Design and Development of Natural Eye Typing Interface

vorgelegt von
M.Sc.
Zhe Zeng

an der Fakultät V – Verkehrs- und Maschinensysteme
der Technischen Universität Berlin
zur Erlangung des akademischen Grades
Doktorin der Naturwissenschaften
- Dr. rer. nat. -
genehmigte Dissertation

Promotionsausschuss:

Vorsitzender: Prof. Dr. Henning Jürgen Meyer
Gutachter: Prof. Dr. Matthias Rötting
Gutachter: Prof. Dr. John Paulin Hansen
Tag der wissenschaftlichen Aussprache: 04. Februar 2022

Berlin 2022

Abstract

With the development of eye tracking technology, gaze interaction has shown great potential to assist day-to-day human computer interaction. Gaze input makes it possible to combine both visual search and action activation into one step, as users can perceive relevant information and activate desired commands, i.e., by focusing their gaze on a corresponding item on the interface. However, the usage scenario is limited by eye tracking accuracy and user acceptance. The aim of this dissertation therefore is to design a dynamic eye typing interface that tolerates eye tracking with low accuracy. It should be easy to understand without extensive training, and can be applied to the use of public screens. To this end, three studies were conducted, they are an offline gaze data analysis, the evaluation for the new typing interface and its iterative design.

The first study focused on the effects of the number of objects and object moving speed in the calibration-free gaze interface based on pursuit movements where objects move linearly. Offline gaze data of 25 participants were collected and analyzed. The results indicate that the number of objects significantly influenced the correct and false detection rates. Participants' performance was better on the interfaces containing 6 and 8 objects compared to 10, 12 and 15 objects. The detection rate was significantly higher for interfaces with faster moving speed than for slower ones.

The second study concentrated on developing a new dynamic eye typing interface. The eye typing interface features two-stage selection and one-point calibration. To activate a command action, the user needs to first look at the corresponding character cluster and then follow the movement of the desired character in this cluster with their eye. With the findings of the first study, this typing interface was designed with eight character clusters, and there are four characters in each character cluster. A user study with 29 participants was conducted to compare four types of feedback for this eye typing interface. The results confirmed that the user can learn how to interact with the system in a short time. The interface with both visual and auditory feedback achieved the highest typing speed.

In the third study, three eye typing interfaces based on the second study were further presented and compared. They are interfaces without language-model support (i.e., the baseline), with letter prediction and with both letter and word prediction, respectively. The results of the user study showed that the interface with letter and word prediction achieved the highest typing speed (5.48 words per minute) and improved by 70 % compared with the baseline. The improvement gradually increased as the user became familiar with the interface.

In this dissertation, several dynamic eye-typing interfaces were developed based on a one-point calibration and the learning costs are relatively low. The results of this dissertation can be used to improve gaze-based human-computer interfaces and support the design of robustly functioning eye-typing interfaces in public spaces.

Zusammenfassung

Entwicklungen im Feld der Eye-Tracking-Technologie eröffnen neue Möglichkeiten zur Nutzung der Blickinteraktion für die Unterstützung der Mensch-Computer-Interaktion im Alltag. Die Blickinteraktion ermöglicht es, visuelle Suche und Auslösen von Aktionen in einem Schritt zu kombinieren, da die Nutzenden relevante Informationen wahrnehmen und gleichzeitig gewünschte Aktionen auslösen können, indem sie beispielsweise ihren Blick auf ein Element im Interface richten. Die Anwendungsmöglichkeiten sind jedoch durch die Genauigkeit der Blickverfolgung und die Akzeptanz der Nutzer begrenzt. Ziel dieser Dissertation ist es daher, ein natürliches Eye-Typing-Interface zu entwickeln, das auch Eye-Tracking mit niedriger Genauigkeit toleriert. Das Interface sollte ohne umfangreiches Training leicht zu verstehen und für die Nutzung von öffentlichen Bildschirmen geeignet sein. Zu diesem Zweck wurden drei Studien durchgeführt: eine Offline-Blickdatenanalyse, eine Evaluationsstudie für ein neues Eye-Typing Interface und die Analyse einer Weiterentwicklung des neuen Interface.

Die erste Studie untersuchte die Auswirkungen der Anzahl von Objekten und der Objektbewegungsgeschwindigkeit in einem augenfolgebewegungsbasierten , kalibrierungsfreien Eye-Typing Interface, in dem sich die angezeigten Objekte linear bewegen. Die offline-Blickdaten von 25 wurden gesammelt und analysiert. Die Ergebnisse zeigen, dass die Anzahl der Objekte im Interface einen signifikanten Einfluss auf die Zahl der korrekten und falschen Detektionen der gezeigten Objekte hat. Die Leistung von Teilnehmenden war beim Interface mit 6 und 8 Objekten besser als beim Interface mit 10, 12 und 15 Objekten. Die Rate der korrekten Detektionen war bei den schnelleren Bewegungsgeschwindigkeiten signifikant höher als bei den langsameren.

Die zweite Studie beschäftigte sich mit der Entwicklung eines dynamischen Eye-Typing Interfaces. Das entwickelte Interface verfügt über eine zweistufige Auswahl und eine Ein-Punkt-Kalibrierung. Um eine Aktion auszulösen, muss der/die Nutzende zunächst auf die entsprechende Zeichengruppe blicken und dann die Bewegung des gewünschten Zeichens in dieser Gruppe mit den Augen verfolgen. Es wurde eine Benutzerstudie mit 29

Teilnehmenden durchgeführt, und vier verschiedene Arten von Feedback für diese Eye-Typing-Interface verglichen. Die Ergebnisse bestätigten, dass der Benutzer den Umgang mit dem Eye-Typing-Interface in kurzer Zeit erlernen kann. Das Interface mit dem kombinierten visuellen und auditiven Feedback erreichte die höchste Tippgeschwindigkeit.

Auf der Grundlage der zweiten Studie wurden in der dritten Studie drei Eye-Typing-Interfaces entwickelt und verglichen. Ein Interface ohne Unterstützung von Sprachmodellen (als Baseline), ein Interface mit Buchstabenvorhersage und ein drittes Interface mit kombinierter Buchstaben- und Wortvorhersage. Die Ergebnisse der Studie zeigen, dass das Interface mit Buchstaben- und Wortvorhersage die höchste Tippgeschwindigkeit erreichte (5.48 Wörter pro Minute), die sich im Vergleich zum Baseline Interface um 70 % verbesserte. Je vertrauter die Nutzenden mit dem Interface wurden, desto stärker verbesserte sich die Tippgeschwindigkeit mit dem System.

In dieser Dissertation wurden mehrere dynamische Eye-Typing-Interfaces basierend auf einer Einpunktkalibrierung entwickelt, deren Nutzung nur geringen Lernaufwand erfordert. Die Ergebnisse dieser Dissertation können genutzt werden um blickbasierte Mensch-Computer-Schnittstellen zu verbessern und unterstützen das Design von robust funktionierenden Eye-Typing Interfaces im öffentlichen Raum.

Acknowledgements

Foremost, I would like to express my deepest gratitude to my supervisor, Prof. Dr.-Ing. Matthias Rötting, for his guidance and support throughout the journey of my doctoral research. His vision, and sincerity have deeply inspired me. It was very fortunate for me to work and study under his supervision. I would also like to thank Prof. Dr. John Paulin Hansen and Prof. Dr.-Ing. Henning Jürgen Meyer for responding immediately and positively to my request for serving as a member of my doctoral committee.

Beyond my supervisor and committee, I got extensive support from Dr. Felix Wilhelm Siebert. I want to express thanks to him for the thorough comments and valuable suggestions. I would like to thank my colleagues from the Chair of Human-Machine Systems, especially Mario Lasch and Stefan Damke, thank you for their support on the hardware. Thanks also to Elisabeth Lange, who always encouraged me to learn German. In addition, I would like to thank Dr. Xixie Zhang, Ying Jiang, Yuan Liu, and Dr. Xiaohan Li, we had endless discussions about research as well as daily life and I will always remember the time we spent together in Berlin.

I am also thankful for my friends who never make me feel lonely when I am away from home. Hans-Georg Mosler, Bing Liu, Dr. Junhua Kang, Dr. Lin Chen, Jing Yang, Dr. Xin Wang, Dr. Chun Yang - thank you!

Finally, the most important thanks go to my family. I would thank my parents for their constant support, understanding, and love. A special thank goes to my husband, Dr. Yu Feng, you are the joy of my life!

Contents

List of Figures

List of Tables

List of Abbreviations

HCI: Human-computer interaction

WPM: Words per minute

KSPC: Keystrokes per character

MSD: Minimum string distance

CER: Corrected error rate

UER: Uncorrected error rate

KS: Keystroke savings

NASA TLX: NASA Task Load Index

DBSCAN: Density-based spatial clustering of applications with noise

ODR: Orthogonal distance regression

ART: Aligned rank transform

CNNs: Convolutional neural networks

HMDs: Head-mounted displays

AR: Augmented reality

VR: Virtual reality

POG: Photo-Oculo Graphy

VOG: Video-Oculo Graphy

POR: Point of regard

Chapter 1

Introduction

The eye is an organ for obtaining environmental information. As an important source of information, the exploration of what the eye focuses on never stops. In the past 100 years, eye-tracking technology has evolved significantly. It enables the detection of eye positions and different types of eye movements and is widely used in medical, marketing, and psychological research. Additionally, when a user interacts with an interface, the eye always looks at the intractable area before activating it. Thus, apart from the registration of eye movements, person's gaze has been used to allow users to control interfaces (Sibert & Jacob, 2000).

Gaze interaction has the potential to become a ubiquitous part of assisting our daily interaction. Gaze input makes it possible to combine both visual search and action activation into one step, as users can perceive relevant information and activate desired commands, e. g., by focusing their gaze on a corresponding item on the interface. The interest in gaze-based interaction design is growing rapidly. The applications, such as eye typing (Majaranta & Räihä, 2002), gaze-controlled web browsing (Menges, Kumar, Müller, & Sengupta, 2017), wheelchair control (Eid, Giakoumidis, & El Saddik, 2016), and gaze selection in virtual reality (Blattgerste, Renner, & Pfeiffer, 2018), have been developed and studied.

As an input modality, gaze has a number of attractive features. Firstly, gaze-based input enables a system to be more accessible. It can help people with physical limitations to communicate with each other. Not only people with permanent disabilities, such as amyotrophic lateral sclerosis (ALS) patients can benefit from this (Hwang, Weng, Wang, Tsai, & Chang, 2014), but also people with a temporary restriction for physical activity, for example, users with arm injuries, or those who are carrying belongings. Additionally, there is an increasing concern about hand hygiene during the Coronavirus Disease 2019

(COVID-19). People are encouraged to use contactless interactions to minimize, and if possible to avoid contacts between users' hands and displays. Even though there are many contactless interfaces using gesture or voice, the gaze is a potential complementary input modality to provide a more hygienic interaction when using public displays/interfaces. In addition, without bystanders hearing voice commands, gaze interaction has a high degree of privacy (Katsini, Abdrabou, Raptis, Khamis, & Alt, 2020). As an example, PIN code input using gaze is much more inconspicuous than voice or gesture control in a public setting.

Although there is an increasing interest in gaze-based interaction, proposed implementations have suffered from a number of limitations. One main challenge in the design of gaze-based interfaces is the lack of accuracy of real-time gaze coordinates, stemming from both biological characteristics of the eye as well as limitations of the eye tracking equipment. Due to miniature eye movements, such as tremor, drift, and microsaccades, there is hardly a moment that the eye is absolutely still. Thus, the use of the gaze as a cursor cannot be as accurate as e. g., using a computer mouse. Second, most gaze interfaces require time-consuming calibration before use. Even if individually calibrated, the accuracy of the eye tracker will gradually decline during use.

To overcome these challenges, pursuit-based interfaces have been proposed by Vidal, Bulling, and Gellersen (2013). Using these interfaces, users can activate an action by following the corresponding moving target with their gaze. An action is activated based on the fitting of the relative motion trajectory, instead of fixed gaze coordinates on the interface, which is why a personal calibration is not necessary for pursuit-based interaction. Hence, this calibration-free method has attracted a lot of interest, as it overcomes the shortcomings arising from low spatial accuracy of the gaze data. Researchers have explored dynamic gaze interfaces based on pursuit movements and developed different applications, such as entering PIN code (Almoctar, Irani, Peysakhovich, & Hurter, 2018; Cymek et al., 2014), eye typing (Lutz et al., 2015), controlling smart watches (Esteves, Velloso, Bulling, & Gellersen, 2015) and selection in VR device (Sidenmark, Clarke, Zhang, Phu, & Gellersen, 2020).

However, despite the advantages of pursuit-based gaze interaction, designing such a dynamic interface needs to engage with the following problems. First, Ludvigh and Miller (1958) reported that visual acuity decreases significantly with the increase in angular velocity of the moving object. In pursuit-based gaze interaction, users need to search for and identify an intended target moving at a certain speed from multiple moving objects to activate a command. Second, when eye gaze is used as an input modality, the basic functions of the eye, such as seeing and perceiving visual information, need to be differentiated from the selection of an action, i. e., Midas touch problem (Majaranta & Räihä, 2002). If this differentiation does not work well, users tend to feel annoyed when their visual search is interrupted by a false activation.

1.1 Aims of the thesis

Designing an eye typing interface for public display is still challenging. Of course, there is solution available to enter text using eye without personal calibration, for example, *SMOOVS* (Lutz et al., 2015). This system works with one-point calibration and features a circle layout in alphabetical order. However, it lacks user acceptance and its typing speed is relatively slow. Furthermore, in this research area of pursuit-based gaze interaction, the use of linear trajectory has not gained much attention, compared with circular trajectory. But, it is common for objects to move in a straight-like line in daily life, such as a moving train. That is, linear pursuit movements are very natural motions for the eye.

Therefore, this thesis starts from the investigation of natural linear pursuit eye movements and aims to develop an eye typing interface without individual calibration, which is simple and easy to understand without extensive training. In particular, the following research questions are tackled.

- What is the appropriate number of objects and moving speed for the gaze interface based on linear motion?
- How to design a dynamic eye typing interface that is easy to learn and user-friendly?
- How to use language models to improve the typing performance of the dynamic eye typing interface?

To address the above research questions, three main studies are conducted.

Firstly, this dissertation investigates the effect of the number of objects and object moving speed in a linear trajectory. The main goal of this study is to develop a deeper understanding for the effect of object number and object moving speeds on the detection performance of gaze interfaces based on linear trajectory smooth pursuit. Besides, three detection methods are compared regarding the detection rates. Based on the findings, some guidelines are formulated for the design of future eye typing interface.

Secondly, a new design concept, i. e., hybrid gaze interface, is proposed. This gaze interface is featured with a two-stage selection and try to combine two selection methods (dwell- and pursuit-based gaze interaction). The interface can be adapted to low-accuracy equipment environments and also minimizes distractions to users' attention. A novel eye typing interface is designed based on this hybrid concept to create a robust gaze-enabled text entry system. Further, this study is interested in providing feedback to achieve more efficient typing performance and a better user experience. In the eye typing system, the information input and output share the visual channel and it's essential to provide the user with the system state. This study would like to address the questions that provision visual feedback is a more direct way or leads to mental overload or that the feedback from another channel, e. g., auditory, is more efficient, or combined modalities are a better solution for such a dynamic gaze interface. Thus, to investigate the effect of feedback, different modalities (visual only, auditory-only, combined visual and auditory, and no feedback) are compared in a user study.

In the third study, the language model is utilized to further improve the typing performance of the hybrid eye typing interface. Three interfaces are designed: with letter prediction, letter+word prediction, and no prediction. Those interfaces are compared in a user study.

1.2 Structure of the thesis

This dissertation comprises seven chapters, and structures as follows:

Chapter 1 briefly introduces this research, which contains the aims of this thesis and outline.

Chapter 2 provides the background information related to this work, including the physiology of the eye, categorisation of eye movements, eye-tracking technology. Besides, the related mathematical theories and evaluation metrics for both objective and subjective measures are described in this chapter.

Chapter 3 presents an overview regarding gaze-based interaction. This chapter introduces different gaze-based applications, and summarizes specific related work about eye typing.

Chapter 4 introduces the study evaluating effects of differential number of objects and moving speeds on the detection of pursuit eye movements. Besides, three detection methods are compared.

Chapter 5 describes the second study, which aims at developing a hybrid eye typing interface. The typing performance and subjective evaluation are compared with four forms of feedback.

Chapter 6 subsequently tests three typing interfaces to gain deeper insight into provision word or letter prediction for this hybrid eye typing interface.

Chapter 7 discusses the studies presented in this thesis, and specifically details the findings regarding design the hybrid eye typing interface. At the same time, the contributions of this thesis are declared. Besides, the outlook is provided regarding the potential application of gaze interaction.

Chapter 2

Background

This chapter briefly explains the physiology of the eye and eye movements (Section 2.1), then moves on to the introduction of eye-tracking technology (Section 2.2). In addition to theoretical reviews, the evaluation metrics and mathematical algorithm are introduced in Section 2.3.

2.1 Physiology of eye and eye movements

The eye is an important body organ, which is used to acquiring visual information in most of the activities, such as reading, sporting, and walking. Although eye is a robust source of information, eye moves all time to obtain a clear vision of an interesting object. The fovea is the area of the retina with the highest visual acuity. Throughout the visible visual range, only the information that falls in the fovea will be processed precisely. The eyeballs often rotate unconsciously, allowing the light to focus on the fovea as much as possible. There are several major patterns of eye movements and they are divided into five basic types: saccadic, smooth pursuit, vergence, vestibular, and physiological nystagmus, i. e., miniature movements of fixation (Robinson, 1968). This chapter focuses on the eye movements related to gaze interactions.

2.1.1 Fixations

Fixations appear at an early stage in the life span (Roucoux, Culee, & Roucoux, 1983), which are relatively static eye movements and related to the perception of a stationary object (Duchowski, 2017, p. 44). The duration of one fixation is generally 200-600 ms (Jacob, 1995).

Fixations do not mean absolute stillness, which are accompanying with miniature eye movements: tremor, drift, and microsaccades (Carpenter, 1988, p. 124-131). Microsaccades are responsible for correcting the displacement of the eye's fixation point caused by tremor and drift, which is generally less than 5° visual angle (Martinez-Conde, Macknik, & Hubel, 2004).

2.1.2 Saccades

Saccades are rapid ballistic eye movements and happen frequently when the fovea repositioned to the next location, for example, people scanning the environment during looking for something in the room. As jerky movements, saccadic movements are considered to be one of the fastest movements of the body and the velocity which are stable in the range of 30 to 40 °/s (Holmqvist et al., 2011, p. 326-328), and the maximum saccadic velocity is ranged from 600 to 900 °/s (Fisher, Monty, & Senders, 1981, p. 33). The duration of saccadic movements fluctuates with the amplitude (as shown in Equation 2.1) (Carpenter, 1988, p. 70-71), which ranged from 30 - 80 ms (Orban de Xivry & Lefevre, 2007). Saccades can not only be intentionally controlled but also be triggered involuntarily by stimuli, such as sudden changes in the peripheral area. The latency of saccades is between 150 and 250 ms (Rashbass, 1961).

$$\text{Saccade duration (ms)} = 2.2 * \text{Saccade amplitude } (^\circ) + 21 \tag{2.1}$$

2.1.3 Smooth pursuits

Smooth pursuit is considered as the intervals eye movements between saccadic movements when the eye follows a moving object (Rashbass, 1961). When people track a moving object with their eyes, those eye movements are known as smooth pursuits which enable people to keep the moving object on the fovea (Robinson, 1965). Meyer, Lasker, and Robinson (1985) defined “when a visual target moves, humans, and other foveate animals, track it, if the head is stationary, with an eye movement called smooth pursuit." Unlike saccades, being both active and inactive response to stimulus, smooth pursuit eye movements happen when eyes proactively follow a moving object or imaginary moving objects (Fukushima, Fukushima, Warabi, & Barnes, 2013). This movement is performed under voluntary control, i. e., the observer can choose whether to follow the moving target or not.

For ramp target motion, eye can keep up with target motion when the velocity of smooth pursuit motion is less than 30 °/s (Robinson, 1965). The maximum pursuit velocity reaches 30 to 40 °/s (Fisher et al., 1981, p. 33). When the target is moving slowly, the pursuit eye movements are more effective and the velocity is closely related to the velocity of the target (Rashbass, 1961; Robinson, 1965). The visual acuity decreases when the velocity of the target is more than 3 °/s (Westheimer & McKee, 1975). Smooth pursuit eye movements tend to maintain eye velocity to match the target velocity and smooth pursuit gain can be defined as the ratio of the eye movement velocity to the target velocity. If the velocity is less than 10 °/s, the eye tends to keep up with the movement of the target. However, the velocity of smooth pursuit is hard to perfectly fit the movement of targets, i. e., the smooth pursuit gain is approximately 1 (Collewijn & Tamminga, 1984; Rashbass, 1961). If the velocity of target motion is too fast (exceed 40 °/s), smooth pursuit eye movement will be interrupted by catch-up saccades to keep up with the moving target (Dodge, 1903; Robinson, 1965).

The latency of evocation smooth pursuit motion ranges from 80 to 150 ms (Lisberger, Morris, & Tychsen, 1987; Rashbass, 1961; Westheimer & McKee, 1978). When the target is moving in an unpredictable direction and speed, the latency of smooth pursuit expands to 100-200 ms. Conversely, when the visual system can anticipate the moving direction of the target, the latency is not only shorter, even negative, i. e., the eyes start moving before the target (Burke & Barnes, 2006; de Hemptinne, Lefevre, & Missal, 2006; Stark, Vossius, & Young, 1962). Due to the latency, the eye accelerates with a ballistic acceleration until it reaches the target speed, which is considered as “open-loop pursuit". Then the “closed-loop pursuit" fallowed within 300 ms after the target starts to move (Holmqvist et al., 2011, p. 432).

Orientations of smooth pursuit eye movements can have an effect on accuracy. Collewijn and Tamminga (1984) found that horizontal pursuit movement performs slightly smoother and more precisely than vertical in general. Ke, Lam, Pai, and Spering (2013) investigated asymmetries in smooth pursuit eye movements and the horizontal and downward moving direction are the most preferred regarding smooth pursuit eye movements.

2.1.4 Comparison saccadic and smooth pursuit movements

Both saccadic and smooth pursuit eye movements refer to the dynamic visual tracking and aim to maintain the target on the fovea. Behavioral and neurophysiological studies have revealed that saccades and pursuit eye movements play a synergistic role in visual tracking. They are two modes of oculomotor control but share a single sensorimotor process (Orban de Xivry & Lefevre, 2007). Saccades are rapid eye movements to align the visual axis with the target, where smooth pursuit eye movements refer to track the relatively slow-moving targets. The latency of pursuit movement is generally shorter than saccadic latency (Fisher et al., 1981, p. 33). Smooth pursuit eye movements usually require a moving target to follow (Young, 1971, p. 433-434), while saccades can be made without stimuli (Duchowski, 2017, p. 40-41). The difference between saccadic and smooth pursuit eye movements is briefly summarized in Table 2.1.

Table 2.1: Comparison of saccadic and smooth-pursuit eye movements

Dimensions	Saccades	Smooth Pursuits
Velocity	30 - 500 °/s	< 30 °/s
Latency	150 - 250 ms	80 - 150 ms
Gather visual information	no	yes
Target stimuli	not necessary	necessary
Sensorimotor process	shared	

2.2 Eye tracking technology

Eye tracking is known as a technology used to measure eye position and eye movement. The device used for eye tracking is called eye tracker. The eye-tracking methods could be divided into four broad categories: Electro-Oculo Graphy (EOG), Scleral contact lens/Search coil, Photo-Oculo Graphy (POG)/Video-Oculo Graphy (VOG), and video-based combined pupil and corneal reflection (Duchowski, 2017, p. 49-56).

Electro-Oculo Graphy is based on the recording of the electrical potential difference between the electrodes around the ocular cavity. The measurements of eye movements are influenced by head movement in this method (Duchowski, 2017, p. 50).

Scleral contact lens/Search coil is one of the most frequently used methods for measuring eye movements, which is to attach a mechanical or optical reference to a contact lens. And people wear directly over the eye to measure the eye movements. Head movements are also involved during measuring (Duchowski, 2017, p. 51).

Photo-Oculo Graphy (POG)/Video-Oculo Graphy (VOG) used real-time image processing technology to record the eye movements. Ocular features (such as the apparent shape of the pupil, the position of the corneal limbus and the corneal reflection) are extracted. Sometimes, chin rest is needed to fix head (Duchowski, 2017, p. 52).

However, the above mentioned methods are hard to measure the point of regard (POR). Video-based combined pupil/corneal reflection is one of the most popular methods for the measurement of the point of regard in real-time. Corneal reflection and the pupil center are two key features of video-based corneal reflection. Corneal reflection was first used for objective eye measurements in 1901 (Robinson, 1968). When light enters the eye, there are four reflections, also know as Purkinje reflection (illustrated in Figure 2.1). Video-based eye tracking is usually based on the first Purkinje reflection (glint). The corneal reflection of a light source (usually near-infrared light) is measured by the relative position of the center of the pupil (Duchowski, 2017, p. 52-54). The eye tracker apparatus of video-based corneal reflection has two forms: table-mounted and head-mounted. The table-mounted eye tracker i. e., Tobii EyeX was used in the following studies, which provided the measurements of the point of regard and tolerated moderately head movements. Besides, the price of this eye tracker apparatus is relatively low (about 100 $). Thus, Tobii EyeX was frequently used in gaze interaction studies (Esteves et al., 2015; Feit et al., 2017; Morimoto & Amir, 2010).

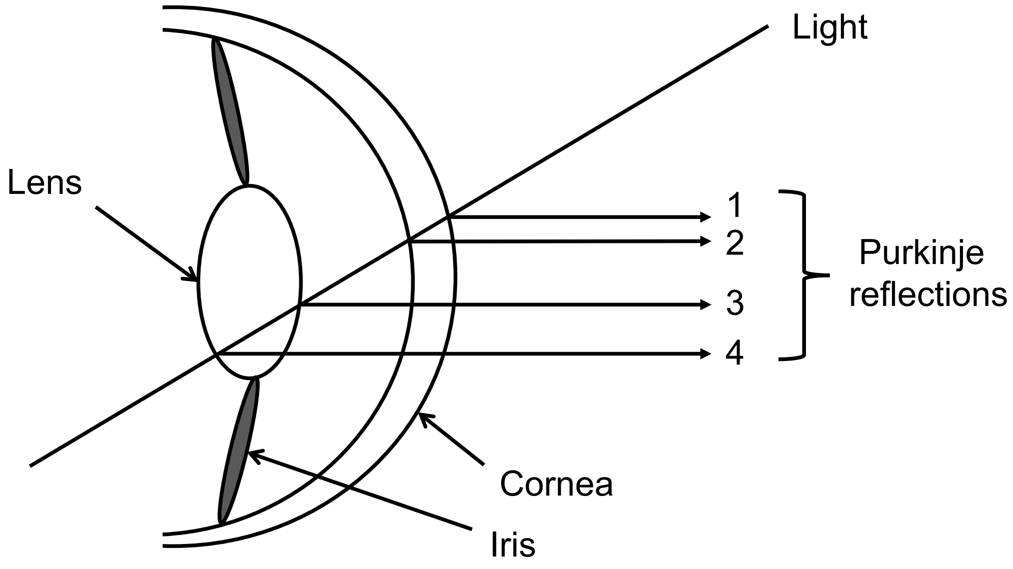


Figure 2.1: Purkinje reflections. Adapted from (Duchowski, 2017, p. 55)

2.3 Evaluation metrics

The evaluation of eye typing could be divided into two parts: performance measures and subjective evaluations. The text enter rate (Subsection 2.3.1), text enter error rate (Subsection 2.3.2) and keystroke savings (Subsection 2.3.3) are the main parts of the performance measures, whereas perceived workload, and open questions are used to evaluate the subjective feedback from participants. The evaluation metrics used in this thesis are explained in the following subsections.

2.3.1 Text entry rate

Text entry rate refers to how fast users can enter text in a typing system. Since the average word length is different in various languages, in this thesis, typing speed is measured in words per minute (WPM), where a word is regarded as five characters, i. e., regardless influence of language type on word length. The counting includes letters, spaces, punctuations, etc. (MacKenzie & Tanaka-Ishii, 2010, p. 48). Words per minute are computed according to the following equation:

$$\text{WPM} = \frac{|T| - 1}{S} \times 60 \times \frac{1}{5} \tag{2.2}$$

where $|T|$ refers to the length of total typed characters in the transcribed string. S represents run time in seconds.

2.3.2 Error rates

Typing error rates mainly include the corrected and uncorrected errors. The corrected errors are the errors that were made but also corrected during typing, whereas the uncorrected errors refer to the errors left in the transcribed text. The used metrics are introduced in the following sections.

Keystrokes per character (KSPC) measures how frequently users correct the entered characters during typing. Keystrokes per character are calculated by the ratio of the sum of letters and including spaces to the final number of characters in the entered string (MacKenzie & Tanaka-Ishii, 2010, p. 52). The formula for computing keystrokes per character is:

$$\text{KSPC} = \frac{|IS|}{|T|} \tag{2.3}$$

In Eq.(2.3), $|IS|$ stands for the length of all characters, including backspaces. As mentioned before, $|T|$ describes the length of total typed characters in the transcribed string. If the user typed perfectly without using backspace and the keystrokes per character value is 1.

Minimum string distance (MSD) is a string metric for measuring the minimum number of error correction operations required to convert a string to another string (Levenshtein, 1966; MacKenzie & Tanaka-Ishii, 2010). MSD error rate is the uncorrected error and reflects how many errors remain in transcribed string. The formula for calculating MSD error rate is:

$$\text{MSD error rate} = \frac{MSD(P,T)}{MAX(|P|,|T|)} \tag{2.4}$$

In Eq.(2.4), P refers the given string. $|T|$ is the length of total typed characters in the transcribed string.

To evaluate the error rates, the keystrokes per character (errors during entry), minimum string distance (errors after entry) are used as dependent variables in the first study. Since, the language model is added for the eye typing interface in the third study, and the number of keystrokes required for typing one word is no longer equal to the length of the word, i. e., if the next possible word is successfully predicted and selected by the user, the number of activated keystrokes is smaller than the length of the word.

Thus, three new evaluation metrics are introduced for the third study, they are uncorrected error rate (UER), corrected error rate (CER), and keystroke savings (KS). One advantage for corrected error rate and uncorrected error rate is that those two metrics share the same denominator, which can be combined to a total error rate (CER + UER). Despite the absence of a description of the total error rate, Chapter 6 analyzes corrected error rate and uncorrected error rate separately.

Corrected error rate (CER) measures the errors corrected during typing, which is similar to the keystrokes per character (MacKenzie & Tanaka-Ishii, 2010, p. 55-56). The formula for calculating the corrected error rate is:

$$\text{Corrected error rate} = \frac{IF}{C + INF + IF} \tag{2.5}$$

In Eq.(2.5), IF refers to all characters backspaced during entry. C stands for all correct characters in the transcribed text. INF represents all incorrect characters in the transcribed text.

Uncorrected error rate (UER) refers to the errors left in the transcribed text, which is similar to the error rate based on minimum string distance (MacKenzie & Tanaka-Ishii, 2010, p. 55-56). The formula for calculating uncorrected error rate is showed in Equation 2.6, the denotation is the same as in Equation 2.5.

$$\text{Uncorrected error rate} = \frac{INF}{C + INF + IF} \tag{2.6}$$

2.3.3 Keystroke savings

Keystroke savings (KS) reflects how many keystrokes are saved with word prediction/completion (Trnka & McCoy, 2008). The formula for calculating KS is:

$$\text{KS} = \frac{Keys_{phrase} - Keys_{with\,prediction}}{Keys_{phrase}} * 100\% \tag{2.7}$$

In Eq.(2.7), $Keys_{phrase}$ refers to the number of keystrokes needed to enter without any word prediction. $Keys_{with\,prediction}$ is the number of the keystrokes entered with word prediction, the backspaces and corrected characters are not included.

2.3.4 Subjective workload

As Rubio, Díaz, Martín, and Puente (2004) writed “The evaluation of mental workload is a key point in the research and development of human-machine interfaces, in search of higher levels of comfort, satisfaction, efficiency, and safety in the workplace." Since the eye is responsible for signal processing of both input and output in the gaze-based interface, the study regarding gaze interaction needs to pay more attention to mental workload during designing the interface.

There are a lot of assessment tools for subjective workload, such as NASA Task Load Index (NASA TLX) (Hart & Staveland, 1988), Subjective Workload Assessment Technique (SWAT) (Reid & Nygren, 1988) and Workload Profile (WP) (Tsang & Velazquez, 1996).

NASA TLX is a subjective, multidimensional assessment tool including mental demand, physical demand, temporal demand, performance, effort, and frustration, in all six dimensions. The score for each dimension ranges from 0 to 100, and a lower score means a better evaluation. SWAT is a workload assessment tool and has three dimensions, they are time load, mental effort load, and psychological stress load. Participants can evaluate with three levels, low, medium and high. It has a pre-task, also known as card sorting procedure, i. e., to develop individual workload scales. Thus, the implementation of this method is considered relatively complex and time consuming. Workload profile method is built from the multiple resource framework proposed by Wickens (2002) and concludes eight workload dimensions. They are perceptual/central processing, response selection and execution, spatial, processing, verbal processing, visual processing, auditory processing, manual output, and speech output.

Taking into account the assessment dimensions corresponding to the typing task and ease of use, the NASA TLX is chosen and used in the third study of this thesis. The NASA TLX questionnaire can be found in Appendix C.1.

2.4 Mathematical methods

This section briefly introduces the mathematical methods used throughout this dissertation, including density-based spatial clustering of applications with noise (DBSCAN) and orthogonal distance regression (ODR).

2.4.1 Density-based spatial clustering of applications with noise (DBSCAN)

DBSCAN is the most common density-based clustering algorithm and is used to remove the spatial outlier points in the following studies. Ester, Kriegel, Sander, and Xu (1996) proposed this method which can be used to cluster a set of points in space, that is, to classify points into core points, reachable points, and outliers. There are two parameters that need to be given for DBSCAN. One is the maximum distance (also known as ϵ), which is the distance between two gaze points, one of which is considered to be in the vicinity of the other. The number of samples (minimum sample size) in the neighborhood where a point is considered a core point. The example is illustrated in Figure 2.2.

The points near A are core points and form one cluster. Points B and C are not core points, but are connected by density through the cluster of A and therefore belong to this cluster. Point N is neither a core point nor density-connected via the cluster of A, so it is classified as noise.

In this thesis, python package “sklearn.cluster.DBSCAN" is used in the one-point calibration process. The methods will return the estimated number of clusters and number

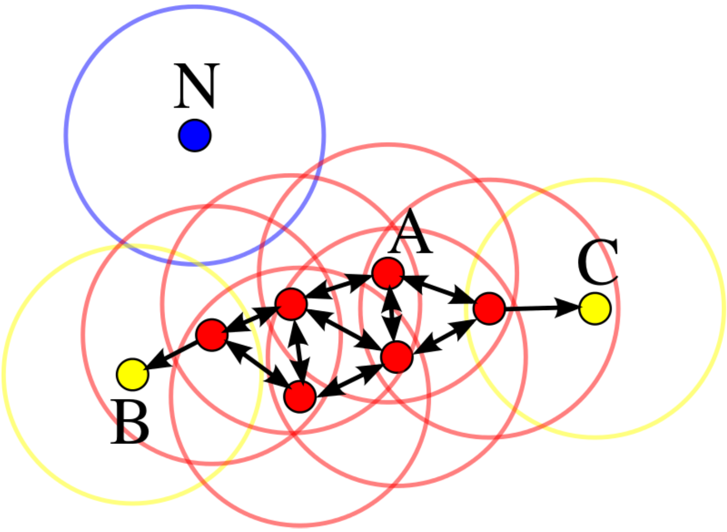


Figure 2.2: Example of DBSCAN (example image under CC BY-SA 3.0)

of noise gaze points. All gaze data are labeled with either cluster number or with “-1" (for noise points).

2.4.2 Orthogonal distance regression (ODR)

Orthogonal distance regression is a special case of total least squares. It aims to minimize the orthogonal distance from data points to a functional or structural model (Boggs & Rogers, 1990).

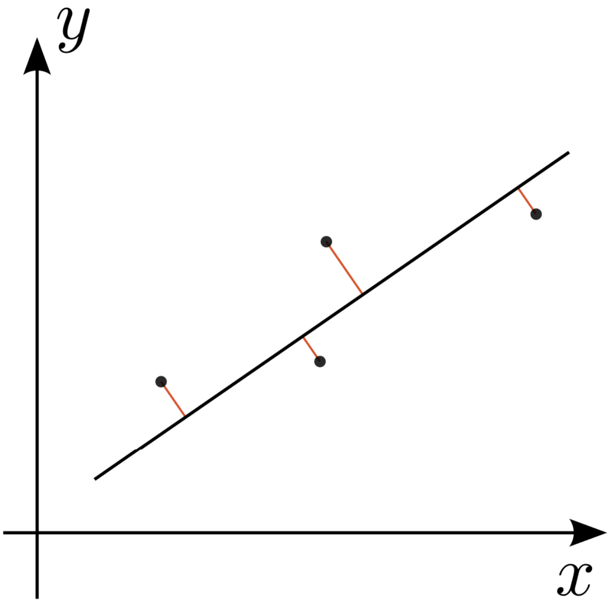


Figure 2.3: Illustration of ODR (example image under CC BY-SA 3.0)

As shown in Figure 2.3, the error in both X and Y coordinates are taken into account in orthogonal distance regression, i. e., the sum perpendicular distances of the points with respect to the estimated line is calculated. The process of estimation can be defined as:

$$\begin{aligned} &X_i = x_i - \delta_i \\ &Y_i = y_i - \epsilon_i \\ &r_i^2 = min\left\{\epsilon_i^2 + \delta_i^2\right\} \\ &\qquad \text{subject to: } Y_i = f\left(X_i + \delta_i; \beta_i\right) - \epsilon_i \end{aligned} \tag{2.8}$$

where $X_i, Y_i, i = 1, ..., n$, are observed random variables with the true values $x_i, y_i, i = 1, ..., n$. δ_i presents the random error associated with x_i, and ϵ_i stands for the random error associated with y_i. r is the orthogonal distance from the point to a linear modal. The error model is estimated by achieving the minimization of the sum of the squared distances of r.

2.5 Summary

This chapter provides an overview of eye movements, including the general classification of eye movements and their characteristics. Besides, the different eye-tracking technology is briefly introduced. Last but not least, the evaluation metrics and mathematical methodologies are described in detail in this chapter.

Chapter 3

Related work

In this chapter, related work focuses on two parts. First, the different gaze interaction designs are reviewed briefly in Section 3.1. Then, the studies of eye typing are summarized in Section 3.2. Based on the previous work, the potential shortcomings of the research area are highlighted in Section 3.3.

3.1 Gaze interaction

Gaze control has a number of advantages over those input methods that rely on physical touch, e. g., hands-free interaction for aseptic environments and increased privacy, since inputs cannot be visually observed by third parties (Cymek et al., 2014). Gaze input provides a natural and fast method for interacting with computers. In some specific situations, the reaction speed can exceed the mouse (Sibert & Jacob, 2000).

Hence, a large number of applications for gaze-based interaction have been developed for everyday human-computer interaction. Gaze typing (Majaranta, MacKenzie, Aula, & Räihä, 2006; Ward, Blackwell, & MacKay, 2000), password input (Cymek et al., 2014; de Luca, Weiss, & Drewes, 2007), controlling smart watch (Esteves et al., 2015), map reading (Göbel, Kiefer, Giannopoulos, Duchowski, & Raubal, 2018), controlling telepresence robots (G. Zhang, Hansen, & Minakata, 2019) and selection in VR/AR device (Esteves et al., 2017; Khamis, Oechsner, Alt, & Bulling, 2018; Sidenmark et al., 2020) have been proposed as use cases.

Although there is an increasing interest in gaze-based interaction, proposed implementations have suffered from a number of limitations. The so-called Midas touch problem represents one of the main challenges for gaze-based interaction. It describes the difficulty to distinguish between visual search patterns and the intentional selection of actionable items on an interface.

In addition, most eye trackers require calibration before use, which is time-consuming and can be inconvenient for spontaneous interaction. And even with individual calibration, eye trackers are still prone to problems with fluctuating spatial accuracy during use. Feit et al. (2017) found that eye-tracking systems need to be re-calibrated multiple times a day to maintain interactable accuracy for most eye typing systems, and the individual difference is relatively large both in accuracy and precision as well.

Another challenge in the design of gaze-based interfaces is the lack of accuracy of real-time gaze coordinates, stemming from both biological characteristics of the eye as well as limitations of the eye-tracking equipment. Due to miniature eye movements, such as tremor, drift, and microsaccades, there is hardly a moment that the eye is absolutely still. Thus, similar to the obvious and well-known "Fat Finger" problem in touchscreen (Bi, Li, & Zhai, 2013), the use of the gaze as a cursor cannot be as accurate as e. g., using a computer mouse.

According to the above mentioned problems, three mechanisms for the selection are prevalent to trigger an action in gaze-based interfaces, namely fixation-based, gesture-based, and smooth-pursuit based selection. This chapter gives a brief overview of the design of gaze interaction methods mentioned above and introduces how people can interact with the interface using their eyes.

3.1.1 Fixation-based gaze interaction

In fixation-based systems, actionable items are fixed on the interface and the user's gaze locates on the presented interface to select it. Actions, e. g., the selection of objects, are triggered through gaze fixation on an actionable item for a set amount of time (typically called dwell time). Gaze-based systems suffer from the Midas touch problem, where actionable items are triggered, although users just look at the actionable item to identify it (Jacob, 1990; Stampe & Reingold, 1995). To overcome the Midas touch problem, the duration of dwell time is set to longer than general fixation (typically 200-600 ms) (Jacob, 1995). The dwell time is varied from 600 to 1000 ms in studies related to fixation-based gaze interface (Majaranta & Räihä, 2002).

Fixation-based systems suffer from two major disadvantages. First, those systems require an individual calibration of the eye-tracker, as they depend upon a high accuracy in the registration of the gaze position, to locate the user's gaze in relation to the objects on the interface. Second, the size of the actionable item is limited due to the size of the fovea and the size of the gaze jitter. The area covering 1° visual angle is considered to be the minimum area required for gaze-based interaction, and this distance corresponds to approximately 1 cm at 65 cm from the display screen. With those limitations, the users have to go through the calibration process before use. Besides, relatively large actionable items with wide separation between objects have to be considered, which limits the number of actionable items on the interface.

3.1.2 Gesture-based gaze interaction

Drewes and Schmidt (2007) introduced gaze gesture method and an action is triggered after completion of a fixed sequence of gaze movements. Gesture-based gaze interaction does not require a graphical interface. This approach is independent of display space and insensitive to eye-tracking accuracy. However, since users need to learn and remember available gestures, gaze gesture-based interaction is not practical for walk-up interaction, where users have no prior knowledge about the interaction.

3.1.3 Pursuit-based gaze interaction

The gaze interfaces based on smooth-pursuit eye movements are composed of moving actionable objects. Instead of an exact location of a user's eye-gaze on the interface, gaze trajectories are used for object identification, which result from the human eye following a moving object. These eye movements are called smooth-pursuit movements, lending the name to these categories of gaze interfaces.

To overcome the calibration challenge, pursuit-based interfaces have been first proposed by Vidal et al. (2013). Using these interfaces, users can activate an action by following the corresponding moving target with their gaze. An action is activated based on the fitting of the relative motion trajectory, instead of fixed gaze coordinates on the interface, which is why a personal calibration is not necessary for pursuit-based interaction. Hence, this calibration-free method has been attracting a lot of interest, as it overcomes the shortcomings arising from the low spatial accuracy of gaze data.

To relate the gaze trajectory to moving objects in the interface, Vidal et al. (2013) utilized Pearson's product-moment correlation in their first implementation of a smooth-pursuit based interface. The parameters, i. e., number of objects, moving trajectory, moving speed, window size, and correlation threshold, are evaluated. They reported that the detection rate starts to decrease when there are eight or more objects on the screen. However, when analyzing the data separately for linear and circular motion, it was found that the detection rate of the circular motion is higher, which allows a more robust detection performance for up to 15 objects. The larger window size (500 ms) has a higher detection performance than the smaller one (100 ms). For slow-moving speed, a larger window size was required. The correlation threshold value is recommended to be set higher than 0.5.

For gaze interface based on smooth pursuits, many studies focus on the circular trajectories. Esteves et al. (2015) explored the circular gaze trajectories in gaze-based smart watch interfaces. The moving speed was given by angular speed, but not in pixels/s or visual angle/s. However, the circular diameters were given (Small: 0.6 cm/0.98° of visual angle, Medium: 1.6 cm/2.62° and Large: 2.6 cm/4.25°). The results have shown that the Pearson correlation-based detection method performs well on up to eight moving orbits. For moving angular speed, the medium angular speed of 120 °/s achieves the best performance in detection rates, whereas the orbits with large diameters have better detection performance than others.

Drewes, Khamis, and Alt (2018) compared two detection methods, namely correlation methods (with or without 5 samples delay) with different moving speeds. The results found that the moving speeds between 6 °/s and 16 °/s result in the higher detection rate with the correlation method than the method based on Euclidean distance, whereas the euclidean method works better when the object moving speed is relatively slow.

Despite their advantages, Vidal et al. (2013) found that the detection rate was lower when objects in the interface move in linear trajectories compared to circular trajectories. The possible reason is that the detection of horizontal and vertical movement using Pearson's product-moment correlation can lead to problems. For trajectories that are purely horizontal or purely vertical, there is zero standard deviation when computing the Pearson correlation. The condition that both the highest correlation coefficient of $corr_x$ and $corr_y$ for the objects need to above a threshold is difficult to meet. Besides, circular trajectories have been found to be subject to increased gaze deviations compared to rectangular trajectories consisting of straight lines (Kosch, Hassib, Woundefinedniak, Buschek, & Alt, 2018).

Thus, apart from circular object movements, researchers have developed interfaces based on linear pursuit eye movement using other algorithms to improve the detection performance. Cymek et al. (2014) investigated moving speed (Slow, Medium, Fast) and minimal object distance (Larger, Small) for gaze interface based on linear smooth pursuit eye movements. The gaze interface contained 16 dynamic elements. Each element moved in three stages, combining horizontal and vertical movements. The detection of targets relies on analyzing the combination and classification of gaze trajectory sequences. The authors reported that in terms of user-friendliness, the most favorable results were obtained in the condition with medium speed (218 pixels/s, about 4.9 °/s) and large distance (39 pixels).

This dynamic interface design based on the linear motion was further utilized by Lutz et al. (2015), who explored an eye typing interface based on two-stage pursuit movement which contains 32 moving objects and four moving speeds was tested, called *SMOOVS*. The best typing performance was achieved with medium movement speed (300 pixels/s, 7.73 °/s). Later, Freytag, Venjakob, and Ruff (2017) compared the gaze interface for PIN input and eye typing (i. e., *SMOOVS*). The results found that the detection is more accurate and quick for the gaze interface containing less moving objects, i. e., the detection performance for PIN input is better than eye typing. So far, there has been no research specifically analyzing the influence of object number and object moving speed in linear trajectory smooth pursuit gaze-based interfaces.

In summary, the large orbits size or long moving trajectory facilitate more successful detection. Besides, a moderate moving speed is appropriate for gaze interface based on smooth pursuit, as Esteves et al. (2015) reported “if it is too slow it becomes a fixation; if it is too fast it turns into repeated saccades".

The results from the studies presented above are summarized in the following table. Some moving speed parameters presented in the table have been estimated based on the figures and descriptions available in the original papers when the actual values were not provided by the authors.

3.1.4 Feedback in gaze interaction

In addition to differences in the basic workings of gaze-based spellers, the design of appropriate feedback can enhance users’ experience (Nielsen, 1994). First, feedback can provide information about the state of the interaction, thereby allowing the user to adjust any erroneous activation before it is registered. Second, feedback can be used to confirm that a selection is registered. Generally, users are relatively tolerant of interfaces with appropriate feedback. However, inappropriate feedback can also be misleading or distracting to users.

Table 3.1: Summary the selected studies of pursuit-based gaze interaction.

	Moving speed[1]	Trajectory	Window size	Methods for detection (threshold)	Number of objects	Distance from display
Vidal et al. (2013)	100-850 pixels/s	circle/linear	100, 500 ms	correlation coefficient (0.2-0.9)	2-20[3]	85 cm
Esteves et al. (2015)	60-240 °/s[2]	circle	1000 ms	correlation coefficient (0.8)	2-16	35 cm
Kangas et al. (2016)	5-7 °/s	circle/linear	500 ms	squared distances (700 pixels)	2	50-70 cm
Drewes et al. (2018)	0.78-25.1 °/s	circle/square diamond	500 ms	correlation coefficient (0.8) euclidean distance (30 pixels)	1	53.5 cm
Cymek et al. (2014)	3.3-9.8 °/s	linear	1000 ms (one direction)	direction-based	16	70 cm
Lutz et al. (2015)	5.67-8.76 °/s	linear	>200 ms	angle-based (58°)	32	60 cm

[1] The speed parameters given in the literature vary and are divided into three categories (pixels/s, visual angle/s and angular speed/s). All speed parameters that can be converted to visual angle/s have been converted.

[2] Angular speed/s.

[3] To prevent the distraction by other moving objects on the screen, only one moving target is visible

Feedback in fixation-based gaze interface

Majaranta, MacKenzie, Aula, and Räihä (2003) found an effect of feedback on typing performance of the fixation-based system. The results indicated that participants typed faster when receiving a combined click and visual feedback than other forms of feedback such as speech only or visual only. Short no-speech sound, like a 'click', was preferred by participants over synthetic speech. The speech feedback is limited in some cases, as it takes time to pronounce a selected letter than just giving a short, no-speech sound as feedback. In addition, Majaranta et al. (2006) found that the feedback needs to be set according to the duration of dwell time.

Feedback in gaze gestures

Since the interaction via gaze gestures is usually facilitated without a graphical user interface, the related studies focus more on vibrotactile feedback (Rantala et al., 2020). The implementation of vibrotactile feedback was found to improve both the typing speed and subjective evaluation of users (Kangas et al., 2014). Köpsel, Majaranta, Isokoski, and Huckauf (2016) compared visual, haptic, and auditory feedback modalities. They evaluated feedback given both during and after the input of a gaze gesture. It was found that there are only slight differences in accuracy and user experience between different feedback modalities in gaze gesture interactions.

Feedback in pursuit-based gaze interface

The influence of feedback has also been considered in research on pursuit-based gaze interaction. Špakov, Isokoski, Kangas, Akkil, and Majaranta (2016) added a “tick" tone in smooth pursuit-based widgets. The comparison of feedback modalities (no feedback, visual, auditory, and haptics) showed that feedback conditions do not significantly affect the performance of pursuits-based gaze interaction. However, most users preferred tactile and auditory feedback (Kangas et al., 2016). To interact with dynamic interfaces with eyes, the cognitive workload is higher than the static, and suitable feedback could be considered in the design of dynamic interfaces. Even though pursuit-based gaze interaction has been studied for years, there is less research on how to design the feedback of the interaction to users.

3.2 Eye typing interface

Eye tracking technology is used in psychology, marketing, and also as an input device for human-computer interaction (Jacob, 1990). For gaze interaction, eye typing is the most widely used application. Using gaze for text entry has several unique benefits, and was initially developed as an important communication tool for people with certain disabilities (Majaranta & Räihä, 2002). Furthermore, systems have been developed to enter text using gaze for public displays (Lutz et al., 2015), and recently have been widely studied as an input method in virtual reality (Rajanna & Hansen, 2018). Dwell time, gestures, or pursuit movements are used to replace the click/tap action to activate a character. The layout of common eye typing interfaces can be divided into two parts: traditional QWERTY layouts and other specific layouts.

3.2.1 QWERTY typing layout

The QWERTY keyboard layout is familiar to most users, hence users can learn the interaction process for gaze-based typing on QWERTY interfaces quickly. There have been a number of studies involving the implementation of QWERTY layout in eye typing.

Dwell-based Typing Interface

For most eye typing systems, the input and output space share the same space and the gaze works as an invisible cursor. Designs based on dwell time are popular, as they allow to control over the Midas touch problem. That is, users can enter a letter by fixating their gaze on it for a pre-defined duration, i.e. the dwell time.

To further facilitate the typing efficiency, individual differences can be taken into account and more flexible dwell duration was investigated, e. g., users were allowed to adjust the dwell time themselves (Majaranta, Ahola, & Špakov, 2009), or systems were capable of automatically adjusting the dwell time based on users' performance (Špakov & Miniotas, 2004) or previously entered text and location of keys (Mott, Williams, Wobbrock, & Morris, 2017; Pi, Koljonen, Hu, & Shi, 2020). However, dwell-time based gaze system depends heavily on the spatial accuracy of the eye tracker, as even slight inaccuracies in tracking can lead to an unintentional input for characters close to the intended target.

Dwell-free typing interface

In dwell-free interfaces, users do not need to fixate their gaze on actionable items for a pre-defined dwell time. For example, on the context-switching interface (Morimoto & Amir, 2010), two QWERTY keyboards are displayed on the screen, and the user swipes between them to enter text. In addition to the character-level text entry method, there is a word-level text entry method based on the analysis of letter sequence using a language model. Users can enter text by looking at a sequence of letters they need to enter with their eyes (Kurauchi, Feng, Joshi, Morimoto, & Betke, 2016; Pedrosa, Pimentel, Wright, & Truong, 2015). When starting and ending a word, an explicit command needs to be given out to distinguish between receiving visual information and selection.

The traditional QWERTY layout has both advantages and disadvantages for gaze-based interfaces. Since it is well known by most users, the QWERTY layout can reduce the cost of learning. However, it was developed for finger typing and is not well adapted for gaze-based typing.

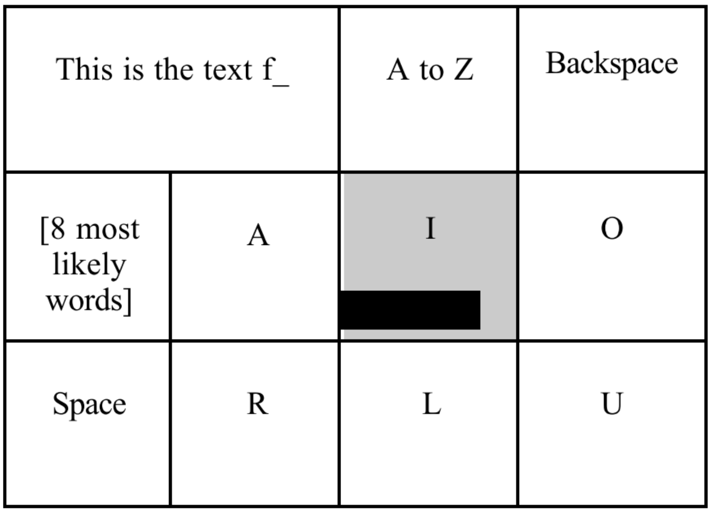
3.2.2 Specific typing interface

In addition to the traditional layout, there are a number of novel keyboard layouts designed specifically for eye typing. Due to the unstable accuracy of gaze data, those layouts try to rearrange the letters' position or cluster letters to facilitate the user experience.

Static-graphic gaze interface

J. P. Hansen et al. (2003a) developed *GazeTalk*, which has more than ten active buttons on the screen (as shown in Figure 3.1). The interface is updated in real-time and what are more likely to be entered are displayed. The characters or words displayed in each button are based on the predictions of a language model. To overcome the limitations of inaccurate gaze data, the size of actionable items is increased, which helps with item distinction, but reduces the number of displayable actionable items or results in the need for bigger displays.

Huckauf and Urbina (2008) proposed the concept of two-stage selection. Their *pEYEs* application takes the form of popup menus, this pie layout is available for menu selection and text entry (see in Figure 3.2). Users first select a cluster and then select the intended character from the selected cluster. In this interface, the displayed characters and the area for activation are separated. The characters are located near the center of the circle and the activation area is located at the edge of the circle.



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

Figure 3.1: Interface of GazeTalk (J. P. Hansen et al., 2003a).

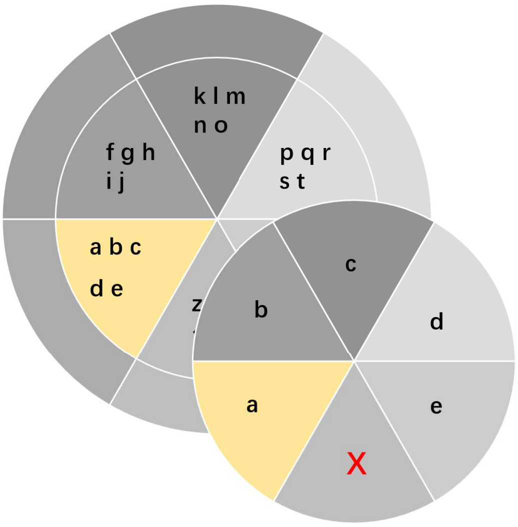


Figure 3.2: Illustration of pEYEs, adapted from Huckauf and Urbina (2007).

Additionally, research has been carried out on gaze gestures to overcome space limitations. Gaze gesture systems such as *Quikwriting* (Perlin, 1998) or *EyeWrite* (Wobbrock et al., 2008) require users to spend a certain amount of time learning and remember the gestures (see in Figure 3.3 and 3.4). Hence, they are not suitable for walk-up-and-use scenarios.

Dynamic-graphic gaze interface

A number of studies have investigated the use of a dynamic interface for gaze typing, i.e., the position of the items is not fixed in such interface. One of the well-known eye typing

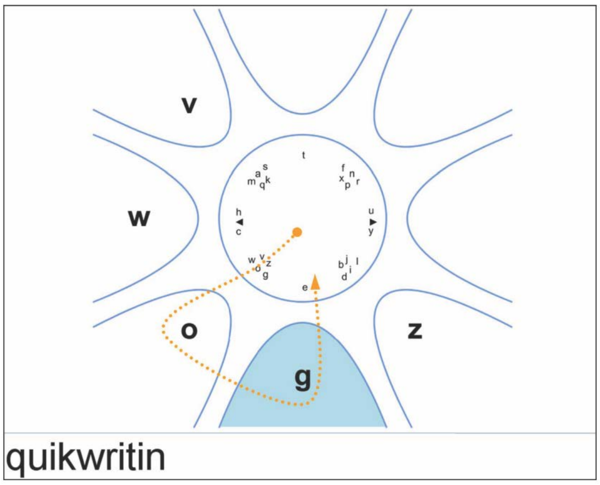


Figure 3.3: Interface of Quikwriting (Bee & André, 2008).

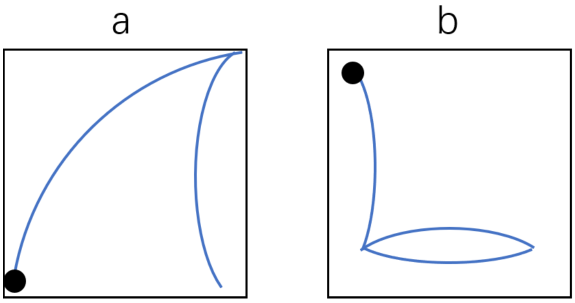


Figure 3.4: The gaze gestures of letter “a" and “b" in EyeWrite, adapted from Wobbrock et al. (2008).

systems is *Dasher* (Ward & MacKay, 2002). In this interface, characters have no fixed position and are placed in a row, according to the probabilities for the frequency of use for each character (as shown in Figure 3.5). The character row is continuously moving from right to left, with the actionable character highlighted. There was one more dynamic eye typing interface, called *StarGazer* (D. W. Hansen et al., 2008), where user could select characters with zooming and panning actions (see in Figure 3.6).

More recently, usage scenarios for gaze interaction have been broadened to include public screens. Studies are dedicated to design interactive interfaces with reduced calibration time or without calibration. Lutz et al. (2015) developed the *SMOOVS* text entry system, a pursuit-based gaze interface with one-point calibration. To avoid the Midas touch problem, the interface is divided into interactable areas and a central area that is not interactable. All elements in the interface are designed to stay static when the gaze cursor is located in the non-interactable middle area of the interface. When the

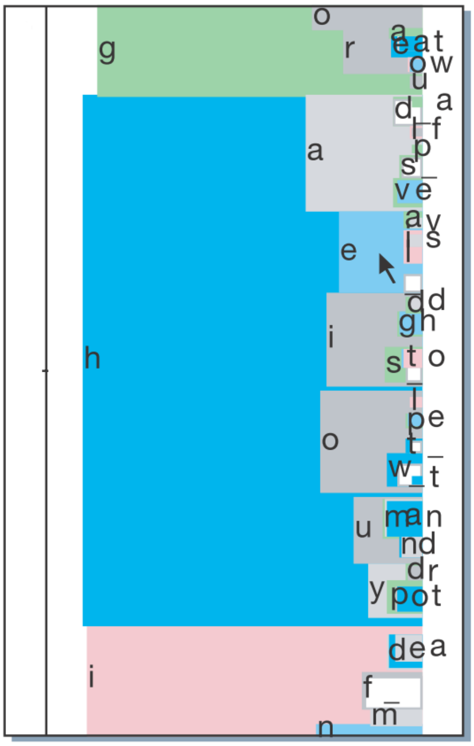


Figure 3.5: Interface of Dasher (Ward & MacKay, 2002).

gaze position is moving out of the deactivated central area, the first moving stage starts. In this stage, all clusters move outward simultaneously. In the second moving stage, the tiles within the selected cluster move apart from each other around the center of the cluster. Although users can quickly understand the interface and learn how to use it, users' attention can be distracted by objects moving at the same time in the first stage. In addition, the text entry rate for *SMOOVS* is relatively low in comparison to other typing systems. A high error rate exacerbates this problem, as users need to frequently correct wrong inputs, often resulting from involuntary initiations during the visual search for the desired character.

In order to reduce the amount of keystrokes for entering a word or sentence, word prediction function was added for *SMOOVS* interface to improve the typing speed to 4.5 WPM (Zeng & Roetting, 2018). In addition to move outward design, there is typing interface featured clockwise or counter-clockwise circular motion, called *EyeTell* (Bafna, Bækgaard, & Paulin Hansen, 2021) and the gaze estimation is based on front-facing camera. The interface of *EyeTell* consists of an inner circle (moving counter-clockwise) and an outer circle (moving clockwise), where the inner circle is formed by characters clusters and the outer circle is the characters in the selected cluster. However, the typing

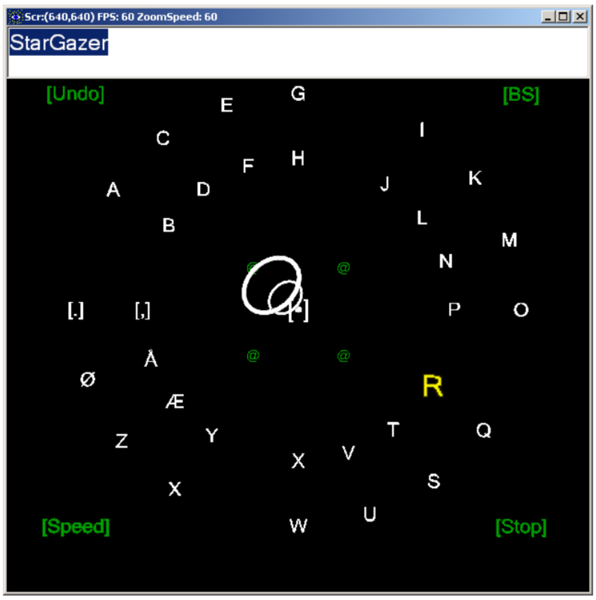


Figure 3.6: Interface of StarGazer (D. W. Hansen et al., 2008).

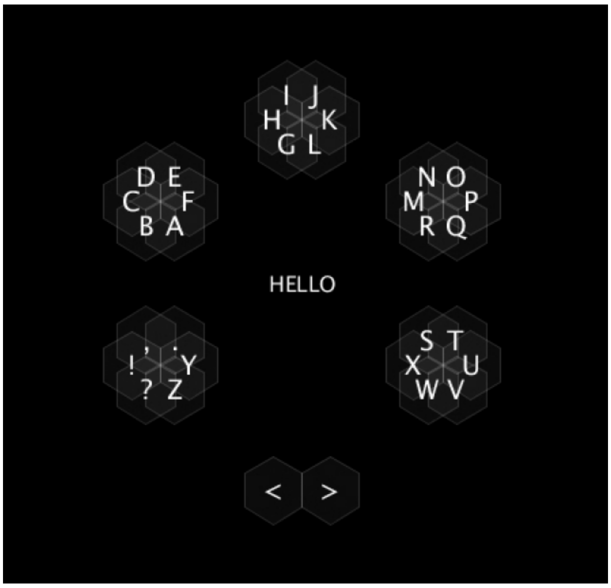


Figure 3.7: Interface of SMOOVS (Lutz et al., 2015).

speed still slow and needs to be improved to meet the needs of daily life.

3.2.3 Language model in eye typing

There has been a large volume of published studies describing the role of language models in text entry tasks. Language model is one of the most widely used methods to improve typing performance on smartphone, and it is also a very important complementary function in the brain-computer interface (BCI) system (Orhan et al., 2012; Speier, Chandravadia, Roberts, Pendekanti, & Pouratian, 2017) and eye typing system (Kristensson & Vertanen, 2012; MacKenzie & Zhang, 2008), where the typing speeds are limited by the input modality.

Provision next possible word/letter prediction

Many studies have explored adding word prediction to improve eye typing efficiency. With the word predictions, the number of keystrokes that a user needs to enter for a certain length of sentence is reduced. The user doesn't have to enter every letter, as a result, the times of wrong activated selection are reduced accordingly. MacKenzie and Zhang (2008) introduced a weight-based algorithm, and compared letter and word prediction. With weighting factors, the system recognized the input letter based on the combination of word stem frequency and the distance from the current gaze position to adjacent letters. In condition letter prediction, the next three possible letters were given and highlighted, and the word prediction function was tested with five candidate words in the word prediction condition. They found that letter prediction provided an effective method to advance the typing performance of the interface with small buttons. And word prediction has an advantage on the interface with big buttons. In addition to the traditional QWERTY layout, there are also new gaze typing interfaces containing word predictions, such as pEYEwrite (Urbina & Huckauf, 2010), GazeTalk (J. P. Hansen, Johansen, Hansen, Itoh, & Mashino, 2003b), and SliceType (Benligiray, Topal, & Akinlar, 2019).

However, all the above mentioned methods are difficult to deploy in public display due to the higher requirement for the accuracy of the gaze position data. The eye tracker needs to be calibrated before use or re-calibrated during use, and this calibration process is relatively boring and time-consuming. Although word prediction was considered in pursuit-based typing interface (Abdrabou, Mostafa, Khamis, & Elmougy, 2019; Zeng & Roetting, 2018). Zeng and Roetting (2018) reported that most participants were willing to use the word prediction, and think it is helpful, but some of the participants tended to ignore the existence of predictive words because it was not salient enough. And prior studies have not been able to achieve a reasonable typing speed.

Word-level eye typing

Due to the limitation of eye-tracking accuracy, adjacent letters are easily activated by mistake. Furthermore, as the number of predicted words increases, the probability of hitting increases, but so does the time and cognitive effort of visual processing. Koester and Levine (1997, 1998) found that the visual search took 150 milliseconds more for each additional predicted word. The time to scan the predicted word list also needs to be taken into account. On the other hand, the number of candidate words is limited by the size of the display (Garay-Vitoria & Abascal, 2006; Garay-Vitoria & González-Abascal, 1997).

Thus, in addition, to provide word predictions, research utilizing language models to improve eye typing performance has focused on the dwell-free eye typing methods (Kristensson & Vertanen, 2012). In such a word-level typing system, users don't have to select letter by letter. The current typing word is predicted according to the gaze path that the user glances at the characters comprising the word. Since the eye is always on, the swipe-based method requires the user to specify the beginning and the ending of a word. Kurauchi et al. (2016) used gestures to distinguish them. Some studies tried to distinguish the beginning and the ending by the bi-modal method, which combined touch and gaze input (Kumar, Hedeshy, MacKenzie, & Staab, 2020).

Some studies have abandoned the keyboard interface, the most famous of which is Dasher (Ward et al., 2000; Ward & MacKay, 2002). It is an efficient dynamic typing interface. According to the prediction of language model, the next possible letters move from the right to the left of the screen. In this method, participants can reach a high typing speed of 17.3 wpm after more than two hours of training (Tuisku, Majaranta, Isokoski, & Räihä, 2008). However, since the typing system relies heavily on the language model, which featured a word-level input. The user could not enter the words which are not contained in the corpus.

Adjusting the detectability based language model

Since eyes need to switch between current typing characters and word prediction during eye typing, there has been a lot of studies describing the role of the position for word predictions. Diaz-Tula and Morimoto (2016) proposed *Augkey*, the related typing suggestions were placed closer with the current selecting item to reduce the eye movements. The results have shown that both typing speed and workload were improved with *Augkey*. Conversely, Sengupta, Menges, Kumar, and Staab (2019) reported that they did not find a significant difference regarding the position of word suggestions.

There was also research focusing on adjusting the detectability based on the language model to achieve a reasonable text entry speed. In such studies, the dwell time can be adjusted according to what has been entered (Mott et al., 2017; Pi et al., 2020).

3.3 Limitations of previous work

As mentioned in Chapter 1, calibration-free gaze interface has been a research topic of great interest in gaze interaction fields. Since smooth pursuit eye movements can be used without calibration in spontaneous gaze interaction, the intuitiveness of the gaze interface design has been a topic of great interest in the human-computer interaction field. However, since most related research focuses on curved smooth-pursuit trajectories (Esteves et al., 2015; Khamis et al., 2018), the design issues of using linear trajectories in gaze interaction are poorly understood. Hence, this dissertation will first explore the user performance of gaze interfaces based on linear smooth pursuit eye movements.

Although research on eye typing has been going on for many years, it still lacks a a user-friendly eye typing interface with high text entry speed. For eye typing interface based on QWERTY, the learning effort is low, and the text entry rate is high. However, personal calibration is necessary before use to ensure acceptable tracking accuracy. Additionally, the accuracy of the gaze cursor is far from the mouse cursor due to the miniature eye movements. Or if the accuracy of the eye tracker is reduced during use, both will cause unintentional selections. To overcome the shortcomings, the size of each item of the eye typing interface need to be designed quite large (Penkar, Lutteroth, & Weber, 2012). For the alternative eye typing interface, some studies designed larger elements to overcome the problem of accuracy (D. W. Hansen et al., 2008; J. P. Hansen et al., 2003a). Recently, a considerable literature has grown up around the theme of designing calibration-free eye typing interfaces for public displays, that aims to shorten or eliminate the personal calibration. However, so far, the text entry rates of calibration-free eye typing interfaces are still too slow. The *SMOOVS* reported that the typing speed ranged from 2.9 to 3.34 WPM, where the text entry speed was improved to 3.41 WPM with a similar layout based on circle pursuit eye movements (Abdrabou et al., 2019). Bafna et al. (2021) tested a new eye typing interface based on pursuit movement only using iPad's front camera, and achieved an average typing speed of 1.27 WPM.

Therefore, this dissertation will seek to design an eye typing interface that is more in line with natural eye movements with high typing efficiency to make up for those shortcomings.

3.4 Summary

This chapter presents an overview regarding the main type of gaze interaction and the design of feedback for gaze interface. Besides, the author summarizes the eye typing literature from two aspects: eye typing interface and utilizing language model to improve the typing performance. Lastly, the limitations of previous work are pointed out.

Chapter 4

Study 1: Gaze interfaces based on linear smooth pursuit

Zeng, Zhe, Felix Wilhelm Siebert, Antje Christine Venjakob, and Roetting, Matthias. Reprinted from, Calibration-free gaze interfaces based on linear smooth pursuit, Journal of Eye Movement Research, 13(1). https://doi.org/10.16910/jemr.13.1.3.

A substantial portion of this chapter is based on the above paper.

4.1 Introduction

This study investigates the effect of number of objects and moving speed on user performance in gaze interfaces where objects move linearly. The main goal of this chapter is to develop a deeper understanding of the effect of number of objects and object moving speed on the eye behavior and detection rates of gaze interfaces. This experiment was conducted with the following hypotheses:

H1: The absolute angle difference between the target movement and gaze trajectory, i. e., orientation error, will increase with an increased number of objects and conversely decrease with a faster moving speed.

H2: The detection rates for objects will be different regarding the number of objects and moving speed.

H3: Users prefer the gaze interface with fewer moving objects and a faster object moving speed.

4.2 Empirical evaluation

This study was conducted in the Eye Tracking Laboratory of the Chair of Human-Machine Systems at the Technische Universität Berlin.

4.2.1 Experimental stimuli

Five interfaces were developed (see Figure 4.1), which were implemented in Python with the Tkinter GUI package.

The interfaces consist of multiple digits arranged in a circle and vary in the number of digits presented. The ordering of the numbers was constant throughout the experiment (increasing clockwise). The digits move outward in a linear trajectory with constant speed. They are systematically placed, in varying degrees in relation to the center point of the display (see Table 4.1 and Figure 4.1).

Table 4.1: The angular range for one object

Number of objects	6	8	10	12	15
Angular division	60°	45°	36°	30°	24°

The diameter of the circles around the digits is 44 pixels (1.13°). The distance from the center of the display to each circle is 150 pixels (3.87°). The interaction with the interface consists of two steps. First, users need to visually perceive the information and search the digit that they want to select. In the second step, users need to follow the given digit with their eyes, while all digits move outward.

Since users' capacity for visual processing is limited, only a certain number of items can be processed at the same time. Research suggests that humans can visually process 20-50 items per second (Wolfe & Horowitz, 2017). Thus, the digits in the tested interface start to move after 800 ms, ensuring that users have appropriate time to search for a target digit. The gaze points were recorded after the objects start to move. Based on similar research by Vidal et al. (2013) and conventional dwell-time based gaze interactions (J. P. Hansen et al., 2003a), the duration of the outward movement of digits is set to 500 ms. The recording was ended when the objects stop to move.

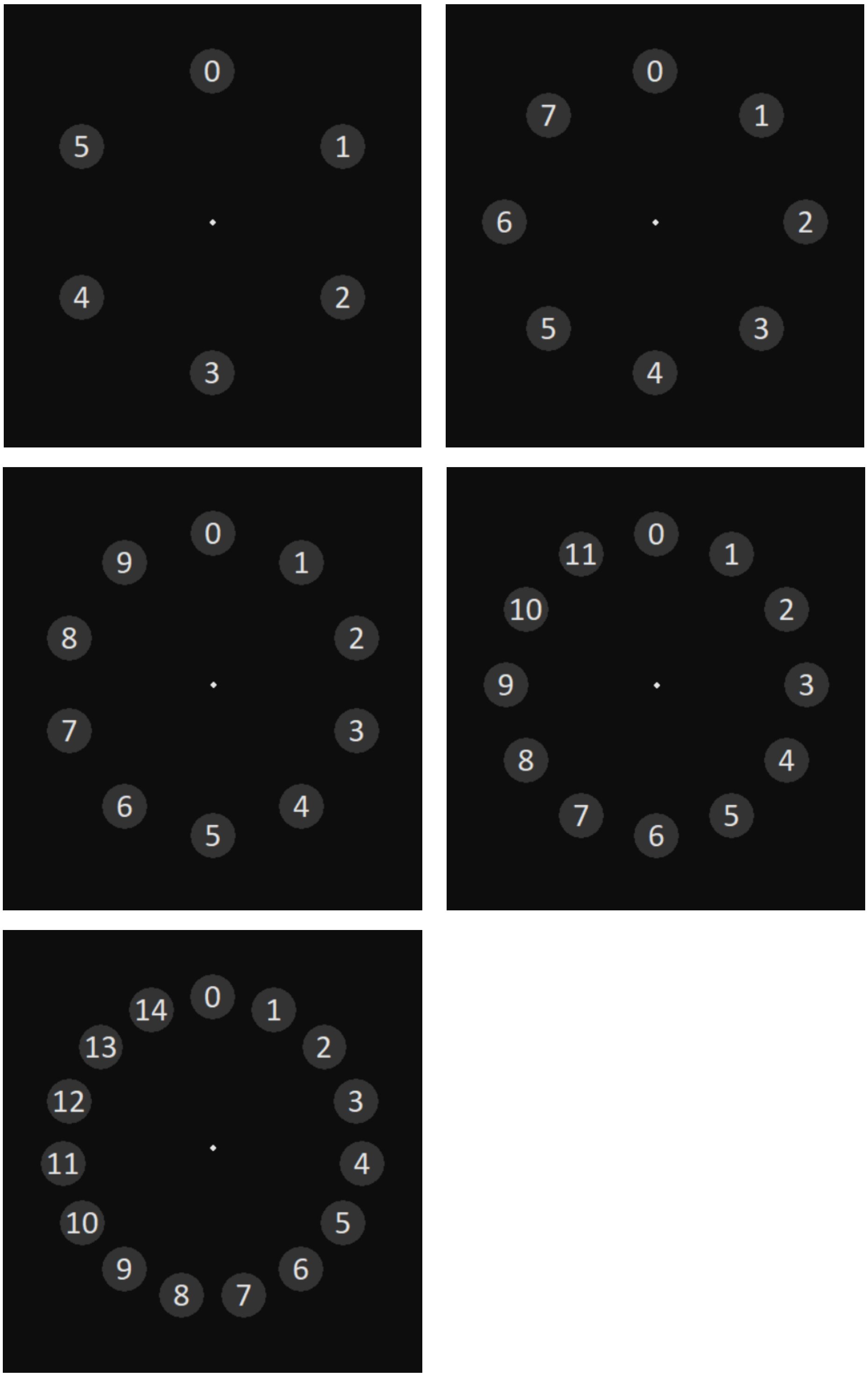


Figure 4.1: Interface layouts before outward movement of digits. The interfaces contain 6, 8, 10, 12, and 15 moving objects, respectively.

4.2.2 Experimental design

This experiment featured a 5×2 within-subjects design. The first factor (number of objects) was varied five folds, i. e., 6, 8, 10, 12, or 15 objects in the interface. The factor (moving speed) was varied two-fold, as objects moved either with 7.73 °/s (300 pixel/s) or with 12.89 °/s (500 pixel/s). All experimental conditions were repeated 12 times - that is to say, each participant performed 120 trials in total.

As dependent variables, two categories of variables were collected, performance measures and subjective experience. For users' performance, orientation error and detection rates were registered. Orientation error describes the absolute angle difference between the target movement trajectory and the regressed line calculated from eye movement data. The correct detection rate is the ratio between the trials of correct detection and total trials for each participant in each condition. The false detection rate refers to the percentage of wrongly detected trials for each participant in each condition, e. g., when the eye trajectory was matched to target "1", although participants were asked to follow target "2". For subjective experience, a semi-structured interview was conducted after the experiment. Participants were asked about their preferences for the interfaces regarding the number of objects and object moving speed. They were also asked about possible reasons for their preference.

4.2.3 Participants

In this study, 25 participants (11 female and 14 male) were recruited. Their age ranged from 21 to 46 years old, with a mean of 29.56 years. 17 participants had normal vision while eight used vision aids during the study (four wore contact lenses and four wore glasses). Six of the participants had previous experience with gaze interaction. Two participants were left-handed, and 23 participants were right-handed. Participants were rewarded with five Euro per visit or alternatively a certification of student experimental hours for attendance.

4.2.4 Apparatus

A Tobii EyeX screen-based remote eye tracker with a sampling rate of 60 Hz was used to record participants' eye-movement data. The eye tracker was mounted beneath a 24-inch Dell monitor with a resolution of 1920 * 1200 pixels (see Figure 4.2). All data were collected without any prior calibration phase for individual users. The eye tracker was calibrated by a third person and this setting was used for all the participants. Across all participants, the average gaze estimation error of the eye data was 4° visual angle (SD = 1.66). The average distance between the participants' eyes to the display was 60 cm (SD = 2.67). A chin rest was attached to the edge of the table, corresponding to the horizontal center of the eye tracker and the monitor. The chin rest was used to prevent participants from leaning too close to the display and maintain a constant viewing distance throughout the experiment.

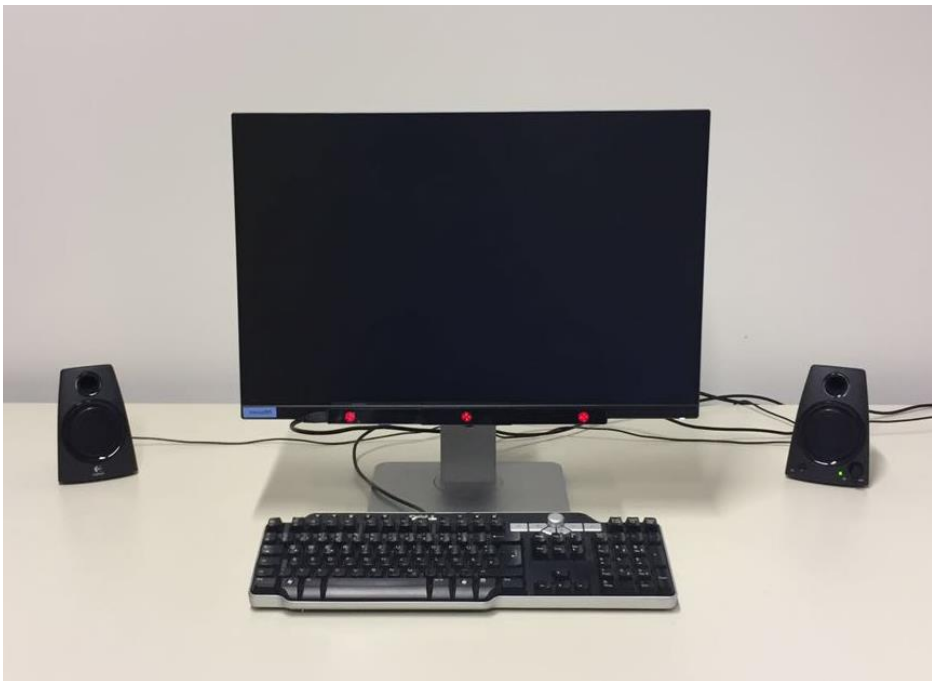


Figure 4.2: Experiment settings

4.2.5 Procedure

The experiment lasted approximately 30 minutes. After being welcomed, participants were provided with written information about the experiment and asked to read it carefully. Then, the experimenter explained the eye tracker to participants. Any questions about the experiment were clarified before the participants signed the Informed Consent Form. Participants were asked to complete questionnaires including demographic information, experience about eye tracking, and gaze interaction. Afterward, participants were instructed to adjust the chin rest to a comfortable height.

The experiment consisted of one training session and two subsequent test sessions. In each session, a target number was displayed in the center of the screen before the start of individual trials. The task for participants was to find the given number in the digits circle and to follow the outward-moving target number with their eyes. The moving objects were displayed after the target number was shown 3 seconds. In the training session, participants could try out the experimental tasks without data being recorded and familiarize themselves with the interface. Once they fully understood the task, the test sessions were started. In order to balance practice and fatigue effect, the sequence of object moving speed and number of objects was fully randomized. For each experimental condition, i. e., for a given number of objects and a given speed, 12 digits had to be selected. The sequence of the 12 digits was randomized between participants to prevent the effects of sequence. For the interface consisting of 15 digits, not all digits presented in the interface had to be selected in the task. To minimize the potential effects of fatigue, participants took a short break between the two test sessions. A semi-structured interview was conducted after the experiment.

4.3 Classification algorithm

A large body of work investigated the detection algorithm used for pursuit-based gaze interface, they mainly include Pearson correlation, Euclidean distance and calculating the angle between two vectors (summarized in Section 3.1.3).

Since the Euclidean distance is more appropriate for detecting slower moving objects, usually slower than 5-6 °/s (Drewes et al., 2018), the testing object moving speeds are relative fast (7.73-12.89 °/s) in this study, this method was not included in the evaluation. In this study, orthogonal distance regression is introduced to analyze orientation error and detection rates regarding number of objects and moving speed. Besides, the method based on orthogonal distance regression is compared with other two other frequently used methods (Pearson correlation, and angle between two vectors) regarding overall correct detection rate.

4.3.1 Detection based on orthogonal distance regression

A regression model was introduced to detect the linear movements to evaluate the gaze trajectory based on smooth pursuit eye movements. This algorithm is based on Orthogonal Distance Regression (ODR). This study utilized ODR to estimate a linear regression model from gaze trajectory. The orthogonal distance r is defined as the distance from the point to a linear model. Using ODR, both errors of X and Y gaze coordinates were taken into account. The function for the calculation of orthogonal distance r with gaze data is:

$$\begin{aligned} r_i^2 =& min\left\{\epsilon_i^2 + \delta_i^2\right\} \\ &\text{subject to: } Y_{eye,i} = f\left(X_{eye,i} + \delta_i; \beta_i\right) - \epsilon_i \end{aligned} \tag{4.1}$$

where $X_{eye} = [x_1, ..., x_n]$ and $Y_{eye} = [y_1, ..., y_n]$ are the horizontal and vertical coordinates of gaze data. The random error δ_i and ϵ_i are corresponding to $X_{eye,i}$ and $Y_{eye,i}$, respectively. And f refers to a function of $X_{eye,i}$ with parameters set β_i.

A linear model based on the gaze data was derived using ODR. The mean μ and standard deviation σ of the orthogonal distances were calculated. The data points are removed when r_i is three standard deviations away from the mean. The program iteratively estimate a linear model until no further data points are removed. The angle θ with regard to the linear model is converted from the following function:

$$\theta = \arctan 2\left(y_{end} - y_{start}, x_{end} - x_{start}\right) \tag{4.2}$$

where x_{end} and x_{start} refer to the horizontal coordinates of end and start gaze points. And y_{end} and y_{start} are the values of the function 4.1 with x_{end} and x_{start} as input of the function, respectively.

An angular criterion was used for target detection. A visualization of the criterion is presented in Figure 4.3. The black lines show the trajectories of moving digits. The colored lines are trajectory examples of pursuit eye movements from one participant. The gray corridors show the angle ranges that are assigned to individual digits. Gaze trajectories with angles located between two gray angle areas will be recognized as not detected. If the detected gaze angle is within a certain range, it will be detected as the corresponding object. A small angle range ($\alpha°$) is defined as a buffer in the middle between two object classification angle areas. If the detected gaze angle is located in this interval, the system will recognize it as not detected, i. e., miss detection.

4.3.2 Detection based on angle between two vectors

Similarly to detection method based on orthogonal distance regression, this vector-based detection refers to the angle between the positive x-axis and the ray to gaze point (x_{eye}, y_{eye})

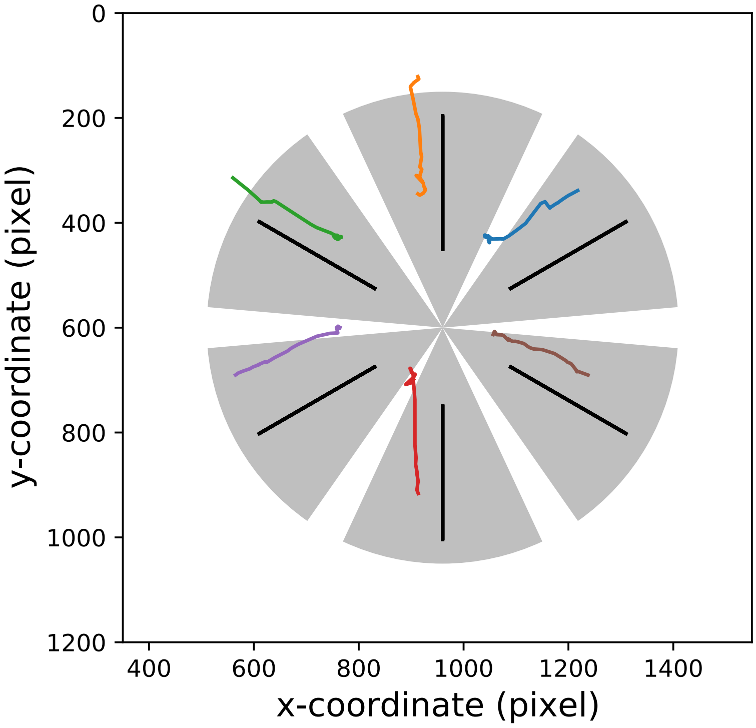


Figure 4.3: Visualization of the gaze and object trajectories

and middle point (x_{mid}, y_{mid}). In this method, y_{mid} and x_{mid} are the average values of x, y gaze coordinates when eye looks the middle of the screen. The *mode* of those angles $[\theta_1, ..., \theta_n]$ is calculated as the selected angle. The angle is calculated by the function:

$$\begin{aligned} &\theta_i = \arctan 2\left(Y_{eye,i} - y_{mid}, X_{eye,i} - x_{mid}\right), \qquad i = 1, ..., n \\ &\theta_{mode} = [\theta_1, ..., \theta_n] \end{aligned} \tag{4.3}$$

4.3.3 Detection based on pearson correlation

Pearson correlation is the most frequently used method to detect gaze input based on smooth pursuit eye movements. To select an object, users need to follow a moving object in the interface with their eyes, and the resulting gaze trajectory is then compared against the trajectories of moving objects present on the display. Since the detection of trajectories is invariant against its origin location, an individual calibration phase is not required (Vidal et al., 2013). The formula for the Pearson correlation is:

$$r_x = \frac{\sum_{i=1}^{n}(X_{eye,i} - \overline{X}_{eye,i})(X_{obj,i} - \overline{X}_{obj,i})}{\sqrt{\sum_{i=1}^{n}(X_{eye,i} - \overline{X}_{eye,i})^2}\sqrt{\sum_{i=1}^{n}(X_{obj,i} - \overline{X}_{obj,i})^2}} \tag{4.4}$$

where $X_{eye,i} = [x_{eye,1}, ..., x_{eye,n}]$ and $X_{obj,i} = [x_{obj,1}, ..., x_{obj,n}]$ are the vertical coordinates of gaze data and one object, separately. The coefficients r_x, r_y are calculated separately for horizontal and vertical positions for all objects, i. e., for the interface containing six moving objects, six pairs of $[r_x, r_y]$ are calculated. The closer the coefficient is to 1, the stronger the correlation between the two time series, and therefore the more similar the trajectory of the eye is to the trajectory of the object. The object with highest r_x, r_y and both r_x, r_y are above a threshold is detected as following target.

The correlations between gaze and objects coordinates were calculated and the division-by-zero error occurs when objects move in horizontal or vertical direction (Drewes et al., 2018). Thus, to both x- and y-coordinates, small random errors were added (M = 0, SD = 0.00001).

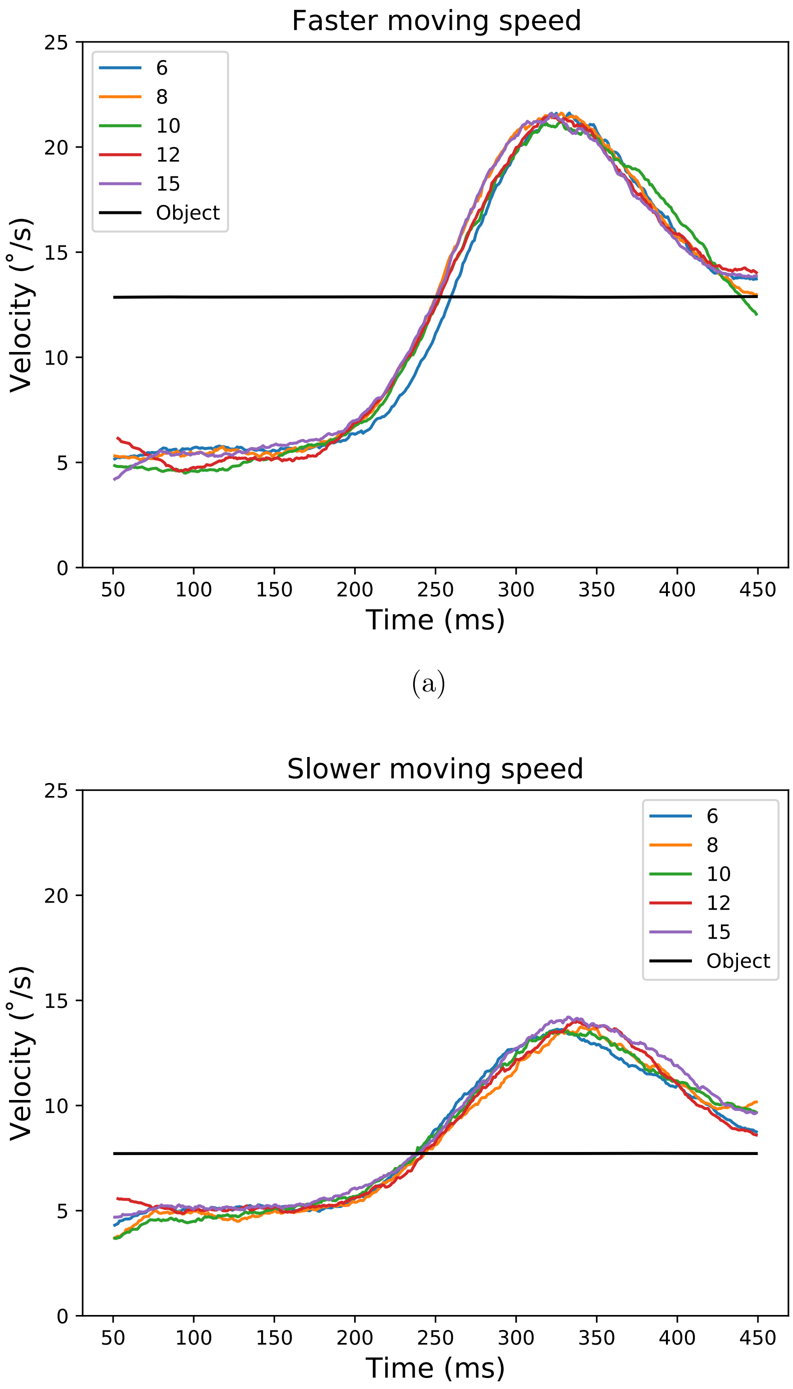
4.4 Results

The gaze data collected during the experiment was analyzed offline. In all, there are 3000 trials (2 moving speeds * 5 number of objects * 12 repeated times * 25 participants). The figures visualized the moving trajectories from one participant for each condition are shown in Appendix A.2. The orientation error, correct detection rate and false detection rate were evaluated with repeated measures ANOVA at a significance level of $\alpha = 0.05$. The Mauchly's test of sphericity was non-significant, so that no correction was needed. A set of Bonferroni corrected t-tests was conducted for pairwise multiple comparisons.

4.4.1 Velocity

The moving velocity of eye and objects is visualized in Figure 4.4. Approximately 150 ms latency was recorded for the onset of pursuit eye movements under both faster and slower moving speed conditions, which contained a certain amount of systematic recording error.

However, the difference in velocity among the number of objects is quite small. Earlier research has shown that the evocation of smooth pursuit eye movements has a latency range from 80 to 130 ms in relation to the start of object movement (Lisberger, 2015; Robinson, 1965). There is a relatively long pursuit latency, and the eye begins to move later than the moving object. Once the eye starts to move, there are several saccadic



(a)

(b)

Figure 4.4: The velocity of eye and moving objects, the black line shows the moving velocity of the object, the colored lines show the moving velocity of eye regarding different number of objects in the interface.

movements, allowing the eye to catch up with the moving object (de Brouwer, Yuksel, Blohm, Missal, & Lefèvre, 2002; Rashbass, 1961). In this study, the classification algorithm took this pursuit latency into consideration. Hence, the first 100 ms gaze data of each trial were discarded, and only the last 400 ms gaze data were analyzed.

4.4.2 Orientation error

Orientation error refers to the absolute angular difference between the target and eye movement trajectory. The orientation error was calculated based on orthogonal distance regression (ODR), the analysis focuses on the performance of smooth pursuit eye movements under a low spatial accuracy. Figure 4.5 shows the distribution of orientation errors for all participants and all trials of the experiment.

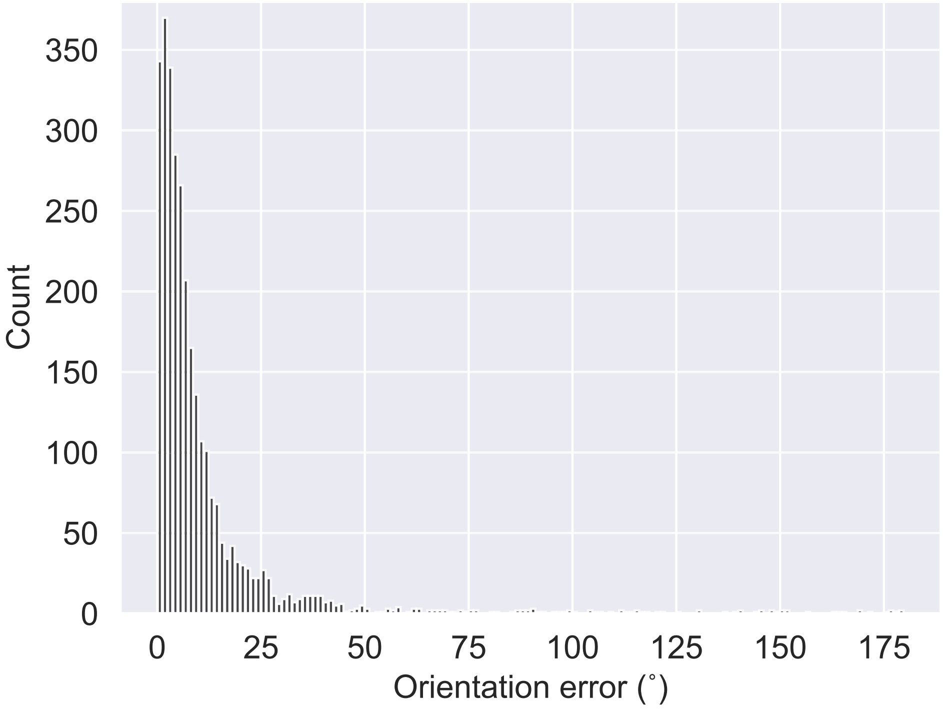


Figure 4.5: The distribution of raw orientation error throughout the experiment

In order to gain further understanding of the orientation error, the outlier data which are three standard deviations away from the mean was excluded. The remaining 2926 trials were used for the analysis for orientation error. The descriptive statistics for all experimental conditions are visualized as boxplots in Figure 4.6.

Descriptively, the mean orientation error decreased from 6 objects to 12 objects, while increasing again for 15 objects. Nevertheless, it can be observed that the mean orientation

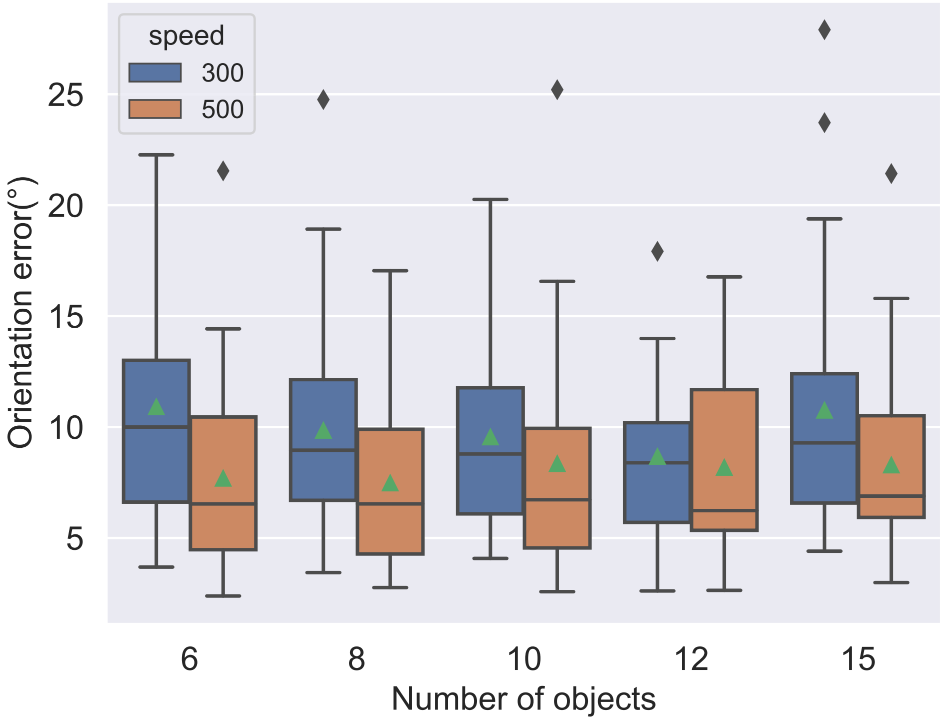


Figure 4.6: Orientation error in the experimental groups for filtered data. The green triangles represent averages.

error was higher in experimental conditions with slower moving speed than in conditions with faster moving speed. This descriptive difference can be observed in overall conditions, irrespective of the number of objects presented in the interface.

The ANOVA confirms this effect, the results show that there was no significant main effect of the number of objects on participants' orientation error ($F(4, 96) = 1.27, p = .29, \eta_p^2 = .05$). However, participants' mean orientation error for slower moving speed was significantly greater than faster moving speed ($F(1, 24) = 30.99, p < .001, \eta_p^2 = .56$). There was no significant interaction between number of objects and moving speed for correct detection rate ($p > .05$).

The mean orientation error for the different objects (i. e., different directions) in each interface is presented in Figure 4.7. The orientation error for faster moving speed was generally smaller than the orientation error for slower moving speed. This effect is most pronounced in interfaces with small numbers of moving objects, but becomes less pronounced in interfaces with a large number of moving objects. In addition, the orientation error of some diagonal directions was found larger than cardinal directions for interfaces with 8, 12 and 15 moving objects.

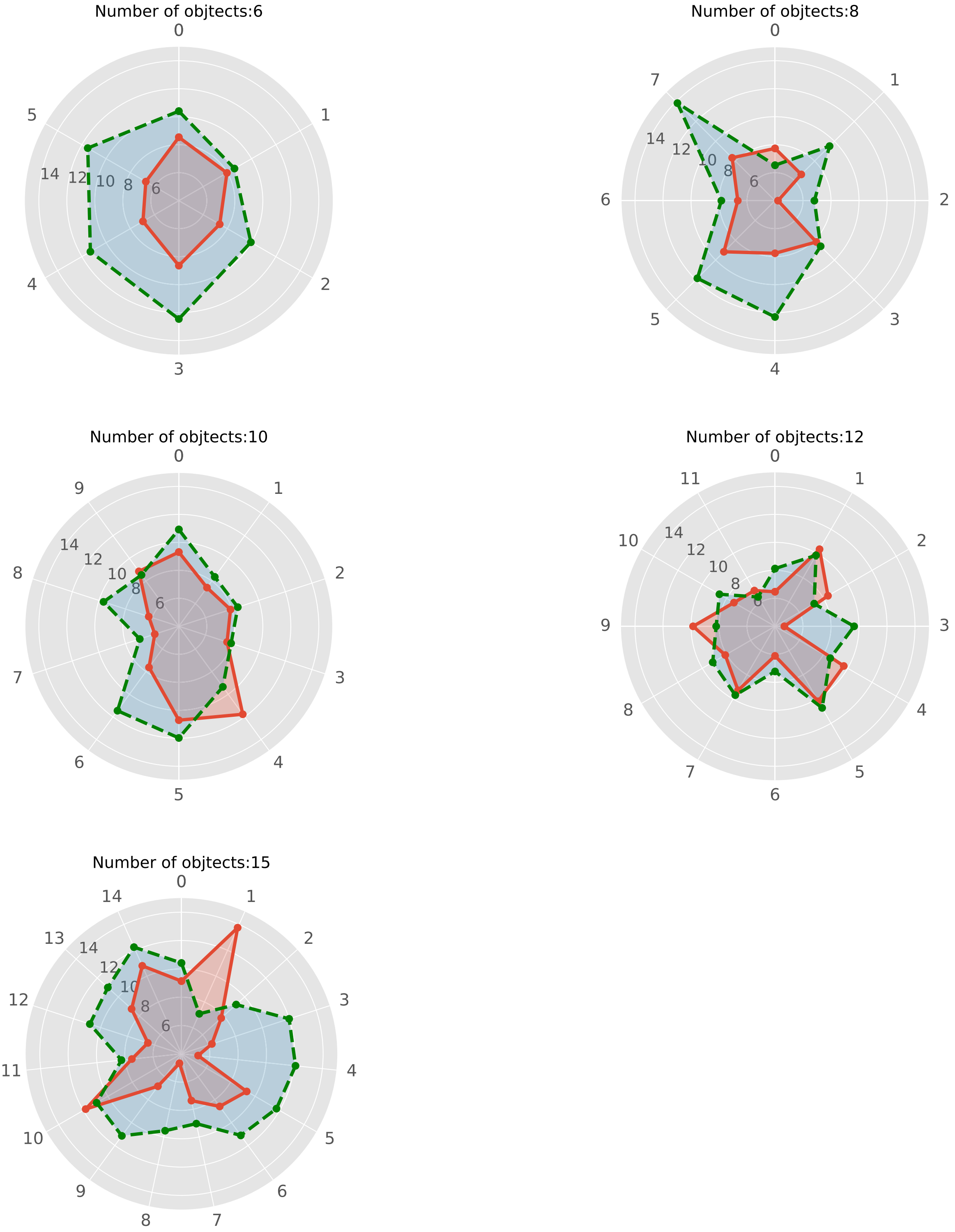


Figure 4.7: Average orientation error for all interfaces. Green dashed line refers to orientation error of slower moving speed, and red line refers to faster moving speed.

4.4.3 Detection rates

This section reports the evaluation whether the target was detected as being followed, i. e., correct detection, detected as another object, i. e., false detection, or no object was detected, i. e., missed detection. The detection rates were analyzed using the raw data set with the consideration of all human errors, i. e., no data was excluded from the analysis.

There are parameters that affect the target detection in pursuit-based interfaces, such as moving speed, number of the moving objects, threshold value of the detection method, and window size for detection. The window size was evaluated in previous study (Vidal et al., 2013), and 500 ms is considered as a proper size. Besides, due to the latency of smooth pursuit eye movements, varied from 80 to 150 ms (Lisberger et al., 1987; Rashbass, 1961; Westheimer & McKee, 1978), the window size is set to all methods 400 ms in this study.

In the detection method based on orthogonal distance regression, a linear model was estimated from gaze data, gaze angle is defined as the angle between the estimated line and the x-axis that is calculated using *arctan2*. When the detected gaze angle is within a certain range, it will be detected as the corresponding object. There is a buffer angle range ($\alpha°$) in the middle between two object classification angle areas to prevent the false selection of boundary objects. If the detected gaze angle is located in this interval, no object will be detected. Five buffer angles were tested in this study, namely 3°, 5°, 7°, 9° and 11°. Figure 4.8 shows that the correct and false detection rates decrease with increasing buffer angle range, whereas the miss detection rate increases with increasing buffer angle range. Since there are up to 15 objects in the gaze interface, a relatively small buffer range ($\alpha = 5°$) was selected for further analysis. The size of detectable range for different interfaces is showed in Table 4.2.

Table 4.2: Detectable range for different interfaces

Number of objects	6	8	10	12	15
Angular division	60°	45°	36°	30°	24°
Size of detectable range	55°	40°	31°	25°	19°

Correct detection rate

The grand mean (M) for the correct detection rate was 0.82. In Figure 4.9, the descriptive statistics of the correct detection rate are presented for each condition.

The mean of correct detection rates generally decreased with an increasing number of

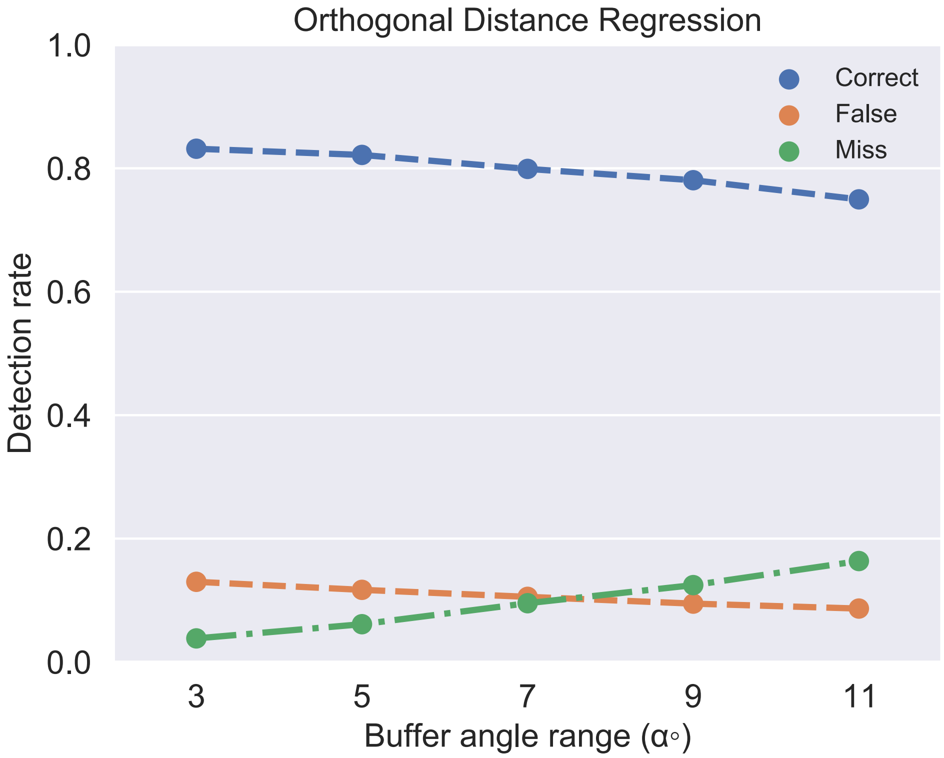


Figure 4.8: The ODR detection rates regarding different buffer angle range

objects. The conditions with faster moving speed had a greater mean for correct detection rate than the conditions with slower moving speed.

The ANOVA shows that there was a significant main effect of the number of objects on participants' correct detection rate $(F(4, 96) = 27.62, p < .001, \eta_p^2 = .54)$. Moreover, participants' mean correct detection rate for faster moving speed was significantly higher than the mean correct detection rate for the slower moving speed $(F(1, 24) = 13.93, p < .01, \eta_p^2 = .37)$. There was no significant interaction between number of objects and moving speed for correct detection rate $(p > .05)$.

As the results showed a significant effect of the number of objects upon correct detection rate, a set of post hoc t-tests was conducted to determine differences between levels. The correct detection rate for 15 objects $(M = 0.69, SD = 0.03)$ was significantly smaller than those for 6 objects $(M = 0.91, SD = 0.02, p < .001)$, 8 objects $(M = 0.89, SD = 0.02, p < .001)$, 10 objects $(M = 0.82, SD = 0.03, p < .001)$, 12 objects $(M = 0.80, SD = 0.03, p < .01)$. The correct detection rate for 6 objects was significantly higher than those from 10 objects $(p < .05)$ and 12 objects $(p < .001)$. Additionally, there were significant differences between 8 and 12 objects $(p < 0.01)$.

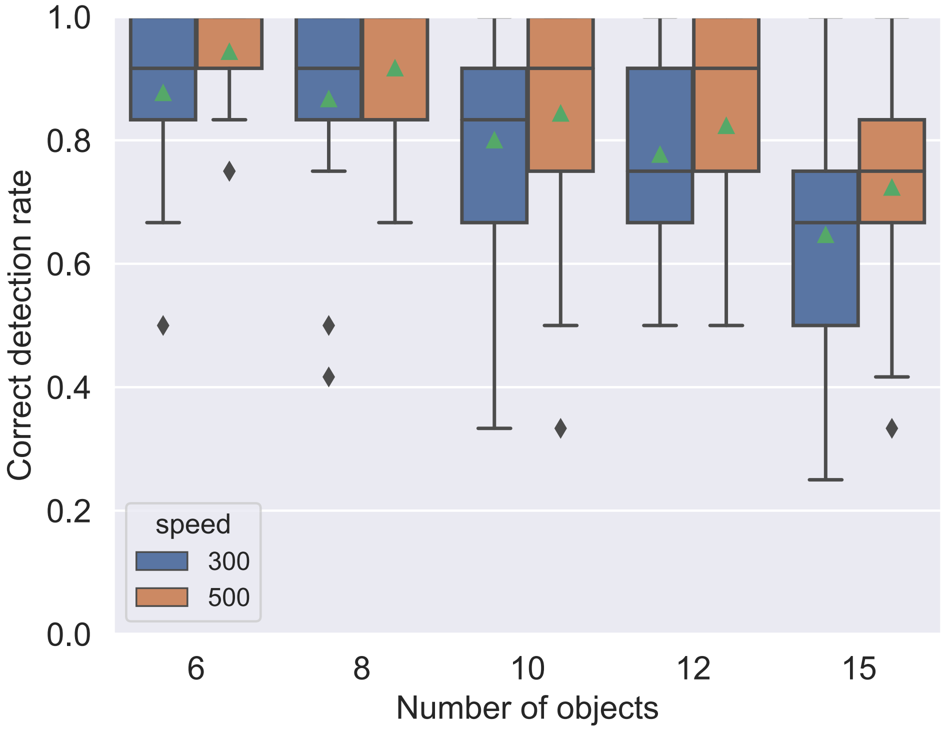


Figure 4.9: Correct detection rate. The green triangles represent averages.

False detection rate

A false detection is registered when a gaze trajectory is detected as an object that is not the target object. The false detection rate describes the ratio of all false detections of all presented trials.

The grand mean (M) for false detection rate was 0.12. The descriptive statistics of false detection rates for all experimental conditions are presented in Figure 4.10. The mean of false detection rates increased with an increasing number of objects. The conditions with slower moving speed had a greater mean for false detection rate than the conditions with faster moving speed.

To gain further understanding of false detected trials, Table 4.3 compares the number of trials which were falsely detected as adjacent digits (i. e., eye followed “2”, but the gaze trajectory was detected as adjacent digits “1” or “3”.) and the number of all trials which were falsely detected.

The ANOVA proved that there was a significant main effect of the number of objects on participants’ false detection rate $(F(4, 96) = 8.85, p < .001, \eta_p^2 = .27)$. Meanwhile, the object moving speed significantly affects the false detection rate, $(F(1, 24) = 6.99, p < .05, \eta_p^2 = .23)$. There was no significant interaction between the number of objects and

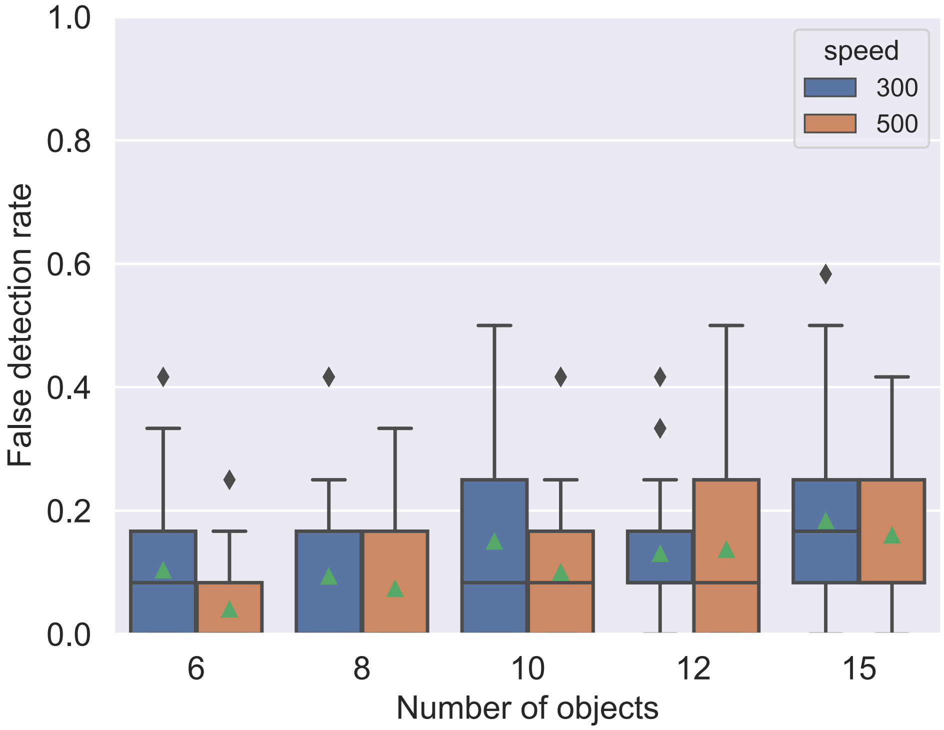


Figure 4.10: False detection rate. The green triangles represent averages.

moving speed for false detection rate ($p > .05$). The results of statistical significance are summarized in Table 4.4.

Pairwise t-tests show that the participants had a significantly higher false detection rate with 15 objects ($M = 0.17, SD = 0.03$) than with 6 objects ($M = 0.07, SD = 0.02, p < .01$), and 8 objects ($M = 0.08, SD = 0.02, p < .01$). Moreover, there were significant differences between 6 and 12 objects ($M = 0.13, SD = 0.02, p < .01$).

4.4.4 Subjective evaluation

Participants were asked to indicate their preference regarding the number of objects and moving speed. They can choose one or more options. Results of participants' preference are presented in Table 4.5. The interface with 12 moving objects was preferred by 10 participants. Open answers to their reasoning behind this preference revealed that participants felt that this interface is similar to the clock face. Five participants chose the interface with 6 moving objects. The four participants who indicated a preference for an interface with less than 6 objects reported that they would find it easier to identify the target objects and expected that it would be easier to follow the objects on the interface if there were less than 6 objects on it. In addition, the interfaces with 10 and 15 moving

Table 4.3: Count for trials which were detected as adjacent digits and all false detected trials for all experimental conditions.

		Number of objects				
		6	8	10	12	15
Slower-300 pixels	adjacent	23	22	35	29	37
	all	31	28	45	39	55
Faster-500 pixels	adjacent	9	19	24	36	37
	all	12	22	30	41	48

Table 4.4: Study 1 - Statistical significance of main and interaction effects regarding objective measures.

	Main effect of number of objects	Main effect of moving speed	Interaction effects
Orientation error	n.s.[1]	$p < .001$	n.s.
Correct detection rate	$p < .001$	$p < .01$	n.s.
False detection rate	$p < .001$	$p < .05$	n.s.

[1] 'n.s.': not significant

objects were chosen by two participants, respectively.

For the subjective experience of the objects' moving speed, more than half of the participants did not report a preference, since they did not perceive a difference in moving speed. Eight participants preferred slower moving speed while three participants chose faster moving speed as favorite. When asked for reasons behind their preferences, those participants with a preference for slower moving speed reported that it easier to follow slower moving objects and require less effort.

4.4.5 Comparison of three detection methods

In addition to detection method based on orthogonal distance regression, two more detection methods are briefly discussed in this study.

Angle between two vectors

Consistent with method based on orthogonal distance regression, five buffer angles were tested in this angle-based detection method, namely 3°, 5°, 7°, 9° and 11°. Figure 4.11

Table 4.5: Subjective feedback from participants

	User Preference	Count
Speed	Faster moving speed	3
	Slower moving speed	8
	Feel no difference in speed	**14**
Number of objects	Less than 6 objects	4
	6 objects	5
	8 objects	4
	10 objects	2
	12 objects	**10**
	15 objects	2
	More than 15 objects	0

shows that the correct and false detection rates decrease slightly with increasing buffer angle range, whereas the miss detection rate increases slightly with increasing buffer angle range. Same as the detection based on orthogonal distance regression, the buffer range (α = 5°) was selected for further analysis. The size of detectable range for different interfaces is showed in Table 4.2.

Pearson correlation

In the offline data analysis, the detection method was computed with five correlation threshold levels: 0.5, 0.6, 0.7, 0.8, 0.9, i. e., the object with highest r_x, r_y is detected as following target, only when both r_x, r_y of are above the predefined threshold. The Figure 4.12 illustrated the correct, false and miss detection rates. Lower thresholds result in both higher correct and false detection rates, whereas higher thresholds lead to higher missing detection rate. The correlation threshold of 0.8 was chosen in many studies (Drewes et al., 2018; Esteves et al., 2015), which also applies to the linear moving data-set in this study. Thus, the threshold of 0.8 was selected for further analysis.

However, this detection method entails major disadvantages that it is hard to detect when there are the objects in the direction of the axis and diagonal lines simultaneously. For some objects, the condition that both the highest correlation coefficient of $corr_x$ and $corr_y$ for the objects need to above a threshold is difficult to meet. The interface with 8 objects was further analyzed regarding the moving directions (see Table 4.6). The detection rates are higher than 0.8 for objects moving in diagonal directions, whereas the

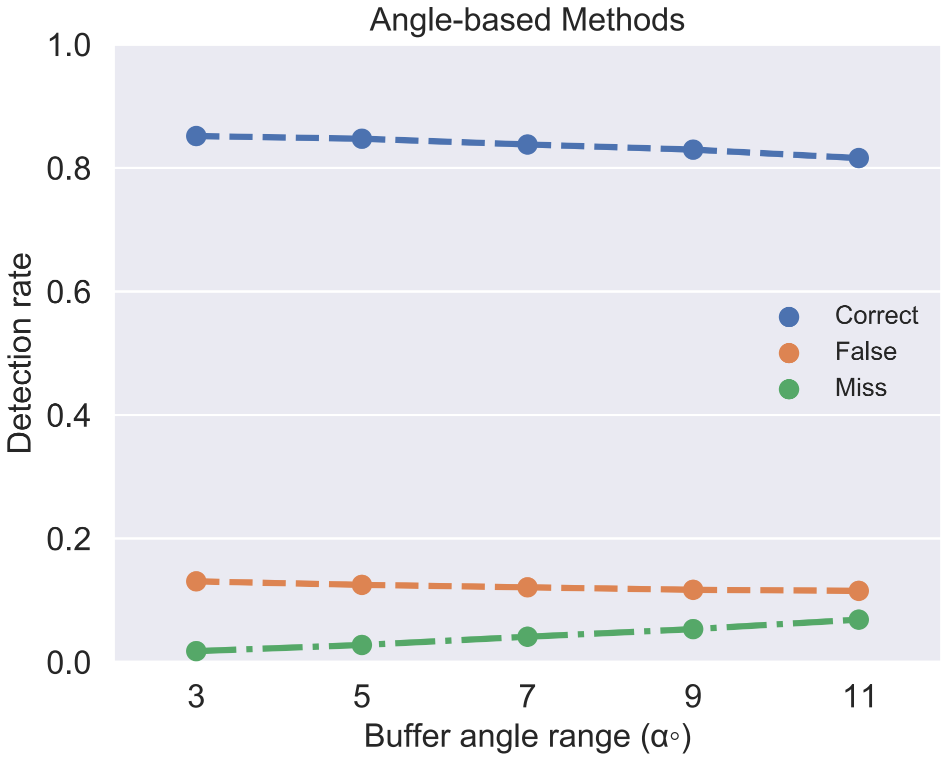


Figure 4.11: The angle-vectors detection rates regarding different buffer angle ranges

Table 4.6: The correct detection rate of interface with 8 objects in directions

Number in interface	Correct detection rate
0	0.0
1	0.87
2	0.0
3	0.82
4	0.0
5	0.88
6	0.0
7	0.82

detection fails to detect the objects moving in cardinal directions.

Comparing the three methods

The correct detection rate was compared regarding the three detection methods, i. e., vector-based, ODR-based and Pearson-correlation-based. From the above comparison of

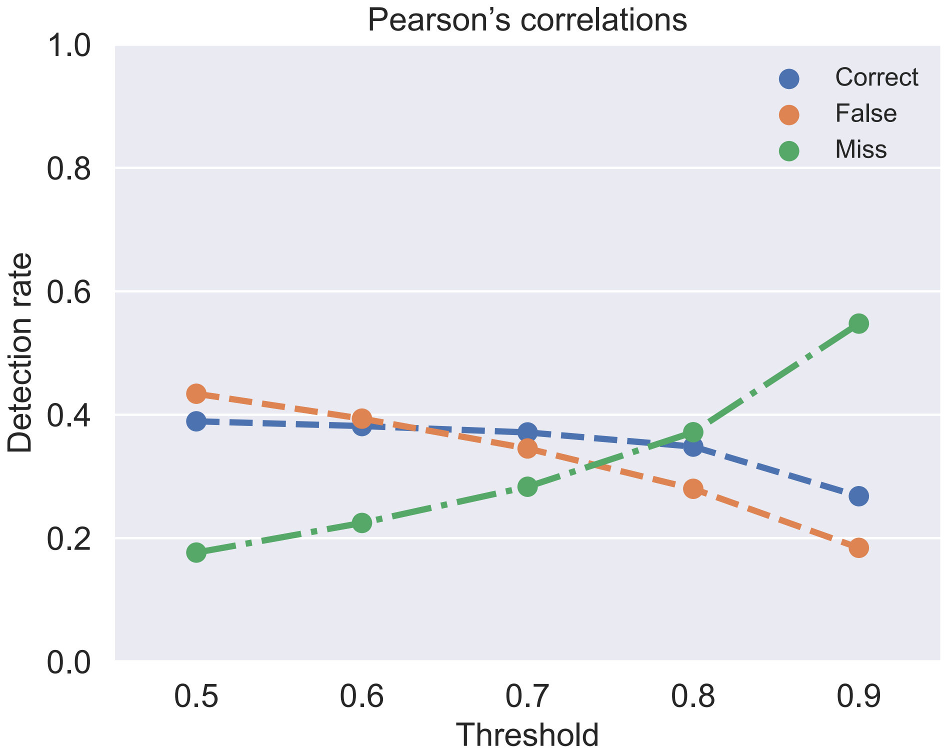


Figure 4.12: The detection rates based on different correlation thresholds

the parameter ranges, the buffer range ($\alpha = 5°$) was selected for vector-based and ODR-based detection. The correlation threshold of 0.8 was selected for Pearson correlation.

The results shows that the correct detection rate continues to decrease as the number of objects increases for all three methods (see Figure 4.13). In particular, the correct detection rate decreases dramatically from 6 to 10 objects for method based on Pearson correlation. The vector-based method achieves the highest correct detection rate. Besides, the correct detection rate of methods based on angle and orthogonal distance regression are much higher than Pearson correlation method. For angle- and ODR-based methods, the interfaces with 6 and 8 moving objects achieve a relative higher detection rate.

4.5 Discussion

This study analyzed how user performance is influenced by the number of objects and object moving speed for gaze interface based linear pursuit movements. Additionally, three detection methods were compared.

The first hypothesis anticipated that the orientation error would increase with an increased number of objects and conversely decrease with moving speed. This hypothesis

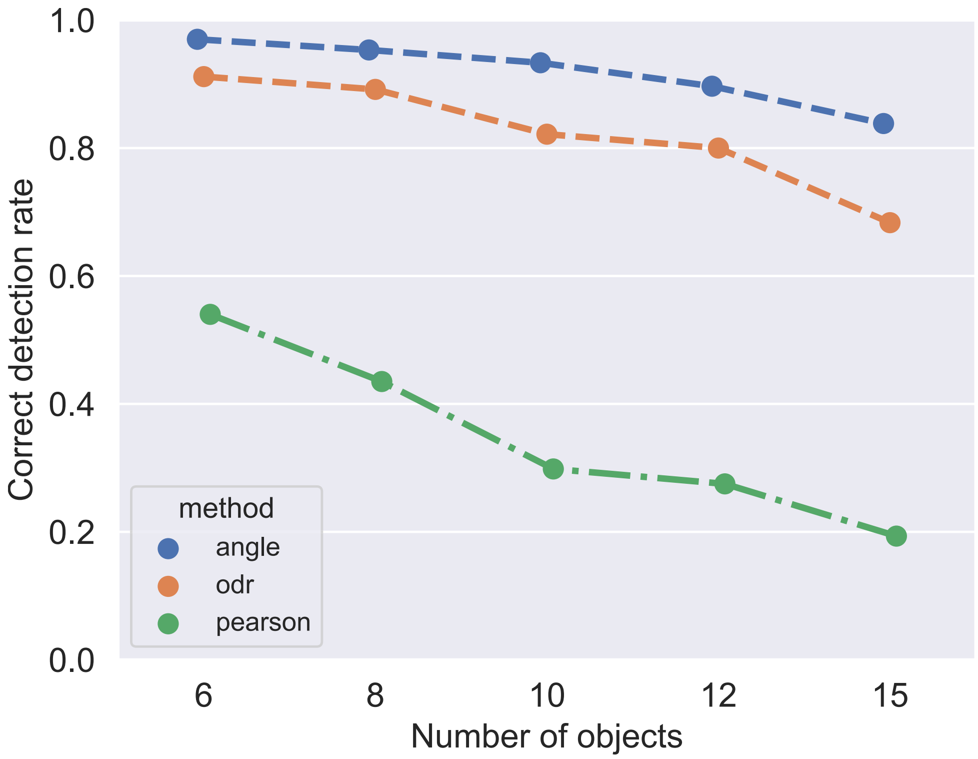


Figure 4.13: The comparison of detection rate among detection methods

was partly confirmed. The results found that the orientation error did not significantly differ between interfaces with varying number of objects. In other words, the pursuit eye movements are not distracted by the increasing number of moving objects and the number of objects has little effect on how well the eye follows the moving target.

At the same time, the orientation error for interfaces with faster moving speed of objects was significantly smaller than for interfaces with slower moving speed. For slower moving objects, the gaze trajectory does not follow the object path as closely as with faster moving objects. A possible explanation for this difference between moving speed conditions might be that with the same recording time and sample size, the moving distance of faster moving condition is longer than the slower one, thus the ODR regression model for trajectories of faster speed performs better than that of the slower ones.

Additionally, most orientation errors are located in an angle range between 0-30°. There is only a small number of orientation errors larger than 30°, which are likely caused by participants' distraction or a participant's inability to locate a target object. Since the gaze data were collected using an eye tracker without individual calibration, the orientation errors occurring within this range are mainly due to the accuracy of measuring equipment.

Although there was no individual calibration for each participant, overall correct detection rates were high. On an individual level, only one participant had a correct detection rate lower than 50%, most likely caused by very thick glasses. For more than two-thirds of false detections, an adjacent digit was detected. The second hypothesis expected that the detection rates for objects will be different regarding number of objects and moving speed. The results found that the correct detection rate decreased significantly while the false detection rate increased significantly with increasing number of objects in the interface. On the question of differences in object moving speeds, this study found that the correct detection rate increased significantly from 300 pixel/s to 500 pixels/s. The false detection rate is significantly lower for faster moving speed than slower one. Furthermore, the ratio of trials detected as adjacent digits to all false detection trials is relatively higher for faster than slower moving speed.

The comparison among levels for the number of objects showed that the decrease in correct detection rate was slow between 6 and 8 objects as well as between 10 and 12 objects. But the decrease was larger between 8 and 10 objects as well as between 12 and 15 objects. The correct detection rate of 15 objects was significantly different compared with interfaces with lower number of objects. No significant difference was found between 6 and 8 numbers for both orientation error and detection rates. These differences in detection rates regarding related to the number of objects in the interface may be caused by trials in which participants were not able to find the position of a target object among other objects presented. However, these differences could also be caused by the decrease in the detectable angle range and the limitation of the spatial accuracy. The detectable range was gradually decreasing with the increasing of objects number. For example, the detectable angle range for each object was 55° for interface with 6 objects, but the range reduced to 19° for interface with 15 objects. Although the target was well followed by the eye, the gaze trajectory could be detected as adjacent objects when the detectable angle was too small and the spatial accuracy was low.

The third hypothesis expected that participants would prefer interfaces with fewer objects and fast-moving speed. Our results were inconsistent with the hypothesis. While some participants preferred interfaces with fewer moving objects, the interface with 12 objects was the most preferred interface. The position of the objects played an important role in the subjective evaluation of the interfaces. The similarity of the 12-object interface to a clock face led a number of participants to report a familiarity between the experimental interface and a clock. Concurrently, this might have helped participants to find the target more easily. Future studies should investigate this influence of interface-familiarity on user preference and interaction performance. While a number of participants did not consciously register the difference in moving speed of objects, some of them preferred slower moving speeds. This is an interesting finding, as it reveals that the subjective experience of users in gaze-based interfaces is not directly linked to a higher performance while using the interface.

While researchers investigated gaze interaction based on smooth pursuits eye movements, they were mainly interested in circular and square-shaped target trajectories (Drewes et al., 2018; Esteves et al., 2015). The detection algorithm used is mainly based on Pearson's product-moment correlation in previous researches. In this study, the linear regression method, vector-based method and Pearson correlation method were compared. The results shows that the detection rates of linear regression method and vector-based method outperform the correlation method regarding detection multiple linear moving objects.

4.6 Summary

In this chapter, a controlled laboratory experiment was conducted to evaluate the effect of objects number and object moving speed on interaction based on linear smooth pursuit eye movements with no individual calibrated eye tracker. When comparing the number of objects, there was only a little difference in orientation error, but the detection rates decreased with an increasing number of objects. The results found the detection of faster moving speed was better than the slower ones. Overall, both the 6 and 8 objects interface with a faster moving speed yielded good user performance. In previous works, six moving directions were frequently used in linear smooth pursuit based interface (Lutz et al., 2015; Zeng & Roetting, 2018). This study shows that the difference between 6 and 8 objects is not significant, both can be well detected by the system. Therefore, it is possible to extend the moving directions of cluster to improve the flexibility of the gaze interface.

Chapter 5

Study 2: Design eye typing interface based on hybrid eye movements

5.1 Introduction

This chapter proposes a new concept to design a hybrid gaze interface, which is featured with a two-stage selection and one-point calibration. To select an object, the user needs first to look at this object for a certain long time to activate it. Then, the user selects this character by following its movement. Figure 5.1 gives a metaphor in daily life and illustrates the interaction process. When people observe a bird on a branch, at that moment, the bird flies off, and people's eyes will naturally follow the trajectory of the flying bird. The eye changes quickly from the state of fixation to catch up saccade, and then to smooth pursuit eye movements. If the bird disappears from the field of view, people will scan the environment by saccadic eye movements to re-position the bird (shown in Figure 5.1a). Similar to the scenario in Figure 5.1a, the user selects a charterer by first looking at the character and then following its movement to select it (shown in Figure 5.1b).

Compared to the previous pursuit-based interfaces, only the objects within the current attention area are moving and interactable, and users can select the target item by following its moving trajectory. The interface can be adapted to low-accuracy equipment environments and also minimizes distractions to users' attention. A novel eye typing interface is designed based on this hybrid concept, to create a robust gaze-enabled text entry system.

Additionally, previous research has mainly focused on interface design and algorithm development, but has neglected to provide feedback for dynamic eye typing interface.

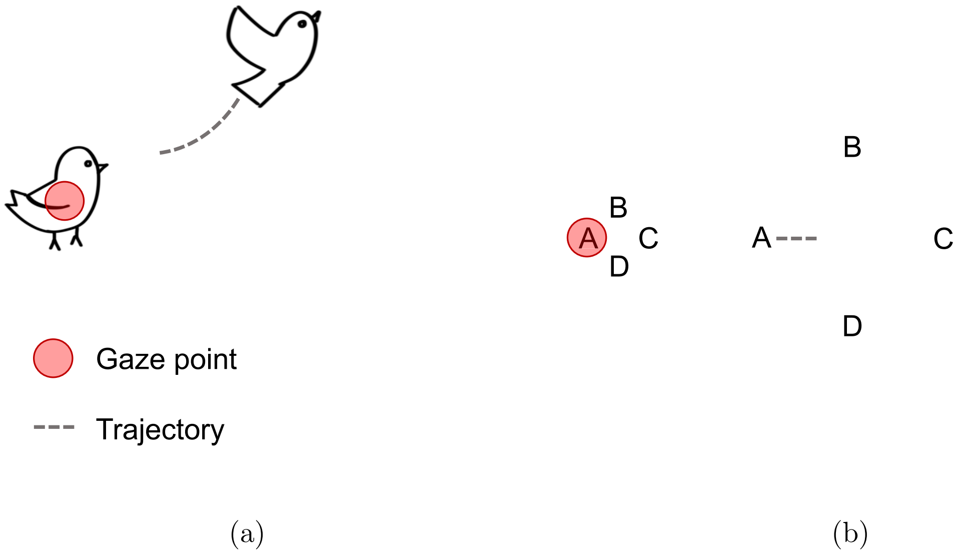


Figure 5.1: The hybrid interaction design. The red circle represents the current gaze position, the gray dashed line represents the moving trajectory of the object.

Thus, this study further investigates the effect of feedback on typing performance and experience of the interface. Different modalities, i. e., visual only, auditory only, combined visual and auditory and no feedback are compared in a user study.

5.2 Interface design

As mentioned before, the visual search for information as well as the selection of intended objects is combined in gaze-based interfaces. In some dynamic gaze interfaces, user can interact with an element by following the movement of this element with the eyes. In *Pursuits* (Vidal et al., 2013), the moving objects are distributed across the screen. The threshold of the time window and correlation coefficient is used to activate a command. That is, if the correlation coefficient exceeds the threshold within the set time window, an element is selected when the eyes gaze at anywhere on the screen. Lutz et al. (2015) further distinguished two areas of the interface. When the gaze point is registered in the deactivated area in the center of the screen, all characters remain static, selection can be triggered only when the gaze is out of the deactivated area. In human-computer interaction, perception happens before reaction in most cases, and interactive content is generally located in area of attention. Thus, in this eye typing interface, the area where the user is currently looking is defined as a dynamic area, which is located in the activated area. Only objects within the attention area move. Other objects stay still to minimize distractions. The iteration process is shown Figure 5.2.

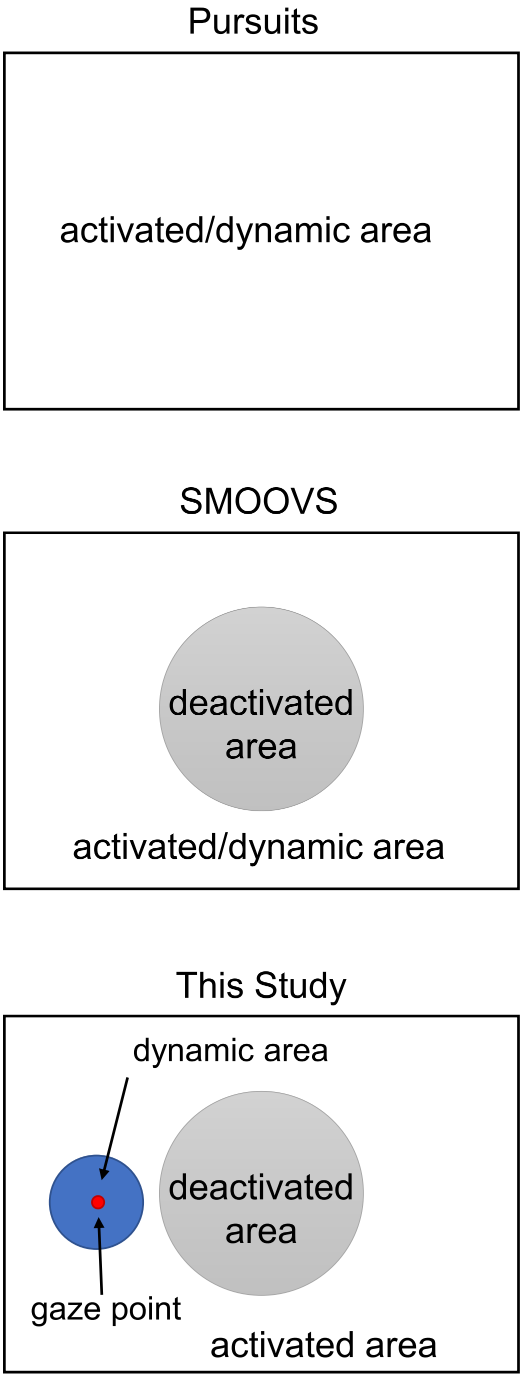


Figure 5.2: Differences in active interface areas in three pursuit-based gaze systems.

In this study, a new gaze typing interface is designed with the goal to facilitate an interaction that is perceived as natural as possible by users. Similar to the layout of the *Hex-o-Spell* (Blankertz et al., 2007), *SMOOVS* (Lutz et al., 2015) and *pEYEs* (Huckauf & Urbina, 2008) interfaces, this new interface utilizes a two-stage concept, where users select a cluster of characters in a first step, followed by the selection of an individual character in a second step.

5.2.1 Layout

The text entry interface is based on an octagon-like layout with eight clusters that allow users to input different characters (see Figure 5.3).

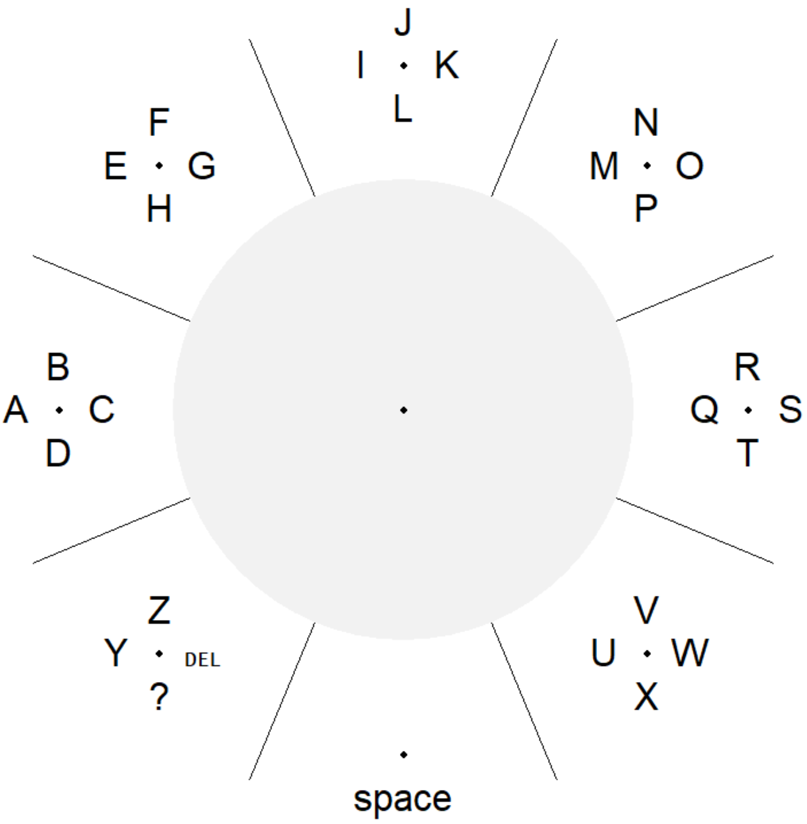


Figure 5.3: The illustration of eye typing interface, the background color is black and the character color is white in real typing system. The gray circular area is idle area, i. e., deactivated area.

The *SMOOVS* interface consists of six clusters, five of which contain six characters.

As Lutz et al. (2015) reported, the accuracy of gaze data decreased significantly from the middle to the edge of the screen after one-point calibration, i. e., the interactions in the central area of the interface are registered with higher accuracy. Hence, a larger number of objects can be distinguished in the first stage of detection which is placed more central in the interface, than in the second stage which is placed on the outside of the interface. Accordingly, the number of clusters could be increased in the first stage, thereby the number of characters in each cluster could be reduced in the second stage. In addition, the Study 1 in Chapter 4 found that the detection rates are higher for both interfaces with the 6 and 8 moving directions. Therefore, in this study, the number of clusters that are selected in the first stage was increased from 6 to 8. With the increased number of clusters, the number of character tiles in a cluster is reduced from 6 to 4, decreasing the required accuracy for character distinction in the second stage. With fewer tiles in the cluster, users also require a shorter time for searching and identifying the desired letters in a cluster.

Regarding the direction of movement, studies have shown that the detection accuracy for smooth pursuit eye movements is influenced by the movement direction. Horizontal and vertical directions are more robustly detected than diagonal directions (Ke et al., 2013; Krukowski & Stone, 2005). Thus, in order to achieve a more robust identification of gaze directions, characters in each cluster are designed to move along the horizontal and vertical axis, e. g., in cluster “ABCD", “A" moves left, “B" moves up, “C" moves right and “D" move down. In this study, 1° visual angle corresponds to 39 pixels at a distance of 60 cm from the user to the screen.

The alphabetical A-Z layout was used in this study, which contains 26 letters, as well as delete, space key, and question mark (see Figure 5.4a). According to (LetterFrequency, 2021), the space key is the most frequently used key, hence it is located in a vertically downward direction of the interface as a single actionable item (not in a cluster). While the key is mainly used to enter a space between words, it also serves as a way to confirm the currently entered word and indicate the start of an entry for a new word. Once the space key is used, typed words are moved from the center to the bottom of the screen (where the word “YOU" located in Figure 5.4a). The delete key is responsible for correcting typed characters.

5.2.2 Interaction design

When a user looks at the central area of the screen, e. g., when reading the typed characters, the interface remains static. When a user's gaze is registered outside of the deactivated central area, the detection of cluster selection is activated.

Stage 0: Idle state.

When the distance between the gaze point and midpoint of the screen is less than the radius of the idle area (see Figure 5.3), no action will be activated. The detection happens only when the eye position is out of the idle area. However, whenever the eyes move back to the idle area during the interaction, all elements in the interface will gradually move back to the their original positions and the system returns to the initial idle stage until user looks at the area beyond the inner idle circle.

Stage 1: Searching and cluster selection.

In this stage, all clusters remain static. The user needs to find the cluster which includes the desired character. The cluster in which the gaze is registered is highlighted (see example "EFGH" cluster in Figure 5.4a). However, if the user's gaze moves out of the area of this cluster, the highlighting is reversed. If the gaze moves to another cluster, that new cluster is highlighted, allowing users to adjust their selection before the activation of an input.

Stage 2: Character selection.

After a cluster is selected, the second stage begins (for more details on the classification algorithm, see Subsection 5.2.3). In this stage, the characters in the selected cluster move outward. Other clusters that were not selected fade out gradually at the same time to avoid visual distraction. That is, the active cluster is highlighted and the entailed characters move outward, while the rest of the clusters fades out. The typing interface at the end of stage 2 is shown in Figure 5.4b. The moving distance of characters is 75 pixels, if the user looks at another cluster when the characters in the selected cluster are moving, those characters will move back to their original position. After the characters in the selected cluster have moved 75 pixels, they fade out and then reappear in their original position (i. e., where they were before moved). If a character is successfully detected, feedback is given simultaneously. Detailed information can be found in the subsection 5.2.2.

The interaction flow is shown in Figure 5.4. When user try to enter the phrase "you are a wonderful example". The typed word ("YOU") is located at the bottom of the screen. The current typed characters ("AR") are displayed in the center of the deactivated area.

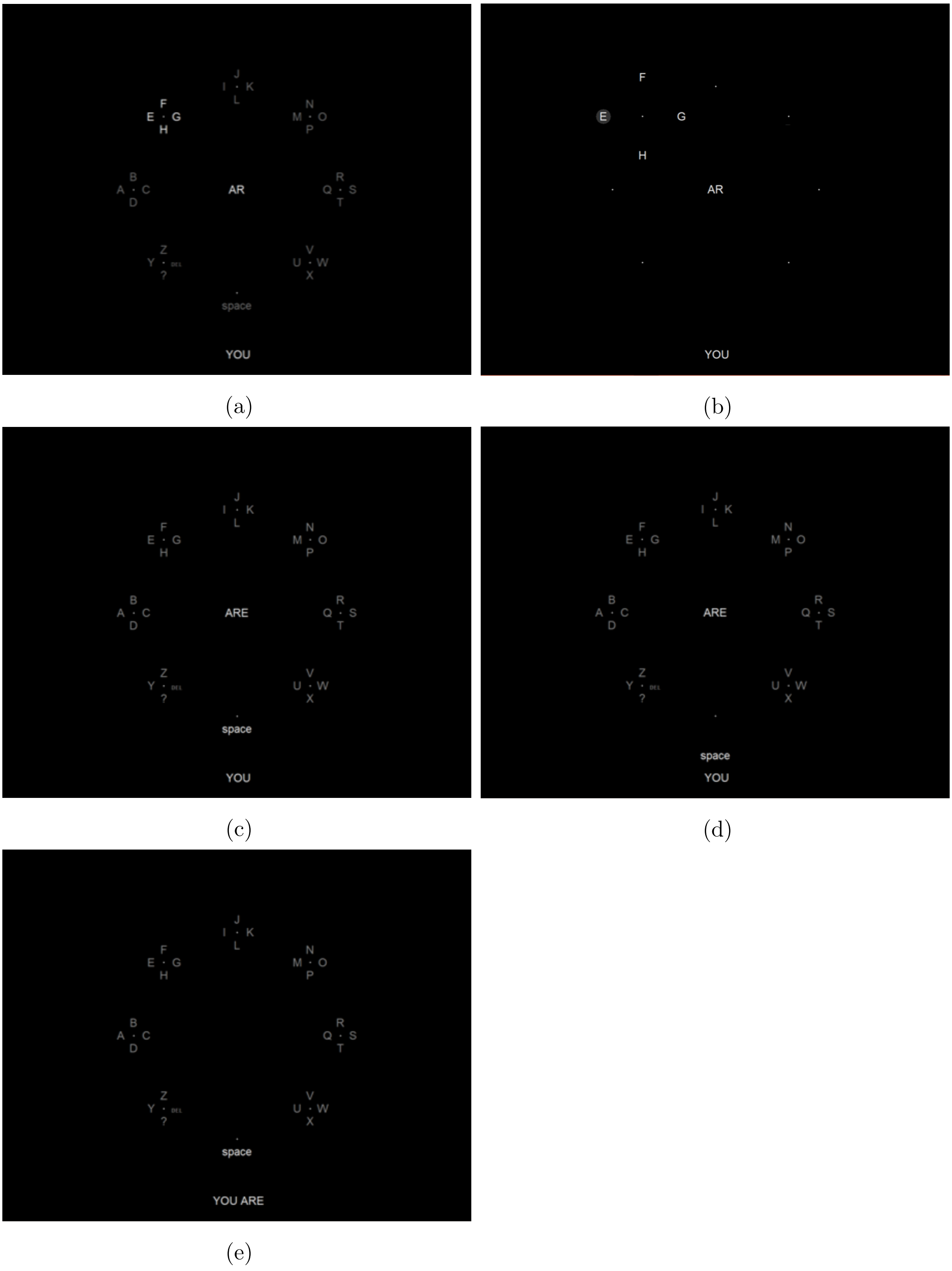


Figure 5.4: The interaction flow

To Enter “E", the eyes first start looking at the “EFGH" cluster, and this cluster is highlighted, illustrated in Figure 5.4a. The characters in selected cluster moved outward in pursuit stage. Eyes need to follow the movement of letter E to enter it, Figure 5.4b shows the end of pursuit stage. If Letter E is correctly detected, it will be shown in the center of the screen (see Figure 5.4c). After that, user can continually follow the space key to move the current typing word to the bottom area and enter space between words (see Figure 5.4d and Figure 5.4e).

Feedback design

Feedback was given to confirm that the system is responding to the input, visualize reactions, and confirm an activated action. The feedback factor consists of four levels, namely no feedback, visual feedback, auditory feedback, and both visual-auditory feedback. Visual feedback appears behind the selected character in a gray circle (the diameter of the circle is 44 pixels, approximately 1°). Auditory feedback was given via computer speakers. The auditory feedback sound was a short continuous beep at 300 Hz. The visual and auditory feedback is synchronized and lasts approximately 200 ms. Figure 5.5 illustrates four feedbacks. To avoid the effects of sequence caused by fatigue and practice, the order of conditions was randomized.

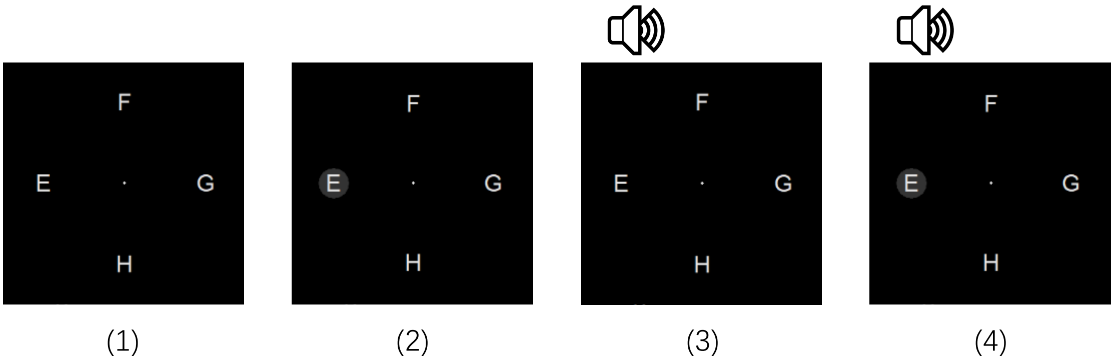


Figure 5.5: The provision of feedbacks is illustrated, from left to right, they are no feedback, only visual feedback, only auditory feedback and both visual-auditory feedback.

5.2.3 Classification algorithm

In the stage 1 (searching and cluster selection), the detection of a cluster is based on the midpoint of screen (M), gaze point (G) and the positive direction of the x-axis (as shown in Figure 5.6). The angle between the vector between two points and the positive x-direction is given by function 5.1 and converted from radian to degree.

$$\theta_g = \arctan 2\left(y_g - y_m, x_g - x_m\right) \tag{5.1}$$

where x_g and y_g are the coordinates of the eye position, x_m and y_m are the midpoint coordinates of a cluster. The detectable range for each cluster is 45° in first stage. When the angle meets the angular criterion of one cluster $\theta_u < \theta_g < \theta_l$, it is recognized as the corresponding cluster. When there are more than two successive measurements detected as the same cluster, i. e., $\theta_{g,i} = \theta_{g,i+1}$, this cluster is highlighted to inform the user about the current selection. When this cluster is further detected for more than 400 ms, then this cluster is activated and the second stage (dynamic stage) starts.

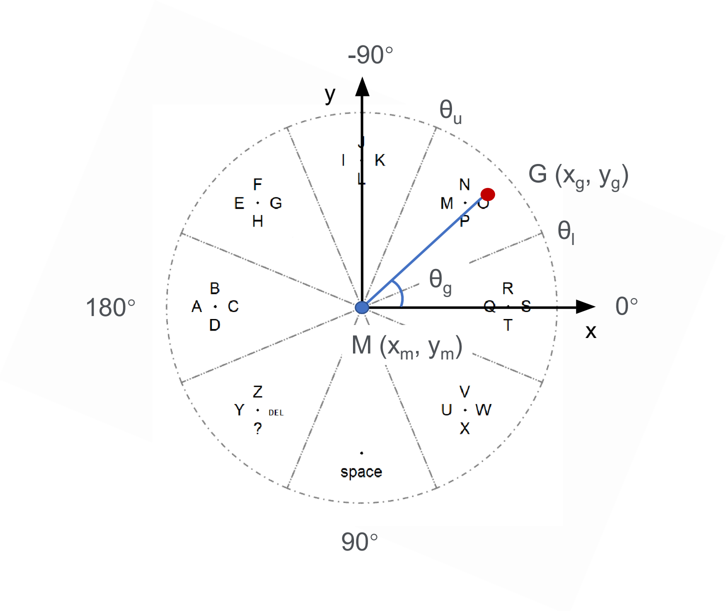


Figure 5.6: Visualization of the angular criterion in first stage. The red dot is gaze point (G), and blue dot is the midpoint of screen (M). The gray dotted line presents the identifiable range of each cluster.

In the stage 2 (character selection), only characters in the selected cluster begin to move. The moving speed of characters is 433 pixels/s (approximately 10°/s visual angle). Users can select the desired character by following the movement of this character with their eyes. But the following action itself does not affect the detection. When for two consecutive gaze points the distance between gaze points and the midpoint of this group exceeds 250 pixels in this stage, the characters in this cluster will move back to the original position, the detection process returns to the first stage. Similar to the first stage, character detection is also based on angular range. After the characters of the selected group have finished moving, 20 gaze points $[(x_{g,1}, y_{g,1}), ..., (x_{g,20}, y_{g,20})]$ will continue to be collected. A set of angles $[\theta_{g,1}, ..., \theta_{g,20}]$ is calculated according to those 20 gaze points and the middle point of this cluster. The *mode* of those angles is calculated and the character corresponding to this angle is selected.

The detectable range of a character for this stage is 85 degrees. In order to reduce false detections, an area of five degrees is defined between the adjacent characters. If the measurement falls into this area, the system skips this input.

5.2.4 One-point calibration

The one-point calibration method developed by Lutz et al. (2015) was used in this study. A number counting down from three appears in the center of the screen and this process lasts three seconds. Users were asked to look at the countdown number in the center of the screen (see in Figure 5.7).

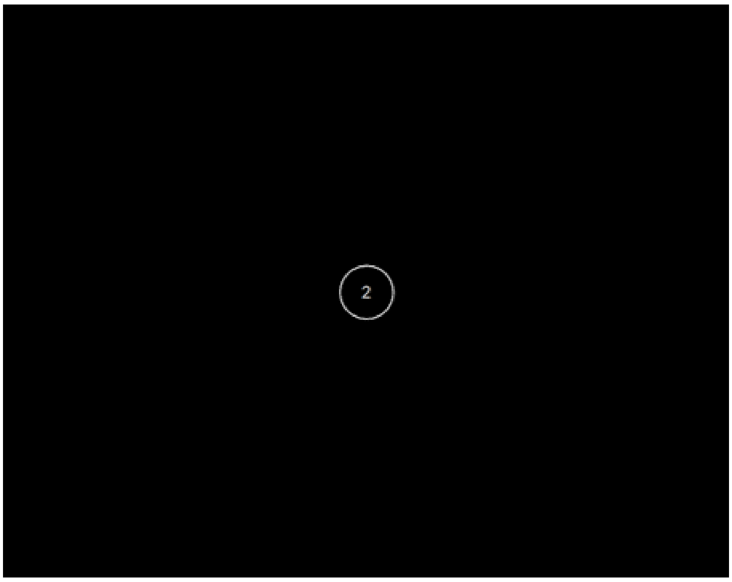


Figure 5.7: The interface of one-point calibration

During the one-point calibration, gaze coordinates were recorded and outliers were removed using DBSCAN (Ester et al., 1996). After the test in the pilot study, the maximum distance between two samples, i. e., one is considered to be in the vicinity of the other is set to 100 pixels (about 2.56° visual angle). The number of samples is set to 50,

i. e., to consider as a core point, the number of samples round the core point need be more than 50. The cluster that aggregates the most gaze data is selected for use for one-point calibration, i. e., other groups and gaze points that are not grouped are considered as outliers (illustrated in Figure 5.8).

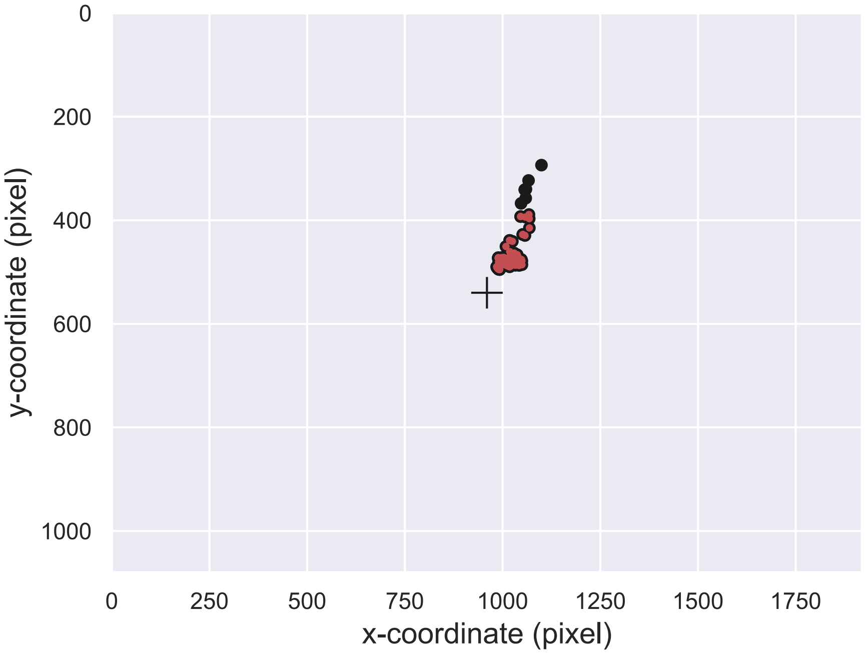
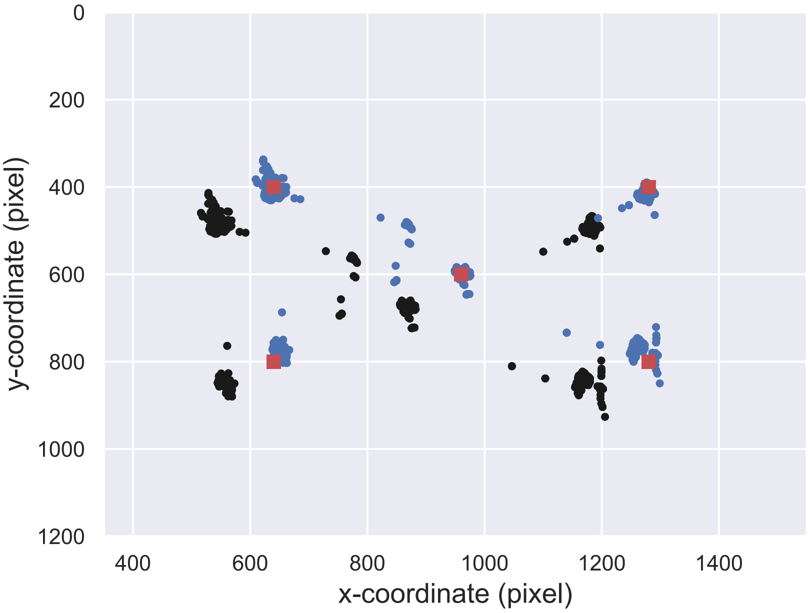


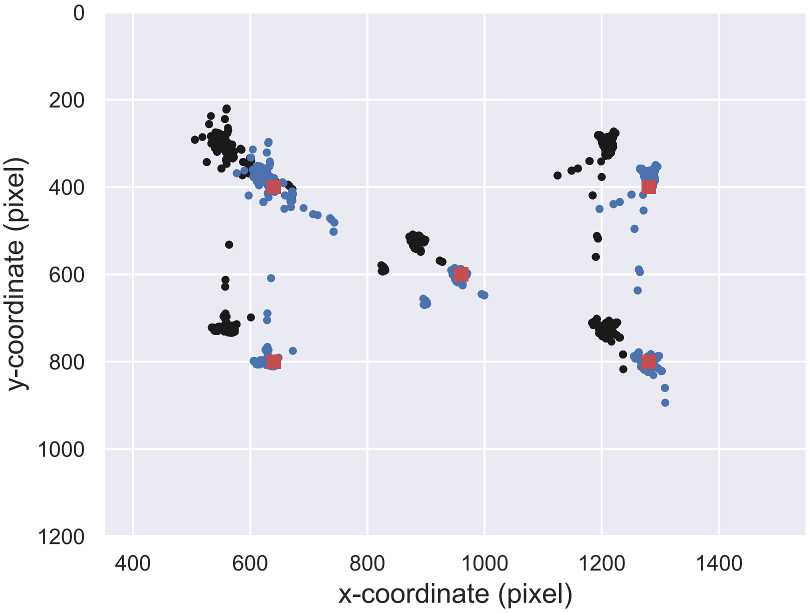
Figure 5.8: Remove outliers using DBSCAN. The black cross is position of the middle calibration Point. The black points are the gaze points during one-point calibration, the red points are the reserved gaze points after discarding outliers.

Based on the retained gaze points, the average means of distance for both x-and y-axis between the midpoint of the screen and gaze positions were calculated. Then, gaze coordinates were translated with the calculated offsets. In similar experiment settings, study 1 reported the average gaze estimation error was 4° (see Chapter 4). Thus if the offset is greater than 4°, the system reminds the participant to adjust their position according to the distance from the screen, and perform the calibration again. According the records during experiment, all participants were able to successfully complete the one-point calibration without re-calibration. On average across all participants, the offset was 55.42 pixels ($SD = 35.79$).

Figure 5.9 visualizes gaze data from the pilot study before and after one-point calibration, where the participant gazes at a stationary point that appeared in sequence at five positions for three seconds on the screen.



(a)



(b)

Figure 5.9: Visualization the gaze data from two participants before and after one-point calibration in pilot study. Red points are target points, the black points visualize the raw gaze data, and blue points visualize adjusted gaze data after one-point calibration.

5.3 Empirical evaluation

To investigate how well do the users enter text using the hybrid eye typing interface, a user study was conducted. In addition, four forms of feedback were compared for their influence on user performance and experience[1].

5.3.1 Evaluation metrics

The words per minute (WPM), keystrokes per character (KSPC) and minimum string distance (MSD) are used to measure typing performance, the calculated metrics are described in detail in Chapter 2.

In addition to the evaluation of typing performance, the participants' opinions were collected via short open questions. After completing the typing tasks for each experimental condition, the participants were asked directly to describe how they experienced the feedback from the system just presented. Once the whole experiment finished, participants were asked to fill out a written questionnaire. The main content of the open questions is shown in Table 5.1.

Table 5.1: The main content in open questions

1.	Please rank the four user interfaces and give reasons.
2.	How did you perceive the speed of the objects.
3.	How did you perceive the size of the objects?
4.	Did you have enough time to search for the desired objects?
5.	Suggestions for improvement

5.3.2 Participants

In this study, 29 participants were recruited (13 female, 16 male) from an online recruitment system. Their ages ranged from 19 to 38 years old, with a mean of 26.7 years. All participants had German as their native language. About half of the participants (69%) reported that they had normal vision and 31% of them wore vision aids during the study. Most participants (83%) had no previous experience with gaze interaction. 72% of participants had no experience with eye tracking.

[1] This experiment was conducted by a master student under supervision of the author of this dissertation (Neuer, 2020)

Both participants and supervisor were required to wear masks and keep a social distance of two meters from each other throughout the experiment due to COVID-19 related hygiene regulations. Participants were rewarded with ten Euros per visit or alternatively a certification of student experimental hours for attendance.

5.3.3 Apparatus

In this experiment, the Tobii EyeX eye-tracker was used to register the gaze location in screen coordinates with a sampling rate of 60 Hz. The eye tracker was mounted beneath a 24" Dell monitor with a resolution of 1920 * 1200 pixels. Task scenarios were created with Python and were presented on the monitor. All data were collected without personal calibration from users. The eye tracker was calibrated once by the experiment supervisor. The average distance between the participants' eyes to the display was 62 cm ($SD = 7.27$).

5.3.4 Procedure

After reading the informed consent form, and hygiene concept, the experiment started with an introduction of the gaze-based text entry system. The experiment consisted of one training and one test phase.

In the training phase, the interface of no feedback was given and five phrases were given for learning how to use the text entry system. If the participants had no further questions about the experimental task, the test phase began. In the test phase, the four text entry interfaces with different feedbacks were tested. Participants went through all four feedback conditions. Each feedback condition was tested with five phrases. Entering a phrase was considered as a trial, i. e., participant typed 5 phrases for each feedback condition and in all, a total of 20 phrases were given per participant. The order of feedback conditions and phrases was randomized. All phrases used in this experiment were selected from a set of 500 phrases (MacKenzie & Soukoreff, 2003) and translated into German. The phrases used in this study are showed in Appendix B.2. The participants were instructed to enter the given phrases as fast and as accurately as possible.

At the beginning of each trial, there was a short one-point calibration, which lasted three seconds. Then a phrase was displayed in capital letters in the center of the screen. The participants then pressed the space bar to start typing after they memorized the phrase. After finishing the tasks for one condition, participants were asked to take short breaks. When the participants completed all the text entry tasks, they were asked to fill out the questionnaire consisting of demographic information and subjective questions.

The whole experiment duration was between 40 and 60 minutes per participant.

5.4 Results

This study featured a one-factor within-subjects design. The results are based on a total of 580 phrases (29 participants * 4 feedbacks * 5 phrases). Typing speed, accuracy and subjective feedback were analyzed.

5.4.1 Text entry rate

The grand mean of typing speed was 4.7 WPM. The fastest typing speed was registered with 8.38 WPM. The descriptive statistics with regard to typing speed for each feedback condition are showed in Figure 5.10 and Table 5.2.

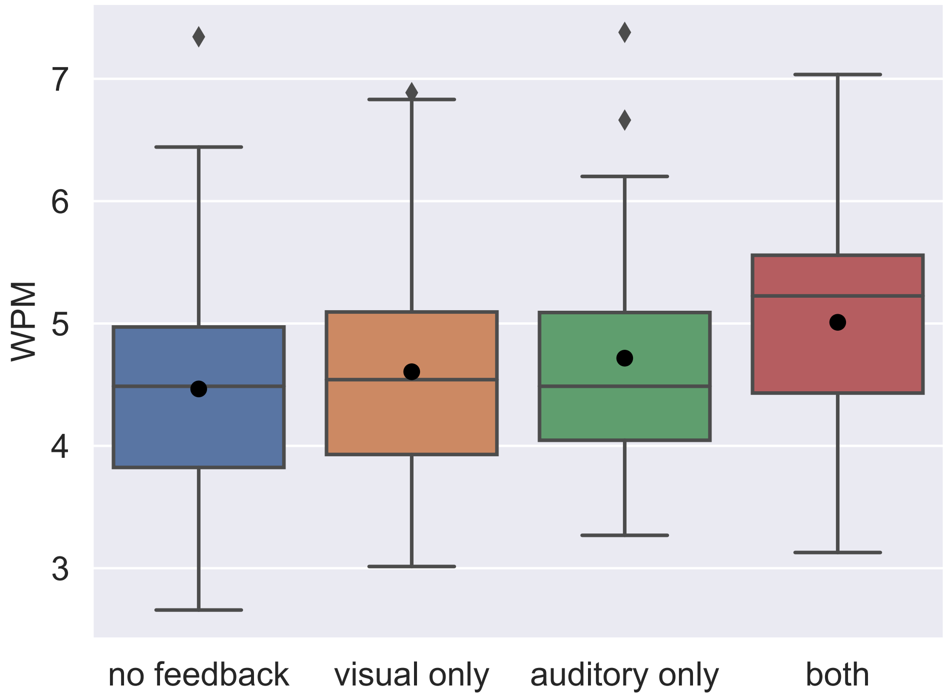


Figure 5.10: Words per minute for each type of feedback, black dots visualize the average values, diamonds are outlier observations.

A repeated-measures ANOVA at a significance level of $\alpha = 0.05$ was applied to analyze data. Mauchly's test of sphericity indicated that no correction was needed. The feedback conditions significantly affected the text entry rate ($F(3, 84) = 4.61, p < .01$). A set of

Bonferroni corrected t-tests were conducted for pairwise multiple comparisons. The Post-hoc comparisons revealed that the text entry rate of both visual-auditory feedback was significantly greater than no feedback ($p < .001$).

5.4.2 Error rates

Keystrokes per character (KSPC)

The grand mean of KSPC was 1.67. The descriptive statistics of KSPC for each feedback condition are presented in Figure 5.11 and Table 5.2. The Kolmogorov-Smirnov test showed that the data did not conform to a normal distribution ($p < .001$), hence the Friedman test was used to assess differences in the KSPC error between feedback conditions. No significant difference was found between the different feedback conditions ($\chi^2(3) = 6.19, p = .10$).

Minimun string distance (MSD)

The grand mean of MSD was 0.1 and the result of the descriptive statistics of MSD was shown in Figure 5.12 and Table 5.2. The Kolmogorov-Smirnov test showed that the data did not conform to a normal distribution ($p < .001$), and the Friedman test was conducted to assess the MSD error. No significant difference was found in the feedbacks ($\chi^2(3) = 2.83, p = .42$).

Table 5.2: Study 2 - Mean absolute values (M) and standard deviations (SD) for each experimental condition.

Feedbacks	WPM		KSPC		MSD	
	M	*SD*	*M*	*SD*	*M*	*SD*
No feedback	4.47	1.13	1.72	0.54	0.10	0.11
Visual only	4.61	0.96	1.70	0.50	0.09	0.06
Auditory only	4.72	0.99	1.69	0.61	0.13	0.16
Both	**5.01**	0.92	1.58	0.41	0.09	0.08

5.4.3 Subjective evaluation

During and after the experiment, participants were asked to report their perceptions and preferences for the four feedback conditions. In addition, the general perceptions about

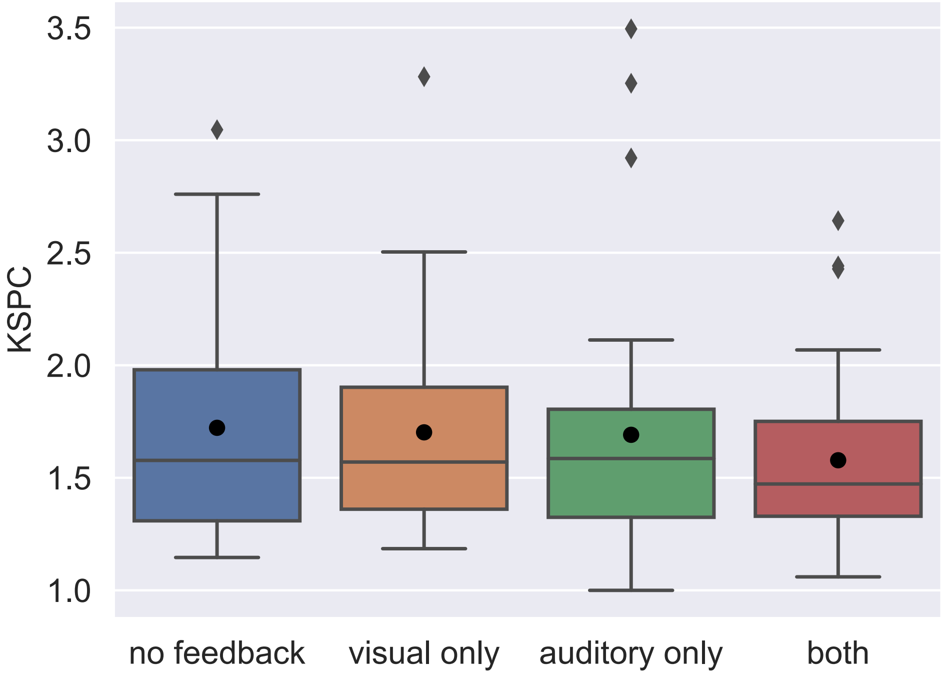


Figure 5.11: Keystrokes per character for each feedback, black dots visualize the average values, diamonds are outlier observations.

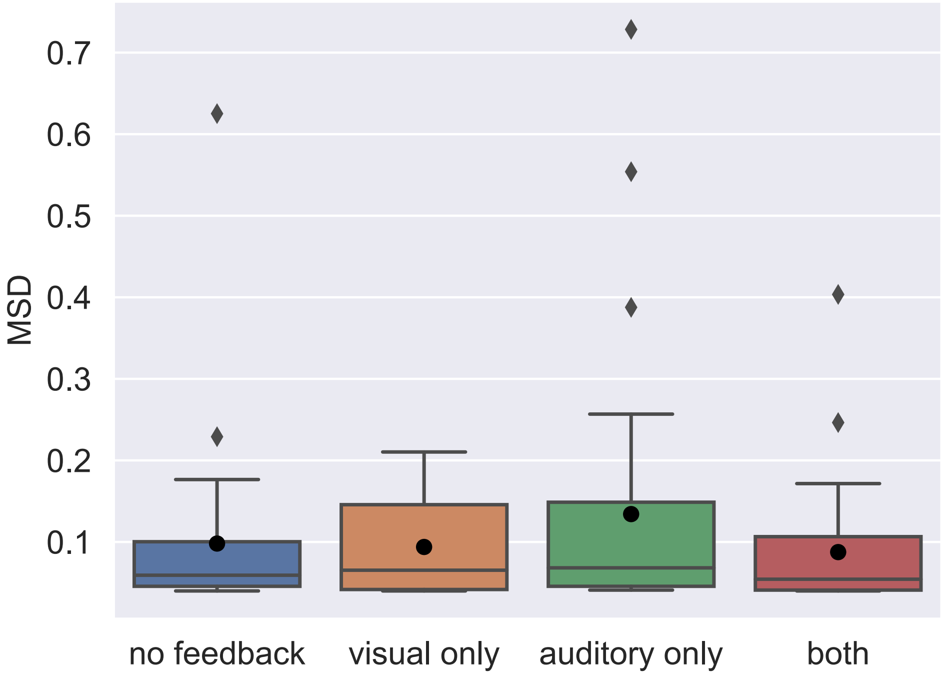


Figure 5.12: Minimun string distance error rate for each feedback, black dots visualize the average values, diamonds are outlier observations.

the text entry system and ideas for improvement were collected and summarized.

All participants reported that they noticed differences in feedback, and almost all of them correctly described these differences. However, the presence of visual feedback was not perceived by the two participants. More than half (55%) of the participants thought that the combined visual-auditory feedback is helpful. Approximately one-third (31%) considered that visual feedback is helpful, and 17% thought that auditory feedback is helpful. Additionally, participants were asked to rank all four feedback conditions. The most preferred feedback was ranked first and the least preferred was ranked fourth. Table 5.3 shows the results of this ranking.

Table 5.3: Ranking results, 1 represents the most preferred, 4 represents the lowest ranking.

		Ranking count			
Feedback	N	1	2	3	4
No Feedback	29	1	3	6	19
Visual only	29	15	9	5	0
Auditory only	29	0	6	15	8
Both	29	13	12	3	1

More than half of participants ranked the sole-visual feedback condition first and nearly half of participants ranked the combined auditory-visual feedback condition first. A Friedman's test revealed that there was a significant difference between the ranking of the four feedback conditions ($\chi^2(3) = 45.68, p < .001$). Post-hoc pairwise comparisons with the Wilcoxon signed-rank test showed that the ranking for combined visual-auditory feedback condition was significantly higher than no feedback condition ($p < .001$) and sole-auditory feedback condition ($p < .001$). Also, the ranking for the sole-visual feedback condition was significantly higher than the sole-auditory ($p < .001$) and no feedback condition ($p < .001$).

The reasons for feedback preference were collected (summarized in Table 5.4). The sole-visual feedback is preferred by most participants because it not only helped the user to confirm the input but also provided information about which character is selected. Some participants found the tone used for the auditory feedback was distracting and disturbing.

The answer regarding the moving speed, visual search time, and item size were clustered (shown in Table 5.5) . About one-third of participants considered the movement

Table 5.4: Reasons behind feedback preference

Feedback	Reasons for ranking	Count
No Feedback	Confirmation of input detection is missing	13
	Need to confirm the input	6
Visual only	Provide direct feedback	6
	Better concentration on typing task	5
	Given confirmation of the input	2
	Checking the input is omitted	2
	Less mistakes	2
Auditory only	Sound is disturbing	6
	Eye response to sound causes errors	2
Both	More robust system interaction	10
	Double support in interaction	2

speed of the interface as too fast which leads to unintentional input. Some participants felt that the moving speed is fast only at the beginning of the use. And some participants stated that the characters which were not easy to find were moving too fast. Besides, there were a few participants who considered the moving speed for some characters as too slow.

At the end of the questionnaire, participants were asked to give suggestions for potential improvement. Multiple participants suggested that the visual feedback should be more salient, and e. g., color should be added to the selected character or the font should be bold, indicating that other visual feedback might be difficult to notice. In addition, participants suggested that the feedback tone should be more user-friendly. Another suggestion was to use different types of auditory feedback for special keys, such as the delete key. Some participants indicated that the delete key could be re-positioned to a better location. And participants were also looking forward to having a QWERTY layout design. A reduction of movement speed and the addition of a word prediction functionality were also mentioned as potential advancements.

Table 5.5: Perception of the text entry system

Topic	Description	Count
Speed	Too fast	11
	Feel fast at the beginning of the use	6
	Too fast for unfamiliar letters	6
	Appropriate	3
	Some keys (e.g., space) are slow	3
Search time	Time is enough to search the desired tiles	14
	Not enough time for searching	13
	After using it for a while, the time is enough	2
Item size	Appropriate	23
	Some keys (e.g., delete) are small	3
	Too small	3

5.5 Discussion

The goal of this study was to develop a novel eye typing interface with a relatively low requirement for spatial accuracy. An experimental laboratory study was conducted to evaluate users' performance and preferences. In addition, multiple feedback stimuli were compared to facilitate immediate input confirmation during the use of the interface.

The results show that the participants were able to learn how to efficiently use the system within an hour. This result has to be seen in the light of the nature of participants in this study, who all were novice users of this novel layout. Most of them had no previous experience with gaze interaction. The average text entry speed reached 4.7 WPM, considerably higher than the *SMOOVS* typing systems (typing speed varies in the range of 2.9 to 3.34 WPM).

However, both correct and uncorrected error rates were relatively high, and participants reported that the system was too sensitive. A lot of participants commented that the moving speed was too fast and they didn't have enough time to search the desired letter, with one participant stating "It's a bit fast when selecting letter groups, which is why it often opens some groups that you don't want to open." The dwell time for cluster selection was set as 400 ms. To achieve a more robust detection, a longer dwell time could be introduced, especially for novices who are not familiar with the layout.

The movement speed of the target is one of the important parameters for the pursuit-based gaze interface and affects the detection performance. When the target moves too fast, smooth pursuit eye movement will be interrupted by catch-up saccades to keep up with the moving target (Dodge, 1903; Robinson, 1965). Moreover, the tracking trajectory is limited for a linear moving target when the size of the display is not big enough, i. e., the target reaches the edge of the screen soon (Vidal et al., 2013). The studies reported that the detection rate for method based on correlation drops when the moving speed is too high or too low, which needs to be controlled in a reasonable range (Drewes et al., 2018; Esteves et al., 2015). Medium angular speed around 120 °/s is considered as proper in orbits trajectory (Esteves et al., 2015), where Drewes et al. (2018) came to a conclusion that the detection rate works well when the moving speed in a range from 6 to 16° visual angle/s. In addition, the studies regarding linear trajectory also came to similar conclusions. Cymek et al. (2014) found that the medium speed (218 pixels/s, about 4.9 °/s) has a higher subjective user evaluation. And the text entry speed is not proportional to moving speed, but peaks at a medium speed (300 pixels/s, approximately 8 °/s). Zeng, Siebert, Venjakob, and Roetting (2020) compared two moving speeds (7.73 °/s and 12.89 °/s), and found that the detection rate for faster-moving speed is higher than slower ones. More important is that there is a general consensus among pursuit-based studies about moving distance, the detection rate or subjective evaluation is higher with a longer trajectory (Cymek et al., 2014; Esteves et al., 2015; Zeng et al., 2020). And our data suggests that the movement speed of the interface in this study was potentially set too high for novices, they can get used to faster speeds as usage time grows.

Besides, the input accuracy might be increased with consideration of learning effects and individual differences. For example, users could be allowed to adjust the parameter themselves (Špakov & Miniotas, 2004) or the system could automatically adjust to meet the need based on past typing records (Mott et al., 2017; Pi et al., 2020).

The results also indicated that the type of feedback used in the interface significantly relates to the typing speed. The highest typing speed was found in the feedback condition with the combined visual and auditory feedback. No significant difference was found either on KSPC or MSD error regarding feedback conditions. Hence, it can be concluded that for typing efficiency in the hybrid eye typing interface, it is optimal to provide combined audio-visual feedback, as the experimental data suggests that it works better than no feedback.

Although the evaluation of user performance indicated that the combined visual and auditory feedback resulted in higher typing speed and lower error rates, the evaluation of user preference showed that 52% of participants ranked the sole-visual feedback to be the most preferred feedback, and slightly fewer participants (45%) preferred the combined visual-auditory feedback most. No participants felt that visual feedback caused extra visual noise that made it hard to concentrate on typing. There was no significant increase in demand for visual resources. Participants appreciated that visual feedback informed them that input was registered and which action was activated. The reason that fewer participants like the auditory feedback were similar to what Špakov et al. (2016) reported about auditory feedback. Participants found the auditory feedback to be useful but distracting in both studies.

Our assessment of effectiveness and efficiency variables, as well as user feedback for the hybrid eye typing interface, shows that the system represents an enjoyable and well-functioning human-machine interaction approach. While the text entry speed count of 4.7 WPM found in this study is not high in comparison to other gaze-based spellers, there are two critical advantages over existing spellers, robustness and user experience.

The hybrid eye typing interface alleviates the challenges of a lack of calibration in walk-up-and-use eye-tracking solutions. Such calibration-free eye typing methods need a shorter time required for the calibration process of eye tracker. Since the angle of the gaze-path is mostly unaffected by a gaze point offset that is sometimes found with one-point calibration, this gaze speller is well suited for immediate interactions. The relatively simple arrangement of usable objects further supports this.

As a second main advantage over existing systems, the exclusive actionability of select object groups through visual and auditory feedback increases user feedback and user understanding during the use of the system. This new eye typing interface is featured with a hybrid design, which combines both dwell-based and pursuit-based gaze interaction. The interaction is dived into two-stages. The cluster selection is corresponding to the static stage, where the cluster looked at by the user is highlighted, but the characters on this interface remain static which enables the user to search the desired character more easily. The characters in one cluster begin to move when the user is fixating on this cluster for a certain time. The selection of characters is done in the dynamic stage. Compared to dwell-based interfaces which have a high requirement for calibration, the calibration process for the proposed hybrid design is reduced to three seconds. There is also no learning of specific eye-movements necessary (like in gesture-based systems), as users do not need to remember or learn gestures to use the hybrid system. Compared to the existing pursuit-based eye typing systems (e. g., *SMOOVS*), the typing efficiency is slightly improved in the proposed hybrid interface.

It can be expected that any introduction of gaze-based interfaces in the public will initially depend on a high level of detection robustness, the simpleness of the interface, and the understand-ability of the interaction design. Since this novel eye typing interface combines these factors, it represents a prime candidate for implementation in public spaces. This hand-free text entry system can help to limit touches and helps to promote hand hygiene. The system is also considered to be more convenient, e. g., for people with disabilities. It could be complementary, where eye input can be used as an alternative interaction method, coexisting with physical buttons and touch screens.

5.6 Summary

This new eye typing interface enables users to enter text by eye after a three seconds calibration process, which shortens the time for calibration. It also provided a hands-free eye typing system with robust typing performance as well as relatively high user acceptance. The interface successfully avoids the Midas touch problem, through the implementation of a static stage, that allows users to search the desired character. The rearrangement of character-clusters in the interface counteracts the effect of lower gaze detection accuracy in the border areas of interfaces. While the combined visual-auditory feedback resulted in the highest typing speed, subjective data revealed that users prefer sole-visual to combined visual-auditory feedback, because the sound was distracting sometimes.

Chapter 6

Study 3: Evaluating eye typing interfaces with language model

6.1 Introduction

A circle-layout eye typing interface that enables enter text only by gaze after a short one-point calibration was proposed in the last chapter. The user could enter a character by following the movement of the desired character by eye. Although this text entry system presents an early investigation of calibration-free eye typing, all novice participants successfully completed the typing task, the maximum typing speed is limited due to the two-stage selection design and is relatively slow.

In order to achieve an acceptable typing speed, utilizing the language model for this eye typing interface is formally investigated. In this chapter, three eye typing interfaces are developed and a user study is conducted to address the following research questions:

1. Whether utilizing language model can significantly improve the typing performance for this eye typing interface.
2. Whether utilizing language model can significantly reduce the workload and improve the user experience during typing.
3. What kind of letter/word prediction designs are more appropriate for such a dynamic eye typing interface.

6.2 Interface design

Consistent with study 2, the eye typing interface is arranged in the order of letters A-Z and contains 26 letters. In addition, there are delete and space keys. In this study, the position of the space key is different from that of in study 2, where space key is located in the vertical downward direction of the interface and, and there is only one item (space key) in this direction. In this study, both delete and space keys are in a group with the letters Y and Z. The position of the cluster moving downward is reserved for the function keys of the word prediction. The currently typed letter is still displayed in the middle of the idle area. When the user checks the typed letters, i. e., the letters displayed in the middle of idle area, no action will be activated. If the user finishes typing a word, they can follow the “SP" key, and the entered word will move from the center to the bottom of the screen.

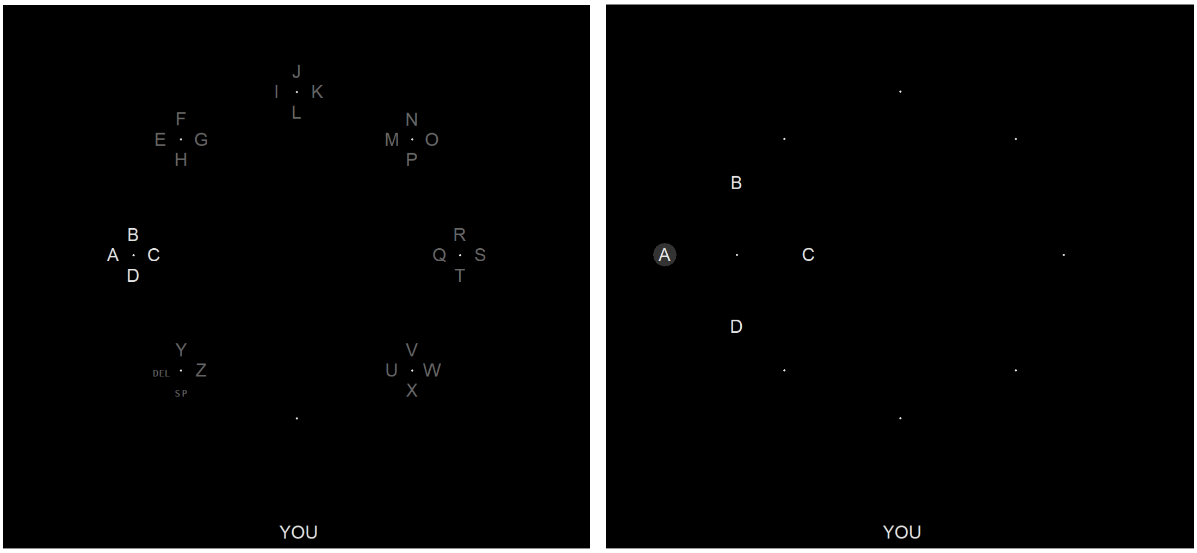
Before entering the typing interface, there is a one-point calibration process, which lasts three seconds, as detailed in Subsection 5.2.4. The process of entering one character is consistent with study 2.

However, according to the results from study 2, participants reflected that the searching time was short and the moving speed is fast. Thus, the dwell time for the stage 1 (searching and cluster selection) was lengthened to 600 ms, i. e., if one cluster is looked over 600 ms, the characters in this cluster start to move outwards. In addition, the moving speed was slowed which is stable at about 250 pixels/s (equivalent to 6.4° visual angle/s). To avoid distraction, when the selected cluster is moving out, other clusters fade out. Figure 6.1b presents the position of the characters when the movement of one cluster is finished. The detection method is the same as study 2. Auditory and visual feedback was given after a character is selected.

Eye typing interfaces with language model

In this study, language model was introduced into the design of the hybrid eye typing interface. Based on the original hybrid eye typing interface proposed in study 2, two more versions, one with letter prediction and the other one with both letter and word prediction, were proposed. Thus, three typing interfaces were tested, they are: (1) No prediction (NoP), served as a baseline, (2) Letter prediction (LP), and (3) Letter + word prediction (L+WP).

The interface of the letter prediction is basically the same in appearance as the No prediction, both contain seven clusters (see Figure 6.1). The difference between No prediction and letter prediction is the moving distance of each letter (see Figure 6.2). In the



(a) Cluster selection

(b) Character selection

Figure 6.1: Screenshot of interfaces of No prediction. (a) when the eyes look at the ABCD cluster, and this cluster is highlighted as feedback. (b) the characters in selected cluster moved outward.

Letter prediction, if the letter belongs to the list of the next four most likely letters to appear, and the cluster containing this letter contains only one letter from the list. The moving distance of this cluster will be reduced from 94 to 68 pixels.

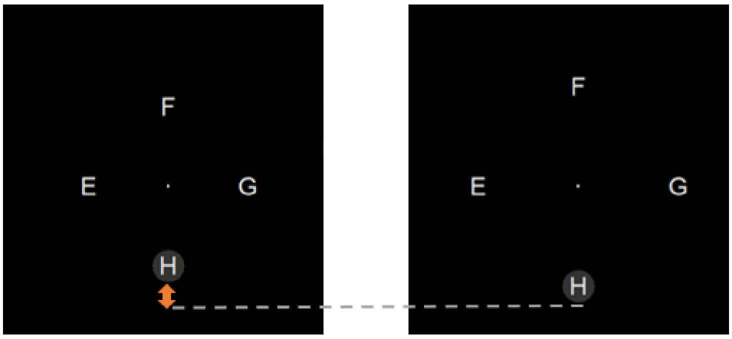


Figure 6.2: The default value of moving distance is 94 pixels (right figure), The figure on the left shows the moving distance when letter prediction activated, i. e., 68 pixels.

The interface with Letter + word prediction contains eight clusters and has one more cluster than No prediction and Letter prediction. In addition to letter prediction function, the three most probable word candidates are presented around the current typed characters (see Figure 6.3). The predicted words are located in the central idle area, and when looking at these predicted words, no action will be triggered. In the cluster containing three arrows, each arrow corresponds to a predicted word in the corresponding position. Since the entered words will be displayed at the bottom of the screen, no character is placed in the downward direction of the arrow cluster to prevent the user from triggering an unintentional selection when reading the entered words.

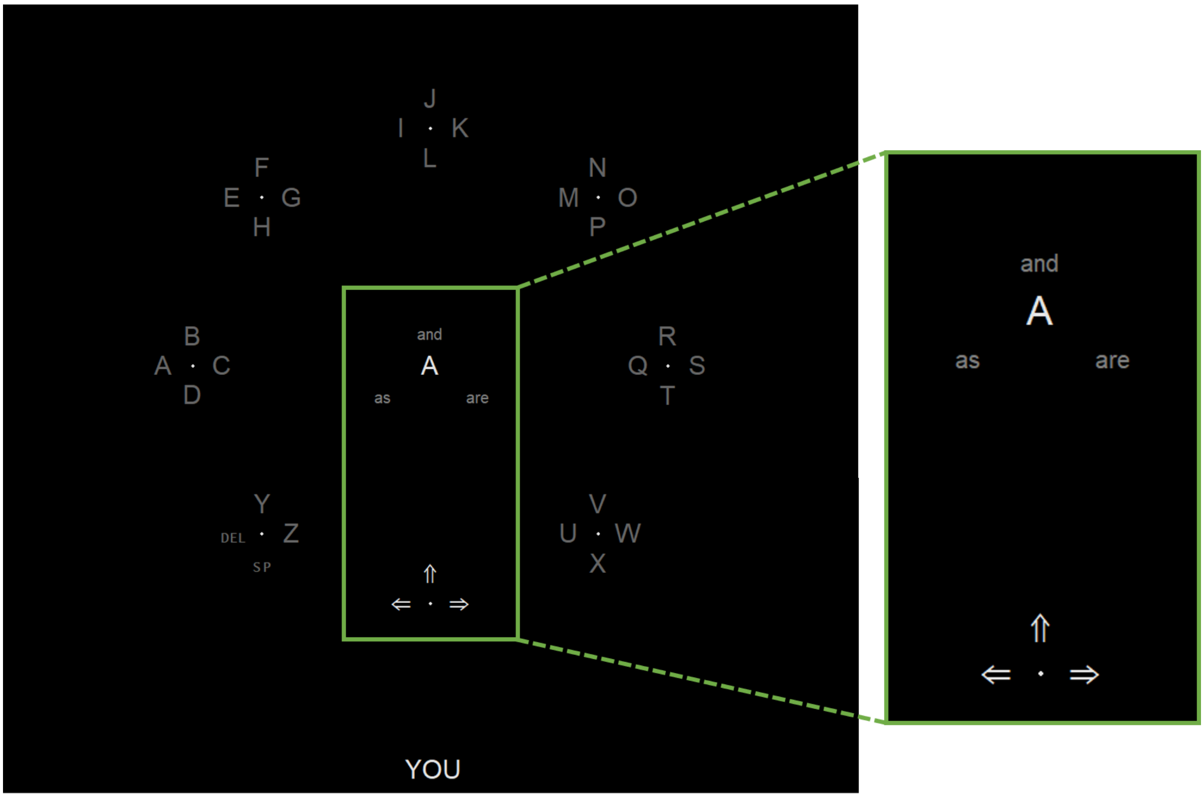


Figure 6.3: The interface for Letter + word prediction. User could follow the corresponding arrow to enter predicted word. In this example, the up arrow stands for "and", the left arrow represents "as", and the right arrow is for "are".

A character-to-word model based on Convolutional Neural Networks (CNNs) proposed by Park (2017) has been chosen for word prediction, which is available to provide both prediction of the current possible word as well as the next possible word. In Letter + word prediction, when a user is typing the current words, three suggestions of word completion are given. And when a word is just entered, three next possible words are provided. The unintentional activation of word prediction leads to more errors and longer correction time, thus, 15 more gaze points are collected to calculate the final angle for the arrow cluster.

6.3 Empirical evaluation

This study executed a user study to collect the data of typing performance and subjective evaluation on the three eye typing interfaces.

6.3.1 Experiment design

The experiment was a 3 * 3 within-subjects design. The *Prediction mode* factor contained three levels (No prediction, Letter prediction, Letter + word prediction) and *Session* is from 1 - 3. The phrases were chosen from the phrases set proposed by MacKenzie and Soukoreff (2003). The order of the conditions was counterbalanced across participants. In summary, 216 trials were retained: 12 participants * 3 interface designs * 3 sessions * 2 phrases = 216 trials.

6.3.2 Evaluation metrics

The words per minute (WPM), uncorrected error rate (UER), corrected error rate (CER) and keystroke savings (KS) were used to measure the typing performance.

Words per minute were used to quantify the typing speed during text entry. One word is considered as five characters, including letters, spaces, punctuations. Uncorrected error rate refers the how many errors remain in transcribed text. And corrected error rate presents how frequently backspace is used during text entry (MacKenzie & Tanaka-Ishii, 2010). Keystroke savings reflects how many keystrokes are saved with word prediction/completion (Trnka & McCoy, 2008).

In addition, participant's perceived workload was measured using NASA Task Load Index (NASA TLX). Besides, subjective feedback about the moving speed, search time, word prediction, and ideas for improvement was also collected in open questions. The main questions are showed in Table 6.1.

6.3.3 Participants

Twelve participants (6 males, 6 females) were recruited for this study. The participants' average age was 28.83 (SD = 4.88). Seven participants wore glasses, one wore contact lenses, four did not wear any visual aids. Two participants knew about gaze interaction before, and the others had had no experience with gaze interaction. And all of them were fluent English speakers, although none of them were native English speakers. Participants

Table 6.1: Main topics for open questions

1.	How did you perceive the moving speed of the objects?
2.	Did you have enough time to search for the desired objects?
3.	Can you easily understand that each arrow corresponds to each predicted word?
4.	Can you successfully select the prediction word?
5.	Do you think word prediction is helpful?
6.	Do you have any suggestions for improvement?

signed an informed consent form before starting the experiment and received monetary compensation after the experiment.

6.3.4 Apparatus

Consistent with the setup of study 2 in Chapter 5, a Tobii EyeX with 60 Hz was attached under a 24-inch monitor with a resolution of 1920 * 1200 pixels. The eye tracker was calibrated beforehand by the experimenter. The participants were asked to sit in front of the monitor. If the participant's eye position moves out of the detection range of the eye tracker device (too close or too far), the participant will be reminded to adjust the position forward or backward until a detectable distance is reached. No other instructions were given. The distance between user and monitor ranged from 44 cm to 72.8 cm (M =58 cm, SD = 6.42). The experimental program were implemented in Python. At a distance of 60 cm from participant to screen, 1° is equivalent to 39 pixels.

6.3.5 Procedure

Prior to the experiment, participants were informed the hygiene concept regarding COVID-19. Both participants and experimenters were required to wear masks and maintain a safe distance of 1.5 meters.

The experiment started with a short introduction about the eye typing system. After that, the participants can freely practice how to use this typing system. After understanding the basic operations, the test began.

At the beginning of each trial, a phrase was displayed in the center of the screen, and participants were asked to remember the phase and press the space key to start the typing task. All selected phrases were easy to understand and remember, such as "time to go shopping", "you must make an appointment", the phrases used in this study are showed in Appendix C.2. There were two trials in each condition, participants were asked to complete the NASA-TLX questionnaire to measure their perceived workload after each condition. They were instructed to enter the given phase as quickly and accurately as possible and correct errors that occurred in the current typing word. Participants were told that they can rest between trials. The demographic questionnaire (e. g., gender, age, and experience about gaze interaction) and open questions were answered after the text entry tasks. The experiment lasted about 40-60 minutes.

6.4 Results

In this section, the typing speed, error rates, and subjective evaluation are analyzed. The descriptive statistics with regard to words per minute, corrected error rate and uncorrected error rate is shown in Figure 6.4 and Table 6.2. The blue circle represents No prediction, the orange triangle stands for Letter prediction, and the green square indicates Letter + word prediction. The descriptive data for keystroke savings is presented in Table 6.3.

Repeated measures ANOVA was used for data analysis, and Bonferroni corrections were used in pair-wise comparisons. Shapiro-Wilk was used to check normality, and for non-normal data, the aligned rank transform (ART) was performed before ANOVA (Wobbrock, Findlater, Gergle, & Higgins, 2011).

6.4.1 Text entry rate

Words per minute (WPM) was used to measured text entry speed. The Letter + word prediction achieved an average text entry rate of 5.48 WPM over sessions, with 4.42 WPM ($SD = 1.31$) in session 1 and 6.19 WPM ($SD = 1.77$) in session 3. The Letter prediction achieved an average text entry rate of 3.42 WPM over sessions, with 3.49 WPM ($SD = 0.61$) in session 1 and 3.33 WPM ($SD = 0.83$) in session 3. The No prediction achieved an average text entry rate of 3.39 WPM over sessions, with 3.19 WPM ($SD = 0.68$) in session 1 and 3.64 WPM ($SD = 0.6$) in session 3.

A Shapiro-Wilk normality test shown that the words per minute were normally distributed ($p > .05$). Since Mauchly's test indicated a violation of sphericity ($p < .05$), the Greenhouse–Geisser correction was used to correct for violation of the assumption of sphericity. The results show a significant main effect of Prediction mode on words per minute ($F(1.19, 13.12) = 52.10, p < .001$). A significant effect of Session on words per

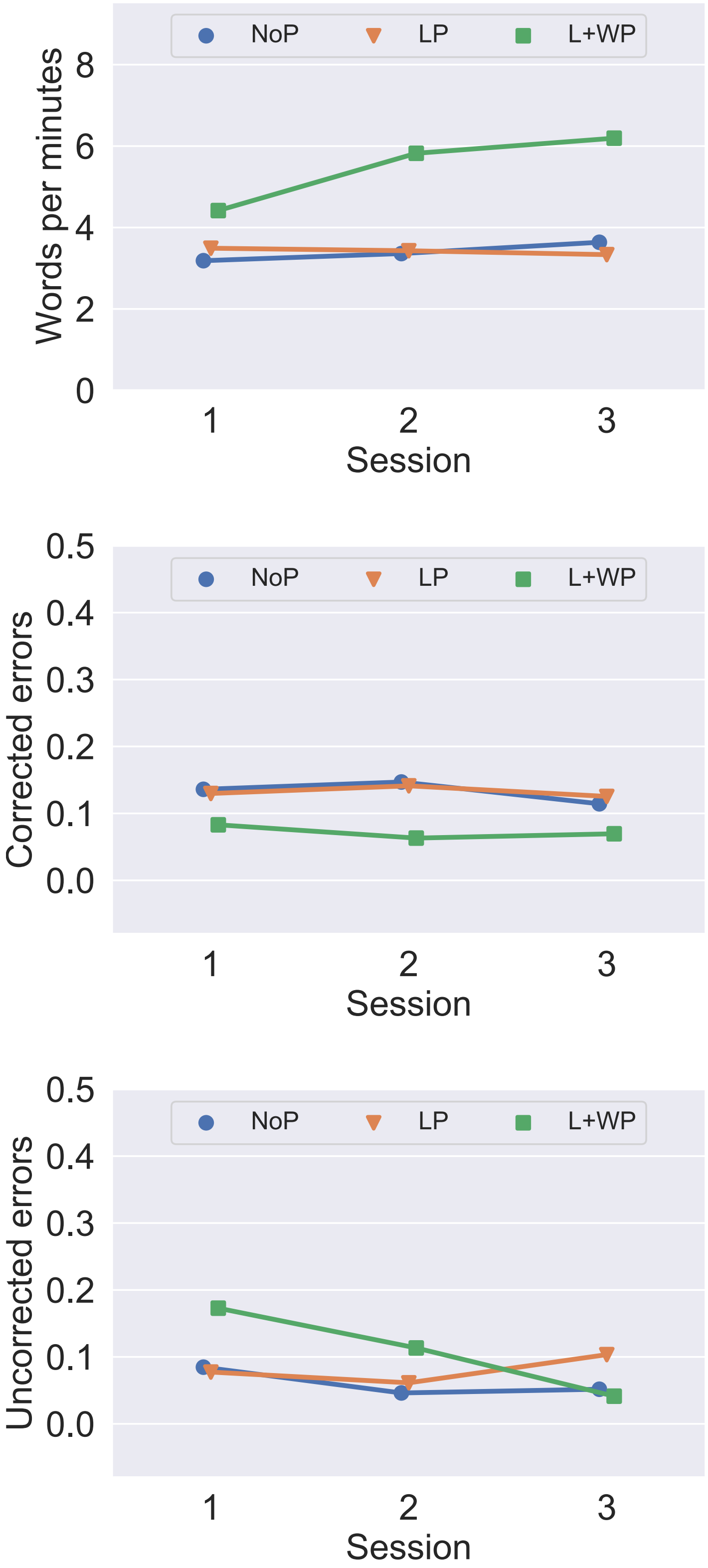


Figure 6.4: Mean of words per minute, corrected error rate, and uncorrected error rate over 3 sessions.

Table 6.2: Study 3 - Mean absolute values (M) and standard deviations (SD) for each experimental condition.

		WPM		CER		UER	
Prediction	Session	*M*	*SD*	*M*	*SD*	*M*	*SD*
L+WP	1	4.42	1.31	0.08	0.06	0.17	0.21
	2	5.82	1.68	0.06	0.05	0.11	0.13
	3	**6.19**	1.77	0.07	0.04	0.04	0.06
LP	1	3.49	0.61	0.13	0.09	0.08	0.09
	2	3.43	0.80	0.14	0.10	0.06	0.11
	3	3.33	0.83	0.13	0.06	0.10	0.14
NoP	1	3.19	0.68	0.14	0.09	0.08	0.13
	2	3.36	0.45	0.15	0.11	0.05	0.09
	3	3.64	0.60	0.11	0.07	0.05	0.10

minute was also found ($F(2, 22) = 7.34, p < .01$). There was a significant interaction between Prediction mode and Session on words per minute ($F(2.2, 24.2) = 4.95, p < .05$). The effect for session interacted with Prediction mode, that is, session affected Letter + word prediction differently than Letter prediction and No prediction. The green line (Letter + word prediction) has a steeper increase from session 1 to session 3 whereas the orange (Letter prediction) and blue (No prediction) lines are much more horizontal. And the pair-wise comparisons shown that there were significant difference between session 1 and 2 ($p < .05$) and between session 1 and 3 ($p < .01$) regarding Letter + word prediction. In addition, there were significant difference between Letter + word prediction and No prediction in session 1 ($p < .01$), between Letter + word prediction and No prediction in session 2 and 3 ($p < .01$), and between Letter + word prediction and Letter prediction in session 2 and 3 ($p < .01$).

6.4.2 Error rates

Corrected error rate (CER) reflects how frequently users correct the unintended or false typed characters using backspace. The Letter + word prediction started with a corrected error rate of 0.08 ($SD = 0.06$) in session 1 to 0.07 ($SD = 0.04$) in session 3. The Letter prediction started with a corrected error rate of 0.13 ($SD = 0.09$) in session 1 to 0.13 ($SD = 0.06$) in session 3. The No prediction started with a corrected error rate of 0.14 ($SD = 0.09$) in session 1 to 0.11 ($SD = 0.07$) in session 3.

A Shapiro-Wilk normality test shown that the corrected error rate was normally distributed ($p > .05$). The Mauchly's Test for Sphericity was not significant regarding the corrected error rate ($p > .05$). A significant main effect of Prediction mode was found $F(2, 22) = 7.17, p < .01$. And there was no significant effect of Session on corrected error rate $F(2, 22) = 0.76, p = .48$. The interaction between these terms was not significant $F(4, 44) = 0.67, p = .62$. From the post-hoc test results, the results found that there were significant differences ($p < .001$) between Letter + word prediction and Letter prediction, and between Letter + word prediction and No prediction. But no difference was found between Letter prediction and No prediction.

Uncorrected error rate (UER) reports how many errors left in the typed sentence. The Letter + word prediction started with a uncorrected error rate of 0.17 ($SD = 0.21$) in session 1 to 0.04 ($SD = 0.06$) in session 3. The Letter prediction started with a uncorrected error rate of 0.08 ($SD = 0.09$) in session 1 to 0.1 ($SD = 0.14$) in session 3. The No prediction started with a uncorrected error rate of 0.08 ($SD = 0.13$) in session 1 to 0.05 ($SD = 0.1$) in session 3.

A Shapiro-Wilk normality test shown that the uncorrected error rate was not normally distributed ($p < .05$), thus, a two-way ART RM-ANOVA was performed. The results show a significant main effect of Prediction mode on uncorrected error rate ($F(2, 88) = 3.45, p < .05$). A significant effect of Session on uncorrected error rate was also found ($F(2, 88) = 3.88, p < .05$). There was no significant interaction between Prediction mode and Session on uncorrected error rate ($F(4, 88) = 1.94, p = .11$). From the post-hoc test results, the results found that there were significant differences between Letter + word prediction and No prediction ($p < .05$) and between session 1 and 3 ($p < .05$).

6.4.3 Keystroke savings (KS)

Keystroke savings measures the percentage of key saving with language model compared to letter-by-letter text entry. Only the data for the Letter + word prediction was analyzed. The Letter + word prediction started with a KS of 0.38 ($SD = 0.2$) in session 1 to 0.44 ($SD = 0.08$) in session 3.

A Shapiro-Wilk normality test shown that the KS was normally distributed ($p > .05$). Since Mauchly's test indicated a violation of sphericity ($p < .05$), the Greenhouse–Geisser correction was used to correct for violation of the assumption of sphericity. The one-way ANOVA indicated that Session had no significant influence on KS ($F(1.27, 13.95) = 1.07, p = .34$).

Table 6.3: Mean and standard deviation for keystroke savings for interface with Letter + word prediction.

Keystroke savings	Session 1	Session 2	Session 3
Mean (M)	0.38	0.44	0.44
Stand deviation (SD)	0.20	0.08	0.08

6.4.4 Subjective evaluation

After completing the task for each technique in each session, participants were asked to fill the NASA TLX questionnaire. Figure 6.5 illustrates the mean scores of the NASA Task Load Index in each dimension. Since these rating results were non-normal, the statistical significance tests were performed using a two-way ART RM-ANOVA.

NASA Task Load Index

Mental demand

The Letter + word prediction started with a mental demand score of 31.1 (SD = 18.0) in session 1 to 16.8 (SD = 19.1) in session 3. The Letter prediction started with a mental demand score of 28.4 (SD = 17.4) in session 1 to 39.6 (SD = 30.8) in session 3. The No prediction started with a mental demand score of 34.6 (SD = 18.4) in session 1 to 25.8 (SD = 18.5) in session 3.

The results show a significant main effect of Prediction mode on mental demand ($F(2, 88) = 3.46, p < .05$). No significant effect of Session on mental demand was also found ($F(2, 88) = 1.5, p = .22$). There was a significant interaction between Prediction mode and Session on mental demand ($F(4, 88) = 3.18, p < .05$). From the post-hoc test results, the results found that the mental demand decreased significantly between session 1 and 3 ($p < .001$) for Letter + word prediction, and decreased also significantly between Letter + word prediction and Letter prediction in session 3 ($p < .001$). Besides, significant differences were found between Letter + word prediction in session 3 and No prediction in session 1 ($p < .001$), and between Letter + word prediction in session 3 and No prediction in session 2 ($p < .05$).

Physical demand

The Letter + word prediction started with a physical demand score of 32.4 (SD = 17.9) in session 1 to 19.2 (SD = 19.4) in session 3. The Letter prediction started with a physical demand score of 33.1 (SD = 16.6) in session 1 to 41.2 (SD = 29.0) in session 3. The No

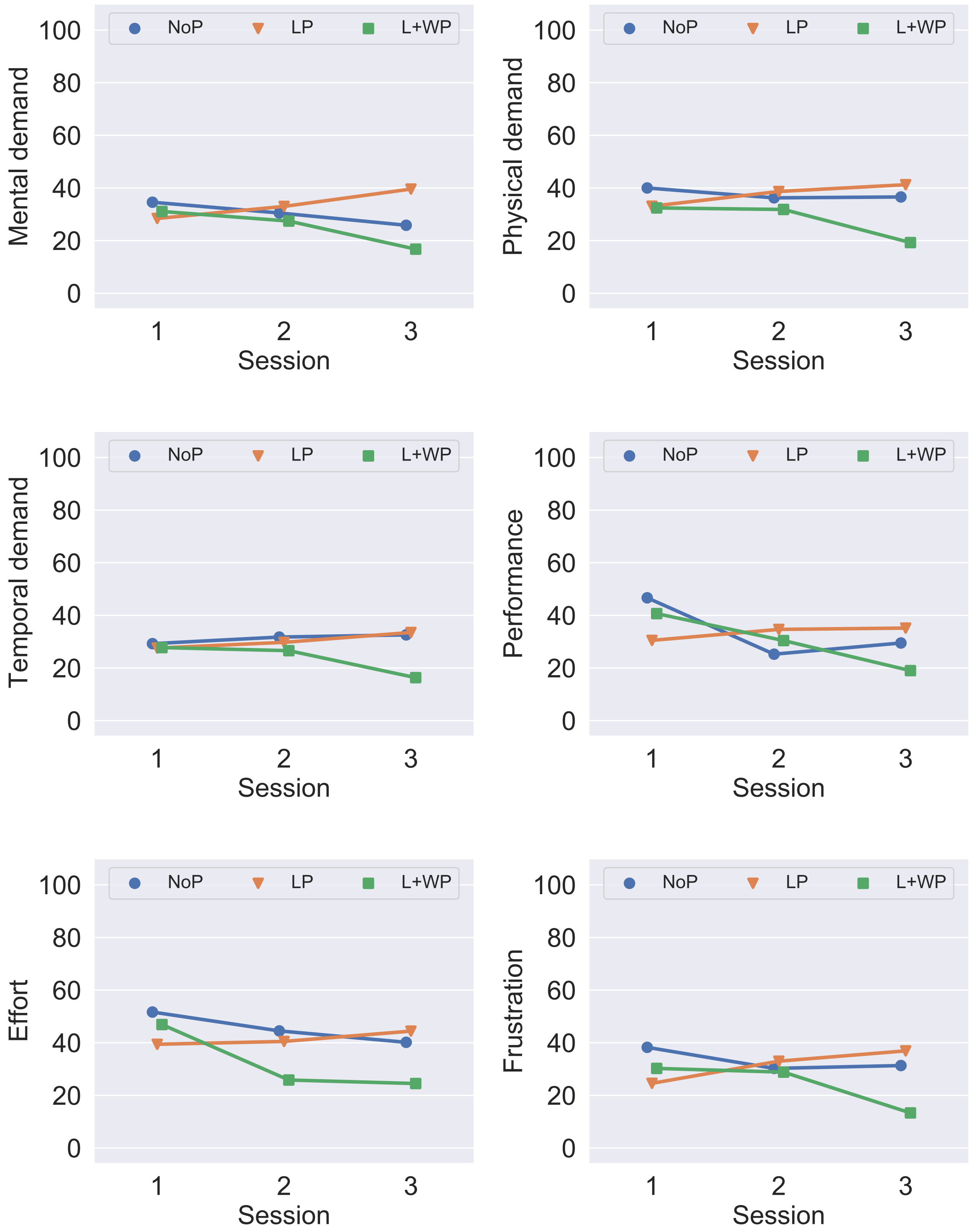


Figure 6.5: Mean of NASA TLX results over sessions, blue circle represents No prediction, orange triangle stands for Letter prediction, and green square indicates Letter + word prediction.

prediction started with a physical demand score of 40 ($SD = 21.3$) in session 1 to 36.6 ($SD = 25.1$) in session 3.

The results show a significant main effect of Prediction mode on physical demand ($F(2, 88) = 6.52, p < .001$). No significant effect of Session on physical demand was also found ($F(2, 88) = 0.85, p = .43$). There was no significant interaction between Prediction mode and Session on physical demand ($F(4, 88) = 1.43, p = .23$). The post-hoc test results show that there were significant declines between Letter + word prediction and Letter prediction ($p < .05$), and between Letter + word prediction and No prediction ($p < .05$).

Temporal demand

The Letter + word prediction started with a temporal demand score of 27.8 ($SD = 23.4$) in session 1 to 16.3 ($SD = 20.3$) in session 3. The Letter prediction started with a temporal demand score of 27.6 ($SD = 20.1$) in session 1 to 33.5 ($SD = 24.6$) in session 3. The No prediction started with a temporal demand score of 29.2 ($SD = 17.1$) in session 1 to 32.6 ($SD = 25.3$) in session 3.

The results show a significant main effect of Prediction mode on temporal demand ($F(2, 88) = 3.6, p < .05$). No significant effect of Session on temporal demand was also found ($F(2, 88) = 0.32, p = .72$). There was no significant interaction between Prediction mode and Session on temporal demand ($F(4, 88) = 1.24, p = .3$). The post-hoc test results show that there was a significant decline between Letter + word prediction and No prediction ($p < .05$).

Performance

The Letter + word prediction started with a overall performance score of 40.7 ($SD = 30.3$) in session 1 to 19 ($SD = 27.7$) in session 3. The Letter prediction started with a overall performance score of 30.5 ($SD = 19.7$) in session 1 to 35.2 ($SD = 27.8$) in session 3. The No prediction started with a overall performance score of 46.7 ($SD = 24.3$) in session 1 to 29.5 ($SD = 17.2$) in session 3.

The results show that there was no significant main effect of Prediction mode on performance ($F(2, 88) = 0.88, p = .41$). A significant effect of Session on performance was also found ($F(2, 88) = 3.3, p < .05$). There was no significant interaction between Prediction mode and Session on performance ($F(4, 88) = 1.87, p = .12$). The post-hoc test results show that participants felt more successful in session 3 than 1 ($p < .05$).

Effort

The Letter + word prediction started with a effort score of 46.9 ($SD = 24.6$) in session

1 to 24.5 ($SD = 23.9$) in session 3. The Letter prediction started with a effort score of 39.4 ($SD = 21.7$) in session 1 to 44.4 ($SD = 32.3$) in session 3. The No prediction started with a effort score of 51.7 ($SD = 23.3$) in session 1 to 40.2 ($SD = 28.5$) in session 3.

The results show a significant main effect of Prediction mode on effort ($F(2, 88) = 5.26, p < .001$). A significant effect of Session on effort was also found ($F(2, 88) = 4.29, p < .05$). There was no significant interaction between Prediction mode and Session on effort ($F(4, 88) = 1.78, p = .14$). The post-hoc test results show that there were significant declines between session 1 and 2 ($p < .05$), between session 1 and 3 ($p < .05$). Besides, the effort score for Letter + word prediction was significantly lower than No prediction ($p < .05$).

Frustration

The Letter + word prediction started with a frustration score of 30.2 ($SD = 23.5$) in session 1 to 13.3 ($SD = 17.6$) in session 3. The Letter prediction started with a frustration score of 24.6 ($SD = 21.1$) in session 1 to 36.9 ($SD = 30.0$) in session 3. The No prediction started with a frustration score of 38.2 ($SD = 31.4$) in session 1 to 31.3 ($SD = 23.2$) in session 3.

The results show no significant main effect of Prediction mode on frustration ($F(2, 88) = 2.65, p = .08$). No significant effect of Session on frustration was also found ($F(2, 88) = 0.15, p = .86$). There was no significant interaction between Prediction mode and Session on frustration ($F(4, 88) = 1.35, p = .26$). The results of statistical significance are summarized in Table 6.4.

Interview

In addition to quantitative evaluation, participants' qualitative feedbacks regarding the moving speed, searching time, word prediction, and improvement suggestions were also collected.

Perceived moving speed

More than half of the participants found the moving speed is appropriate (P1, P3-6, P9-10). And four participants (P2, P8, P11-12) thought the speed is too fast, and one of them (P2) reported that “the moving speed is too fast only at the beginning of the experiment, and I can adapt to this speed after a period of practice." Only one participant (P7) considered that the speed is too fast.

Table 6.4: Study 3 - Statistical significance of main and interaction effects regarding performance and subjective measures.

	Main effect of prediction mode	Main effect of session	Interaction effects
Typing performance:			
Words per minutes	$p < .001$	$p < .01$	$p < .05$
Corrected error rate	$p < .01$	n.s[1]	n.s.
Uncorrected error rate	$p < .05$	$p < .05$	n.s.
Keystroke savings[2]	-	n.s.	-
NASA-TLX:			
Mental demand	$p < .05$	n.s.	$p < .05$
Physical demand	$p < .001$	n.s.	n.s.
Temporal demand	$p < .05$	n.s.	n.s.
Performance	n.s.	$p < .05$	n.s.
Effort	$p < .001$	$p < .05$	n.s.
Frustration	n.s.	n.s.	n.s.

[1] 'n.s.': not significant

[2] Only applied for data from letter and word prediction condition.

Searching time

Four participants (P2, P4, P6, P12) felt that they had enough time to search the desired letter. Other eight participants (P1, P3, P5, P7-11) reported that they did not have enough time only at the beginning, after practicing a few phrases they could have time to find the desired letter.

Views on word prediction

Participants were asked three questions regarding their views on word prediction. The first question about the degree of difficulty to understand the design that arrows in different directions corresponding to the predicted word at the corresponding position. All participants considered that it is very easy to understand, and one of them (P10) suggested adding color for different predicted words. The second question is about whether each arrow can be selected successfully. Six participants (P7-12) thought that they can select the arrows successfully, and one of them (P7) reported that arrows were easier to be selected than letters. On the other hand, there are six participants (P1-P6) who considered that predictive words can easily be chosen by mistake. The third question is whether the word prediction is helpful. Almost all users responded that word prediction function is very helpful. P9 appreciated the word prediction, special for lengthy words.

Suggestions

Five participants (P6, P7, P9, P11-12) suggested separating the space and delete keys because misselections often occur between these two keys. Those misselections make them nervous, which leads to more misselections. Four participants (P2-4, P6) proposed that the sensitivity of the system could be reduced to avoid unintentional selection. P1 commented that "Hope that there can be an automatic calibration mechanism to make the typing system more and more accurate".

6.5 Discussion

In this study, three prediction modes were compared, the results were analyzed regarding the text entry performance, NASA Task Load Index, and qualitative feedback.

For the first research question, the results show that there was a significant improvement in typing speed by adding the word predictions. With the provision of the word prediction function, the typing speed was faster than Letter prediction and No prediction over sessions. The typing speed of Letter + word prediction increased greatly from session 1 to session 3, whereas the typing speed of Letter prediction and No prediction grew more modestly. And the typing speed of Letter prediction and No prediction were similar over sessions in this user study. This suggests that participants were able to learn the novel interface after a short practice, and Letter + word prediction also exhibits a more efficient typing speed. However, participants need more time to get familiar with the interface with word predictions to reach relatively stable text entry speed.

The results also indicate that the corrected error rate of Letter + word prediction was significantly lower than Letter prediction and No prediction. Using word prediction reduces the number of keystrokes required for users to enter phrases of a certain length, and at the same time, reduces the possibility of unintentional input to decrease the number of corrections correspondingly.

For uncorrected error rate, there was a significant difference in prediction modes, mainly between Letter + word prediction and No prediction. In this experiment, participants were told that they don't have to correct the errors for word prediction which was incorrectly entered, because this study wants to know whether the word prediction can be selected correctly. The results found that the uncorrected error rate is quite high for Letter + word prediction in sessions 1 and 2 compared to Letter prediction and No prediction. However, the uncorrected error rate of Letter + word prediction decreased gradually over sessions, The gap with No prediction was already very small in session 3, and even lower than Letter prediction. For beginners, the rate of unintentional selection of word prediction is relatively high, thus, the uncorrected error rate is relatively high in the first two sessions. However, those errors were significantly reduced with short practice sessions. Besides, although there was no significant difference on KS over session in Letter + word prediction, the trend indicates that somewhat more keystrokes were saved from sessions 1 to 2.

For the second research question, the results of the NASA TLX test reveals that Letter + word prediction had a lower score than Letter prediction and No prediction on the mental, physical, temporal demand, and effort dimensions. Those differences were substantially increased from sessions 1 to 3. And participants evaluated that the performance was getting better and the efforts devoted to accomplishing the task were decreased over sessions.

Those results also confirm the findings of the text entry performance. When the user is not familiar with the text entry system, the participants' cognitive load is relatively high, and they tend to focus on the task of finding letters, thereby ignoring the update of the predicted words. After participants became familiar with the interaction, they could find the correct word prediction more proficiently and quickly, and therefore relied more on the word prediction.

For the last research question, the results show that adding word prediction can significantly improve typing performance and reduce workload level. However, not all designs embedded language models will bring significant improvements. MacKenzie and Zhang (2008) introduced weight based on language model to assist the gaze typing, and no significant improvement was found compared with baseline. In this study, different from what was expected, the typing interface containing letter prediction did not show a better typing performance and lower workload index than No prediction, and even worse in some situations. For example, in the mental demand, the scores of Letter + word prediction and No prediction were slightly decreasing with practice, however, the score of Letter prediction is gradually increasing. Even in the session 3, there was a significant difference between Letter + word prediction and Letter prediction on mental demand. Two possible reasons are speculated. First, similar to the findings of several studies, the control of rhythm is very necessary to be taken into account (Majaranta et al., 2006; Mott et al., 2017). Maintaining the rhythm of typing brings a smooth typing experience. In the Letter prediction, the moving distance of letter with high probability was reduced by almost one third. This obvious difference greatly interrupts the typing rhythm of the participants. For subsequent improvements, a more moderate approach for letter prediction is needed, such as referencing this literature (Mott et al., 2017) to keep this change (in dwell time or moving distance) within a certain range and gradually increase or decrease it depending on the predicted probability. Second, in the case of letter prediction, a letter with a high probability will be “easier" to be selected. If the letter is what should be entered, the cost of activating it can be reduced. However, if this letter is not the desired one, it also increases the probability of making a mistake. It can be expected that the prediction results of the language model will be more accurate, but for some words with lower occurrence probability, the prediction of the language model has a correspondingly lower probability. This leads to when users want to enter the letters with low probability, the letters with higher probability are easier to be unintentionally activated. Therefore, the above two points need to be considered in the follow-up work.

There are also some limitations to this study. The algorithm of detection needs to be further optimized to avoid unintentional triggers in order to achieve a better user experience. Another limitation is that the participants are not native English speakers, this may have a certain impact on the text entry efficiency. Besides, participants reported that the space and delete keys are too close to each other and often selected by mistake, so they hope to separate those two keys. Studies have indicated that cognitive workload can be measured using eye movement metrics (Duchowski et al., 2018; Kosch et al., 2018). In this study, participants also reported that when they feel relaxed, the typing speed is relative faster, but when unintentional activation happened repeatedly, typing efficiency will decrease. Thus, the position of functional keys should be addressed in subsequent iterations.

6.6 Summary

In summary, this chapter explored how to use language model to improve typing performance and reduce the workload of the eye typing interface. A user study was conducted and three prediction modes (Letter + word prediction, Letter prediction, No prediction) were compared.

From the results, this study offers important insights into providing word/letter prediction. First, the provision of word prediction significantly outperformed the other two prediction modes (Letter prediction, No prediction) in terms of typing speed and error rates. Besides, Letter + word prediction also got a better rating in terms of NASA TLX.

Although adding word prediction is a more efficient way for eye typing, participants need more time to familiarize themselves with the operation than without word prediction. For public display, where users do not have a longer time to learn the system, the number of predicted words can be appropriately reduced to simplify the operation, for example, only to offer a one-word candidate (Zeng & Roetting, 2018). In addition, the control of rhythm also needs to be considered when the activation threshold is changed based on the probability predicted by the language model. In this study, participants used the typing system only for a relatively short time, in the case of Letter + word prediction, the text entry speed has not yet reached a plateau.

In further work, a longitudinal study could be executed to provide more comprehensive results on the learning curve. To find out how long it takes for a novice to learn to reach a plateau in typing performance, and how fast can it achieve. In addition, a field study can also be performed in real public display (Freytag, 2020; Khamis, Alt, & Bulling, 2015).

Besides, there is individual difference, e. g., novice, expert, regarding the preference and typing efficiency. Thus, the parameters (such as moving speed, dwell-time) could be set to automatically adjust based on the typing performance (Špakov & Miniotas, 2004). The provision of word prediction for this hybrid eye typing interface would also be considered to further speed up the typing efficiency.

Chapter 7

Conclusions and outlook

The last chapter summarizes the conclusions that have been derived along the way and provides an outlook for future research.

7.1 Conclusions

This thesis proposed a circle layout hybrid eye typing interface, including offline data analysis, detection methods comparison, developing the interface, and improving the typing performance with a language model. Interpretations of the design issues are scattered throughout the three studies, i. e., Chapter 4 - 6. The conclusions of this thesis are discussed and summarized in the following paragraphs.

For the first research question **“What is the appropriate number of objects and movement speed for the gaze interface based on linear motion?"** The first study specifically analyzed the influence of object number (6, 8, 10, 12, or 15) and object moving speed (7.73 °/s vs. 12.89 °/s) in linear trajectory smooth pursuit gaze-based interfaces. Offline gaze data of pursuing linearly moving objects was collected. To estimate the range of parameters and comparison detection algorithms for robust detection, the gaze orientation and detection rates were compared with respect to the number of objects and the speed of object movement. Besides, three detection methods were compared regarding the correct detection rate. Results indicate that the number and speed of the displayed objects influence users' performance with the interface. The number of objects significantly affected the correct and false detection rates when selecting objects in the display. Participants performed better on the interfaces containing 6 and 8 objects than on the interfaces containing 10, 12, and 15 objects. Detection rates and orientation error were significantly influenced by the moving speed of displayed objects. Faster moving speed

(12.89 °/s) resulted in higher detection rates and smaller orientation error compared to slower moving speeds (7.73 °/s). When it comes to the comparison among detection methods, the angle-based method achieved the highest detection rate in this data set. The findings of this study can help to develop an accessible calibration-free gaze interface based on linear moving design.

To address the second research question **“How to design a dynamic eye typing interface that is easy to learn and user-friendly?"** Based on the findings of the first study, the second study presented a circle-layout eye typing interface, which is a two-stage design and combined dwell- and pursuit-based selection. The alphabet is clustered in groups of four characters, and users select a cluster by gazing at it in the first stage (all items are static) and then select the desired character by following its movement with their eyes in the second stage (only the characters in desired cluster are moving). A user study was conducted to explore the impact of auditory and visual feedback on typing performance and user experience of this novel interface. Results show that participants can quickly learn how to use the system, and an average typing speed of 4.7 WPM can be reached without lengthy training. The subjective evaluation of participants revealed that users preferred visual feedback over auditory feedback while using the interface. Results indicate that this eye typing interface can be used for walk-up-and-use interactions, as it is easily understood and robust to eye-tracking inaccuracies.

The last research question is **“How to use language models to improve the typing performance of the dynamic eye typing interface?"** The third study aimed to improve the typing performance of the circle-layout eye typing interface proposed in the second study. Language model was utilized in this study to improve the typing performance, and three interfaces were designed and compared, which are letter prediction, letter+word prediction and no prediction. The results of the user study show that participants achieved an average text entry speed of 5.48 WPM for letter+word prediction, 3.42 WPM for letter prediction, 3.39 WPM for no prediction. Typing speed of letter+word prediction is 70.3% faster than without word prediction. And almost all participants can easily understand the design for word prediction, and thought this function was very helpful. And letter+word prediction also received a better grade on the NASA task load index. However, compared to letter prediction and no prediction, participants needed more time to familiarize themselves with the interface including letter+word prediction in order to reach a plateau regarding text entry speed.

7.2 Outlook

Gaze is a potential input modality in interaction design because it provides a direct way to interact with the interface, as Jacob (1990) wrote “what you look at is what you get". The possibilities for further improvements, as well as potential areas of gaze-based application, are discussed in this section.

Robust estimation

The gaze-based interaction suffers from some shortcomings. First of all, the accuracy issue comes from the eye tracker device on the one hand, and from the characteristics of the eye (miniature eye movements) on the other hand. Most existing eye trackers require calibration before use. The calibration process is time-consuming which should be simplified to meet the need for spontaneous interaction. To achieve a more natural calibration process, future work could investigate using pursuit calibration to further improve the calibration accuracy (Drewes, Pfeuffer, & Alt, 2019; Pfeuffer, Vidal, Turner, Bulling, & Gellersen, 2013). Moreover, the detection algorithm for low spatial gaze data could also be studied, such as using a linear regression-based algorithm (Drewes, Khamis, & Alt, 2019), profile matching and 2D correlation (Velloso, Coutinho, Kurauchi, & Morimoto, 2018), to allow the interface to accommodate more targets and make the detection more robust.

Gaze cursor in VR/AR

In human-computer interaction, the way of interacting with the interface varies with the device (see Figure 7.1). The user controls the computer’s graphical user interface through the computer mouse. When it comes to mobile device, such as tablet computer and smartphone, touch is most frequently used to control those devices. As technology advances and devices are updated, more new ways of interaction are emerging and being accepted by users, such as gesture, voice control. The past decade has seen the rapid development of head-mounted display (HMDs) for augmented or virtual reality. Hand gesture and gaze cursor are becoming potential control methods for HMDs. Since the tracking accuracy is reported that ranged from 2-4.5° visual angle in VR device (Rajanna & Hansen, 2018), the size of the character keys should be designed to be large enough to prevent unintentional activation. In this case, the eye typing interface proposed in this thesis could be one potential solution for entering text in head-mounted display, which has a relatively low requirement for the eye-tracking accuracy.

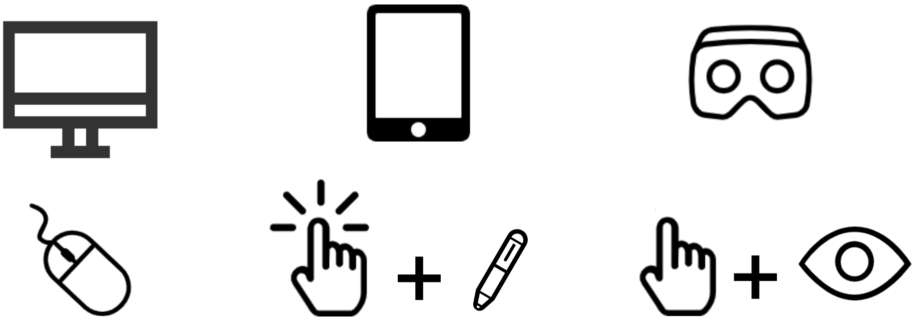


Figure 7.1: Pointers on the display for different device.

Multimodal interaction

Another possible area of future research is to explore the combination with other input modalities to make up for each other's shortcomings to get more efficient and natural user experiences. For example, adding head movement (Feng, Zou, Kurauchi, Morimoto, & Betke, 2021; Sidenmark, Mardanbegi, Gomez, Clarke, & Gellersen, 2020) and touch (Kumar et al., 2020).

Gaze estimation based on webcam

Additionally, since the eye tracker based gaze estimation is subject to the limitation in the detectable tracking range, a webcam-based gaze estimation method could be used to increase the available area for gaze estimation. Compared with the eye tracker used in this study (Tobii EyeX), a webcam-based gaze estimation extends the maximum tolerance distance range from 75 cm to 180 cm (X. Zhang, Sugano, & Bulling, 2019). In addition, it would considerably reduce the price of equipment and also broaden the usage scenarios, such as *Eyetell* (Bafna et al., 2021).

References

Abdrabou, Y., Mostafa, M., Khamis, M., & Elmougy, A. (2019). Calibration-free text entry using smooth pursuit eye movements. In *Proceedings of the 11th acm symposium on eye tracking research & applications.* New York, NY, USA: Association for Computing Machinery. Retrieved from `https://doi.org/10.1145/3314111.3319838` doi: 10.1145/3314111.3319838

Almoctar, H., Irani, P., Peysakhovich, V., & Hurter, C. (2018). Path word: A multimodal password entry method for ad-hoc authentication based on digits' shape and smooth pursuit eye movements. In *Proceedings of the 20th acm international conference on multimodal interaction* (p. 268–277). New York, NY, USA: Association for Computing Machinery. Retrieved from `https://doi.org/10.1145/3242969.3243008` doi: 10.1145/3242969.3243008

Bafna, T., Bækgaard, P., & Paulin Hansen, J. P. (2021). Eyetell: Tablet-based calibration-free eye-typing using smooth-pursuit movements. In *Acm symposium on eye tracking research and applications.* New York, NY, USA: Association for Computing Machinery. Retrieved from `https://doi.org/10.1145/3448018.3458015` doi: 10.1145/3448018.3458015

Bee, N., & André, E. (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* (pp. 111–122). Retrieved from `https://doi.org/10.1007/978-3-540-69369-7_13`

Benligiray, B., Topal, C., & Akinlar, C. (2019). Slicetype: fast gaze typing with a merging keyboard. *Journal on Multimodal User Interfaces*, *13*(4), 321–334. Retrieved from `https://doi.org/10.1007/s12193-018-0285-z`

Bi, X., Li, Y., & Zhai, S. (2013). Ffitts law: Modeling finger touch with fitts' law. In *Proceedings of the sigchi conference on human factors in computing systems* (p. 1363–1372). New York, NY, USA: Association for Computing Machinery. Retrieved from `https://doi.org/10.1145/2470654.2466180` doi: 10.1145/

2470654.2466180

Blankertz, B., Krauledat, M., Dornhege, G., Williamson, J., Murray-Smith, R., & Müller, K.-R. (2007). A note on brain actuated spelling with the berlin brain-computer interface. In *International conference on universal access in human-computer interaction* (pp. 759–768). Retrieved from `https://doi.org/10.1007/978-3-540-73281-5_83`

Blattgerste, J., Renner, P., & Pfeiffer, T. (2018). Advantages of eye-gaze over head-gaze-based selection in virtual and augmented reality under varying field of views. In *Proceedings of the workshop on communication by gaze interaction.* New York, NY, USA: Association for Computing Machinery. Retrieved from `https://doi.org/10.1145/3206343.3206349` doi: 10.1145/3206343.3206349

Boggs, P. T., & Rogers, J. E. (1990). Orthogonal distance regression. *Contemporary Mathematics*, *112*, 183–194.

Burke, M., & Barnes, G. (2006). Quantitative differences in smooth pursuit and saccadic eye movements. *Experimental brain research*, *175*(4), 596–608.

Carpenter, R. H. (1988). *Movements of the eyes, 2nd rev.* Pion Limited.

Collewijn, H., & Tamminga, E. P. (1984). Human smooth and saccadic eye movements during voluntary pursuit of different target motions on different backgrounds. *The Journal of physiology*, *351*(1), 217–250. Retrieved from `https://doi.org/10.1113/jphysiol.1984.sp015242`

Cymek, D. H., Venjakob, A. C., Ruff, S., Lutz, O. H.-M., Hofmann, S., & Roetting, M. (2014). Entering pin codes by smooth pursuit eye movements. *Journal of Eye Movement Research*, *7*(4), 1. Retrieved from `https://doi.org/10.16910/jemr.7.4.1`

de Brouwer, S., Yuksel, D., Blohm, G., Missal, M., & Lefèvre, P. (2002). What triggers catch-up saccades during visual tracking? *Journal of neurophysiology*, *87*(3), 1646–1650. Retrieved from `https://doi.org/10.1152/jn.00432.2001`

de Hemptinne, C., Lefevre, P., & Missal, M. (2006). Influence of cognitive expectation on the initiation of anticipatory and visual pursuit eye movements in the rhesus monkey. *Journal of neurophysiology*, *95*(6), 3770–3782. Retrieved from `https://doi.org/10.1152/jn.00007.2006`

de Luca, A., Weiss, R., & Drewes, H. (2007). Evaluation of eye-gaze interaction methods for security enhanced pin-entry. In *Proceedings of the 19th australasian conference on computer-human interaction: Entertaining user interfaces* (pp. 199–202). New York, NY, USA: ACM. Retrieved from `http://doi.acm.org/10.1145/1324892.1324932` doi: 10.1145/1324892.1324932

Diaz-Tula, A., & Morimoto, C. H. (2016). Augkey: Increasing foveal throughput in eye typing with augmented keys. In *Proceedings of the 2016 chi conference on human factors in computing systems* (p. 3533–3544). New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi.org/10.1145/2858036.2858517 doi: 10.1145/2858036.2858517

Dodge, R. (1903). Five types of eye movement in the horizontal meridian plane of the field of regard. *American journal of physiology-legacy content*, *8*(4), 307–329. Retrieved from https://doi.org/10.1152/ajplegacy.1903.8.4.307

Drewes, H., Khamis, M., & Alt, F. (2018). Smooth pursuit target speeds and trajectories. In *Proceedings of the 17th international conference on mobile and ubiquitous multimedia* (pp. 139–146). New York, NY, USA: ACM. Retrieved from http://doi.acm.org/10.1145/3282894.3282913 doi: 10.1145/3282894.3282913

Drewes, H., Khamis, M., & Alt, F. (2019). Dialplates: Enabling pursuits-based user interfaces with large target numbers. In *Proceedings of the 18th international conference on mobile and ubiquitous multimedia.* New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi.org/10.1145/3365610.3365626 doi: 10.1145/3365610.3365626

Drewes, H., Pfeuffer, K., & Alt, F. (2019). Time- and space-efficient eye tracker calibration. In *Proceedings of the 11th acm symposium on eye tracking research & applications.* New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi.org/10.1145/3314111.3319818 doi: 10.1145/3314111.3319818

Drewes, H., & Schmidt, A. (2007). Interacting with the computer using gaze gestures. In *Ifip conference on human-computer interaction* (pp. 475–488). Retrieved from https://doi.org/10.1007/978-3-540-74800-7_43

Duchowski, A. T. (2017). *Eye tracking methodology: Theory and practice (3rd ed.).* Springer.

Duchowski, A. T., Krejtz, K., Krejtz, I., Biele, C., Niedzielska, A., Kiefer, P., ... Giannopoulos, I. (2018). The index of pupillary activity: Measuring cognitive load vis-à-vis task difficulty with pupil oscillation. In *Proceedings of the 2018 chi conference on human factors in computing systems* (p. 1–13). New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi.org/10.1145/3173574.3173856 doi: 10.1145/3173574.3173856

Eid, M. A., Giakoumidis, N., & El Saddik, A. (2016). A novel eye-gaze-controlled wheelchair system for navigating unknown environments: Case study with a person with als. *IEEE Access*, *4*, 558-573. Retrieved from 10.1109/ACCESS.2016.2520093

Ester, M., Kriegel, H.-P., Sander, J., & Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In *In proceedings of the second international conference on knowledge discovery and data mining* (p. 226–231). AAAI Press.

Esteves, A., Velloso, E., Bulling, A., & Gellersen, H. (2015). Orbits: Gaze interaction for smart watches using smooth pursuit eye movements. In *Proceedings of the 28th annual acm symposium on user interface software & technology* (p. 457–466). New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi.org/10.1145/2807442.2807499 doi: 10.1145/2807442.2807499

Esteves, A., Verweij, D., Suraiya, L., Islam, R., Lee, Y., & Oakley, I. (2017). Smoothmoves: Smooth pursuits head movements for augmented reality. In *Proceedings of the 30th annual acm symposium on user interface software and technology* (p. 167–178). New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi.org/10.1145/3126594.3126616 doi: 10.1145/3126594.3126616

Feit, A. M., Williams, S., Toledo, A., Paradiso, A., Kulkarni, H., Kane, S., & Morris, M. R. (2017). Toward everyday gaze input: Accuracy and precision of eye tracking and implications for design. In *Proceedings of the 2017 chi conference on human factors in computing systems* (p. 1118–1130). New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi.org/10.1145/3025453.3025599 doi: 10.1145/3025453.3025599

Feng, W., Zou, J., Kurauchi, A., Morimoto, C. H., & Betke, M. (2021). Hgaze typing: Head-gesture assisted gaze typing. In *Acm symposium on eye tracking research and applications.* New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi.org/10.1145/3448017.3457379 doi: 10.1145/3448017.3457379

Fisher, D. F., Monty, R. A., & Senders, J. W. (1981). *Eye movements: Cognition and visual perception (psychology library editions: Perception)* (Vol. 8). Routledge.

Freytag, S.-C. (2020). Sweet pursuit: User acceptance and performance of a smooth pursuit controlled candy dispensing machine in a public setting. In *Acm symposium on eye tracking research and applications.* New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi.org/10.1145/3379156.3391356 doi: 10.1145/3379156.3391356

Freytag, S.-C., Venjakob, A. C., & Ruff, S. (2017). Applicability of smooth-pursuit based gaze interaction for older users. In *Proceedings of the cogain symposium. cogain* (Vol. 17).

Fukushima, K., Fukushima, J., Warabi, T., & Barnes, G. R. (2013). Cognitive processes involved in smooth pursuit eye movements: behavioral evidence, neural substrate and clinical correlation. *Frontiers in systems neuroscience*, *7*, 4. Retrieved from `https://doi.org/10.3389/fnsys.2013.00004`

Garay-Vitoria, N., & Abascal, J. (2006). Text prediction systems: a survey. *Universal Access in the Information Society*, *4*, 188–203. Retrieved from `https://doi.org/10.1007/s10209-005-0005-9`

Garay-Vitoria, N., & González-Abascal, J. (1997). Intelligent word-prediction to enhance text input rate (a syntactic analysis-based word-prediction aid for people with severe motor and speech disability). In *Proceedings of the 2nd international conference on intelligent user interfaces* (p. 241–244). New York, NY, USA: Association for Computing Machinery. Retrieved from `https://doi.org/10.1145/238218.238333` doi: 10.1145/238218.238333

Göbel, F., Kiefer, P., Giannopoulos, I., Duchowski, A. T., & Raubal, M. (2018). Improving map reading with gaze-adaptive legends. In *Proceedings of the 2018 acm symposium on eye tracking research & applications* (pp. 29:1–29:9). New York, NY, USA: ACM. Retrieved from `http://doi.acm.org/10.1145/3204493.3204544` doi: 10.1145/3204493.3204544

Hansen, D. W., Skovsgaard, H. H. T., Hansen, J. P., & Møllenbach, E. (2008). Noise tolerant selection by gaze-controlled pan and zoom in 3d. In *Proceedings of the 2008 symposium on eye tracking research & applications* (p. 205–212). New York, NY, USA: Association for Computing Machinery. Retrieved from `https://doi.org/10.1145/1344471.1344521` doi: 10.1145/1344471.1344521

Hansen, J. P., Johansen, A. S., Hansen, D. W., Itoh, K., & Mashino, S. (2003a). Command without a click: Dwell time typing by mouse and gaze selections. In *Interact* (Vol. 3, pp. 121–128).

Hansen, J. P., Johansen, A. S., Hansen, D. W., Itoh, K., & Mashino, S. (2003b). Language technology in a predictive, restricted on-screen keyboard with ambiguous layout for severely disabled people. In *Proceedings of eacl workshop on language modeling for text entry methods* (p. 59–66).

Hart, S. G., & Staveland, L. E. (1988). Development of nasa-tlx (task load index): Results of empirical and theoretical research. In *Advances in psychology* (Vol. 52, pp. 139–

183). Elsevier. Retrieved from https://doi.org/10.1016/S0166-4115(08)62386-9

Holmqvist, K., Nyström, M., Andersson, R., Dewhurst, R., Jarodzka, H., & Van de Weijer, J. (2011). *Eye tracking: A comprehensive guide to methods and measures*. OUP Oxford.

Huckauf, A., & Urbina, M. (2007). Gazing with peye: New concepts in eye typing. In *Proceedings of the 4th symposium on applied perception in graphics and visualization* (p. 141). New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi.org/10.1145/1272582.1272618 doi: 10.1145/1272582.1272618

Huckauf, A., & Urbina, M. H. (2008). Gazing with peyes: Towards a universal input for various applications. In *Proceedings of the 2008 symposium on eye tracking research & applications* (p. 51–54). New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi.org/10.1145/1344471.1344483 doi: 10.1145/1344471.1344483

Hwang, C.-S., Weng, H.-H., Wang, L.-F., Tsai, C.-H., & Chang, H.-T. (2014). An eye-tracking assistive device improves the quality of life for als patients and reduces the caregivers' burden. *Journal of motor behavior*, *46*(4), 233–238. Retrieved from https://doi.org/10.1080/00222895.2014.891970

Jacob, R. J. K. (1990). 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* (p. 11–18). New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi.org/10.1145/97243.97246 doi: 10.1145/97243.97246

Jacob, R. J. K. (1995). Eye tracking in advanced interface design. In *Virtual environments and advanced interface design* (p. 258–288). USA: Oxford University Press, Inc.

Kangas, J., Akkil, D., Rantala, J., Isokoski, P., Majaranta, P., & Raisamo, R. (2014). Gaze gestures and haptic feedback in mobile devices. In *Proceedings of the sigchi conference on human factors in computing systems* (p. 435–438). New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi.org/10.1145/2556288.2557040 doi: 10.1145/2556288.2557040

Kangas, J., Špakov, O., Isokoski, P., Akkil, D., Rantala, J., & Raisamo, R. (2016). Feedback for smooth pursuit gaze tracking based control. In *Proceedings of the 7th augmented human international conference 2016.* New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi.org/10.1145/2875194.2875209 doi: 10.1145/2875194.2875209

Katsini, C., Abdrabou, Y., Raptis, G. E., Khamis, M., & Alt, F. (2020). The role of eye gaze in security and privacy applications: Survey and future hci research directions. In *Proceedings of the 2020 chi conference on human factors in computing systems* (p. 1–21). New York, NY, USA: Association for Computing Machinery. Retrieved from `https://doi.org/10.1145/3313831.3376840` doi: 10.1145/3313831.3376840

Ke, S. R., Lam, J., Pai, D. K., & Spering, M. (2013). Directional asymmetries in human smooth pursuit eye movements. *Investigative ophthalmology & visual science*, *54*(6), 4409–4421. Retrieved from `https://doi.org/10.1167/iovs.12-11369`

Khamis, M., Alt, F., & Bulling, A. (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* (p. 863–872). New York, NY, USA: Association for Computing Machinery. Retrieved from `https://doi.org/10.1145/2800835.2804335` doi: 10.1145/2800835.2804335

Khamis, M., Oechsner, C., Alt, F., & Bulling, A. (2018). Vrpursuits: Interaction in virtual reality using smooth pursuit eye movements. In *Proceedings of the 2018 international conference on advanced visual interfaces* (pp. 18:1–18:8). New York, NY, USA: ACM. Retrieved from `http://doi.acm.org/10.1145/3206505.3206522` doi: 10.1145/3206505.3206522

Koester, H. H., & Levine, S. (1997). Keystroke-level models for user performance with word prediction. *Augmentative and Alternative Communication*, *13*(4), 239–257. Retrieved from `https://doi.org/10.1080/07434619712331278068` doi: 10.1080/07434619712331278068

Koester, H. H., & Levine, S. (1998). Model simulations of user performance with word prediction. *Augmentative and Alternative Communication*, *14*(1), 25–36. Retrieved from `https://doi.org/10.1080/07434619812331278176` doi: 10.1080/07434619812331278176

Köpsel, A., Majaranta, P., Isokoski, P., & Huckauf, A. (2016). Effects of auditory, haptic and visual feedback on performing gestures by gaze or by hand. *Behaviour & Information Technology*, *35*(12), 1044–1062. Retrieved from `https://doi.org/10.1080/0144929X.2016.1194477` doi: 10.1080/0144929X.2016.1194477

Kosch, T., Hassib, M., Woundefinedniak, P. W., Buschek, D., & Alt, F. (2018). Your eyes tell: Leveraging smooth pursuit for assessing cognitive workload. In *Proceedings of the 2018 chi conference on human factors in computing systems* (p. 1–13). New

York, NY, USA: Association for Computing Machinery. Retrieved from https://doi.org/10.1145/3173574.3174010 doi: 10.1145/3173574.3174010

Kristensson, P. O., & Vertanen, K. (2012). The potential of dwell-free eye-typing for fast assistive gaze communication. In *Proceedings of the symposium on eye tracking research and applications* (p. 241–244). New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi.org/10.1145/2168556.2168605 doi: 10.1145/2168556.2168605

Krukowski, A. E., & Stone, L. S. (2005). Expansion of direction space around the cardinal axes revealed by smooth pursuit eye movements. *Neuron*, *45*(2), 315–323. Retrieved from https://doi.org/10.1016/j.neuron.2005.01.005

Kumar, C., Hedeshy, R., MacKenzie, I. S., & Staab, S. (2020). Tagswipe: Touch assisted gaze swipe for text entry. In *Proceedings of the 2020 chi conference on human factors in computing systems* (p. 1–12). New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi.org/10.1145/3313831.3376317 doi: 10.1145/3313831.3376317

Kurauchi, A., Feng, W., Joshi, A., Morimoto, C., & Betke, M. (2016). Eyeswipe: Dwell-free text entry using gaze paths. In *Proceedings of the 2016 chi conference on human factors in computing systems* (p. 1952–1956). New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi.org/10.1145/2858036.2858335 doi: 10.1145/2858036.2858335

LetterFrequency. (2021, August). *Computer qwerty keyboard key frequency.* http://letterfrequency.org/.

Levenshtein, V. I. (1966). Binary codes capable of correcting deletions, insertions, and reversals. In *Soviet physics doklady* (Vol. 10, pp. 707–710).

Lisberger, S. G. (2015). Visual guidance of smooth pursuit eye movements. *Annual review of vision science*, *1*, 447–468. Retrieved from https://doi.org/10.1146/annurev-vision-082114-035349

Lisberger, S. G., Morris, E., & Tychsen, L. (1987). Visual motion processing and sensory-motor integration for smooth pursuit eye movements. *Annual review of neuroscience*, *10*(1), 97–129. Retrieved from https://doi.org/10.1146/annurev.ne.10.030187.000525

Ludvigh, E., & Miller, J. W. (1958). Study of visual acuity during the ocular pursuit of moving test objects. i. introduction. *Journal of the Optical Society of America*, *48*(11), 799–802. Retrieved from https://doi.org/10.1364/JOSA.48.000799

Lutz, O. H.-M., Venjakob, A. C., & Ruff, S. (2015). Smoovs: Towards calibration-free text entry by gaze using smooth pursuit movements. *Journal of Eye Movement Research*, *8*(1), 2. Retrieved from `https://doi.org/10.16910/jemr.8.1.2`

MacKenzie, I. S., & Soukoreff, R. W. (2003). Phrase sets for evaluating text entry techniques. In *Chi '03 extended abstracts on human factors in computing systems* (p. 754–755). New York, NY, USA: Association for Computing Machinery. Retrieved from `https://doi.org/10.1145/765891.765971` doi: 10.1145/765891.765971

MacKenzie, I. S., & Tanaka-Ishii, K. (2010). *Text entry systems: Mobility, accessibility, universality.* Elsevier.

MacKenzie, I. S., & Zhang, X. (2008). Eye typing using word and letter prediction and a fixation algorithm. In *Proceedings of the 2008 symposium on eye tracking research & applications* (p. 55–58). New York, NY, USA: Association for Computing Machinery. Retrieved from `https://doi.org/10.1145/1344471.1344484` doi: 10.1145/1344471.1344484

Majaranta, P., Ahola, U.-K., & Špakov, O. (2009). Fast gaze typing with an adjustable dwell time. In *Proceedings of the sigchi conference on human factors in computing systems* (p. 357–360). New York, NY, USA: Association for Computing Machinery. Retrieved from `https://doi.org/10.1145/1518701.1518758` doi: 10.1145/1518701.1518758

Majaranta, P., MacKenzie, I. S., Aula, A., & Räihä, K.-J. (2003). Auditory and visual feedback during eye typing. In *Chi '03 extended abstracts on human factors in computing systems* (p. 766–767). New York, NY, USA: Association for Computing Machinery. Retrieved from `https://doi.org/10.1145/765891.765979` doi: 10.1145/765891.765979

Majaranta, P., MacKenzie, I. S., Aula, A., & Räihä, K.-J. (2006). Effects of feedback and dwell time on eye typing speed and accuracy. *Universal Access in the Information Society*, *5*(2), 199–208. Retrieved from `https://doi.org/10.1007/s10209-006-0034-z`

Majaranta, P., & Räihä, K.-J. (2002). Twenty years of eye typing: Systems and design issues. In *Proceedings of the 2002 symposium on eye tracking research & applications* (pp. 15–22). New York, NY, USA: ACM. Retrieved from `http://doi.acm.org/10.1145/507072.507076` doi: 10.1145/507072.507076

Martinez-Conde, S., Macknik, S. L., & Hubel, D. H. (2004). The role of fixational eye movements in visual perception. *Nature reviews neuroscience*, *5*(3), 229–240. Retrieved from `https://doi.org/10.1038/nrn1348`

Menges, R., Kumar, C., Müller, D., & Sengupta, K. (2017). Gazetheweb: A gaze-controlled web browser. In *Proceedings of the 14th web for all conference on the future of accessible work.* New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi.org/10.1145/3058555.3058582 doi: 10.1145/3058555.3058582

Meyer, C. H., Lasker, A. G., & Robinson, D. A. (1985). The upper limit of human smooth pursuit velocity. *Vision research*, *25*(4), 561–563. Retrieved from https://doi.org/10.1016/0042-6989(85)90160-9

Morimoto, C. H., & Amir, A. (2010). Context switching for fast key selection in text entry applications. In *Proceedings of the 2010 symposium on eye-tracking research & applications* (p. 271–274). New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi.org/10.1145/1743666.1743730 doi: 10.1145/1743666.1743730

Mott, M. E., Williams, S., Wobbrock, J. O., & Morris, M. R. (2017). Improving dwell-based gaze typing with dynamic, cascading dwell times. In *Proceedings of the 2017 chi conference on human factors in computing systems* (p. 2558–2570). New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi.org/10.1145/3025453.3025517 doi: 10.1145/3025453.3025517

Neuer, E. (2020). *Improving a gaze speller based on smooth pursuit eye movements by implementing multimodal feedback.* (unpublished thesis, TU Berlin)

Nielsen, J. (1994). *Usability engineering.* Morgan Kaufmann.

Orban de Xivry, J.-J., & Lefevre, P. (2007). Saccades and pursuit: two outcomes of a single sensorimotor process. *The Journal of physiology*, *584*(1), 11–23. Retrieved from https://doi.org/10.1113/jphysiol.2007.139881

Orhan, U., Erdogmus, D., Roark, B., Oken, B., Purwar, S., E. Hild, K., ... Fried-Oken, M. (2012). Improved accuracy using recursive bayesian estimation based language model fusion in erp-based bci typing systems. In *2012 annual international conference of the ieee engineering in medicine and biology society* (p. 2497-2500). doi: 10.1109/EMBC.2012.6346471

Park, K. (2017, October). *Word prediction using convolutional neural networks.* https://github.com/Kyubyong/word_prediction.

Pedrosa, D., Pimentel, M. D. G., Wright, A., & Truong, K. N. (2015, March). Filteryedping: Design challenges and user performance of dwell-free eye typing. *ACM Trans. Access. Comput.*, *6*(1). Retrieved from https://doi.org/10.1145/2724728 doi: 10.1145/2724728

Penkar, A. M., Lutteroth, C., & Weber, G. (2012). Designing for the eye: Design parameters for dwell in gaze interaction. In *Proceedings of the 24th australian computer-human interaction conference* (pp. 479–488). New York, NY, USA: ACM. Retrieved from `http://doi.acm.org/10.1145/2414536.2414609` doi: 10.1145/2414536.2414609

Perlin, K. (1998). Quikwriting: continuous stylus-based text entry. In *Proceedings of the 11th annual acm symposium on user interface software and technology* (pp. 215–216).

Pfeuffer, K., Vidal, M., Turner, J., Bulling, A., & Gellersen, H. (2013). Pursuit calibration: Making gaze calibration less tedious and more flexible. In *Proceedings of the 26th annual acm symposium on user interface software and technology* (p. 261–270). New York, NY, USA: Association for Computing Machinery. Retrieved from `https://doi.org/10.1145/2501988.2501998` doi: 10.1145/2501988.2501998

Pi, J., Koljonen, P. A., Hu, Y., & Shi, B. E. (2020). Dynamic bayesian adjustment of dwell time for faster eye typing. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, *28*(10), 2315-2324. doi: 10.1109/TNSRE.2020.3016747

Rajanna, V., & Hansen, J. P. (2018). Gaze typing in virtual reality: Impact of keyboard design, selection method, and motion. In *Proceedings of the 2018 acm symposium on eye tracking research & applications.* New York, NY, USA: Association for Computing Machinery. Retrieved from `https://doi.org/10.1145/3204493.3204541` doi: 10.1145/3204493.3204541

Rantala, J., Majaranta, P., Kangas, J., Isokoski, P., Akkil, D., Špakov, O., & Raisamo, R. (2020). Gaze interaction with vibrotactile feedback: Review and design guidelines. *Human–Computer Interaction*, *35*(1), 1–39. Retrieved from `https://doi.org/10.1080/07370024.2017.1306444`

Rashbass, C. (1961). The relationship between saccadic and smooth tracking eye movements. *The Journal of Physiology*, *159*(2), 326. Retrieved from `https://doi.org/10.1113/jphysiol.1961.sp006811`

Reid, G. B., & Nygren, T. E. (1988). The subjective workload assessment technique: A scaling procedure for measuring mental workload. *Advances in psychology*, *52*, 185–218. doi: https://doi.org/10.1016/S0166-4115(08)62387-0

Robinson, D. A. (1965). The mechanics of human smooth pursuit eye movement. *The Journal of Physiology*, *180*(3), 569–591. Retrieved from `https://doi.org/10.1113/jphysiol.1965.sp007718`

Robinson, D. A. (1968). The oculomotor control system: A review. *Proceedings of the IEEE*, *56*(6), 1032–1049. doi: 10.1109/PROC.1968.6455

Roucoux, A., Culee, C., & Roucoux, M. (1983). Development of fixation and pursuit eye movements in human infants. *Behavioural brain research*, *10*(1), 133–139. Retrieved from https://doi.org/10.1016/0166-4328(83)90159-6

Rubio, S., Díaz, E., Martín, J., & Puente, J. M. (2004). Evaluation of subjective mental workload: A comparison of swat, nasa-tlx, and workload profile methods. *Applied psychology*, *53*(1), 61–86. Retrieved from https://doi.org/10.1111/j.1464-0597.2004.00161.x

Sengupta, K., Menges, R., Kumar, C., & Staab, S. (2019). Impact of variable positioning of text prediction in gaze-based text entry. In *Proceedings of the 11th acm symposium on eye tracking research & applications.* New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi.org/10.1145/3317956.3318152 doi: 10.1145/3317956.3318152

Sibert, L. E., & Jacob, R. J. K. (2000). Evaluation of eye gaze interaction. In *Proceedings of the sigchi conference on human factors in computing systems* (p. 281–288). New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi.org/10.1145/332040.332445 doi: 10.1145/332040.332445

Sidenmark, L., Clarke, C., Zhang, X., Phu, J., & Gellersen, H. (2020). Outline pursuits: Gaze-assisted selection of occluded objects in virtual reality. In *Proceedings of the 2020 chi conference on human factors in computing systems* (p. 1–13). New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi.org/10.1145/3313831.3376438 doi: 10.1145/3313831.3376438

Sidenmark, L., Mardanbegi, D., Gomez, A. R., Clarke, C., & Gellersen, H. (2020). Bimodalgaze: Seamlessly refined pointing with gaze and filtered gestural head movement. In *Acm symposium on eye tracking research and applications.* New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi.org/10.1145/3379155.3391312 doi: 10.1145/3379155.3391312

Speier, W., Chandravadia, N., Roberts, D., Pendekanti, S., & Pouratian, N. (2017). Online bci typing using language model classifiers by als patients in their homes. *Brain-Computer Interfaces*, *4*(1-2), 114–121. Retrieved from https://doi.org/10.1080/2326263X.2016.1252143

Stampe, D. M., & Reingold, E. M. (1995). Selection by looking: A novel computer interface and its application to psychological research. *Studies in Visual Information Processing*, *6*, 467–478. Retrieved from https://doi.org/10.1016/S0926-907X(05)80039-X

Stark, L., Vossius, G., & Young, L. R. (1962). Predictive control of eye tracking movements. *IRE Transactions on human factors in electronics*(2), 52–57. doi: 10.1109/THFE2.1962.4503342

Trnka, K., & McCoy, K. F. (2008). Evaluating word prediction: framing keystroke savings. In *Proceedings of acl-08: Hlt, short papers* (pp. 261–264).

Tsang, P. S., & Velazquez, V. L. (1996). Diagnosticity and multidimensional subjective workload ratings. *Ergonomics*, *39*(3), 358–381. Retrieved from `https://doi.org/10.1080/00140139608964470`

Tuisku, O., Majaranta, P., Isokoski, P., & Räihä, K.-J. (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* (p. 19–26). New York, NY, USA: Association for Computing Machinery. Retrieved from `https://doi.org/10.1145/1344471.1344476` doi: 10.1145/1344471.1344476

Urbina, M. H., & Huckauf, A. (2010). Alternatives to single character entry and dwell time selection on eye typing. In *Proceedings of the 2010 symposium on eye-tracking research & applications* (p. 315–322). New York, NY, USA: Association for Computing Machinery. Retrieved from `https://doi.org/10.1145/1743666.1743738` doi: 10.1145/1743666.1743738

Velloso, E., Coutinho, F. L., Kurauchi, A., & Morimoto, C. H. (2018). Circular orbits detection for gaze interaction using 2d correlation and profile matching algorithms. In *Proceedings of the 2018 acm symposium on eye tracking research & applications.* New York, NY, USA: Association for Computing Machinery. Retrieved from `https://doi.org/10.1145/3204493.3204524` doi: 10.1145/3204493.3204524

Vidal, M., Bulling, A., & Gellersen, H. (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* (p. 439–448). New York, NY, USA: Association for Computing Machinery. Retrieved from `https://doi.org/10.1145/2493432.2493477` doi: 10.1145/2493432.2493477

Špakov, O., Isokoski, P., Kangas, J., Akkil, D., & Majaranta, P. (2016). Pursuitadjuster: An exploration into the design space of smooth pursuit based widgets. In *Proceedings of the ninth biennial acm symposium on eye tracking research & applications* (p. 287–290). New York, NY, USA: Association for Computing Machinery. Retrieved from `https://doi.org/10.1145/2857491.2857526`

Špakov, O., & Miniotas, D. (2004). On-line adjustment of dwell time for target selection by gaze. In *Proceedings of the third nordic conference on human-computer* *interaction* (p. 203–206). New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi.org/10.1145/1028014.1028045 doi: 10.1145/1028014.1028045

Ward, D. J., Blackwell, A. F., & MacKay, D. J. (2000). Dasher—a data entry interface using continuous gestures and language models. In *Proceedings of the 13th annual acm symposium on user interface software and technology* (pp. 129–137).

Ward, D. J., & MacKay, D. J. (2002). Fast hands-free writing by gaze direction. *Nature*, *418*(6900), 838–838. Retrieved from https://doi.org/10.1038/418838a

Westheimer, G., & McKee, S. P. (1975). Visual acuity in the presence of retinal-image motion. *Journal of the Optical Society of America*, *65*(7), 847–850. Retrieved from https://doi.org/10.1364/JOSA.65.000847

Westheimer, G., & McKee, S. P. (1978). Stereoscopic acuity for moving retinal images. *Journal of the Optical Society of America*, *68*(4), 450–455. Retrieved from https://doi.org/10.1364/JOSA.68.000450

Wickens, C. D. (2002). Multiple resources and performance prediction. *Theoretical issues in ergonomics science*, *3*(2), 159–177. Retrieved from https://doi.org/10.1080/14639220210123806

Wobbrock, J. O., Findlater, L., Gergle, D., & Higgins, J. J. (2011). The aligned rank transform for nonparametric factorial analyses using only anova procedures. In *Proceedings of the sigchi conference on human factors in computing systems* (p. 143–146). New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi.org/10.1145/1978942.1978963

Wobbrock, J. O., Rubinstein, J., Sawyer, M. W., & Duchowski, A. T. (2008). Longitudinal evaluation of discrete consecutive gaze gestures for text entry. In *Proceedings of the 2008 symposium on eye tracking research & applications* (p. 11–18). New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi.org/10.1145/1344471.1344475 doi: 10.1145/1344471.1344475

Wolfe, J. M., & Horowitz, T. S. (2017). Five factors that guide attention in visual search. *Nature Human Behaviour*, *1*(3), 0058. Retrieved from https://doi.org/10.1038/s41562-017-0058

Young, L. R. (1971). *The control of eye movements*. Academic Press New York.

Zeng, Z., & Roetting, M. (2018). A text entry interface using smooth pursuit movements and language model. In *Proceedings of the 2018 acm symposium on eye tracking research & applications* (pp. 69:1–69:2). New York, NY, USA: ACM. Retrieved from http://doi.acm.org/10.1145/3204493.3207413 doi: 10.1145/3204493.3207413

Zeng, Z., Siebert, F. W., Venjakob, A. C., & Roetting, M. (2020). Calibration-free gaze interfaces based on linear smooth pursuit. *Journal of Eye Movement Research*, *13*(1), 3. doi: 10.16910/jemr.13.1.3

Zhang, G., Hansen, J. P., & Minakata, K. (2019). Hand- and gaze-control of telepresence robots. In *Proceedings of the 11th acm symposium on eye tracking research & applications* (pp. 70:1–70:8). New York, NY, USA: ACM. Retrieved from `http://doi.acm.org/10.1145/3317956.3318149` doi: 10.1145/3317956.3318149

Zhang, X., Sugano, Y., & Bulling, A. (2019). Evaluation of appearance-based methods and implications for gaze-based applications. In *Proceedings of the 2019 chi conference on human factors in computing systems* (pp. 1–13).

Appendix A Documents

A.1 Questionnaire for study 1

Fragebogen zu demographischen Angaben

Personencode: _______________

Bitte füllen Sie einige Angaben zu Ihrer Person aus.

Bitte geben Sie Ihr Alter an: __________

Beruf ______________________________

Geschlecht

- □ Weiblich
- □ Männlich
- □ Anderes

Haben Sie bereits Erfahrungen oder Vorkenntnisse im Bereich Blickbewegungsmessung?

- □ Ja, ____
- □ Nein

Haben Sie bereits Erfahrungen mit Blickinteraktion?

- □ Ja, ____
- □ Nein

Tragen Sie Sehhilfe?

- □ Ja, ____
- □ Nein

Sind Sie von Natur aus Rechts- oder Linkshänder?

○ Rechtshänder ○ Linkshänder

Sind Ihre Augen zur Zeit in höherem Maße beansprucht?

○ nein, gar nicht ○ eher nein ○ teils, teils ○ eher ja ○ ja, genau

Bitte geben Sie Ihre Einschätzung ab. Kreuzen Sie Zutreffendes an.

Wird sich die Anzahl der Objekte auf die Verfolgung der Ziele auswirken?

- □ Nein
- □ Wenn Ja

  Welche der folgenden Benutzerschnittstelle bevorzugen Sie? Die Benutzerschnittstelle besteht aus:

  - □ weniger als 6 bewegten Objekte
  - □ 6 bewegten Objekte
  - □ 8 bewegten Objekte
  - □ 10 bewegten Objekte
  - □ 12 bewegten Objekte
  - □ 15 bewegten Objekte
  - □ mehr als 15 bewegten Objekte

Warum? ____

Wird sich die Geschwindigkeit von bewegten Objekten auf die Verfolgung der Ziele auswirken?

- □ Nein
- □ Wenn Ja

  Welches gefällt Ihnen besser?

  - □ Schnellere Geschwindigkeit
  - □ Langsamere Geschwindigkeit

Warum? ____

Wird sich die Größe von bewegten Objekten auf die Verfolgung der Ziele auswirken?

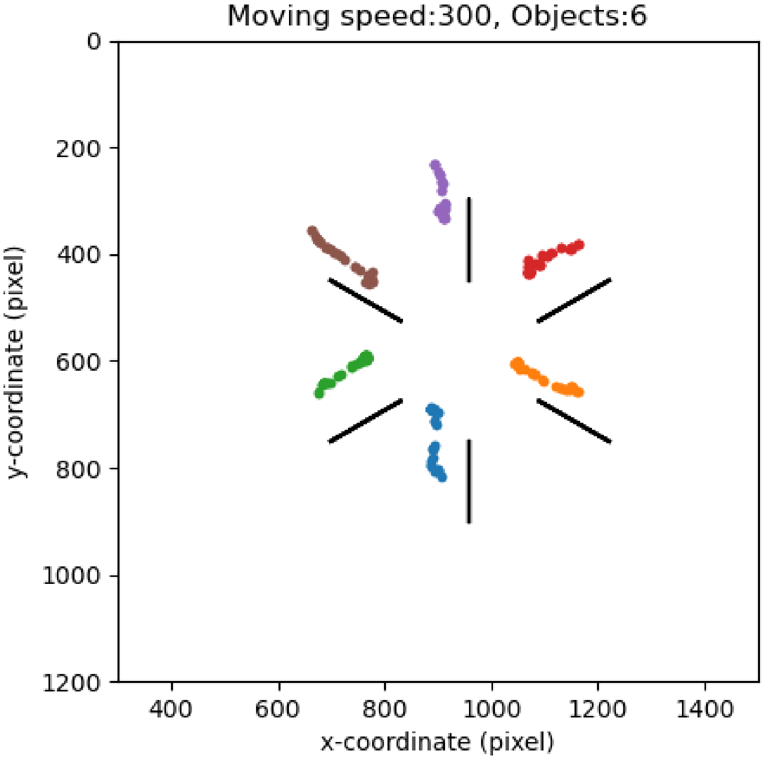
- □ Nein
- □ Wenn Ja

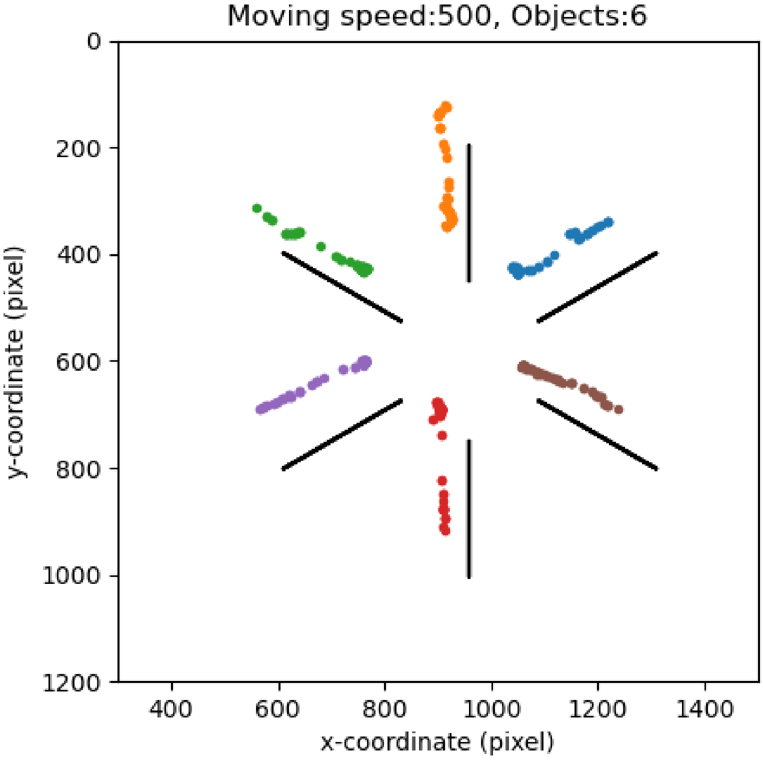
  Wie empfinden Sie die Größe der Objekte im Experiment?

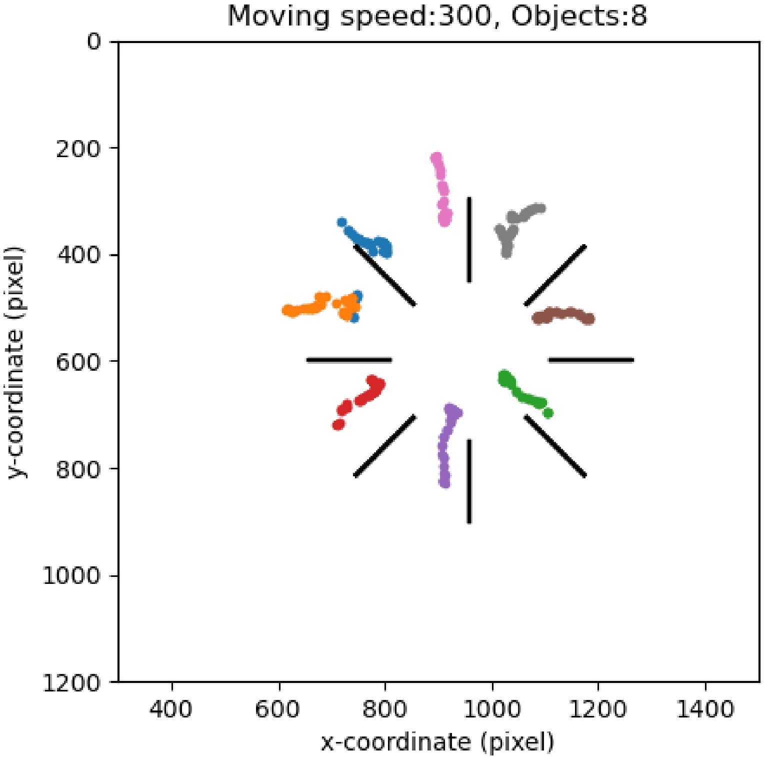
  zu klein □ □ □ zu groß

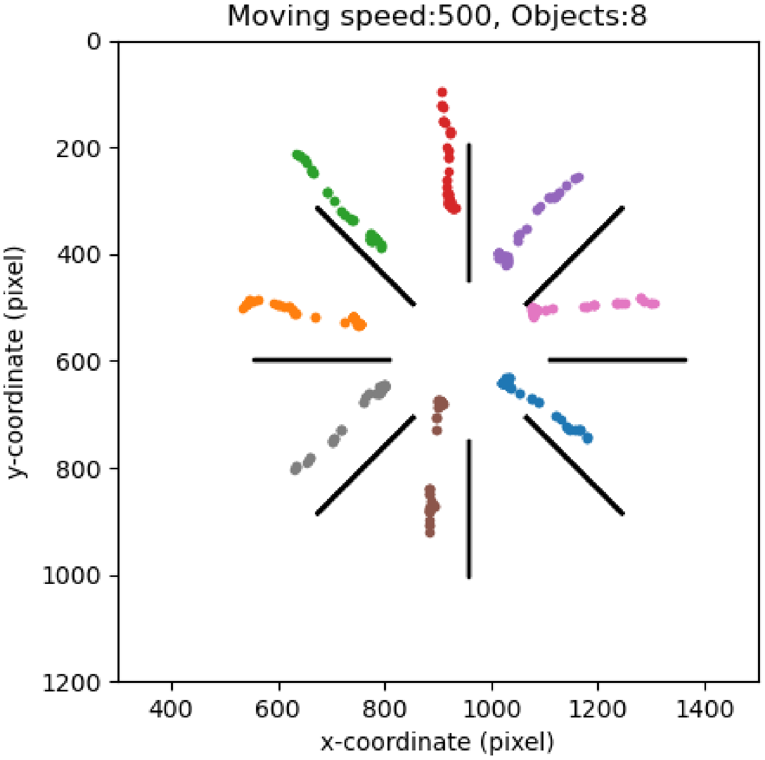
Warum? ____

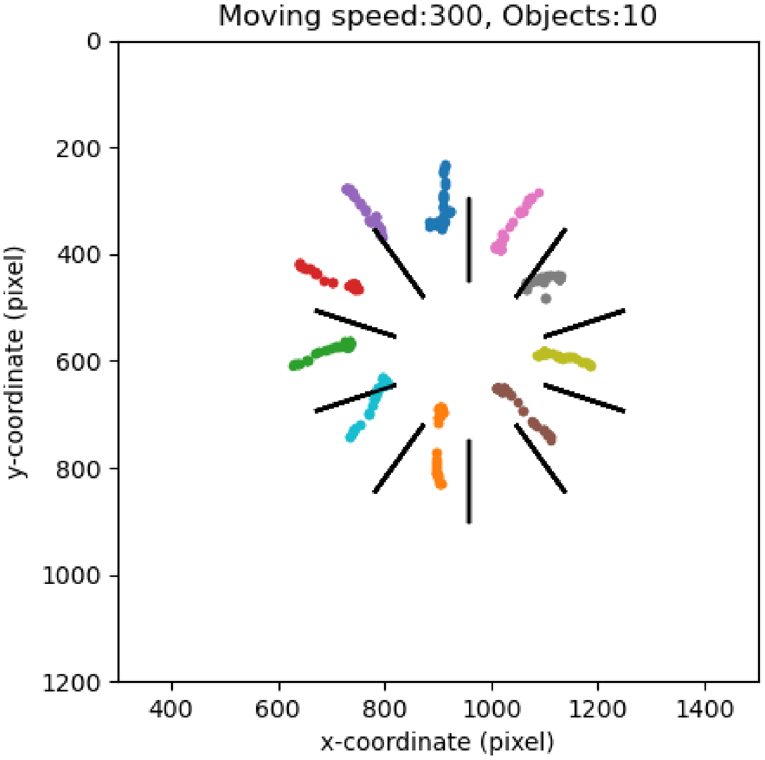
A.2 Visualization moving trajectories from one participant

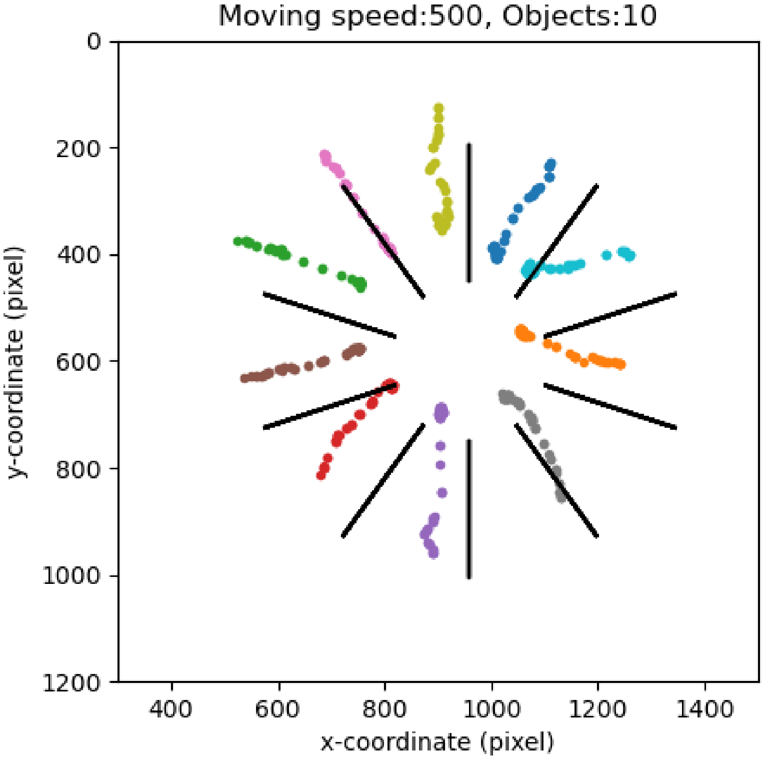

Moving speed:300, Objects:6
0
200
400
600
800
1000
1200
y-coordinate (pixel)
400
600
800
1000
1200
1400
x-coordinate (pixel)

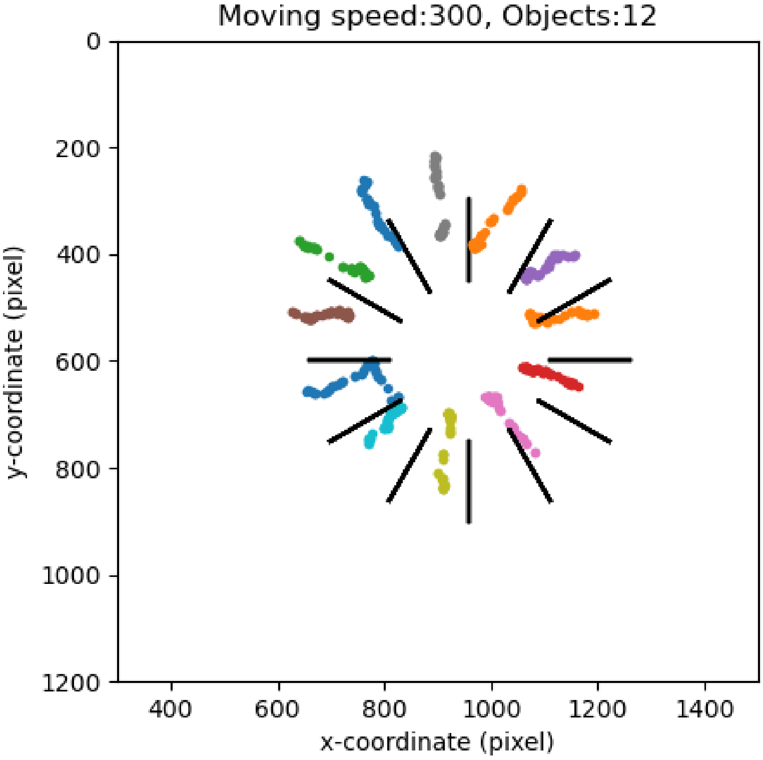

Moving speed:500, Objects:6
0
200
400
600
800
1000
1200
y-coordinate (pixel)
400
600
800
1000
1200
1400
x-coordinate (pixel)

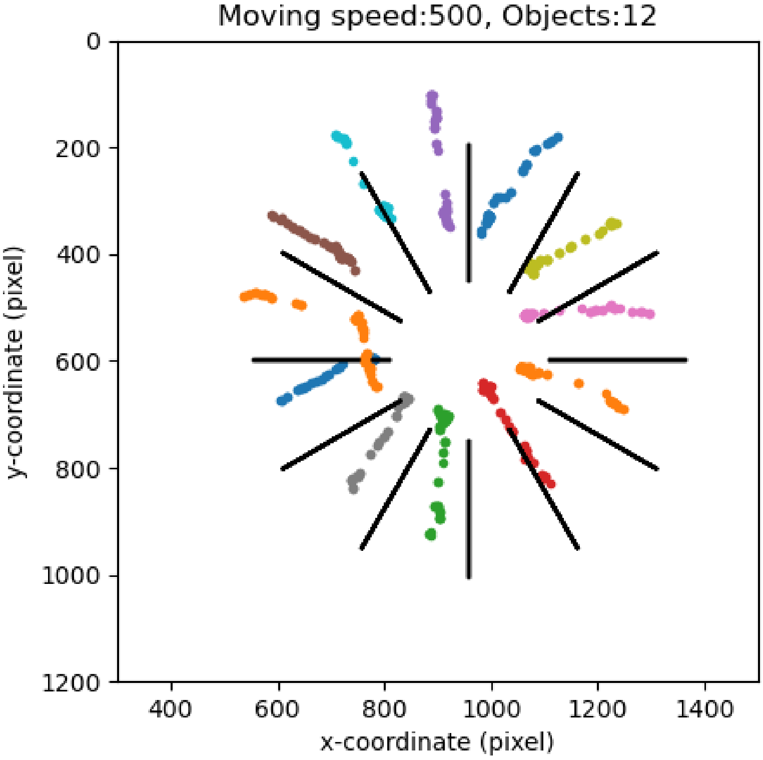

Moving speed:300, Objects:8
y-coordinate (pixel)
0
200
400
600
800
1000
1200
x-coordinate (pixel)
400
600
800
1000
1200
1400

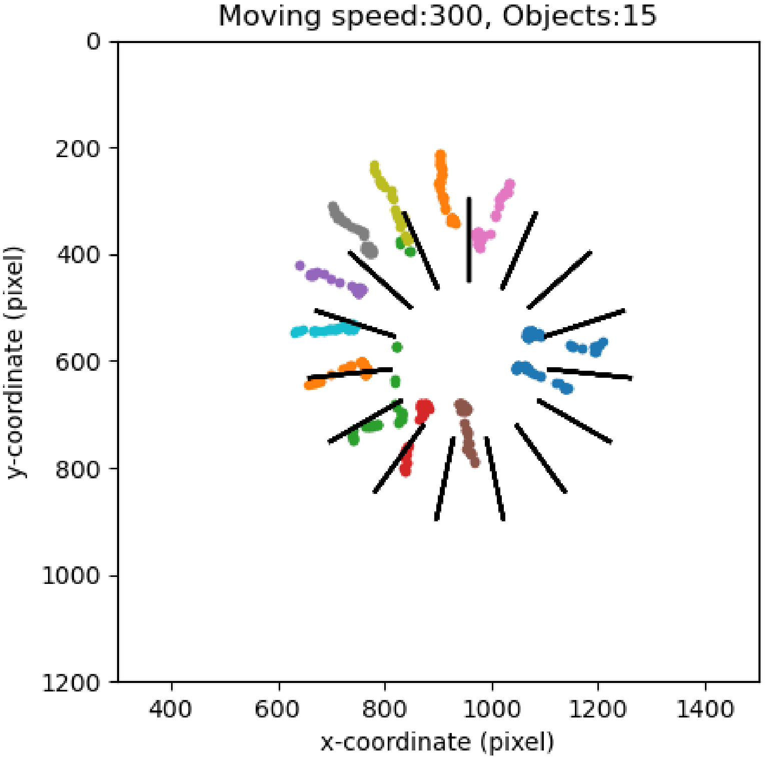

Moving speed:500, Objects:8
y-coordinate (pixel)
0
200
400
600
800
1000
1200
x-coordinate (pixel)
400
600
800
1000
1200
1400

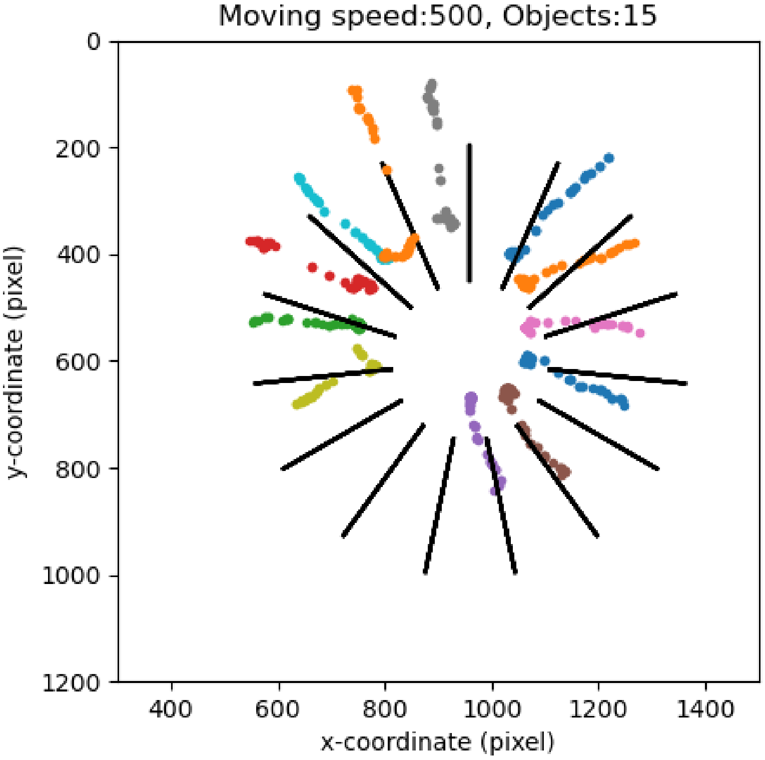

Moving speed:300, Objects:10
y-coordinate (pixel)
x-coordinate (pixel)
0
200
400
600
800
1000
1200
400
600
800
1000
1200
1400


Moving speed:500, Objects:10
y-coordinate (pixel)
x-coordinate (pixel)
0
200
400
600
800
1000
1200
400
600
800
1000
1200
1400


Moving speed:300, Objects:12
y-coordinate (pixel)
0
200
400
600
800
1000
1200
x-coordinate (pixel)
400
600
800
1000
1200
1400


Moving speed:500, Objects:12
y-coordinate (pixel)
0
200
400
600
800
1000
1200
x-coordinate (pixel)
400
600
800
1000
1200
1400


Moving speed:300, Objects:15
0
200
400
600
800
1000
1200
y-coordinate (pixel)
400
600
800
1000
1200
1400
x-coordinate (pixel)


Moving speed:500, Objects:15
0
200
400
600
800
1000
1200
y-coordinate (pixel)
400
600
800
1000
1200
1400
x-coordinate (pixel)

Appendix B

B.1 Open questions for study 2 - German version

Sind Ihnen Unterschiede in den vier Benutzerschnittstellen aufgefallen?

□ Nein

□ Wenn Ja

1. Beschreiben Sie bitte was Ihnen aufgefallen ist.

2. Bringen Sie bitte die vier Benutzerschnittstellen in eine Rangfolge. Beginnen Sie mit der für Sie präferierten Benutzerschnittstelle:

_____ Kein Feedback
_____ Auditives Feedback
_____ Visuelles Feedback
_____ Auditiv-Visuelles Feedback

Beschreiben Sie bitte warum Ihnen diese Schnittstelle am meisten gefallen hat:

Beschreiben Sie bitte warum Ihnen diese Schnittstelle am wenigsten gefallen hat:

Haben Sie Verbesserungsvorschläge für zukünftige Benutzerschnittstellen dieser Art? Wenn ja, welche?

Wie haben Sie die Geschwindigkeit der Objekte wahrgenommen? War diese zu schnell / zu langsam / genau richtig?

Wie haben Sie die Größe der Objekte wahrgenommen? War diese zu groß/klein/genau richtig?

Hatten Sie genug Zeit die gewünschten Objekte zu suchen?

B.2 Phrases for study 2

Phrases used in study 2.

1. Kanada hat zehn Provinzen
2. Der Bus war sehr voll
3. Du bist ein gutes Beispiel
4. Dein Vortrag war interessant
5. Diese Uhr ist zu teuer
6. Er hat sieben Mal angerufen
7. Morgen scheint die Sonne
8. Du musst einen Termin machen
9. Gehst du gerne Zelten?
10. Das Bad ist gut zum Lesen
11. Ich stimme dir zu
12. Das ist eine sehr gute Idee
13. Bitte folge den Regeln
14. Ich spiele gerne Tennis
15. Willst du das wirklich?
16. Diese Lederjacke ist zu warm
17. Der Benzinpreis ist hoch
18. Die Kinder spielen
19. Ich treffe dich gegen Mittag
20. Heute ist es sehr windig

Appendix C Documents

C.1 NASA TLX Scale

NASA Task Load Index

Hart and Staveland's NASA Task Load Index (TLX) method assesses work load on five 7-point scales. Increments of high, medium and low estimates for each point result in 21 gradations on the scales.

Code	Task	Date

Mental Demand — How mentally demanding was the task?

Very Low — Very High

Physical Demand — How physically demanding was the task?

Very Low — Very High

Temporal Demand — How hurried or rushed was the pace of the task?

Very Low — Very High

Performance — How successful were you in accomplishing what you were asked to do?

Perfect — Failure

Effort — How hard did you have to work to accomplish your level of performance?

Very Low — Very High

Frustration — How insecure, discouraged, irritated, stressed, and annoyed were you?

Very Low — Very High

C.2 Phrases for study 3

Phrases used in study 3.

1. thank you for your help
2. the trains are always late
3. I can play much better now
4. you must make an appointment
5. what you see is what you get
6. players must know all the rules
7. time to go shopping
8. this is a very good idea
9. this equation is too complicated
10. I like to play tennis
11. can we play cards tonight
12. have a good weekend
13. I agree with you
14. healthy food is good for you
15. I will meet you at noon
16. my fingers are very cold
17. you are a wonderful example
18. the children are playing