidal, Andreas Bulling, and Hans Gellersen. 2013. 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*. ACM, 439–448.
- Michel Wedel and Rik Pieters. 2008. A review of eye-tracking research in marketing. In *Review of marketing research*. Emerald Group Publishing Limited, 123–147.