DESIGNING GAZE-BASED INTERACTION FOR PERVERSIVE PUBLIC DISPLAYS

DISSERTATION
an der Fakultät für Mathematik, Informatik und Statistik
der Ludwig-Maximilians-Universität München

vorgelegt von
M.Sc. Computer Science and Engineering
Mohamed Khamis
geboren am 4. September 1989

München, den 14. Mai 2018
Erstgutachter: Prof. Dr. Florian Alt
Zweitgutachter: Prof. Dr. Andreas Bulling
Drittgutachter: Prof. Dr. Marc Langheinrich

Tag der mündlichen Prüfung: 7th of August, 2018
Abstract

The last decade witnessed an increasing adoption of public interactive displays. Displays can now be seen in many public areas, such as shopping malls, and train stations. There is also a growing trend towards using large public displays especially in airports, urban areas, universities and libraries.

Meanwhile, advances in eye tracking and visual computing promise straightforward integration of eye tracking on these displays for both: 1) monitoring the user’s visual behavior to evaluate different aspects of the display, such as measuring the visual attention of passersby, and for 2) interaction purposes, such as allowing users to provide input, retrieve content, or transfer data using their eye movements. Gaze is particularly useful for pervasive public displays. In addition to being natural and intuitive, eye gaze can be detected from a distance, bringing interactivity to displays that are physically unreachable. Gaze reflects the user’s intention and visual interests, and its subtle nature makes it well-suited for public interactions where social embarrassment and privacy concerns might hinder the experience.

On the downside, eye tracking technologies have traditionally been developed for desktop settings, where a user interacts from a stationary position and for a relatively long period of time. Interaction with public displays is fundamentally different and hence poses unique challenges when employing eye tracking. First, users of public displays are dynamic; users could approach the display from different directions, and interact from different positions or even while moving. This means that gaze-enabled displays should not expect users to be stationary at a specific position, but instead adapt to users’ ever-changing position in front of the display. Second, users of public displays typically interact for short durations, often for a few seconds only. This means that contrary to desktop settings, public displays cannot afford requiring users to perform time-consuming calibration prior to interaction.

In this publications-based dissertation, we first report on a review of challenges of interactive public displays, and discuss the potential of gaze in addressing these challenges. We then showcase the implementation and in-depth evaluation of two applications where gaze is leveraged to address core problems in today’s public displays. The first presents an eye-based solution, EyePACT, that tackles the parallax effect which is often experienced on today’s touch-based public displays. We found that EyePACT significantly improves accuracy even with varying degrees of parallax. The second is a novel multimodal system, GTmoPass, that combines gaze and touch input for secure user authentication on public displays. GTmoPass was found to be highly resilient to shoulder surfing, thermal attacks and smudge attacks, thereby offering a secure solution to an important problem on public displays.

The second part of the dissertation explores specific challenges of gaze-based interaction with public displays. First, we address the user positioning problem by means of active eye tracking. More specifically, we built a novel prototype, Eye-
Scout, that dynamically moves the eye tracker based on the user’s position without augmenting the user. This, in turn, allowed us to study and understand gaze-based interaction with public displays while walking, and when approaching the display from different positions. An evaluation revealed that EyeScout is well perceived by users, and improves the time needed to initiate gaze interaction by 62% compared to state-of-the-art. Second, we propose a system, Read2Calibrate, for calibrating eye trackers implicitly while users read text on displays. We found that although text-based calibration is less accurate than traditional methods, it integrates smoothly while reading and thereby more suitable for public displays. Finally, through our prototype system, EyeVote, we show how to allow users to select textual options on public displays via gaze without calibration. In a field deployment of EyeVote, we studied the trade-off between accuracy and selection speed when using calibration-free selection techniques. We found that users of public displays value faster interactions over accurate ones, and are willing to correct system errors in case of inaccuracies.

We conclude by discussing the implications of our findings on the design of gaze-based interaction for public displays, and how our work can be adapted for other domains apart from public displays, such as on handheld mobile devices.


Diese kumulative Dissertation überprüft zunächst die Herausforderungen interaktiver öffentlicher Displays und diskutiert das Potenzial von Blickbasierten Interaktion zu deren Bewältigung. Anschließend wird die Implementierung und eingehende Evaluierung von zwei Beispielhaften Anwendungen vorgestellt, bei denen Nutzer durch den Blick und öffentlichen Displays interagieren. Daraus ergeben sich weitere greifbare Vorteile der blickbasierten Interaktion für öffentliche Display-Kontexte. Bei der ersten Anwendung, EyePACT, steht der Parallaxeneffekt im Fokus, der heutzutage häufig ein Problem auf öffentlichen Displays darstellt, die über Berührung (Touch) gesteuert werden. Die zweite Anwendung ist ein neuartiges multimodales System, GTmoPass, das Gaze- und Touch-Eingabe zur sicheren Benutzerauthenti-
zierung auf öffentlichen Displays kombiniert. GTmoPass ist sehr widerstandsfähig sowohl gegenüber unerwünschten fremden Blicken als auch gegenüber sogenannten thermischen Angriffen und Schmierangriffen. Es bietet damit eine sichere Lösung für ein wichtiges Sicherheits- und Datenschutzproblem auf öffentlichen Displays.

There are many people that I would like to thank for their support in the last 4 years. I would like to thank my supervisors Florian Alt, and Andreas Bulling, whom I was very fortunate to work under their supervision. I consider myself very lucky for having had the chance to work with both of you. Although each of you has a unique approach in research, both of you are exemplary researchers, mentors and supervisors – you both complemented each other and this made the co-supervision flawless. I would like to thank you for your continuous support on personal and professional levels. You did not only support, guide, and refine my work, but you also significantly helped in shaping me as an academic and helped me make the most out of those 4 years. The three of us make an great team! I would also like to thank Marc Langheinrich for responding immediately and positively to my request for being my external examiner, and traveling all the way for my defense.

I am very grateful for the help I got from all the professors in the Chair for Media Informatics. Thank you Heinrich Hussmann, Andreas Butz, and Albrecht Schmidt for your continous support ever since I have known you. You always provided constructive and helpful feedback in the doctoral colloquiums, and helped me in a lot of situations where I needed expert opinions from senior academics.

I would like to thank Rainer Fink, who ensures that all technicalities run smoothly, and Franziska Schwamb for her help with paperwork and bureaucratic hurdles.

I have made a lot of friends during my time at the LMU. Thank you Malin Eiband for being such a nice, truthful and supporting friend. Our “shoulder surfing” project remains one of my most pleasant collaboration experiences. Thank you for introducing me to my favorite bottle. Thank you Mariam Hassib for being such a thoughtful friend and helping out in all sorts of matters, ranging from introducing me to Florian 4 years ago, to offering me to stay over until I find an apartment in Munich. I have always enjoyed our collaborations. I wish you, Amr and your little one all the best, and I am happy to say “dude, we are no longer screwed”. Thank you Nada Terzimehić for being a great friend. It was always very nice to chat, enjoying Sababa’s Falafel together, and it was a pleasure to build Lunchocracy together. Thanks for all the coffee and food you offered me! may you have a lot of “Yummy!” moments. Thanks Sarah Theres Völkel for all the inspiration you spread with your activeness and engagement. I am sure you will do very well in your PhD – you are and you will continue to be a very successful person. I would like to thank Danniel Buschek for all the interesting discussions and successful collaborations. It is always nice to work with you. Thank you Christina Schneegass and Pino Colada for spreading happiness in the chair. And thank you Christina for organizing the first German HCI booklet and the awesome CHI party. Thanks Sarah Aragon Bartsch for all the interesting discussions about languages, and for bravely embracing ERASMUS. Thank you Ceenu George for the nice collaborations, it was
a pleasure to work with you. Let’s hope we do make it to Kazakhstan one day. Thanks Christian Mai for the many recent collaborations and taking care of all the VR stuff. Thank you Renate Häuslschmid for being an awesome office mate and for the many moments of laughter watching funny videos. Thank you Hanna Schneider for being a cool office mate, and an awesome travel mate in Heidelberg. Thank you Tobias Seitz for being such a cool workmate, and for the opportunity to work on PasswordBlur together. Thank you Axel Hösl for your great help with EyeScout, it was nice to work with you. Thank you Thomas Kosch, Jakob Karolus, Matthias Hoppe, and Pascal Knierim for all the nerf gun fights and the cool moments, as well as the interesting crazy researchie discussions. Thank you Sylvia Rothe for the collaborations, it was a pleasure to work with you. Thanks Maria Fysaraki for bringing all the Greek food and sweets. Thank you Emanuel von Zeszchwitz for the fruitful discussions and successful collaborations. It is always great to collaborate with you and I have learned a lot from you. Thank you Bastian Pfleging for all the help with finding the best deals and offers. Thank you Daniel Ullrich for being my movie nerd companion, and the awesome ice coffee. Thanks Alexander Wiethoff for the helpful feedback on my first IDC. Thank you Heiko Drewes for the nice collaborations and interesting discussions, it was a pleasure to work with you. Thank you to all the external PhD students: Christian Lachner, Veronika Gampfer, David Englmeier, Gesa Wiegand, and Michael Braun. Thank you and best of luck to the new PhD students, Kai Holländer, Sarah Prange, Lukas Mecke, Beat Rossmy. Finally I’d like to thank them all again for the amazing graduation hat!

I extend my thanks to all external collaborators Stefan Schneegass, Niels Henze, Sven Mayer, Huy Viet Li, Yomna Abdelrahman, Menna Essam, Christian Sturm, Ville Mäkelä, Diako Mardanbegi, Vito Gentile, Salvatore Sorce, and Hans Gellersen. I also thank my students whom I had the privilege to work with.

I would like to thank my friends who were very close to me and supportive during the last 4 years. Thank you to the “Gang”: Mostafa Ismail, Mohamed Anwar, Lojayna Sherif. Thank you Passant El Agroudy, and all my friends in Kaiserslautern especially: Yasmina Adams, Mohamed Selim, and Bahaa Beih.

The guidance and support of my academic mentors Slim Abdennadher, Christian Sturm, and Fatma Meawad extends beyond these 4 years. Thank you.

I cannot find enough words to thank my mother Eiman Hafez, sisters Engy and Lamiaa, and father Mohsen Khamis. Thank you all for being the best family ever, for your love, and unconditional support throughout my life, as well as encouraging and motivating me to strive for the best.

Last but most definitely not least, I would never be able to thank my wife, Maha Badreldin, enough for her support, understanding, and encouragement. Thank you for your support, love, patience, and encouragement, without which I would have neither got this degree, nor the lectureship position in the University of Glasgow. Thank you for standing by me.
# Table of Contents

List of Figures xv

Publications and Declaration of Co-Authorship 1

1 Introduction and Background 5

1 Introduction 7
   1.1 Research Objectives ............................................. 9
   1.2 Research Questions ................................................ 9
   1.3 Summary of Contributions ........................................ 10
      1.3.1 Practical Contributions ..................................... 10
      1.3.2 Theoretical Contributions ................................... 11
      1.3.3 Methodological Contributions ................................. 11
   1.4 Dissertation Overview ........................................... 12
   1.5 Full Publications List ........................................... 14

2 Background and Related Work 21
   2.1 The Advent of Pervasive Public Displays .......................... 22
      2.1.1 Uniqueness of Public Displays .............................. 22
      2.1.2 Interacting with Public Displays ............................. 23
         2.1.2.1 Touch .................................................... 23
         2.1.2.2 Mid-air Gestures ....................................... 23
      2.1.3 Mobile Device Interaction .................................... 24
      2.1.4 Gaze .......................................................... 24
   2.2 The Eyes in Human-Computer Interaction ........................... 24
      2.2.1 Opportunities of Eye Tracking ............................... 25
      2.2.2 General Limitations of Eye Tracking ......................... 25
   2.3 Previous Work on Gaze-enabled Displays ........................... 26
      2.3.1 Attention Monitoring on Public Displays ..................... 26
2.3.1.1 Methods to Monitor Attention ........................................ 26
2.3.1.2 Attention Monitoring in Practice ..................................... 27
2.3.2 Implicit Gaze Interaction with Public Displays ................. 27
  2.3.2.1 Eye-Contact Aware Implicit Interaction ......................... 28
  2.3.2.2 Gaze-Behavior Aware Implicit Interaction ....................... 28
2.3.3 Explicit Gaze Interaction with Public Displays ................. 28
  2.3.3.1 Gaze-only Interaction ............................................. 29
  2.3.3.2 Gaze-supported (Multimodal) Interaction ....................... 29
2.4 Trends and Limitations of Previous Work ......................... 29

II SUMMARY OF PEER-REVIEWED PUBLICATIONS .......................... 31

3 Opportunities of Gaze on Public Displays ............................. 33
  3.1 EyePACT: Implicit Eye-based Parallax Correction ................. 34
  3.2 GTmoPass: Explicit Gaze-supported Authentication ............... 35

4 Challenges of Gaze-enabled Public Displays ......................... 37
  4.1 EyeScout: Movement and Position Independent Gaze Interaction . 38
  4.2 TextPursuits: Implicit Calibration and Calibration-free Gaze Interaction 39
  4.3 EyeVote in the Wild: Understanding the Trade-off between Accuracy and User Experience ......................... 40

III CONCLUSION AND FUTURE WORK ....................................... 43

5 Closing Remarks ............................................................. 45
  5.1 Design Recommendations ............................................... 46
  5.2 Future Work ............................................................... 46
    5.2.1 Our Methodology Beyond Public Displays ....................... 47
    5.2.2 There is More to Gaze Interaction than Accuracy ............... 47
    5.2.3 Privacy Implications of Gaze-enabled Public Displays ........... 48
# TABLE OF CONTENTS

IV  **OPPORTUNITIES OF GAZE-ENABLED PUBLIC DISPLAYS**  49

6  **Tackling Challenges of Interactive Public Displays using Gaze**  51

7  **EyePACT: Eye-Based Parallax Correction on Touch-Enabled Interactive Displays.**  57

8  **GTmoPass: Two-factor Authentication on Public Displays Using Gaze-Touch passwords and Personal Mobile Devices.**  77

V  **CHALLENGES OF GAZE-ENABLED PUBLIC DISPLAYS**  87

9  **EyeScout: Active Eye Tracking for Position and Movement Independent Gaze Interaction with Large Public Displays.**  89

10  **TextPursuits: Using Text for Pursuits-Based Interaction and Calibration on Public Displays.**  103

11  **EyeVote in the Wild: Do Users bother Correcting System Errors on Public Displays?**  117

Bibliography  125

Curriculum Vitae  143
LIST OF FIGURES

1.1 Our Methodology .......................................................... 12
3.1 EyePACT: Eye-Based Parallax Correction on Touch-Enabled Interactive Displays ......................................................... 34
4.1 EyeScout: Active Eye Tracking for Position and Movement Independent Gaze Interaction with Large Public Displays. ................. 38
4.2 TextPursuits: Using Text for Pursuits-Based Interaction and Calibration on Public Displays ...................................................... 39
4.3 EyeVote in the Wild: Do Users bother Correcting System Errors on Public Displays? .......................................................... 41
Publications and Declaration of Co-Authorship

All the following papers were peer-reviewed.

Chapter 6 corresponds to the following workshop paper:


This paper is based on my initial PhD research proposal. The work was discussed with my supervisors: Florian Alt and Andreas Bulling. However both the initial and camera ready versions of the paper were written by me.

Chapter 7 corresponds to the following journal paper:


I conceived the idea and the original research contributions of EyePACT after noticing the issue on a public display in Münchner Freiheit, in Munich, Germany. The concept was discussed with my supervisors: Florian Alt and Andreas Bulling. Daniel Buschek assisted with the regression analysis reported in section 5 of the paper (≈ half a page). Tobias Thieron assisted in the implementation reported in the second paragraph of section 3 (≈ 5 lines), which was done as part of his master thesis under my supervision. He conducted the study, which I designed and supervised. I conducted the analysis, reporting and interpretation of the results. I wrote the initial draft of the article and did the subsequent corrections after feedback from my supervisors: Florian Alt and Andreas Bulling.

Chapter 8 corresponds to the following conference paper:

I received the Google IoT research award based on the idea of GTmoPass. I conceived the idea as a followup project on my Late-Breaking-Work article at CHI 2016. I carried out the full implementation of GTmoPass. Regina Hasholzner conducted the user study using my implementation and reported on the initial analysis as part of her internship at the chair of media informatics. I developed the study’s design and procedure, which Regina carried out under my close supervision. I significantly extended the analysis of the results, and performed the interpretation and reporting of the results. The work was discussed with my supervisors: Florian Alt and Andreas Bulling.

**Chapter 9** corresponds to the following conference paper:


I conceived the idea and the original research contributions of EyeScout. Axel Hösl and I co-supervised the master theses of Martin Reiss and Alexander Klimczak. They both built the initial prototype under my close supervision. They also conducted the study, whose design and procedure was developed by me. I wrote the initial draft and performed the analysis, interpretation and reporting of the results. The work was regularly discussed with Florian Alt, Andreas Bulling, and Axel Hösl. The three revised the first version of the article, and Axel produced figure 1. I did the subsequent corrections.

**Chapter 10** corresponds to the following conference paper:


I conceived the idea and original research contributions of TextPursuits, which constitutes two systems: EyeVote and Read2Calibrate. The idea of EyeVote was described in my initial PhD proposal. The implementation was partially realized by Katharina Stolz as part of her bachelor thesis. I conducted the analysis, interpretation and reporting of the results. The second system, Read2Calibrate, was implemented by Ozan Saltuk as part of his master thesis, who also wrote part of the implementation subsection ($\approx 1/6$ of page 6). I closely supervised both theses. Alina Hang co-supervised Ozan’s thesis, and suggested improvements to the first draft of the paper. I wrote the first draft, and made the majority of the subsequent corrections. My supervisors, Florian Alt and Andreas Bulling, helped me in improving the draft.
Chapter 11 corresponds to the following conference paper:


I conceived the idea and original research contributions of this work, which was a follow up on the TextPursuits paper (Chapter 10). I developed the study’s design and procedure. The study was then conducted by Ludwig Trotter, Markus Tessmann, and Christina Dannhart in the context of the “Advanced Topics in HCI” module\(^1\). The students customized the existing implementation of EyeVote for this work’s user study. I closely supervised the deployment of the field study. The work was discussed with my supervisors: Florian Alt and Andreas Bulling. Ludwig Trotter wrote the first draft of the implementation and results sections. I significantly revised the sections, wrote the first draft of the entire paper, and made the vast majority of the subsequent corrections. Florian Alt assisted me in improving the framing of the story.

Munich, 14.05.2018

Mohamed Khamis

---

\(^1\) [http://www.medien.ifi.lmu.de/lehre/ss16/ath/](http://www.medien.ifi.lmu.de/lehre/ss16/ath/)
INTRODUCTION AND BACKGROUND
1

Introduction
Many aspects of Mark Weiser’s vision for computers in the 21st century [124] are already part of our daily lives. Computing devices are seamlessly integrated into environments around us. These devices come in many forms, including pervasive public displays. In fact, our present day public displays resemble what Weiser and his colleagues referred to as boards – one of the key components of ubiquitous computing. Public interactive displays, and in particular large ones, are increasingly becoming ubiquitous. They can now be found deployed in public areas, such as train stations, airports, malls, supermarkets, and universities. They bring a lot of tangible benefits to passersby. For example, they are used to deliver information (e.g., train schedules and news) and personalized content [21], allow purchase of tickets and services, encourage civic engagement [114], and nudge behavior change [94]. These applications are largely driven by advances in sensing technologies, which in turn enable a myriad of modalities to empower public displays. One particularly promising modality that is increasingly gaining attention is eye gaze.

Being a subtle, natural and intuitive modality, the use of eye gaze for interaction aligns nicely with the Mark Weiser’s vision of pervasive technologies that can weave themselves into everyday life. Humans use their eyes intuitively to express visual interests, reveal their intentions, and regulate conversations. In addition to being valuable to human-computer interaction in general, eye gaze is exceptionally attractive for pervasive public displays. Being fast [100], gaze has a lot of potential in delivering “immediate usability” – a requirement of pervasive public displays [81]. Gaze can be used from a distance, hence supplying users with a contact-free and hygienic method for interacting in public, even if the display is physically unreachable (e.g., behind glass window or above user height [20]), or if the user is wearing gloves and hence cannot interact via touch.

Acknowledging the potential of gaze on public displays, researchers proposed novel methods that leverage gaze to improve the user’s experience [104, 105, 118, 128, 131]. While these works presented a leap towards gaze-enabled public displays, there are several gaps in previous work in this area. First, prior work mainly focused on introducing gaze interaction techniques, or improving gaze estimation accuracy. That being said, one gap is that there is a lack of gaze-based solutions to problems that are specific to public displays. A second gap is that to date, eye tracking technologies and techniques that are intended for desktop settings are still being used for public displays. As a result, there is a need to adapt eye tracking to the many unique aspects of public displays, which are under-investigated in the context of gaze-based interaction. Examples of said aspects are the facts that opposed to desktop settings, users of public displays could be interacting from different positions and even while moving, and that public displays cannot afford requiring time-consuming eye tracker calibration. There is also a lack of understanding of the needs of users of gaze-enabled public displays; e.g., how far are users striving for quick interactions? and how can we design systems that optimize interaction speed at the expense of accuracy, yet still provide a pleasant user experience?
1 Introduction

1.1 Research Objectives

To address these gaps, this dissertation reports on multiple novel concepts (2 methods, and 3 systems), and results from a series of lab studies (N=133) and field studies (N=49–106). More specifically, this dissertation closes the aforementioned gaps by addressing the following research objectives:

1. Understanding the potential of eye gaze in the context of pervasive public displays (Chapter 6).
2. Designing, implementing, and evaluating gaze-based interfaces that address key challenges of pervasive public displays (Chapters 7 and 8).
3. Exploring and addressing the unique aspects and challenges arising from integrating gaze with pervasive public displays (Chapters 9, 10, and 11).

To fulfill Objective 1, we first report on a review of challenges of pervasive public displays that can be addressed by leveraging the passerby’s eye gaze (Chapter 6).

As for Objective 2, we report on the design, implementation, and evaluation of two systems that employ gaze to address core challenges on pervasive public displays. Namely, in Chapter 7 we introduce the concept, implementation and evaluation of EyePACT, an effective approach for mitigating the parallax effect on public displays, which is a fundamental problem with many of today’s public displays. Second, in Chapter 8 we introduce the concept, implementation and evaluation of a novel approach for secure user authentication on public displays.

Finally, we address Objective 3 by studying two unique properties of gaze interfaces on pervasive public displays: eye tracking on the move, and calibration-free gaze interaction. We do so by introducing novel systems, and evaluating them in lab and field studies. More specifically, we introduce EyeScout in Chapter 9, the first system to leverage active eye tracking to expand the areas from which gaze interaction with large displays is possible, while significantly decreasing the time required to initiate gaze interaction. Second, in Chapter 10, we show how reading text on public displays can be used to enable calibration-free gaze interaction, or to calibrate eye trackers implicitly. Finally in Chapter 11, we report on a field study that shows that passersby would rather use gaze interaction techniques that are fast even if they were inaccurate, as long as usable input correction mechanisms are provided.

1.2 Research Questions

By addressing the aforementioned objectives, we answer the following research questions:
1.3 Summary of Contributions

This dissertation makes practical, methodological, and theoretical contributions.

1.3.1 Practical Contributions

This dissertation provides a number of practical contributions for improving the design of gaze-enabled public displays. In this work, we have designed various approaches that leverage gaze to address challenges on today’s public displays, such as parallax-free touch interaction and secure authentication. We have also presented the design, implementation and evaluation of forward-looking systems that facilitate gaze interaction on the move, implicit eye tracker calibration, and calibration-free interaction. The proposed concepts are novel yet applicable on today’s public displays, and can be key to enabling wide-scale adoption of gaze on public displays. We conclude with recommendations for designing gaze-based interaction for public displays (Section 5.1)
1.3.2 Theoretical Contributions

This dissertation contributes a body of knowledge that augments our understanding of gaze-based interaction with public displays. First, our work has identified opportunities brought forth by integrating gaze into pervasive public displays. Second, we explored core challenges that are unique to gaze-enabled public display settings. This dissertation proposed systems that leverage said opportunities, and proposed systems and studies that address and deepen our understanding of challenges of gaze-enabled public displays.

At the same time, this work lays a foundation on which future studies and systems can build on: 1) many of the identified opportunities are waiting to be explored and leveraged by researchers and practitioners. 2) Our exploration of unique aspects of public displays has set the scene for the upcoming research challenges for gaze-based interaction on public displays.

1.3.3 Methodological Contributions

In addition, this dissertation also makes research methodology contributions. Namely, we propose a methodology for studying gaze-based interaction while moving (e.g., walking across large displays, or approaching them from random positions). Work in gaze-based interaction has mostly considered a stationary user – there had been little to no experimental research on gaze interaction while on the move. Our methodology contribution is demonstrated in Chapter 9, where we designed the first user study in which users perform gaze-based selections while walking. This study design can be used as a basis for follow up work on gaze-based interaction while moving. This is not only relevant to public displays, but also for gaze interaction with mobile, wearable, and head mounted devices. For example, our methodology was adopted in follow up work on gaze-based interaction while walking in virtual environments [56].

Another core contribution is the overarching methodology adopted in this work to explore and understand the use of eye gaze in the domain of pervasive public displays (Figure 1.1). The methodology adopted in this dissertation is as follows: The first step is to identify challenges users face in this domain – in Chapter 6, we do that for pervasive public displays. The second step is to study how gaze can address core challenges in this domain – in Chapters 7 and 8, we report on methods for addressing core problems on public displays using gaze. The third step is to understand the novel challenges arising from integrating gaze in this domain. The fourth step is to design, implement, and evaluate solutions to address these challenges. We identify a novel challenge of using gaze in this domain in Chapters 9, 10 and 11, and then we design, implement, and evaluate solutions to address them.
Figure 1.1: Our methodology is as follows: First, we identify the challenges in the field of pervasive displays. Second, we study how gaze can address these challenges. Third, we investigate the novel challenges arising from the use of gaze in this domain. Finally, we design, implement, and evaluate solutions to address these challenges. This methodology can be adopted for exploring gaze-based interaction on other platforms, such as handheld mobile devices.

In section 5.2.1, we discuss how this methodology can be transferred to other domains, such as mobile devices, mixed reality, and internet of things.

1.4 Dissertation Overview

This cumulative dissertation comprises 6 peer-reviewed scientific publications, and is complemented with the following chapters:

- **Chapter 2: Background and Related Work** This chapter summarizes the relevant background in both pervasive public displays, and eye tracking. It also discussed previous work in gaze-enabled public displays, and highlights its trends and limitations.

- **Chapter 3: Opportunities of Gaze on Public Displays** This chapter summarizes the peer-reviewed publications presented in Chapters 6, 7, and 8.

- **Chapter 4: Challenges of Gaze-enabled Public Displays** This chapter summarizes the peer-reviewed publications presented in Chapters 9, 10, and 11.

- **Chapter 5: Closing Remarks**: This chapter concludes and summarizes the outcomes of this work, presents design recommendations for gaze-enabled public displays, and highlights directions for future research.

- **Chapters 6–11**: each of those chapters corresponds to the a peer-reviewed publication as shown in Table 1.2 on the following page.
<table>
<thead>
<tr>
<th>Chapter</th>
<th>Publication</th>
</tr>
</thead>
</table>

**Table 1.2:** The dissertation comprises 6 peer reviewed papers published as a journal paper at IMWUT, conference papers at UIST 2017, PerDis 2017, UbiComp 2016, MUM 2016, and a workshop paper at UbiComp 2015.
1.5 Full Publications List

Apart from the papers included in this dissertation, below is a full list of all 38 publications published in my academic career.

Journal papers (4)


Conference papers (21)

2018


2017


HONORABLE MENTION AWARD


20. Mohamed Khamis, Ludwig Trotter, Markus Tessmann, Christina Dannhart, Andreas Bulling, and Florian Alt. 2016. EyeVote in the wild:


2015


2013


Technical reports, adjunct papers (16)


2

Background and Related Work

Work on gaze-enabled public displays builds on two strands of work: (1) Pervasive Public Displays, and (2) Eye Tracking in HCI. While there are in depth reviews of prior work in public displays [20], and eye tracking [26, 27, 69], in this chapter we focus on the developments that led to the emergence of gaze-enabled public displays. We first provide an overview on the history of displays, and how they developed to become the public, pervasive, and interactive displays that we encounter nowadays. Second, we discuss how the tracking of eyes developed from a method to study and understand humans, to an effective modality that comes with challenges and opportunities for 1) attention monitoring, 2) implicit interaction, and 3) explicit interaction. Finally, we focus exclusively on previous work in gaze-enabled public displays, and conclude with trends and limitations in previous work.
2.1 The Advent of Pervasive Public Displays

For thousands of years, humans utilized different types of media to publicly display content. Drawings were painted on cave walls in prehistoric ages. Mesopotamians, Pharaohs, and Mesoamericans engraved text and drawings on walls, pillars, and temples. Ancient Egyptians made wall posters out of papyrus [12], and advertising plates that date back to ancient times were found in China [66]. Public displays in the form of signs were also used in the middle ages for advertising and making announcements. Public displays thrived in the 1800s, when they appeared in the form of painted billboards to advertise circuses and theatres [109].

Fast-forward to today, and we will find that digital displays have become ubiquitous in almost all public spaces. Industries saw the benefits of distributed networks of interactive displays, and rushed to deploy interactive displays as ticket vending machines in train and bus stations, interactive information boards in airports [8], and self check out terminals in supermarkets [77]. Universities and governments started to deploy displays in city centers [85].

HCI researchers took it upon themselves to explore novel uses of public displays. As early as 1980, Galloway and Rabinowitz introduced their prototype “Hole in Space” [32]. Hole in space was presented as an art installment, with the core idea of linking passersby around a display in New York with others around a display in Los Angeles. The opportunities arising from connecting distant places using interactive public displays is still pursued up until this decade [36]. Researchers explored other uses of public displays. For example, they were used for collecting votes in public with the aim to encourage civic discourse [103, 114]. Other displays aimed at promoting community interaction and place awareness [76], nudging behavior change [94], and visualizing urban information [17, 58]. Public displays were also found to be effective platforms for crowdsourcing [34] and serious games [48].

2.1.1 Uniqueness of Public Displays

Today’s interactive public displays correspond to what Weiser [124] described as boards. Indeed, the unique aspects of public displays justify the recognition of “boards” as a distinct computing device. Although public displays share some similar aspects with mobile devices and desktop computers, they are profoundly different in many ways. Researchers have studied the distinct audience behavior around public displays. For example, several works reported on the honeypot effect [14, 20], which refers to the influence that users, who are engaged with the display, induce on surrounding passersby. Another unique behavior is the landing effect [83], which describes cases in which a passerby realizes that a display is interactive after walking by the display, and then returns back to examine it.
When it comes to interaction, public displays are again unique compared to other platforms. For example, unlike desktop computers, users spend much less time interacting with public displays; average interaction durations with public displays were reported to be between 23 and 26 seconds in several deployments [33, 83]. Consequently, public displays need to be “immediately usable” [81]. Another distinction is that unlike desktop settings where users are mostly stationary and interact from the same position every time, users approach public displays from different positions, and could perceive content or even interact while they are moving [82].

For these reasons, we argue that the unique aspects of pervasive public displays warrant an in-depth investigation of how gaze-based interaction can be leveraged and enabled in this distinct domain.

2.1.2 Interacting with Public Displays

Present day’s deployments of public displays feature interactivity in different forms. Some displays are implicitly interactive, that is, they react to the user’s natural behavior, for example, when they approach the display. Previous deployments displayed silhouettes that mimic users’ movements in an effort to make the display noticeable and fight interaction blindness [55, 83]. Other displays allowed explicit input through modalities beyond keys and buttons, such as touch [87], mid-air gestures [72], mobile devices [84], and recently also eye gaze [130].

2.1.2.1 Touch

Touch interfaces were a significant improvement over physical hardware (e.g., keypads, buttons and joysticks) since they expand the entropy of interaction possibilities, and allow faster software-based updates of the user interface. However touch comes with its short-comings. Namely, touch interfaces have to be physically reachable, while public displays are often mounted above user’s height for visibility, or placed behind shop windows. Touch interactions also come with hygienic concerns, and could leave traces that can be maliciously exploited to extract the user’s interactions [9, 79]. This means that deploying touch-based displays requires making them reachable and thereby protecting them from 1) theft, 2) unauthorized access control, 3) intentional or unintentional damage [20].

2.1.2.2 Mid-air Gestures

Interacting via mid-air gestures was proposed as an alternative modality to touch. It allows at-a-distance interaction, and can extend interaction durations due to their playful nature [3]. On the downside, mid-air gestures require teaching passersby how to perform them [2, 121], and can be socially embarrassing in public [80].
2.1.2.3 Mobile Device Interaction

Several works proposed using mobile devices (e.g., smartphones or smartwatches) as gateways to public displays. They are promising platforms for carrying profiles of their owners, hence paving the way for straightforward personalization of public displays [21, 59, 75]. They can also be used for gesture-based interaction with remote displays [23, 93]. A disadvantage of mobile device interaction is that it often requires configuring the personal device (e.g., installing an app, or connecting to a server) prior to interaction – hence the user needs to be “prepared” for interaction. Users might be reluctant to do that if they will only occasionally interact with the display, or if they do not see a clear benefit of the display yet. In case of smartphones, users might be unwilling to take it out of their pocket or bag [71].

2.1.2.4 Gaze

Recently, eye gaze has been gaining attention as an alternative and promising modality for interaction with public displays. Unlike touch, gaze can be used for at-a-distance interaction. Unlike mid-air gestures, gaze is subtle and less embarrassing to use in public [53]. Unlike mobile device interaction which often require installing app or taking devices out of one’s pockets, gaze is fast and intuitive [100]. Gaze provides a lot of opportunities in addressing many environmental and technical challenges on public displays, such as the observability of user’s input, and the parallax effect – we summarize some of these opportunities in chapter 3, and discuss them in details in chapters 6, 7 and 8.

We discuss the advantages and limitations of eye tracking in the following section.

2.2 The Eyes in Human-Computer Interaction

Since the 19th century, eye-gaze tracking has been employed in numerous fields, such as optics, medicine, as well as behavioral and psychological research. The first studies were done using direct observations [41]. For example, Javal identified saccades for the first time by studying eye movements while subjects were reading using a mirror [45]. In 1978, Rosen et al. published a report on an eye-based system for non-vocal communication for the disabled [95]. The earliest published research about the use of eyes for interaction with computers was by Bolt [13] in 1981. Bolt introduced the “World of Windows”, in which users could interact by gazing at one of multiple displays to zoom into its content. Multiple follow up works about eye tracking in HCI contexts were published afterwards [40, 65, 123]. Jacob then explored eye gaze in further details for human-computer interaction [42, 43], and identified the so-called “Midas Touch” problem on gaze-based interfaces, which refers to the problem of distinguishing whether a user is looking at something in order to perceive it, or in order to control it.
2 Background and Related Work

2.2.1 Opportunities of Eye Tracking

When tracked, the eye movements can be leveraged for understanding users. Eye movements can reflect many aspects that go beyond visual interests, such as cognitive processes, thoughts, activities and intentions. For example, eye behavior can communicate how tired, or how much cognitive demand the user is under [90]. Eye movements can give hints on a user’s language proficiency [50]. Last but not least, gaze monitoring is widely used for usability evaluations [91].

In addition to tracking, eye movements can also be leveraged for interaction. Two types of gaze-based interaction are distinguished: explicit gaze-based interaction, and implicit gaze-based interaction. Explicit gaze-based interaction refers to intentionally moving the eyes to signal commands. One example of explicit gaze-based interaction is eye typing [70], where users dwell at letters to type them. While the majority of work on gaze for explicit interaction was intended for desktop settings, researchers explored other contexts including mobile devices [126], smart homes [115], and smart watches [31]. On the other hand, implicit gaze-based interaction refers to cases where the system reacts to the user’s natural eye movements, i.e., the user does not control the eyes intentionally. For example, Smith et al. proposed implicit interaction by turning screens on when the user gazes at them [102], and Mariakakis et al. proposed SwitchBack, which tells the user which line they were reading before looking away from the display [73].

Researchers have already summarized eye tracking applications and opportunities [27, 69]. In our work we focus on eye tracking on public displays (see Section 2.3).

2.2.2 General Limitations of Eye Tracking

Eye tracking technologies do not come without limitations. As mentioned earlier, one of the most prominent problems of gaze-based interfaces is the Midas touch [42]; the eyes are perceptual organs, hence systems need to distinguish whether the user is looking at content to control it, or to merely perceive it. To counter that problem, researchers employed different methods, such as interaction by dwell time—gazing at the target for some time before activating it. Others proposed leveraging special eye movements, such as smooth pursuit [118], optokinetic nystagmus [44], and gaze gestures [25]. A second problem is eye tracking accuracy. Usually accuracy is improved if the user goes through a calibration process. However calibration is often tedious and tiring to perform [89]. Furthermore, even with perfect calibration, it is not feasible to distinguish which part of the user’s 2° of visual view is being attended to [69]. The reason is that humans switch attention within these two degrees without moving their eyes. Another issue is that today’s eye tracking technologies are influenced by light conditions – sunlight disturbs the infrared sensors in commercial eye trackers, and video-based eye tracking is not reliable in the dark.
2.3 Previous Work on Gaze-enabled Displays

Gaze has been utilized in numerous ways in HCI. Previous works classified gaze-based systems into different clusters. For example, Duchowski classified them into interactive and diagnostic applications [27], while Majaranta and Bulling classified them into explicit gaze-based systems, attentive user interfaces, gaze-based user modeling, and passive eye monitoring [69]. Similarly, eye tracking can be employed in different ways for public displays [53]. We classify the uses of gaze on public displays into three categories: (1) Attention Monitoring, (2) Implicit Gaze-based Interaction, and (3) Explicit Gaze-based Interaction.

2.3.1 Attention Monitoring on Public Displays

The first category is attention monitoring: which refers to systems that monitor the gaze behavior of passersby to quantify attention [5, 19, 106]. These systems are built with the aim of understanding where passersby look. This means that in contrast to the other categories, they do not react to the user’s gaze, but rather silently monitor visual attention for post-hoc analysis and diagnostic purposes.

Many of the models that describe the behavior of the audience around public displays distinguish a phase where users start to become aware of the presence of a display [14, 78, 81]. Knowledge about when, how, and why users transition to this phase can help public display owners make key design decisions to optimize for the display’s use case. For example, if a certain type of animation was found to attract the passerby’s attention, designers could program the displays to show these animations more often or when someone approaches. Eye tracking is hence valuable on public displays since it can provide a reliable measurement of visual attention. This, in turn, can be used to assess the methods for attracting attention, and the display’s overall effectiveness [81]. Furthermore, public displays have long been missing an equivalent for a user “clicking through” content [20]. Eye tracking can provide an alternative metric by quantifying visual attention to elements on the display.

There are two lines of previous work in this area: (1) works that proposed technical methods to quantify attention by leveraging eye gaze, and (2) works where gaze behavior detection was applied and used to understand passerby’s attention.

2.3.1.1 Methods to Monitor Attention

Previous works tried to estimate attention to the display using the head-pose [10, 101, 120]. More recently, some works incorporated the gaze direction to estimate attention to public displays [5, 106]. Alt et al. used a geometric-based gaze estimation method [5]. Geometric-based approaches first detect facial and eye features such as the pupil’s center and eye corners, and then try to fit a geometric
model of the eye onto the user’s eye. A gaze vector is then extended from the center of the estimated eye ball to the pupil’s center, to eventually intersect the display at a certain point. This point is then deemed to be the gaze point. Geometric approaches are sometimes referred to as feature-based approaches if they rely solely on the features without model fitting. While geometric approaches can often yield highly accurate gaze estimates, they are susceptible to changing lighting conditions.

On the other hand, works such as AggreGaze [106] employ appearance-based gaze estimation approaches. Appearance-based approaches directly map images of the user’s eyes and/or face to gaze estimates based on training datasets and machine learning, e.g., convolutional neural networks [130]. Appearance-based approaches are often more robust against different lighting conditions, and can even be optimized by artificially changing the illumination levels of the images in the training dataset [125]. Their downside is that they require training data.

2.3.1.2 Attention Monitoring in Practice

Apart from technical works, in this section we discuss findings from actual deployments that employed eye tracking to remotely monitor the audience’s attention. For example, Ravnik and Solina [92] mounted a camera on a public display and measured attention time to find that passersby gazed at the display for an average of 0.7 seconds. They found that men and children are more likely to gaze longer at the display compared to women and adults respectively. They also found that showing dynamic content results in higher attention time. Using face detection and machine learning, ReflectiveSigns [82] scheduled content to be displayed based on previous experiences of which content attracted passersby attention the most.

Other works investigated the use of mobile eye tracking while recruited participants walked in front of displays. For example, researchers have run studies to quantify attention to public displays in transportation and shopping streets [39, 97]. Dalton et al. [19] recruited 24 participants, and asked them to walk through a shopping mall while wearing mobile eye trackers. They found that users gazed at displays for brief durations of approximately 0.3 seconds on average. Their results also suggest that the architecture of the building in which the displays are deployed influences the gaze behavior.

2.3.2 Implicit Gaze Interaction with Public Displays

Second, rather than requiring users to intentionally control their eyes, implicit gaze interaction refers to adapting the experience based on the user’s natural eye behavior. This area is sometimes referred to as attentive user interfaces [69, 98, 116]. Implicit gaze interaction can bring a myriad of benefits to interactive public displays. The user’s gaze could reflect interests [44, 68], and intentions [61].
vantage of implicit interaction is that it does not require training the user to interact via gaze, since no explicit eye-based actions are needed.

This type of gaze-based interaction can be classified to two categories: (1) systems that react once the user establishes eye contact with the system, and (2) systems that continuously react to the estimated gaze point based on the user’s eye movements.

2.3.2.1 Eye-Contact Aware Implicit Interaction

A straightforward way to exploit implicit gaze-based interaction, is to program displays to be attentive to the user’s gaze, i.e., react once users look at the display [117]. Relatively less research has been done in this area. Early works detected eye contact to mobile devices [22, 117], and home appliances [99]. Similarly, Smith et al. [102] and Zhang et al. [127] proposed methods for detecting eye contact on public displays, and accordingly trigger actions, such as turning the display on.

2.3.2.2 Gaze-Behavior Aware Implicit Interaction

While the aforementioned works focused on eye contact detection, other works estimated which on-screen objects the user is looking at and adapted the experience accordingly. For example, PeepList [49] is a pervasive display that estimates the perceived importance of on-screen information through the user’s gaze. The system dynamically generates a list of content sorted by importance. Mubin et al. [80] developed an intelligent shopping window where the system responded to user’s gaze towards products, which was determined via head tracking. Brudy et al. [15] and Eaddy et al. [28] exploited eye tracking to protect public display users from shoulder surfing by hiding private content that passersby gaze at. Schmidt et al. [96] displayed text that follows the user’s eye gaze as they walk past large public displays. Lander et al. developed the collaborative newspaper [63], a system that allows multiple users to read different parts of the same article simultaneously. Karolus et al. [50] detected language proficiency on public information displays through eye gaze, and accordingly showed translations of the on-screen text.

2.3.3 Explicit Gaze Interaction with Public Displays

The final category is explicit gaze-based interaction, which refers to explicitly employing eye movements to provide input on public displays.

Explicit gaze interaction is attractive for public displays because it is fast [100], and natural in most of the times [118]. Furthermore, gaze-based input can be provided at a distance, which is particularly useful for public displays that are physically unreachable; users can still interact by gaze or a combination of gaze and mid-air gestures even if the display is placed behind glass windows or above user’s height. Finally, gaze-based input has several advantages over existing input methods. For
example, input via touch is associated with hygienic concerns, while mid-air gestures are sometimes embarrassing to perform in public [72]. Explicit Gaze interaction can be further dissected into (1) explicit gaze-only interaction, where gaze is the sole input method, and (2) explicit gaze-supported interaction, where gaze is used to alongside another input modality to improve it.

2.3.3.1 Gaze-only Interaction

Gaze has also been used as the sole input modality for interaction with public displays. Sippl et al. [101] estimated the user’s horizontal and vertical gaze, to determine which of four quadrants the user is looking at. Zhang et al. [128, 130, 131] estimated the horizontal gaze direction to browse content, such as news and albums. Vidal et al. [118] introduced the use of smooth pursuit eye movements for calibration-free gaze interaction with public displays. They referred to their method as “Pursuits”, and since then it has been used in multiple applications [18, 67]. Jalaliniya and Mardanbegi proposed using Optokinetic Nystagmus for interaction, i.e., an eye movement that features multiple saccades followed by an extended smooth pursuit movement after finding the target of interest [44]. While they do not evaluate their system on a public display, they present public displays as one of the main domains where their method is promising. In GazeProjector [62], users interacted via eye gaze with surrounding displays by wearing a mobile eye tracker. Gaze was estimated on displays with the help of an edge detection algorithm that identifies borders of surrounding displays.

2.3.3.2 Gaze-supported (Multimodal) Interaction

Gaze can be effective when used alongside other input modalities. For example, Stellmach and Dachselt [104, 105], as well as Turner et al. [110] combined gaze with touch to facilitate the manipulation of targets on large public displays. Gaze has been combined with other modalities for transferring content from public displays to personal devices. For example, Mäkela et al. combined mid-air gestures with gaze to retrieve content from public displays [72], Turner et al. used multimodal touch and gaze input to transfer content on public displays [111, 112, 113]. Gaze was even combined with feet for interaction with situated displays in operation rooms [37].

2.4 Trends and Limitations of Previous Work

The early days of gaze-enabled displays focused on the technical realization of eye tracking and gaze-based interaction. With the advancements in hardware and visual computing, researchers and practitioners moved from using the head-pose and face orientation [80, 96, 101, 120], to using the actual gaze point estimated based on images of the user’s eyes [5, 128].
Previous work focused on specific aspects of gaze-enabled displays, such as how to estimate gaze based on face orientation and head-pose \([80, 96, 101, 120]\), methods for calibration-free gaze interaction \([25, 44, 118, 129]\), using mobile eye trackers for interaction with public displays \([62, 64, 111]\), measuring the audience attention \([5, 19, 39, 92, 97, 106]\), and inventing novel concepts for gaze-supported interaction \([72, 88, 104, 105, 110]\). However, existing work is lacking in several directions.

First, there is a lack of interactive systems that leverage gaze to address “practical issues” that are specific to public displays. In our work, we propose both implicit and explicit gaze-based approaches to address important problems on public displays, such as the parallax effect (Chapter 7) and secure authentication (Chapter 8).

Second, there is a lack of work in enabling remote gaze-based interaction for users of large public displays who are, in contrast to desktop settings, typically approaching the display from different directions, and even interacting while moving. We explore this unique aspect of public displays in our implementation and evaluation of EyeScout (Chapter 9), which enables gaze-based interaction from spontaneous positions in front of the display, as well as while moving past the display.

Third, with the exception of the work of Pfeuffer et al. \([89]\), there is a lack of work in intuitive calibration methods that are suitable for public display deployments. We propose a novel calibration method that calibrates eye trackers while users read text on public displays (Chapter 10), which is one of the most common types of content on public displays \([6, 38, 57, 86, 107, 114, 119]\).

Fourth, when designing gaze-based interaction, it is often the case that the configuration that results in the highest accuracy is not the most usable one (e.g., in terms of interaction speed). We make this observation in our TextPursuits project (Chapter 10), and we find this trade-off as well when choosing between highly accurate calibration methods that take time and faster ones that are inaccurate (cf., 9-point calibration vs 5-point calibration). It is not clear if public display users would prefer highly accurate methods, or would rather tolerate the inaccuracies of more usable and faster methods. We investigate this issue in our in-the-wild field study of a gaze-based voting system (Chapter 11).
II

SUMMARY OF PEER-REVIEWED PUBLICATIONS
Opportunities of Gaze on Public Displays

Eye tracking can be leveraged in many ways to bring tangible benefits to public displays. In Chapter 6, we provide a review of challenges of pervasive public displays, and discuss how gaze shows promise in addressing many of them. For example, detecting the passerby’s attention is a challenge at the outset of interaction with displays [47, 83]; gaze can be an indicator of the user’s attention towards the display [19, 102]. Passersby are trained to ignore displays in public and often do not know that they are interactive [60, 74, 83], but gaze can be leveraged to communicate that the display is interactive by showing call-to-action labels [20] right where the user is looking, hence making it unlikely the user will miss it. In many cases displays are behind glass windows for security, or above the user’s height for visibility; gaze can be used at-a-distance and hence allows interaction with displays that are physically unreachable. Gaze is generally subtle and hence less embarrassing to use in public compared to other modalities, such as mid-air gestures [14, 72]. Finally, interactive public displays require “immediate usability” [20, 81]; gaze is fast (e.g., faster than pointing [100]) and hence promising for enabling immediate usability.

In this work we focus on gaze-based interaction, that is, allowing users to intentionally or unintentionally provide input that is either supported by eye gaze or entirely done using the eyes. We address issues that are particular to interacting with displays in public space. Namely, the parallax effect, and the observability of the sometimes sensitive user input. In the following, we summarize our contributions in implicit and explicit gaze-based interaction with public displays.

• Section 3.1 describes EyePACT, an implicit gaze-based method that overcomes the parallax effect, which is a recurrent problem on today’s touch-based public displays.

• Section 3.2 describes GTmoPass, a secure multimodal approach that combines gaze and touch input for explicit user authentication on public displays.
Figure 3.1: Public displays are often augmented with thick layers of glass to protect them against vandalism and harsh weather. This results in a gap between the touchscreen and the display, which in turn introduces the parallax effect illustrated in (A). The parallax effect refers to the displacement between the perceived touchpoint and the active one, and can be observed on today’s public displays. (B) shows a situation where parallax was observed on a display in Münchner Freiheit station in Munich, Germany. EyePACT solves this problem by leveraging the eye’s position.

3.1 EyePACT: Implicit Eye-based Parallax Correction

Problem: Public displays that employ touch interaction utilize touch sensitive (capacitive) surfaces at the outset of the display. A problem arises when displays are deployed in public space; public displays are often augmented with thick layers of glass to protect them against vandalism and harsh weather conditions. These protective layers are placed in between the touchscreen and the display, resulting in a relatively large distance between the surface being touched and the display elements. As shown in Figure 3.1A, the distance between the display and touchscreen causes a displacement between the perceived touch point (red area in Figure 3.1) and the touch point detected by the system (green area in Figure 3.1). This effect is referred to as the Parallax effect.

Opportunity: As described in section 2.3.2 knowledge about the user’s gaze direction can implicitly complement interactions when using other modalities, such as touch. We exploited implicit gaze-based interaction to address the parallax effect, a prominent problem on today’s interactive displays.

EyePACT: EyePACT is an Eye-based Parallax Correction technique for Touch-enabled displays. EyePACT is a simple, yet effective, method to correct for parallax by estimating a gaze vector that starts from the user’s eyes, intersecting the user’s touch point, and then finally intersecting the display at a certain point. The touch event is then initiated at the point the user is intending to touch. This work is the first to systematically evaluate the concept of eye-based parallax correction; we built an apparatus to create and control the parallax effect in our lab.
3 Opportunities of Gaze on Public Displays

Figure 3.2: GTmoPass employs explicit gaze-based interaction to allow users to securely authenticate on public displays using multimodal passwords consisting of gaze and touch input, and a personal mobile device.

Evaluation: We carried out multiple studies (N=38) to evaluate the method’s effectiveness and how well it is perceived by users in case of single and multiple users, different target sizes, and different levels of parallax. When evaluating EyePACT for multiple users, we also found that it prevents interfering fingers when multiple users interact with the same on-screen target via touch (see Figure 3.1C). This presents an opportunity that could enable novel multiuser interaction concepts.

Findings: In summary, we found that EyePACT significantly improves the accuracy of touch-enabled displays even with varying gap distances between the touch surface and the display, that it adapts to different levels of parallax, and that it maintains a significantly large distance between multiple users’ fingers when interacting with the same target. This project is discussed in more details in Chapter 7.

3.2 GTmoPass: Explicit Gaze-supported Authentication

Problem: There are many situations in which users need to securely authenticate on public displays. In addition to ATMs, users could authenticate at public terminals when buying tickets for museums, buses, etc. Furthermore, the increasing demand for personalized and context-related experiences on public displays [21] underlines the need for secure user authentication to ensure that only the legitimate user accesses their potentially sensitive data. However, many of today’s concepts for authentication are vulnerable to various types of side-channel attacks, such as shoulder surfing [29], smudge attacks [9], and thermal attacks [1].

Opportunity: As explained in section 2.3.3, in addition to the implicit use of gaze, gaze can also be employed explicitly to provide input. An advantage stemming from the subtleness of eye movements, is that it can be employed for inconspicuous authentication in public space. This benefit is amplified when gaze is combined
with another modality, since each additional modality translates to an additional channel of input that attackers need to observe. In our work, we leveraged explicit gaze-based interaction for secure authentication on public displays.

**GTmoPass:** Users authenticate using GTmoPass by providing multimodal pass-words on their mobile devices. The password consists of gaze and touch input (see Figure 3.2A). The password is then securely transmitted to a server through Bluetooth beacons. The proposed system is a two-factor authentication scheme where the multimodal password is the knowledge factor (i.e., something the user knows) and the mobile device is the possession factor (i.e., something the user has). Gaze input is detected through the front facing camera of an off-the-shelf mobile device, and touch input is directly provided on the touchscreen. This way, in order for an adversary to attack the user, the adversary would need to (1) observe the user’s eyes to find the gaze input, (2) observe the phone’s screen to find the touch input, (3) combine the observations to generate the password, and (4) acquire the mobile device (e.g., by theft).

**Evaluation:** We evaluated the usability and security of the concept (N=32). Usability of GTmoPass was evaluated in the lab by inviting 16 participants to enter Gaze-Touch passwords on their mobile devices (Figure 3.2B and 3.2C). Afterwards we conducted a security study with 16 different participants, who took the roles of attackers and tried to observe passwords by watching videos of users authenticating using GTmoPass (Figure 3.2D).

**Findings:** We found that although authentication using GTmoPass is slightly slower than traditional methods, it is highly resilient against the aforementioned threats. This project is discussed in more details in Chapter 8.

**Acknowledgement:** This project was supported by Google IoT research award received in March 2016.
Challenges of Gaze-enabled Public Displays

Although research in gaze detection and gaze-based interaction is already well established for desktop settings, the domain of gaze-enabled public displays is unique and imposes novel challenges that are under-investigated. In our work, we identified three main unique aspects of gaze-enabled public displays that require further investigation. Namely, in contrast to desktop settings where the user is (1) positioned in front of the eye tracker, (2) mostly stationary, and (3) interacting for a long period of time, public displays (1) expect users to interact from different positions, (2) expect users to move while interacting, and (3) cannot afford requiring time-consuming eye tracker calibration.

We focus on the use of remote eye trackers; that is, we investigate gaze-based interaction with public displays without augmenting the user. While head-mounted eye trackers are becoming affordable [51], and show promise in tackling the aforementioned challenges [62], they are still special-purpose equipment that require augmenting individual users [64]. Indeed there is a vision of having eye trackers already-integrated into daily Eyewear [16], and also the vision of having pervasive interconnected display networks in the future [20]. Accordingly, it is possible that in the future passersby would be wearing eye trackers that are able to communicate the gaze behavior to surrounding displays. However, a pervasive integration on such a big scale would require taking concepts from lab settings to the field, which is currently challenging to investigate using mobile eye trackers unless participants are explicitly hired [19]. Until wearing mobile eye trackers becomes a norm, there is a need to study the natural user behavior on gaze-enabled public displays using remote eye tracking.

In the following we discuss how we addressed these unique aspects of gaze-enabled public displays in our research.

1. Section 4.1 describes our prototype system, EyeScout, that allows movement and position independent gaze-based interaction via active eye tracking.

2. Section 4.2 summarizes our investigation of implicit calibration (Read2Calibrate) and explicit interaction (EyeVote) with textual content on public displays via gaze.

3. Section 4.3 summarizes an in the wild deployment of EyeVote, that allowed us to investigate the trade off between accurate and fast gaze interaction techniques, and the implications on willingness of passersby to correct system errors resulting from inaccuracy on gaze-enabled displays.
Figure 4.1: EyeScout is an active eye tracking system that allows users to interact via gaze from random positions in front of large public displays (A), or while moving past the display (B).

4.1 EyeScout: Movement and Position Independent Gaze Interaction

Problem: Public displays expect users to interact from different locations, distances, and orientations relative to the display [83, 122]. Large public displays in particular allow passersby to interact while on the move [96]. On the other hand, most existing remote eye tracking approaches require users to keep their head facing the tracker in a relatively confined tracking box.

EyeScout: In our work, we explored the use of “active eye tracking” for the first time for public displays to address these issues. Active eye tracking refers to the automatic adaptation of the eye tracking hardware to adapt to the user’s state rather than restricting the user’s movements. We do so by presenting EyeScout, a novel active eye tracking system that enables gaze interaction on large public displays from different lateral positions in front of the display and while on the move (see Figure 4.1). EyeScout consists of two main modules: a body tracking module, and an eye tracking module that is mounted on a rail system. The system detects the user’s position, and then dynamically adapts the eye tracker’s position by moving it to face and following the user. This way, EyeScout (1) allows gaze-based interaction with large public displays for users interacting from different positions within a certain distance in front of the display (Figure 4.1A), and (2) allows users to interact via eye gaze while moving in front of the display (Figure 4.1B). We evaluated EyeScout for scenarios in which users “Walk then Interact” (to test for position independence), as well as “Walk and Interact” (to test for movement independence).

Findings: We found that EyeScout is well-suited for both interaction modes, it is well perceived, and it reduces the interaction initiation time by 62% compared to state of the art systems [7, 130]. Based on these positive results, we proposed several contexts in which EyeScout can be particularly useful, such as large displays along escalators and moving pathways, as well as large cylindrical displays. This project is discussed in more details in Chapter 9.
4 Challenges of Gaze-enabled Public Displays

Figure 4.2: In our project TextPursuits, we implemented and evaluated two systems: (A) Read2Calibrate allows users to implicitly calibrate the eye tracker while reading text on public displays, and (B) EyeVote allows users to vote on public displays by picking one of several text-based options by following them with their eyes. Through two user studies, we identified several factors that influence the accuracy and user perception, as well as a trade-off between accuracy and perceived user experience.

4.2 TextPursuits: Implicit Calibration and Calibration-free Gaze Interaction

Problem: Immediate usability is one of the main requirements of interactive public displays [81]. Users interact with public displays for very short durations [33, 81, 83], hence it is very important to make sure that the time needed to kick off interaction on public displays is as fast as possible. On the other hand, eye trackers typically need to be calibrated for each user [69]. Calibration is a procedure in which the user is asked to gaze at several fixed points on the screen to allow the eye tracker to collect eye images, and establish a mapping of eye movements to gaze points on the screen. While calibration results in more accurate gaze estimation, the process is tedious and time-consuming to the user [89]. Investing time to calibrate an eye tracker in a desktop setting might be acceptable because the user is going to interact for a relatively long period of time. In contrast, spending a big portion of the interaction duration solely to calibrate the eye tracker is unacceptable in public settings [53]. In the following, we discuss two prototypes: Read2Calibrate and EyeVote that demonstrate two approaches to address this problem.

Read2Calibrate: One way to reduce the overhead caused by calibration is to make the process easier by interleaving it into the user’s usual interactions. Read2Calibrate is an implicit calibration method, with which users calibrate eye trackers simply by reading text on a public display (Figure 4.2B). The idea is motivated by the abundance of text-based content on public displays that the users read anyway. Read2Calibrate shows animated text on the screen and feeds the gaze data to an algorithm that gradually enhances the calibration of the gaze tracking. The concept builds over previous work in Pursuits, in which moving targets were shown to calibrate as users follow the moving targets with their eyes [89].
Findings: In a user study (N=18), we found that text-based calibration is not as precise as state-of-the-art calibration procedures, but it is sufficient to identify the gaze area within 6° of visual angle. The results show that the inclination of the shown text and the speed at which it is gradually revealed have a strong effect on accuracy. Most interestingly, the findings indicate that there is a trade-off between the configuration leading to the most accurate results and the users’ preferred configuration.

EyeVote: Another approach to reduce the overhead caused by calibration is to completely eliminate it and use calibration-free gaze techniques, such as Pursuits [118]. While Pursuits has been used for many use cases, such as games [52, 118], authentication [18], text entry [67], and smart rooms [115], it had never been explored with text prior to our work. Pursuits is not necessarily straightforward to employ with textual stimuli: reading is typically associated with saccadic eye movements rather than smooth pursuit. Hence, eye movements performed while reading a piece of text that is moving could overlay the smooth pursuit movement, which could in turn result in difficulty in correlating movements of the target with that of the eyes. Also, due to the Midas effect, gaze-based systems need to distinguish users reading textual content from interacting with it. To better understand this, we implemented EyeVote (Figure 4.2A), a survey application that allows users to cast their votes on public displays by selecting one of several text-based options.

Findings: Results from a user study (N=19) revealed that representing text in short concise form makes it suitable for Pursuits-based interaction. We also found that moving text-based targets in circular motion results in the highest accuracy, but worst user experience since circular eye movements are demanding and tiring. These projects are discussed in more details in Chapter 10.

4.3 EyeVote in the Wild: Understanding the Trade-off between Accuracy and User Experience

Problem: An outcome of both systems that were discussed in Section 4.2 is that the configuration that results in the highest accuracy is not necessarily the one that results in the best perceived user experience. In fact, there seems to be a trade-off between accuracy and user experience. For example, in Read2Calibrate, users preferred to read flat text because it is more natural to read, but text inclined at 45° yielded the highest calibration accuracy because it spans across both the x- and y-axes. Similarly, in EyeVote, users found targets moving in circular motion to be the most demanding to select. Yet the fact that circular eye movements is not common also meant that circular trajectories result in the least false positives. This presents designers with a challenge: should we design interfaces that are highly accurate even though it might be ill perceived? or should we use the less accurate methods and allow users to correct or confirm their input? While similar questions
Figure 4.3: We deployed EyeVote in a field study to understand the willingness of public display users to correct system errors. EyeVote was configured to occasionally show intentionally falsified feedback and prompt users to correct their vote. Results from a field study show that most participants do correct system errors when the correction method is straightforward.

were investigated in several domains, public display systems pose very unique design challenges in terms of error recovery. Public display users interact for very short amounts of time and are believed to abandon the display when interrupted or forced to deviate from the main task [35, 121].

EyeVote in the Wild: To answer these questions, we conducted an in-the-wild field study of EyeVote where we deployed the system in a university campus. To experiment with the users’ willingness to correct system errors, we programmed EyeVote to intentionally evoke errors to see if users correct them. We did that by showing falsified feedback; this was done by showing an answer in the recap view that was not among the options the user had available to choose from.

Findings: We found that public display users are willing to correct system errors provided that the correction is fast and straightforward. For example, we found that users are less likely to correct errors if the correction mechanism requires a change of interaction modality. This project is discussed in more details in Chapter 11.
III

Conclusion and Future Work
5
Closing Remarks

This dissertation investigated the design of gaze-based interaction for pervasive public displays. This was done through two lines of work. First, based on an in-depth review of challenges of pervasive public displays, we identified a manifold of opportunities eye gaze could bring to this setting. We then proposed the concept, implementation, and in-depth evaluation of two systems that address profound problems on today’s pervasive public displays, namely, these systems offer accurate parallax correction (Chapter 7), and secure user authentication (Chapter 8) on pervasive public displays. Second, we identified core challenges of gaze-enabled public displays that are unique to this setup and are very different compared to previously investigated challenges. Specifically, we identified 1) the problem of supporting interaction from different user positions, including cases where the user is moving (Chapter 9), and 2) the issues pertaining to calibration, where we investigated implicit calibration (Chapter 10), calibration-free interaction (Chapters 10 and 11), and input correction in case of fast but inaccurate gaze-based selections (Chapter 11).
5.1 Design Recommendations

In addition to the individual recommendations in each of Chapters 9, 10, and 11, we highlight the following recommendations for designing gaze-based interaction on pervasive public displays:

1. **Employ Active Eye Tracking for Gaze Interaction in Public Space**: Gaze-enabled displays need to adapt to dynamic users who could approach the display from different directions or interact from different positions and while moving. In EyeScout (Chapter 9), we employed active eye tracking, i.e., dynamically adapting the tracker to the user’s position, with success. Active eye tracking can be achieved in many other ways, such as by using drones, or using Pan-Tilt-Zoom cameras for eye tracking.

2. **Use Implicit Calibration, or Calibration-free Gaze Interaction Techniques**: Calibration is tedious and unacceptable for public interactions, especially if the interaction duration is expected to be short. Employing implicit calibration by integrating it into interaction reduces the perceived calibration overhead. This can be done by integrating calibration into reading text as we did in Read2Calibrate (Chapter 10). Alternatively, gaze interaction techniques that do not require calibration, such as Pursuits [118] and gaze gestures [25], should be used.

3. **Optimize for Fast Gaze Interaction, Even if it is at the Expense of Accuracy**: In many of our projects, we found a trade-off between gaze interaction speed and gaze interaction accuracy. We found that passersby are willing to correct system errors on public displays as long as correction methods are fast and straightforward (Chapter 11). This means that using faster interactions techniques despite their low accuracy, and allowing users to correct incorrect detections would yield better results in public space.

5.2 Future Work

Apart from public displays, the methodology followed in this dissertation can be adopted for other pervasive domains, such as mobile devices. Furthermore, while this dissertation closes multiple gaps, it highlights opportunities for future work relevant to researchers and practitioners working in eye tracking, public displays, and gaze-enabled displays.
5.2.1 Our Methodology Beyond Public Displays

Advances in ubiquitous computing brought forth a plethora of novel devices. At the same time, eye tracking technology is continuously advancing, hence nudging researchers to explore the opportunities it brings to new domains in which eye tracking is relatively less explored. For example, researchers have started studying eye tracking in domains such as handheld mobile devices [54], on smart watches [30], internet of things [115], as well as AR [31] and VR [56].

As briefly mentioned in section 1.3.3, the methodology we adopted in this work can be adapted for studying the use of eye gaze for other domains. Our methodology is as follows: 1) we identify challenges in the field of pervasive displays, 2) study how gaze can address them, 3) investigate novel challenges arising from the use of gaze in this domain, and then 4) design, implement and evaluate solutions to address these challenges. This methodology can be adapted for other domains for which eye tracking is being explored. For example, researchers interested in studying eye tracking and gaze-based interaction for handheld mobile devices can follow a similar approach: The first step is to identify the challenges users currently face on mobile devices. An example of such challenge is the difficulty of reaching and tapping on targets that are at the top of the mobile device’s screen. The second step is to experiment how eye gaze can help solve these problems. Following the same example, a solution could be to employ multimodal interaction where the user gazes at an unreachable target, and then taps on a specific area that is easily reachable. Finally, challenges that are particular to gaze-enabled handheld mobile devices need to be investigated. For example, unique challenges in this domain include the shaky environments in which mobile devices are used, the susceptibility to motion blur, and the fact that the user’s holding posture influences the visibility of their eyes in the front-facing camera’s view. In fact, we started employing this methodology for gaze-enabled handheld mobile devices in a number of side projects, which has resulted in a better understanding of the opportunities and challenges of eye gaze in this domain [54].

5.2.2 There is More to Gaze Interaction than Accuracy

Research in eye tracking has traditionally focused on improving the accuracy of gaze estimation. While improving accuracy is important, our findings suggest that there are aspects that could sometimes be more important than accuracy.

Many of our studies shed light on a trade-off between gaze interaction accuracy and gaze interaction speed – the setup that results in the highest accuracy is usually the slowest. In the particular, we found that compromising accuracy for the sake of faster interactions is more suitable on public displays; users are willing to correct
system errors of a fast inaccurate system rather than interacting with a slow accurate one. We discuss this in details in Chapters 10 and 11.

In this work, we have extensively used techniques that leverage gaze behavior (e.g., gaze gestures [25] and Pursuits [118]). These techniques are fundamentally different compared to the traditionally used gaze fixations, since they offer accurate means for interaction even though they rely on inaccurate gaze estimates. This makes Pursuits and gestures not only suitable for spontaneous calibration-free interaction, but also for interaction in dynamic setups, such as when walking along displays, where gaze estimates are likely to be inaccurate. We discuss this in more details in Chapter 9. This demonstrates another example where gaze estimates can be useful despite their inaccuracy.

While building highly accurate gaze estimation methods is important, future work in gaze-enabled displays should accept this compromise. Instead of focusing only on improving accuracy, we should additionally build methods that embrace inaccurate gaze estimates to enable highly usable interaction.

### 5.2.3 Privacy Implications of Gaze-enabled Public Displays

In this dissertation, we leveraged the user’s eyes to build systems and methods that improve interaction with public displays. However, being a rich source of information about the user, tracking the user’s eyes in public space comes with implications on privacy. Namely, eye movements do not only reflect the user’s visual attention, but also potentially sensitive information that the user might not wish to share, such as the mental disorders [46], psychiatric diseases [108], and could even reflect political temperaments [24]. Additionally, since eye trackers are basically cameras, users might be concerned about the misuse of the video feed. For example, users could think that the space owner shares their behavioral patterns with third parties to generate targeted advertisements [55].

This is a challenge for future gaze-enabled public displays. One direction to address this issue is to enforce a policy, similar to that of certificate authorities used for HTTPS, that requires space owners to certify gaze-enabled public displays that do not violate the user’s privacy. Another approach would be to communicate to the user how the video feed is being processed, or what is being done with the gaze data. To this end, gaze-enabled public displays can borrow concepts used for mobile devices; there has been work on nudging users when private data or permissions (e.g., the user’s location) is shared with a mobile application [4, 11].
OPPORTUNITIES OF GAZE-ENABLED PUBLIC DISPLAYS
Tackling Challenges of Interactive Public Displays using Gaze
Tackling Challenges of Interactive Public Displays using Gaze

Mohamed Khamis
Media Informatics Group
University of Munich
Munich, Germany
mohamed.khamis@ifi.lmu.de

Andreas Bulling
Perceptual User Interfaces Group
Max Planck Institute for Informatics
Saarbrucken, Germany
bulling@mpi-inf.mpg.de

Florian Alt
Media Informatics Group
University of Munich
Munich, Germany
florian.alt@ifi.lmu.de

Abstract
Falling hardware prices led to a widespread use of public displays. Common interaction techniques for such displays currently include touch, mid-air, or smartphone-based interaction. While these techniques are well understood from a technical perspective, several remaining challenges hinder the uptake of interactive displays among passersby. In this paper we propose addressing major public display challenges through gaze as a novel interaction modality. We discuss why gaze-based interaction can tackle these challenges effectively and discuss how solutions can be technically realized. Furthermore, we summarize state-of-the-art eye tracking techniques that show particular promise in the area of public displays.

Author Keywords
Gaze; gaze-based interaction; public displays; pervasive displays; digital signage

Introduction
Public displays have become ubiquitous in public spaces, such as shopping malls or transit areas in airports and train stations. Equipped with an increasing number of sensors, their interactive capabilities promise informative, entertaining and engaging applications that provide tangible benefits to users. Such sensors include, for example, touch screens, cameras, and depth sensors, thereby enabling interaction
6 Tackling Challenges of Interactive Public Displays using Gaze

Based on touch, mid-air gestures, smartphones, and recently also gaze. Nevertheless, the uptake of interactive public displays has been slowed due to many challenges.

This work suggests using gaze to tackle core challenges of display interaction. Gaze-based interaction has numerous advantages: gaze is intuitive [21], natural to use [22], indicates visual attention and usually precedes action [13].

Eye Tracking Techniques for Public Displays
Classical eye tracking techniques require performing time-consuming and cumbersome calibration [13]. This prerequisite has slowed the adoption of gaze for public displays. Previous work overcame this by estimating gaze with low accuracy based on head tracking and face detection [5].

More sophisticated calibration-free eye tracking methods, that show promise in the domain of public displays, were recently introduced. Examples include the work by Zhang et al. [23] that relies on relative eye-movements (e.g. distance between the pupil and eye corner). Vidal et al. [22] proposed leveraging the smooth pursuit eye movement to enable spontaneous gaze-based interaction. Pursuits has been evaluated on a public display and was shown to be well-perceived by passersby [11]. The method has also been used for flexible calibration [18]. Nagamatsu et al. [17] enabled calibration-free eye tracking using a sophisticated hardware setup (2 cameras and 8 LEDs).

With our work we aim to identify which of the techniques are suitable for particular applications and situations.

Addressing PD challenges using Gaze
Gaze can be used both implicitly and explicitly to enhance the user experience with public displays. By looking into existing work, six main challenges that hinder the uptake of interactive public displays have been identified, in this section we shed light on why and how gaze can be superior over existing techniques in addressing these challenges.

Detecting the user’s attention
Attracting and detecting the user’s attention are core challenges at the outset of the interaction process with public displays [7]. Previous work presented readily interactive displays and tried to attract the passerby’s attention using physical objects [10] or user representations such as mirrored silhouettes [16]. Although inferring “attention” to a display is complicated, the passerby’s gaze indicates overt visual attention and in many cases precedes action [13].

Recent research used wearable eye trackers to detect visual attention to displays [6]. Also remote eye trackers can be used to detect a user’s gaze, which can be a plausible indication of his/her attention to the display, particularly when combined with head orientation and body posture.

Communicating Interactivity to the Passerby
In order for the passers-by to distinguish interactive displays from static advertising screens, a public display needs to communicate that it is interactive [7]. Existing approaches include flipping an edge of the display [12], using call-to-action labels, using signs next to display [14] or assigning someone to invite passersby to interact [9]. Based on gaze-data, it is possible to show a concise call-to-action label right where the passerby is looking, at the moment s/he attends to the display. This makes it less likely to be overlooked compared to existing approaches.

Accessibility and Social Embarrassment
Another challenge is the accessibility of the displays. Touch-based interaction is not always possible due to the display’s location (e.g. in many cases the display is behind a glass window or mounted above head-height for visibility and security [7]). Gesture-based interaction is often difficult due to the lack of a generally agreed-upon gesture-to-action map.
pings. Moreover, mid-air gestures were found to be embarrassing for users in public [4], particularly if visible from afar. By using remote eye trackers, interaction via gaze becomes very subtle and can be hardly recognized by others in public; thus overcoming the embarrassment problem, while maintaining the advantage of at-a-distance interaction.

Immediate Usability
When it comes to interaction, there is the requirement of immediate usability [7]. Interaction time with a public display is often short (in seconds) [15]. So far this has been addressed by using interaction concepts that require a low learning curve, and by using call-to-action labels [14]. Gaze-based interaction is fast [19] and intuitive [21]. Gaze can also be used alongside other interaction modalities to improve their usability. Combining midair gestures and/or touch interaction with gaze tracking in public displays holds promise by, for example, adapting UIs based on users’ visual attention.

Privacy in Public Spaces
As displays become more interactive there is a need to enable personalization and to allow users to input data (e.g. add a post [2]). Consequently, displays need to deal with sensitive data (e.g. passwords), that users will be skeptical to provide in a public environment. This problem is currently mitigated by asking users to exchange sensitive data through their mobile devices [3]. Using gaze-based authentication was shown to be more secure than classical methods [8]. Moreover, previous work has demonstrated the feasibility of content exchange across devices via gaze [20]. This makes content-exchange less prone to observations and less likely to leave exploitable smudge traces.

Gaze as a Performance Indicator
Unlike websites, public displays have no equivalent of a user “click stream”, which makes it difficult to track user actions for evaluation purposes [7]. Gaze can offer metrics to quantify the performance of displays. Such metrics include, among others, dwell time and number of fixations; these can be used as indicators of attention, perception, understanding, and interest.

Limitations
Eye tracking could be challenging outdoors as the trackers can be influenced by varying light conditions. Moreover, eye trackers have usually been intended for desktop settings, where a single user interacts at a time from the same distance. However, public displays expect multiple users of different heights to interact from different positions. Recent work suggested guiding passersby to certain positions in front of displays using on-screen visual cues [1].

Conclusion
In this paper we discussed why we believe gaze-based interaction to be a promising modality for tackling many challenges related to interactive public displays and introduced different gaze tracking techniques. In addition we provided pointers for future research in this area.

References
EyePACT
Eye-Based Parallax Correction on Touch-Enabled Interactive Displays.
EyePACT: Eye-Based Parallax Correction on Touch-Enabled Interactive Displays

MOHAMED KHAMIS, Ubiquitous Interactive Systems Group, LMU Munich, Germany
DANIEL BUSCHEK, Ubiquitous Interactive Systems Group, LMU Munich, Germany
TOBIAS THIERON, Ubiquitous Interactive Systems Group, LMU Munich, Germany
FLORIAN ALT, Ubiquitous Interactive Systems Group, LMU Munich, Germany
ANDREAS BULLING, Max Planck Institute for Informatics, Saarland Informatics Campus, Germany

The parallax effect describes the displacement between the perceived and detected touch locations on a touch-enabled surface. Parallax is a key usability challenge for interactive displays, particularly for those that require thick layers of glass between the screen and the touch surface to protect them from vandalism. To address this challenge, we present EyePACT, a method that compensates for input error caused by parallax on public displays. Our method uses a display-mounted depth camera to detect the user’s 3D eye position in front of the display and the detected touch location to predict the perceived touch location on the surface. We evaluate our method in two user studies in terms of parallax correction performance as well as multi-user support. Our evaluations demonstrate that EyePACT (1) significantly improves accuracy even with varying gap distances between the touch surface and the display, (2) adapts to different levels of parallax by resulting in significantly larger corrections with larger gap distances, and (3) maintains a significantly large distance between two users’ fingers when interacting with the same object. These findings are promising for the development of future parallax-free interactive displays.

CCS Concepts: • Human-centered computing → User interface design; User centered design;

Additional Key Words and Phrases: Public Displays; Parallax; Touch screens; Gaze

ACM Reference Format:

1 INTRODUCTION

Interactive displays are becoming ubiquitous and are increasingly deployed in public areas, such as universities, train stations, airports, or shopping malls. While advances in sensing technology enable new types of interaction, such as with mid-air gestures [29] or gaze [20], touch remains among the most commonly employed modalities [4]. In the early days of touch screens, infrared sensors were mounted above the screen surface to detect where the user’s finger was touching. Hence, a relatively large distance existed between the touch screen and the display...
that led to the so-called parallax effect [34], i.e., a difference between the perceived location at a particular UI element and users’ actual touch location (see Figure 1A). This difference becomes larger as the angle between the display and the user’s eyes increases (e.g., when looking at far corners of the display). The parallax effect can trigger unwanted input actions that can negatively impact usability and user experience [31].

Advances in capacitive touch screens have resulted in the screens becoming more compact, thereby reducing the parallax effect. However, the effect has become prominent again with the widespread use of public displays (see Figure 2). The reason is that on one hand, manufacturers typically protect displays against vandalism by using thick layers of glass. On the other hand, displays are often deployed behind shop windows and touch input is enabled through a transparent foil attached to the window. This increases the distance between touch screen and display and, in turn, introduces parallax (cf. Figure 1B).

Compensating for the error caused by the parallax effect is important because it makes interaction cumbersome and time-consuming which can, in turn, lead to users abandoning the displays [13, 44]. The negative impact on user experience is particularly prominent in interfaces that require several accurately located touch events, where an inaccurate touch event could result in unintended input [31]. Examples of such interfaces include interactive maps (see Figure 1B), on-screen keyboards used to provide text input or passwords (see Figure 2), calendars for choosing departure dates, etc. These interfaces would benefit from parallax correction to overcome inaccurate selections which impacts the user experience. Although a common approach is to use larger buttons aimed at covering the active and perceived touch areas, the use of larger buttons may not be feasible in the aforementioned applications. On the other hand, software-based solutions, such as training and using offset regression models like the one we discuss in Section 5, would not compensate for parallax accurately without knowledge of the user’s height and position, as well as the number of users. Note that public displays often expect multiple users interacting simultaneously [11, 19, 29].

In this work we propose EyePACT, a simple, yet effective, method to overcome the parallax effect. The technique corrects for the parallax offset by estimating a gaze vector that intersects the user’s touch point, hence estimating where on the screen the user intended to touch. Our approach relies on a depth camera, hence EyePACT is unlikely to require a significant cost, particularly since an increasing number of displays has integrated cameras to measure audience attention [2, 35], for security purposes (e.g., ATMs), and for interaction [29]. While similar ideas have been described in some patents [3, 5, 6, 18, 40], eye-based parallax compensation has never been evaluated and it therefore remains unclear how well such compensation can work and how well it is perceived by users. This work makes the following contributions: (1) We introduce the concept and implementation of...
Fig. 2. The parallax effect has recently become prominent again with the widespread use of public displays that are augmented with thick layers of glass to protect against vandalism. The figure shows multiple cases where the parallax effect is prominent on real-world deployments. The red area is the touched area from the user’s perspective, while the green one is the area the system mistakenly activated due to parallax.

EyePACT, a simple yet effective approach for compensating for parallax. (2) We report on our findings of an accuracy evaluation of EyePACT, which show that EyePACT brings the touch points significantly closer to the target even with varying degrees of parallax. (3) We demonstrate the effectiveness of EyePACT in allowing multiple users to interact on parallax-enabled displays without overlapping fingers.

2 BACKGROUND AND RELATED WORK

Our work builds on three strands of prior research, namely (1) parallax in HCI research, (2) parallax on touch-based public displays, and (3) eye detection on public displays.

2.1 Parallax in HCI research

While we focus on one way the parallax effect could occur, there are many conditions that could lead to a similar effect. As the distance between a touch surface and a display increases, so does the discrepancy between what each of the eyes sees. This is often referred to as binocular parallax [25]. Previous work showed that when pointing at 3D stereoscopic objects, users touch between their dominant and non-dominant eye position with a bias towards the dominant eye position [39]. Addressing binocular parallax is out of the scope of this work. (1) In our work, we focus on the parallax effect on public displays, where an increase in the distance between the touch surface and the display is rarely more than 7 cm [14, 15, 30, 49]. On the other hand, binocular parallax occurs at larger distances (e.g., 30 cm between the touchscreen and the UI element [25]). (2) Employing gaze or mid-air gestures would be more suitable for contexts where the public display is physically unreachable or too far away from the user [4, 20]. And (3), our participants did not report experiencing binocular parallax in our setup.

In contrast, motion parallax refers to the depth cues that humans perceive while moving; as we move or as objects move around us, we perceive closer objects to be moving faster than farther ones. Humans use this cue (i.e., the perceived speed of objects) to estimate how far the objects are compared to each other [32]. Parallax is often exploited for simulating depth perception by using so-called parallax barriers [23], which enables stereoscopic rendering by limiting the observer’s point of view [41]. The idea is to use two stacked screens, with the front one showing an array of pinholes. By doing so, the setup exploits binocular and motion parallax cues to present the user with the illusion of depth.

2.2 Parallax in Public Displays

In contrast to the aforementioned types of parallax, we focus on the parallax error induced by large distances between the touchscreen and the display, which results in a displacement between the perceived and detected touch point on a touch-enabled public display. This was previously referred to as “parallax error” [28] or “visual
Parallax was a major issue in touch-based interaction in the early days of touchscreens [34]. The problem was significantly reduced with the advancements in Gorilla Glass and thin touchscreens. However, public displays (see Figure 1B) are still affected by parallax due to the use of vandalism-proof glass which increases the distance between the touch layer and the screen [14, 15, 30, 49].

A common approach used frequently for ATMs is to use larger buttons aimed to cover the active and perceived touch areas. Another approach is to incline the display to reduce the angle relative to the user’s view and hence reduce the parallax effect. Forcing designers to use bigger buttons is not an optimal solution and might not be feasible in interfaces where there is multitude of accurately located touch events (e.g., interactive map, month calendar, or on-screen keyboard). While inclining the display is not always feasible, in particular with large displays. Previous patents discussed solutions [3, 5, 6, 18, 40]. These methods utilize a front-facing camera of a mobile device to estimate the viewing vector. However, it is not clear how well these methods perform and how they are perceived by users. Moreover, the use case covered by these patents involve a single user interacting with a touchscreen of a mobile device. In contrast, we study parallax correction on public displays for single and multiple users. Migge et al. presented a parallax-correction approach that relied on a Wiimote’s tracking IR-markers that are attached to the user’s head [28]. On the other hand, our approach does not augment the user but relies only on a remote camera, which is readily integrated into existing public displays.

2.3 Eye and Attention Detection on Public Displays
To detect the user’s eye position in front of a public display, it is essential to first detect the user’s face. Several approaches have been proposed for face detection, examples are the active shape model (ASM) [10], the active appearance model (AAM) [9], the gradient-based approach [38] and the Viola-Jones detection framework [43]. After detecting the user’s face, face landmarks are determined. These landmarks include features of the user’s eyes (e.g., eye corners, pupils, etc.). In our work, we use an optimized version of the AAM algorithm that incorporates depth-based face segmentation by using an RGBD camera (kinect) [36].

There is an increasing interest in the detection of eyes in the vicinity and attention to public displays [2, 35, 37]. Determining where on the screen the user is looking requires estimating a gaze vector. A time-consuming and tedious calibration process preceding the actual interaction is required to obtain highly accurate gaze estimates [27]. Therefore, recent work either proposed to blend calibration into the application (e.g., as users read text [21]), or to use gaze interaction techniques that do not require accurate gaze estimates and are therefore calibration-free [42, 47].

In summary, while highly accurate gaze estimates would enable perfect parallax correction, current state-of-the-art techniques either require calibration or estimate an inaccurate gaze vector. Instead, our approach does not require an accurate gaze estimate but uses the position of the eyes in 3D space in front of the display to generate a vector that intersects the user’s touch point and the display. The intersection is deemed to be the point the user is trying to touch.

3 EYEPACT—CONCEPT AND IMPLEMENTATION
To study the parallax effect, it was necessary to build an apparatus where the distance between the display and the touch screen is adjustable, hence recreating and controlling the parallax effect (see Figure 3). To do that, we laminated a touch foil\(^1\) on a 110 cm × 60 cm × 8 mm Plexi glass. Figure 3 shows how we fixed the laminated touch foil to the display while keeping the distance between them adjustable. In particular, we fixed four metal rods into the stand of a 42 inch display (1366 × 768 pixels). We then used four 3D-printed holders to hold the display to the rods. The gap distance can be adjusted by sliding the rods and holders.

\(^1\)http://www.visualplanet.biz/touchfoil/

Our approach does not utilize accurate gaze point estimation and hence does not require calibration. Instead, a depth camera locates the user’s eyes in the 3D space in front of the display. To do this, we mounted a Kinect 360 on a tripod positioned behind the display. We continuously track the user’s pupils by employing the approach by Smolyanskiy et al., which augments the Active Appearance Model (AAM) algorithm by using depth-based face segmentation detected by an RGBD camera (kinect) [36].

Whenever a touch is detected on the touchscreen (Active2D), the point is converted from a 2D point on the touchscreen surface to a 3D point in the 3D space in front of the display (Active3D). Afterwards, a 3D ray is extended from the middle of both eyes and intersecting point Active3D to eventually intersect the display at point Perceived3D. Perceived3D is then converted from a 3D point to a 2D point (Perceived2D) on the display’s surface. The system blocks the original touch event that is orthogonal to Active2D (green dot in Figure 1A), and triggers a touch event at Perceived2D instead (red dot in Figure 1A). Visual feedback appears where Perceived2D is triggered.

When correcting parallax for multiple users (Figure 1C), the system has to determine which user performed the action to estimate the perceived point from his/her perspective. A straightforward approach might seem to correct for the user closest to the target. This approach would fail if, for example, a user extends his/her arm to touch a point that is closer to another user.

Instead, EyePACT determines which pair of eyes to correct for depending on the closest arm to the display at the time of the touch event. This is done by utilizing skeletal tracking [48], which provides the position of users’ joints in the 3D space in front of the display. The positions of the hand, wrist and elbow joints are then used to determine which user is closest to the touchscreen at the time of the touch action. Once the user is determined, EyePACT corrects for parallax from that user’s perspective.

4 STUDY1: ACCURACY EVALUATION
To evaluate EyePACT’s parallax correction accuracy we asked participants to touch the center of crosshairs with different levels of parallax, target states, and user positions (Figures 4A and 4B).
Fig. 4. In the accuracy evaluation study, participants were asked to touch the center of multiple crosshairs. (A) In the stationary target condition, one out of 15 equally distributed crosshairs was highlighted at a time. Participants had to aim at touching the center of the highlighted one (yellow dot) twice before another untouched crosshair was highlighted. The blue dot is the uncorrected touch point (Active2D), while the red dot is the corrected touch point inferred by our system (Perceived2D). (B) In the moving target condition, participants had to touch a moving crosshair, that bounced against edges, twice before it reappeared at a different starting position. We found that EyePACT results in significantly shorter distances to the target (Line AC) compared to the uncorrected distances (Line AB).

4.1 Design

We experimentally manipulated several independent variables that could impact the performance of EyePACT:

1. The user’s state: the participant could be stationary (standing 50 cm away from the display center) or move freely.
2. The target’s state: the target could be (T1) stationary or (T2) moving (Figures 4A and 4B respectively).
3. The gap distance: the distance between the touch screen and the display could be (G1) short, (G2) medium, or (G3) long (4 cm, 5.4 cm and 6.8 cm respectively).

The experiment was designed as a repeated measures study, where all participants performed all conditions. To make sure participants are not influenced by the predefined stationary user position, all participants started with the moving user condition. To minimize learning effects, we alternated the order at which participants performed the conditions of the target state. For example, the first participant started with the stationary target condition, while the second user started with the moving one. The same was done for the third variable.

In the stationary target condition, a random crosshair out of 15 equally distributed ones was highlighted at a time. Participants had to touch the center of the highlighted one (yellow dot in Figure 4A) twice before another untouched crosshair was highlighted. In the moving target condition, participants had to touch a moving crosshair twice before it reappeared at a different starting position. Each participant touched 15 moving crosshairs twice. All crosshairs moved at a constant speed, but in different directions with different starting positions.

We covered three gap distances to experiment with different levels of parallax. The manufacturer’s recommendation is to use a distance of 4.0 cm between the display and the touch screen to overcome the electrical noise caused by the display, which could interfere with the capacitive touch sensors. We used this as our minimum distance in addition to two larger distances: 5.4 cm and 6.8 cm. These values were determined by adding the recommended air-gap distance and the typical thicknesses of current vandalism-resistant glass. For example, at the time of submission of this paper, companies manufacture security glass that range in thickness from 6.5 mm to 25 mm [14], and from 11.5 mm to 39 mm [30].
EyePACT: Eye-Based Parallax Correction on Touch-Enabled Interactive Displays

Fig. 5. Figure A shows that: (1) The uncorrected distance (blue) increases depending on the gap distance. This indicates that parallax was successfully created in our setup. (2) The corrected distance to the target (red) is always shorter than the uncorrected one. This means that EyePACT significantly improves accuracy for all gaps. (3) The portion of the uncorrected distance that is decreased (i.e., difference between the red and blue bars) becomes larger as the gap distance increases. This means that EyePACT’s correction adapts to the size of the gaps. Figure B shows that although EyePACT reduces the distance to both moving and stationary targets significantly, the improvement is significantly higher with stationary targets than with moving ones.

4.2 Participants
We recruited 18 participants (5 females) with ages between 21 and 36 ($M = 25.2, SD = 4.4$) through mailing lists and social network groups. The height of the participants ranged from 172 cm to 189 cm ($M = 178.3, SD = 5.1$). They were compensated with online shop vouchers or participation points.

4.3 Procedure
The experimenter explained the study. Each participant performed 3 blocks with each block covering one of the three gap distances and 4 conditions (2 user states $\times$ 2 target states), amounting to 12 conditions $\times$ 15 targets $\times$ 2 touches = 360 touches. The study was concluded with a questionnaire and a semi-structured interview.

4.4 Results
At every touch event, we logged the uncorrected touch point, the corrected touch point, and the center of the target, which correspond to the blue dot, red dot, and yellow region/dot in Figures 1A and 4A respectively. We logged 7406 touch points – slightly more than what we expected, because participants sometimes performed more than two touches when uncertain if they hit the target. This mostly happened for moving targets. To address this we measured the average Euclidean distance between the points and the target for each condition. We compared the Euclidean distance between the uncorrected touch point and the target (Line AB in Figure 4A) and between touch point corrected by EyePACT and the target (Line AC in Figure 4A). Before analyzing the data, we excluded 122 out of 7406 measurements as outliers ($> \mu + 2.5 \times StandardDeviation$).

For analysis we used repeated measures ANOVA to test for significance in case of parametric data. Post-hoc pair-wise comparisons were done using t-tests. For non-parametric data (e.g., not normalized), we used Friedman’s test for significance, and Wilcoxon signed-rank tests with for pair-wise comparisons. In all cases, the p-value was corrected using Bonferroni correction to counteract the multiple comparisons problem.

Effect of Gap Distance on Accuracy. A significant main effect was found for the gap distance on the distance to the target $F_{2,34} = 22.28, p < 0.001$. Post-hoc analysis revealed significant differences between all pairs ($p < 0.05$). This means that our setup successfully recreated a parallax effect that significantly affects accuracy.
Effect of EyePACT Correction on Accuracy. A significant main effect was found for EyePACT correction on distance to target $F_{1,17} = 87.06, p < 0.001$. Post-hoc analysis showed a significant difference ($p < 0.001$) between corrected distance ($M = 29.84, SD = 13.15$) and uncorrected distance ($M = 46.07, SD = 15.08$). This means that EyePACT results in a significantly shorter distance to the target.

Effect of EyePACT Correction on Accuracy With Respect to the Gap Distances. To study the effect of the gap distance further, we ran paired-samples t-tests and found that the corrected distance to the target is significantly shorter compared to the uncorrected one in all gap distances: $G1 t(17) = -6.059, p < 0.001$, $G2 t(17) = -8.154, p < 0.001$ and $G3 t(17) = -8.89, p < 0.001$.

We calculated the difference between the corrected and uncorrected distances (Diff) which is represented by the green bar in Figure 5A. We also found that Diff is significantly different depending on the gap distance $\chi^2(2) = 20.333, p < 0.001$. Post-hoc analysis with Wilcoxon signed-rank tests with Bonferroni correction showed significant differences between all pairs ($p \leq 0.01$). The largest correction was apparent in DiffG3 ($M = 20.94, SD = 9.99$), followed by DiffG2 ($M = 15.81, SD = 8.22$) then DiffG1 ($M = 11.97, SD = 8.38$). This implies that EyePACT adapts to different levels of parallax, resulting in larger corrections when the parallax effect is higher.

The results are visualized in Figure 5A, and summarized in its caption.

Effect of Target State on Accuracy. A significant main effect was found for target state on distance to the target $F_{1,17} = 51.2, p < 0.001$. Post-hoc analysis revealed that the distance in case of stationary targets ($M = 32.82, SD = 16.73$) is significantly shorter ($p < 0.001$) than in the case of moving targets ($M = 43.09, SD = 14.14$). This motivated us to further investigate how well EyePACT performs with respect to the target state.

Running paired-samples t-tests, we found significant differences between the corrected distance ($M = 21.35, SD = 8.92$) and the uncorrected distance ($M = 44.29, SD = 14.74$) in the case of a stationary target $t(17) = -13.084, p < 0.001$, and also between the corrected distance ($M = 38.32, SD = 11.07$) and the uncorrected distance ($M = 47.86, SD = 15.1$) in the case of a moving target $t(17) = -4.416, p < 0.001$. This means that EyePACT significantly reduces the distance to stationary and moving targets. However the improvement is significantly higher in case of a stationary target. Results are visualized in Figure 5B and summarized in its caption.

Effect of User’s Height on Accuracy. Compared to short users, taller ones are expected to experience a stronger parallax effect due to the steeper angle to the target. Indeed, we found a strong positive correlation between the participant’s height and the distance between the uncorrected point and the center of the target $|r| = 0.535$ ($p < 0.05$) – data was shown to be linear according to a Shapiro-Wilk’s test ($p > 0.05$). On the other hand, we did not find any indicators that the height of the user, or distance between the eyes and the screen, influences or correlates with the accuracy of EyePACT. A Pearson’s product-moment correlation was run and showed a small correlation ($|r| = 0.105$), that was not statistically significant ($p > 0.05$), between the participant’s height and the distance between the corrected touchpoint and target.

On the other hand, a repeated measures ANOVA showed significant main effect of the position of the target’s row on the accuracy of EyePACT $F_{2,14} = 10.7, p < 0.001$. Pairwise comparisons using Bonferroni correction showed that only the accuracy at the topmost row ($M = 18.8, SD = 4.5$) and at the lowermost row ($M = 24.9, SD = 9.9$) are significantly different ($p < 0.005$). This implies that EyePACT’s accuracy drops as the target’s position is towards the bottom of the display. This is because the parallax grows further as the target is farther away from the user (see Figure 6).

In summary, we neither found an influence of height on EyePACT’s accuracy, nor a relationship between height and our method’s accuracy. However, not finding significant differences does not mean that there is none. In fact, the significant effect of the position of the target on accuracy means that the distance between the user’s eyes and the target has an influence. Hence whether the user’s height influences EyePACT’s accuracy remains an
open question for future research. This can be investigated by a large scale study with balanced heights, and performing statistical power analysis to estimate the probability of accepting the null hypothesis (i.e., that the user’s height has no influence on accuracy) [12].

4.5 Observations and Feedback
When asked about the accuracy of the system on a 5-point scale (1=very imprecise; 5=very precise), responses ranged from 2 to 5 and were more inclined towards high precision ($M = 3.28$, $SD = 0.83$). Almost all participants indicated to find it easier to select stationary targets, which is reflected in the quantitative analysis. P9 mentioned that touching the moving crosshair became easier over time. P3 and P4 noted that they found the system to be highly accurate. P3 was surprised that the system worked despite the touch screen being far from the display. P11 and P18 remarked that they perceived no difference in accuracy between the smallest and medium gap distances. This is in-line with our findings which show that EyePACT corrections are equally good for different gap distances (see Figure 5A).

Additionally we asked participants if they attempted to compensate for the parallax effect themselves. While some needed an explanation of what parallax is, some others indicated that although they did at the beginning, they quickly realized that the system responds more intuitively if they touch where they see the target. On a 5-point scale (1=very rarely; 5=very often), responses varied from 1 to 4 and indicate that participants rarely corrected for parallax ($M = 2.56$, $SD = 1.04$).

Although participants were explicitly asked to move freely in the moving user conditions, participants often remained in the center. Some highlighted that they were too lazy to move and targets were easily reachable from the center. Others did not think that they would perform better by moving. While P12 said he would avoid walking while looking at the screen lest he trips or steps on someone’s feet. This explains the lack of significant effects of user state on the distance to the target.

Fig. 6. Offset regression models for stationary (left) and moving targets (right). Arrows point from touch to target locations, revealing 2D offset patterns across the screen.

5 TOUCH OFFSET REGRESSION ANALYSIS
Beyond overall accuracy, it is also insightful to evaluate the offset patterns underlying the observed touch interactions. Related work analyzed such patterns on mobile devices with regression models [7, 8, 45, 46]. We also trained linear offset models (see [7, 8]) on the data of the accuracy study to analyze offset patterns.
Figure 6 visualizes models for stationary and moving targets: Users had large vertical offsets, which grow with the target location towards the bottom of the screen. Smaller horizontal offsets grow towards the left/right screen edges. This is explained by geometry – parallax grows the further users have to look down and to the side. Thus, these patterns visually explain why knowing the eye position relative to the display is highly valuable information for correcting parallax.

Since offset models map touches to targets, they can also be applied for parallax correction. We used leave-one-user-out cross-validation to train and test the models. ANOVA showed that the factor model (with levels: no model, hardware, software, both) has a significant effect on touch offsets (corrected by Greenhouse-Geisser: $F_{1, 82,32,64} = 175.54, p < 0.001$; moving: $F_{1, 71,29,04} = 51.29, p < 0.001$). Post-hoc tests revealed that both hardware and software approach significantly reduce offsets, and that combining their predictions significantly reduces offsets further. This results in a total improvement in offset RMSE of 56.6% (over the baseline) for fixed crosshairs and 29.7% for moving crosshairs.

Note, however, that the software approach requires (1) training data for the models, (2) might not work for users with heights different from those the data was trained with, and (3) cannot distinguish multiple users.

6 STUDY2: SUPPORTING MULTIPLE USERS
In a following step, we investigated how well EyePACT corrects parallax for multiple users and in turn allowing them to interact with the same on-screen object. We invited participants in pairs to participate in a collaborative multiplayer game where balloons would appear at the bottom and float to the top (Figure 7). The task was to collaboratively pop the balloons by (1) one player touching the balloon to stop it and (2) the second player touching it to pop it.

6.1 Design
This study was designed as a repeated measures experiment where all participants went through all conditions. We studied the effect of two independent variables:

1. Target size: we experimented with small, medium and large balloons with diameters 30, 60 and 90 pixels ($2.15^\circ$, $4.29^\circ$, $6.44^\circ$ of visual angle respectively).
2. Gap distance: similar to the accuracy study, we experimented with 4 cm, 5.4 cm and 6.8 cm gap distances.

The order of conditions was counterbalanced using a Latin-square. Participants swapped roles, that is, if participant A stopped a balloon, and participant B popped it, the following time a balloon appeared participant B stopped it, while participant A popped it. Each pair popped 8 balloons per condition.

6.2 Participants
We invited 20 participants (10 pairs, 19-55 years, ($M = 25.7, SD = 7.77$)), 160 to 191 cm tall ($M = 177.3, SD = 8.7$). Participants were recruited using mailing lists and social networks. They were compensated with online shop vouchers or participation points.

6.3 Procedure
Participants filled in consent forms and were explained the study. Pairs were then asked to take positions (Figure 7), which they maintained till the end of the study. Each pair performed 3 blocks with each block covering one of the three gap distances and 3 conditions. This means each pair performed 3 gap distances $\times$ 3 target sizes $\times$ 8 balloons = 72 balloon bursts.

6.4 Results
At each stop and pop action, we logged the uncorrected and corrected touch points. This allowed us to measure:
We investigated how the conditions influence these two distances and the difference between them (see Figure 7B).

- The \textit{inter-fingers distance}: the distance between the two uncorrected points.
- The \textit{perceived distance}: the distance between the two corrected points.

Effect of Parallax Correction. A repeated measures ANOVA showed significant main effect of EyePACT correction on the distance between the two points \(F_{1,9} = 56.61, p < 0.001\). Post-hoc analysis using Bonferroni correction showed that the \textit{perceived distance} \((M = 23.26, SD = 13.07)\) is significantly shorter than the \textit{inter-fingers distance} \((M = 58.12, SD = 23.83)\). This means that EyePACT brings the two touch points significantly closer to each other when multiple participants touch the same object.

Effect of Gap Distance. After excluding 7 outliers \((\mu + 3 \times \text{Standard Deviation})\), a repeated measures ANOVA showed significant main effect of the gap size on the \textit{inter-fingers distance} \(F_{2,18} = 11.2944, p < 0.005\). Post-hoc analysis using Bonferroni correction showed that there is a significant difference \((p < 0.005)\) between the shortest \((M = 45.2, SD = 14.86)\) and the largest gap distance \((M = 72.3, SD = 23.62)\). The other pairs were not significantly different. Yet, Figure 8 suggests that the larger the gap distance, the larger the \textit{inter-fingers} distance. From this we conclude that the larger the parallax, the farther users’ fingers will be when touching the screen while looking at the same object.

On the other hand, no significant main effects were found for the gap distance on the \textit{perceived distance} \((p > 0.05)\). This means that there is no evidence that the gap distance influences the distance between the corrected points. Figure 8 shows that the distance between the two touch points corrected by EyePACT is almost the same across the different gap distances. This suggests that EyePACT corrects equally well for multiple users at different levels of parallax.

Effect of Target Size on Perceived and \textit{Inter-fingers} distances. A repeated measures ANOVA showed significant main effect of target size on the perceived distance \(F_{2,18} = 106.32, p < 0.001\). Post-hoc analysis using Bonferroni correction revealed significant differences between all pairs \((p < 0.001)\), indicating that the distance was shortest for small targets \((M = 12.56, SD = 4.67)\), followed by medium targets \((M = 22.86, SD = 7.53)\), and longest for large targets \((M = 34.38, SD = 9.95)\). This is expected as both corrected touch points need to be closer to each other to fit in smaller targets. We found no significant effect of target size on distance between the two uncorrected points (inter-fingers distance). This means there is no evidence that the target size influences the distance between the fingers.
Fig. 8. The figure shows that as the gap between the touch screen and the display increases, so does the distance between the users’ fingers when they both touch the same object (inter-fingers distance). The figure also shows that despite the increasing gap distance, which in turn results in a stronger parallax effect, the distance between the two corrected touch points (perceived distance) does not vary a lot.

<table>
<thead>
<tr>
<th>Gap Distance (cm)</th>
<th>Distance between the two corrected points (in pixels)</th>
<th>Distance between the two uncorrected points (in pixels)</th>
</tr>
</thead>
<tbody>
<tr>
<td>G1 (4.0 cm)</td>
<td>23.19</td>
<td>60.87</td>
</tr>
<tr>
<td>G2 (5.4 cm)</td>
<td>22.48</td>
<td>64.12</td>
</tr>
<tr>
<td>G3 (6.8 cm)</td>
<td>24.11</td>
<td>73.39</td>
</tr>
</tbody>
</table>

Fig. 9. The figure shows the effect of the target’s size on the distance between the two corrected points (perceived distance), and the distance between the two uncorrected points (inter-fingers distance). As expected, the smaller the target, the smaller is the distance between the two corrected points within the target. The figure also suggests that the Inter-fingers distances are random and are not influenced by the target’s size. The value between brackets denotes the distance in centimeters.

<table>
<thead>
<tr>
<th>Target Size (px)</th>
<th>Distance between the two corrected points (in pixels)</th>
<th>Distance between the two uncorrected points (in pixels)</th>
</tr>
</thead>
<tbody>
<tr>
<td>T1 (30 px)</td>
<td>12.55</td>
<td>66.15</td>
</tr>
<tr>
<td>T2 (60 px)</td>
<td>22.86</td>
<td>61.50</td>
</tr>
<tr>
<td>T3 (90 px)</td>
<td>34.38</td>
<td>70.73</td>
</tr>
</tbody>
</table>

Effect of Target Size on the Distance to Target Center. Although the main aim of the second study was to evaluate EyePACT in case of multiple users, it was also possible to study the influence of the target size on EyePACT’s accuracy for each individual user, since the task involved touching balloons of different sizes. We did not find any

significant effect of target size on the distance between the target center and the uncorrected touchpoint \( (p > 0.05) \). On the other hand, a repeated measures ANOVA with Greenhouse-Geisser correction revealed a significant main effect of target size on distance between the target center and the corrected touchpoint \( F_{1.2,10.9} = 39.97, p < 0.001 \). Pairwise comparisons with Bonferroni correction showed significant differences between all pairs. The distance between the corrected touchpoint and the small target \( (M = 10.16, SD = 4.63) \) was the shortest, followed by that between the corrected touchpoint and the medium target \( (M = 15.45, SD = 8.34) \), and finally that between the corrected touchpoint and the large target \( (M = 24.13, SD = 14.07) \).

This means that although the distance between the uncorrected touchpoint and the target center remained almost unchanged (see Figure 10), the distance between the corrected touchpoint and the target center adapted based on the target size. This provides evidence for an additional usability advantage made available by EyePACT. That is, although smaller targets typically require more precise pointing by users, EyePACT users do not need to be more precise in order to touch smaller targets. Figure 10 illustrates this advantage: our participants did not need to be more precise when touching smaller targets compared to larger ones (blue bars); EyePACT made their touchpoints more precise and corrected them to trigger on the targets (red bars).

7 DISCUSSION

Our findings from the first study show that EyePACT (1) brings the touch point significantly closer to the target for all gap distances, (2) results in significantly higher accuracy for stationary targets, (3) adapts to parallax by resulting in significantly larger corrections when the gap distance is increased. The second study shows that EyePACT (1) maintains a significantly large distance between two users’ fingers when interacting with the same object, and (2) is flexible with regard to targets of different sizes.

Technical Realization. While we are starting to see a rising adoption of interaction via mid-air gestures and gaze, touch remains the most prominent interaction technique for public displays [11]. Many of these displays have readily integrated cameras. For example, ATM machines are often equipped with cameras for security reasons. This makes the adoption of EyePACT straightforward; our approach only requires detection of the user’s eyes. We use a depth camera for multi-user scenarios to identify which arm is closest to the display at the time of interaction. Although achieving the same with high accuracy using an RGB camera would result in higher
processing demand, which could in turn result in a delay in the system’s response, advances in hardware and processing power make the use of an RGB camera for the same purpose also feasible in the near future.

Eliminating the Error Completely. Although EyePACT significantly improves touch accuracy based on the user’s eye position, it does not completely eliminate the error caused by parallax. This is due to several reasons. One reason is that even with perfect eye tracking, knowing the exact pixel the user is focused on is almost impossible; users can move attention within the fovea, which is around 2 degrees of visual angle, without moving their eyes [27]. Another reason is that the user’s touch tends to be biased towards the dominant eye [39]. An interesting direction for future work is to estimate the user’s dominant eye based on these biases and adapt the parallax correction accordingly.

Manual Parallax Correction and Showing Visual Feedback. In general, users do not expect parallax correction. This was observed in our studies where several users upon the first touch overcompensated since they expected a parallax effect. In the two studies visual feedback was provided. Hence, participants quickly understood where the touch point was registered and stopped compensating. From this we learn that upon deployment in a public space, visual feedback is crucial in order not to confuse users. Furthermore, future versions of EyePACT can detect such over compensations and use them as a form of calibration to improve the correction.

Other Application Opportunities. In this work we focused on addressing the problem of the displacement between the perceived and detected touch points on a public display. However, this is not the only issue resulting from the large distance between the touch surface and the display. For example, binocular parallax occurs when the distance between the touchscreen and UI elements is very large (e.g., 30 cm [25]). While this is not a realistic setting for public displays that are in the scope of this work, and binocular parallax was not reported by our participants, EyePACT can be optimized to address binocular parallax as well. In particular, previous work showed that when pointing at 3D stereoscopic objects, users touch between their dominant and non-dominant eye position with a bias towards the dominant eye position [39]. Valkov et al. developed guidelines to address this issue for interacting with 3D stereoscopic objects, extrapolating their 3D solution to the 2D case could be a direction for future work to optimize EyePACT for contexts where this issue occurs (e.g., transparent displays [25, 26]).

Reaching Farther Objects. While the parallax effect comes with its shortcomings, it could also offer opportunities that can improve the user experience. For example, by intentionally introducing a large gap between the touch surface and the display, the angle between the user’s gaze vector and the touch surface plane becomes wider. This can, in turn, reduce the user’s effort to reach out to distant objects since the user does not have to be in a position that is orthogonal to the object (see Figures 1A, 1C, and Figure 7). A typical problem with touch screens is the reachability of distant content [16, 17]. This problem is amplified as large displays become more common [1, 22, 33], which results in hardly reachable on-screen objects. The use of EyePACT with a large gap can also allow users to interact with a larger portion of the display while being stationary. Nevertheless, interacting with far-away objects is likely to cause binocular parallax. Furthermore, a large gap can result in making it difficult for the user to focus on his/her fingers and the displayed object at the same time. An interesting direction for future work is to investigate starting at which gap distances focusing becomes uncomfortable for the user.

7.1 Limitations and Future Work
In our experiments, the speed and direction of the moving target was randomly decided at the beginning of each trial. Future work could investigate the impact of speed and direction of moving targets on the accuracy of EyePACT. Furthermore, as indicated in section 4.4, we plan to conduct a large scale study in which we balance the height of participants to better understand the impact of the user’s height on the parallax error and correct it accordingly. Finally, an interesting direction for future work is to study how knowledge about the user’s dominant
eye and dominant hand can help in improving the accuracy of EyePACT. In the future we plan to further extend EyePACT to address different types of parallax (e.g., Binocular parallax). Furthermore, EyePACT can be exploited to support interaction by, for example, allowing users to reach farther objects, addressing binocular parallax, and allowing novel collaborative multi-user interactions by eliminating finger interferences.

8 CONCLUSION

In this work we presented EyePACT, a low-computation yet effective approach for overcoming parallax on public displays. First, we discussed why parallax is an issue on public displays nowadays and the need for research in this space. Second, we described a prototype with which we created and controlled the parallax effect. Third, we described the implementation of EyePACT. Forth, we demonstrated EyePACT’s accuracy in a first user study to show that it (1) significantly improves accuracy in the presence of parallax, (2) adapts to different levels of parallax, (3) improves accuracy for both stationary and moving targets, but the improvement is more significant for stationary ones, and (4) although future work is needed to determine if it is not affected by the user’s height, we did not find any significant effects in our study. Fifth, we described a software-based regression model that was trained using the data from the first study, and described why it is not sufficient in the case of users with different heights and in multiuser scenarios. Finally, in a second user study we showed that EyePACT (1) significantly improves accuracy for multiple users and also adapts to different levels of parallax, (2) enables multiple users to interact with the same objects while keeping their fingers far apart, (3) does not require users to be more precise when touching smaller targets. In addition we utilized this method to enable novel ways of interaction that are otherwise infeasible.

ACKNOWLEDGEMENTS

This work was funded, in part, by the Cluster of Excellence on Multimodal Computing and Interaction (MMCI) at Saarland University, Germany, and by the Bavarian State Ministry of Education, Science and the Arts in the framework of the Centre Digitisation.Bavaria (ZD.B).

REFERENCES

7 EyePACT: Eye-Based Parallax Correction on Touch-Enabled Interactive Displays.

M. Khamis et al.


EyePACT: Eye-Based Parallax Correction on Touch-Enabled Interactive Displays


7 EyePACT: Eye-Based Parallax Correction on Touch-Enabled Interactive Displays.

Received February 2017; revised August 2017; accepted October 2017
GTmoPass
GTmoPass: Two-factor Authentication on Public Displays Using Gaze-Touch passwords and Personal Mobile Devices

Mohamed Khamis¹, Regina Hasholzner¹, Andreas Bulling², Florian Alt¹
¹Ubiquitous Interactive Systems Group, LMU Munich, Germany
²Max Planck Institute for Informatics, Saarland Informatics Campus, Germany
{mohamed.khamis, florian.alt}@ifi.lmu.de, bulling@mpi-inf.mpg.de

ABSTRACT
As public displays continue to deliver increasingly private and personalized content, there is a need to ensure that only the legitimate users can access private information in sensitive contexts. While public displays can adopt similar authentication concepts like those used on public terminals (e.g., ATMs), authentication in public is subject to a number of risks. Namely, adversaries can uncover a user’s password through (1) shoulder surfing, (2) thermal attacks, or (3) smudge attacks. To address this problem we propose GTmoPass, an authentication architecture that enables multi-factor user authentication on public displays. The first factor is a knowledge-factor: we employ a shoulder-surfing resilient multimodal scheme that combines gaze and touch input for password entry. The second factor is a possession-factor: users utilize their personal mobile devices, on which they enter the password. Credentials are securely transmitted to a server via Bluetooth beacons. We describe the implementation of GTmoPass and report on an evaluation of its usability and security, which shows that although authentication using GTmoPass is slightly slower than traditional methods, it protects against the three aforementioned threats.

Author Keywords
Multi-factor Authentication, Pervasive Displays, Eye Gestures

ACM Classification Keywords
K.6.5 Computing Milieux: Security and Protection: Authentication

INTRODUCTION
Public displays deliver various kinds of tangible benefits and are now deployed in train stations, airports, and streets. Meanwhile, there is an increasing demand for displays to offer personalized, context-specific content [7, 21, 23].

There are many cases in which users need to securely authenticate at public displays. For example, while a group of tourists examine places to visit on a large public display, the system could allow users to buy tickets for museums, buses, etc. One or more users could then authenticate in parallel by entering their passwords on their mobile devices. Using the mobile device’s MAC address and the provided password, the system validates the user’s credentials and charges the correct account for the ticket fee. While there are scenarios where it might be acceptable to continue the purchase and interaction on the mobile device, in many cases it is favorable to keep the user at the display to resume the primary task. In the aforementioned example, tourists could then be shown further suggestions for activities at the given place.

When exchanging sensitive data with a public display (e.g., login credentials), users are prone to several types of threats. Namely, adversaries can uncover a user’s password in public space through: (1) Shoulder surfing attacks: observing the user while authenticating [14], (2) Thermal attacks: exploiting the heat traces resulting from the user’s interaction with the interface [1, 24], and (3) Smudge attacks: exploiting the oily residues left after authenticating on touchscreens [3]. While the latter two risks were demonstrated to be feasible, shoulder surfing was shown to occur in daily contexts [14].

In this work we introduce GTmoPass, an authentication architecture that enables multi-factor user authentication on public displays...
displays. GTmoPass uses a shoulder-surfing resilient multimodal scheme that combines Gaze and Touch for password entry as a knowledge factor. Additionally, it uses personal mobile devices as a possession factor (see Figure 1). After entering the password on the mobile device, the password is then securely transferred to an authentication server whose URL is communicated to the mobile device via Bluetooth beacons. The use of BLE beacons alleviates the need to manually enter URLs, or scan QR-codes. This means that when interacting with public display that employs GTmoPass for the first time, users do not have to do anything other than launching the app and entering the password.

The results of our evaluation show that users authenticate relatively fast (2.8 – 4.9 seconds), and that the authentication process is resilient to the aforementioned threats. Even if the password is leaked, the architecture of GTmoPass requires the adversary to additionally acquire the user’s mobile device.

BACKGROUND AND RELATED WORK

Authentication Factors
Researchers and practitioners have developed different ways for users to authenticate in order to be granted access to private or sensitive information. Three of the most popular authentication forms are (1) knowledge-based authentication, (2) possession-based authentication, and (3) inherence-based authentication (also known as biometric authentication).

Knowledge Factor
Knowledge-based authentication schemes rely on “something the user knows”. It is perhaps the most commonly used factor [2]. Examples are passwords, PINs, and graphical passwords. Researchers also developed ways to authenticate using eye movements [8, 12, 13, 16], mid-air gestures [25], and by recalling photographs [26]. Knowledge-based schemes allow changing passwords, and can be integrated into any system that accepts any kind of user input. On the other hand, knowledge-based passwords can be guessed by illegitimate users. Attackers can find the password by observing users during authentication [14]. Smudge attacks are possible when passwords are entered through touchscreens [3]. Graphical passwords such as Android lock patterns are particularly vulnerable to smudge attacks [29, 40]. Furthermore, many knowledge-based schemes are also vulnerable to thermal attacks, where heat traces resulting from the user’s interactions with the interface are exploited to find the password [1].

Possession Factor
The possession factor relies on “something the user possesses”. Physical keys and scannable personal IDs are examples of possession-based authentication. Researchers experimented with identifying users on public displays through the MAC address of their smartphone [28]. Davies et al. [7] exploit the user’s mobile device to identify the list of applications the user wants to interact with on a display. Others approaches include using Bluetooth devices [6] and wearable shutter glasses [32]. While this type of schemes does not require users to remember passwords, it requires keeping possession of the access token. A drawback of using this factor alone, is that if an attacker gets hold of the access token, the attacker can impersonate the user and gain unauthorized access.

Inherence Factor
The inherence factor relies on biometric data, such as fingerprints, user behavior (e.g., behavior on a touchscreen [10] or eye-gaze behavior [33]) and face detection [15]. While biometric authentication can be easy and fast to use, it is accompanied with a number of problems. Biometric passwords cannot be changed; once a user’s biometric data (e.g., fingerprint or iris scan) is leaked, there is no way the user can invalidate the leaked data. Face recognition can be bypassed using pictures of the legitimate user [22], and users leave fingerprints everywhere as they interact with surrounding objects. Furthermore, users are oftentimes concerned about disclosing their biometric data to third-party companies [27], especially after it was found that biometric data can be stolen remotely [34, 42].

Multifactor Authentication
Multifactor authentication refers to the use of two or more of the aforementioned factors for improved security. This approach is adopted by ATM machines; users have to know a PIN, and have to possess an ATM card. The approach has also been adopted by multiple Internet services, such as Google, Facebook, Microsoft, and Dropbox; users have to know their username and password, and have to possess a previously identified mobile device on which they receive an additional one-time password, or confirm their log-in attempt. Researchers developed systems where users authenticate at ATMs by entering PINs on their mobile phones [4, 30, 31]. De Luca and Fraudendienst introduced PocketPIN, where users can enter credit card data on their phones before being securely transmitted to public terminals [9].

The advantage of GTmoPass is that it employs multimodal authentication as a knowledge factor, and personal mobile devices as a possession factor. Multimodal authentication was shown to be highly resilient to shoulder surfing [19]. Furthermore, thermal and smudge attacks normally require the attacker to inspect the interface after the user had left [1, 3]. Our architecture complicates these attacks by relying on the user’s mobile device for input. This means that an attacker can only perform these attacks by stealing the mobile device fast enough before the heat or smudge traces can no longer be traced. And even by doing so, the attacker would not be able to identify the gaze-input.

Protecting Privacy on Public Displays
In addition to the aforementioned works by Davies et al. [6, 7] and Schaub et al. [28], other works exploited proxemics for showing content on public displays. For example, Vogel and Balakrishnan show private content on public displays only when the user is very close to the display [36]. Brudy et al. proposed some concepts to hide private data on public displays by partially hiding the private data from the observer’s view estimated by a Kinect [5].

GTMOPASS
GTmoPass is an authentication architecture that enables secure multifactor user authentication on public displays. In the
following we describe the concept and implementation of GTmoPass, and which threat models it is optimized against.

**Concept**

GTmoPass relies on two factors: (1) a knowledge factor: we use a multimodal authentication scheme that combines gaze and touch input, and (2) a possession factor: users enter the multimodal passwords on their personal mobile devices.

**The Knowledge Factor**

For the knowledge factor in GTmoPass, we employ a modified version of GazeTouchPass [19], a state-of-the-art authentication scheme that is highly resilient to shoulder surfing attacks. GazeTouchPass is a multimodal scheme that employs combinations of touch-based PINs (0-9) and eye movements (gaze gestures to the left and to the right). This means that it uses a theoretical password space of $12^n$ (10 digits + 2 gaze gestures) where $n$ is the length of the password. A password could be: Gaze (left), Touch (2), Gaze (right), Touch (1).

The strength of GazeTouchPass lies in its use of two input modalities. This adds complexity to shoulder surfing attacks, as it requires the attacker to observe (1) the user’s input on the touchscreen, and (2) the eye movements of the user.

**The Possession Factor**

For the possession factor in GTmoPass, we rely on the user’s personal mobile device. The multimodal passwords are entered on the mobile device; touch input is detected through the touchscreen, and gaze input is detected through the front-facing camera. The mobile device then communicates securely with an authentication server, that validates the password and signals the display to show the private information.

**Implementation**

GTmoPass consists of two main components: (1) the authentication server, and (2) the user’s mobile device (client).

**Authentication Server**

Implemented in NodeJS, the authentication server is configured to receive HTTP requests. The server runs on a computer (e.g., in our setup we used a WiFi router. The IP address of the server is broadcasted using a BEEKS BLE beacon in Google’s Eddystone protocol. The IP is broadcasted in Eddystone-URL data packets at a 10 Hz rate (i.e., it broadcasts once every 100 ms), with a transmission power of 0 dBm ($\approx$ 50 meters).

**The Client**

GazeTouchPass [19] was implemented as an Android application. It uses the OpenCV library $^3$ and the Viola-Jones classifier [35] to detect the user’s face and eyes. Afterwards, in a manner similar to previous work [41, 43], the gaze direction is estimated depending on the distance between each eye and the center of the user’s face.

We further extended GazeTouchPass to communicate with the authentication server. As soon as the modified app launches, it communicates the user’s IP is broadcasted in Eddystone-URL data packets at a 10 Hz rate (i.e., it broadcasts once every 100 ms), with a transmission power of 0 dBm ($\approx$ 50 meters).

**EVALUATION**

We previously evaluated the usability and observation resistance of GazeTouchPass, to find that although it requires 1.6 more seconds than traditional PINs [39], it is significantly more secure against shoulder surfing compared to traditional PINs [19]. We also found that the structure of a multimodal password has an influence on its security. Namely, passwords that contain several switches from one modality to another, are more difficult to observe compared to those that have less switches. For example, a password such as “left-2-right-1” has three modality switches, and is hence harder to observe compared to “left-right-2-1”, which has only one modality switch. The reason is that attackers would have to switch attention between the user’s fingers and the user’s eyes more often in the case of passwords with more modality switches.

Another interesting insight from our previous evaluation of GazeTouchPass is that multimodal passwords that start or end with gaze input were perceived by participants to be more difficult to observe. While our previous study was designed to focus on the effect of the number of switches in input modalities on the observability of the password, in the currently presented work we focus on the influence of the position of the gaze input in the password on the security of the GazeTouch password.

**Usability Study**

The goals of this study are to: (1) collect feedback about the use of GazeTouchPass in the proposed setup, and (2) understand the influence of the gaze-input’s position on usability.
Figure 3. Participants were recorded during the usability study as they enter passwords using an HD video camera. The recorded videos were used in the subsequent security study to simulate shoulder surfing attacks. Figure A shows a user entering a touch input Touch(B), while figure B shows a user performing a gaze gesture Gaze(Left).

**Apparatus**
We used a 48 inch Samsung TV (1920×1080 px) as a display. We connected a laptop computer running a NodeJS server that accepts HTTP requests. The server validates the received passwords and updates the display’s content accordingly. We recorded participants as they enter passwords from the side (see Figure 3). An HD camera was positioned such that it is close enough to show the touchscreen, and also the user’s eyes. These videos were recorded to be used in the subsequent security study, to simulate shoulder surfing attacks.

**Design**
Since we wanted to investigate the influence of the position of the gaze input in the password, we experimented with four conditions: (1) passwords that start with gaze input (GazeStart), (2) passwords that end with gaze input (GazeEnd), (3) passwords that start and end with gaze input (GazeStartEnd), and (4) passwords with gaze input in the middle (GazeMiddle).

The study was designed as a repeated measures experiment. Each participant performed 16 authentications (4 passwords × 4 conditions) using randomly generated passwords. In case of conditions GazeStart and GazeEnd, participants entered two passwords with one gaze input at the start/end of the password, while the other two passwords had two gaze inputs at the start/end of the password. Table 1 shows sample passwords.

**Participants**
We invited 16 participants to our lab (6 females), recruited through mailing lists and social networks. Participants were awarded with online shop vouchers or participation points. All participants had normal or corrected-to-normal vision.

**Procedure**
The experimenter first described the study and asked the participants to sign a consent form. She then handed the participant a mobile device with the modified version of GazeTouchPass installed, and explained how it works. Each participant was allowed to perform a training run per condition to get acquainted with the system. The trial attempts were not included in the analysis. At each authentication attempt, the experimenter read out the password to be entered according to a previously generated list that was randomized. The participant would then enter the password, and observe the feedback on the display, which indicated whether or not the correct password was detected. Afterwards participants were interviewed to learn about their feedback, ideas and concerns about GTmoPass.

**Results**
To measure the impact of the position of gaze input on the usability of the passwords, we measured the entry time and the error count while authenticating.

**Entry time** was measured starting from the first input until the last input was recognized. Figure 4 illustrates the time taken to enter a password at every condition. A repeated measures ANOVA (with Greenhouse-Geisser Correction due to violation of sphericity) showed a significant effect of the gaze input’s position on the time it takes to enter a password (F1, 8, 27.5 = 9.1, p < 0.05). Post-hoc analysis with Bonferroni correction (α = 0.05 / 6 comparisons = 0.0083) showed significant differences in entry time between GazeStart (M = 2863 ms, SD = 1525 ms) and GazeMiddle (M = 4959 ms, SD = 3141 ms), between GazeEnd (M = 3892 ms, SD = 3045 ms) and GazeMiddle (M = 4959 ms, SD = 3141 ms), and between GazeStartEnd (M = 3757 ms, SD = 3852 ms) and GazeMiddle (M = 4959 ms, SD = 3141 ms). The other pairs were not significantly different (p > 0.05). This means that passwords with gaze in the middle are significantly slower than other cases.

**Error count** reflects the number of times the participant entered the password incorrectly. Errors could occur either due to entering the wrong gaze or touch input, or due to the system detecting an incorrect gaze input due to poor lighting conditions. Figure 5 shows the number of errors at each condition.

**Qualitative Feedback** collected at the end of the study through semi-structured interviews revealed positive feedback towards the system. Many participants reported that they liked the idea of detecting eye movements through the smartphone’s camera, and would imagine using it to authenticate on ATMs instead of using cards and PINs. Some participants suggested using the system to open security doors. One participant suggested using it for authentication on other digital devices.
Unlike existing systems, GTmoPass is resilient to thermal and smudge attacks by design. Heat traces and oily residues can only uncover the touch part of the password, but not the gaze input. Therefore in this study we focus on GTmoPass’s resilience to observation attacks.

Although previous work evaluated the impact of the number of modality switches on the security of passwords [19]. Our aim in this study was to understand the influence of gaze input’s position on the observation resistance of the password.

Using the video material produced in the first study, we conducted a second observability study that simulated a shoulder surfing attack against a user authenticating using GTmoPass. To do this, we invited 16 different participants and asked them to simulate shoulder surfers by watching the recorded videos, and trying to find the entered passwords (see Figure 6).

**Threat Model**

In our threat model, the user and the attacker are in public space. The attacker is familiar with GTmoPass and how it works. The attacker observes the user from an angle that allows seeing both the touch input on the mobile device, and the eye movements (see Figure 3). The distance between the attacker and the user is close enough to see the touchscreen, but far enough to reduce the effort of switching focus back and forth between the user’s eyes and the device’s touchscreen. After unveiling the password, the attacker tries to get hold of the device and authenticate at the display.

**Results**

To understand the impact of the position of gaze input on the observability of the passwords, we measured the binary success rate and the Levenshtein distance.

**Success rate** reflects how many passwords of each condition were successfully observed. The attackers were explained how the system works, had a chance to try the application themselves, and were allowed to take notes while watching the videos.

**Participants and Reward mechanism**

We invited 16 participants (9 females) through mailing lists and social networks. None of them had participated in the usability study. All participants were awarded either an online shop voucher or participation points. In addition, they took part in a lottery for an additional 10 EUR online shop voucher, where the chance of winning increases as the participant successfully attacks more passwords. This was done to encourage participants to put a lot of effort in finding the passwords.

**Design**

This study also followed a repeated measures within subjects design. Each participant watched 4 videos of successful authentications from each condition (4 videos × 4 conditions = 16 videos in total), each of which is a recording from a different usability study participant.

**Procedure and Apparatus**

After arriving at our lab and filling out the consent form, the experimenter explained the concept and the reward mechanism. The videos were displayed and controlled by the experimenter on a computer (1920 px × 1080 px). Participants were given a pen and a paper and were allowed to take notes while watching the videos (see Figure 6). Since each video was attacked once, it was watched once and hence the duration of the attack depends on the length of the video. They were also allowed to try the application themselves. After watching the video once, they provided up to three guesses, but were not told whether their guesses were correct or not to avoid influencing the perceived difficulty. We concluded with a questionnaire to learn more about the perceived difficulty of attacks.

**Figure 4. Mean authentication times.**

**Figure 5. Number of attempts before a successful entry.**

In the process of this study, we ensured that the participants were not influenced by the perceived difficulty of attacks.

<table>
<thead>
<tr>
<th>Mean values</th>
<th>GazeStart</th>
<th>GazeEnd</th>
<th>GazeMiddle</th>
<th>GazeStartEnd</th>
</tr>
</thead>
<tbody>
<tr>
<td>Binary success rate (0%=all incorrect;100%=all correct)</td>
<td>19%</td>
<td>16%</td>
<td>13%</td>
<td>16%</td>
</tr>
<tr>
<td>Levenshtein distance (0=very similar;4=completely different)</td>
<td>1.36</td>
<td>1.18</td>
<td>1.21</td>
<td>1.27</td>
</tr>
<tr>
<td>Perceived Difficulty (1=very easy;5=very hard)</td>
<td>4</td>
<td>3</td>
<td>3.5</td>
<td>4</td>
</tr>
</tbody>
</table>

Table 2. We measured the binary success rate (i.e., whether or not the attacker’s guess is correct) and the Levenshtein distance (i.e., how similar the guesses are to the correct password). A low success rate and high Levenshtein distance indicate high observation resistance and hence higher security. The most secure passwords are the GazeMiddle, where only 13% of the passwords were observed successfully, and guesses were 55% similar to the correct ones. Participants reported the perceived difficulty of attacks before knowing whether their guesses were correct.

Table 2 shows the average success rate and Levenshtein distance for each condition. While participants succeeded the most when attacking GazeStart, their guesses had the longest average distance from the correct password. Guesses against GazeEnd were closest to the correct password (i.e., low distance) possibly because the attackers knew that each password consists of 4 inputs, and after seeing two or three touch-inputs they foresaw that the following inputs are gaze-based.

We also collected the perceived difficulty which participants indicated through a Likert scale (5-points;1=very easy;5=very hard). Table 2 shows that participants found all types difficult to observe, but GazeStart and GazeStartEnd were perceived to be more difficult compared to GazeEnd and GazeMiddle. These values are in-line with the Levenshtein distances.

DISCUSSION
The proposed architecture enables usable and secure authentication on public displays. Users simply approach the display with their unmodified personal mobile device that has our app installed. The app retrieves the IP of the server that is broadcasted by the BLE beacon, allowing the user to directly authenticate without entering URLs or scanning QR codes.

A usability study shows that the use of multimodal passwords in that setup is feasible, well perceived, and is only slightly slower than the less secure PINs (von Zeeuw et al. report 1.5 seconds for PIN-entry [39]). While users make few errors, previous work has shown that users are willing to correct errors on public displays [20]. A security study showed that authentication is robust against observation attacks (only 13–19% successful attacks in optimal conditions), which is more secure than PINs and several recently proposed systems. For example, attacks against EyePassShapes [8], EyePIN [12], GazeTouchPass [19] and XSide [11] had success rates of 42%, 55%, 19% and 9% – 38% respectively. Furthermore, the fact that gaze-input does not leave traces on the device makes GTmoPass secure against thermal and smudge attacks even if the attacker gets hold of the mobile device.

Trade-off between Usability and Security
While authentication using PINs is fast, it is known to be insecure and highly vulnerable to observation attacks [19, 37] and thermal attacks [1]. On the other hand authentication using multimodal passwords is more secure, but takes longer time compared to PINs. While even a slight increase in authentication time on mobile devices has a big impact considering that users unlock their mobile devices more than 50 times a day [18], we argue that authentication on public displays does not happen as often and hence a slight increase (between 1.3 and 2.5 seconds in our case) is not very significant.

In addition to the overall trade-off, we found that having gaze-input in the middle of the password (GazeMiddle) is the least likely to be successfully attacked, but also requires the longest time to enter. In general, it was found that providing consecutive gaze inputs results in longer authentication times. This was the case in GazeStartTwo and GazeEndTwo and GazeMiddle (see examples in Table 1). This is due to the time it takes to perform a gaze gesture, look to the front again, then perform another gaze gesture. On the other hand, guesses against GazeEnd are the closest to the actual password. We expect that after observing two or three touch inputs, participants foresaw that the following inputs could be gaze-based.

Perceived Difficulty of Shoulder Surfing
It is interesting that the perceived difficulty of attacks reported by participants was more in-line with the Levenshtein distances rather than with the binary success rate. The Levenshtein distance metric evaluates how similar a guess is to the actual password, which means that it also reflects how many times digits or gaze gestures were observed correctly. This means that unlike the binary success rate, participant’s confidence in identifying particular inputs can be a valid indicator of low Levenshtein distances.

It is not surprising that the final success rate does not correlate with the perceived difficulty. In fact, previous work reported that attackers often underestimate the perceived difficulty of shoulder surfing. For example, in the work by George et al. [17], the perceived difficulty of performing shoulder surfing attacks changed drastically after trying to perform attacks.

CONCLUSION
In this work we showed that GTmoPass offers a secure authentication architecture for public displays. A usability and a security study showed that GTmoPass is usable and secure against shoulder surfing. We also discussed how thermal and smudge attacks are infeasible by design.

In the future, we want to evaluate more complicated threat models. For example, a combination of a thermal attack to uncover touch input and an observation attack to uncover gaze input, or multiple consecutive observations by insiders (e.g., family members or work colleagues). Another interesting threat model is the case of having two attackers: one observing the eyes, while the other observes the touchscreen. We also intend to conduct a field study to better understand how users perceive GTmoPass in the wild. A further direction for future work is to include a third inference factor. This can be done by scanning the fingerprint or by face detection using the front-facing camera.

ACKNOWLEDGEMENT
This work was supported by a Google IoT Technology Research Award.
REFERENCES


Challenges of Gaze-enabled Public Displays
EyeScout
Active Eye Tracking for Position and Movement Independent Gaze Interaction with Large Public Displays.
EyeScout: Active Eye Tracking for Position and Movement Independent Gaze Interaction with Large Public Displays

Mohamed Khamis¹, Axel Hoesl¹, Alexander Klimeczak¹, Martin Reiss¹, Florian Alt¹, Andreas Bulling²

¹Ubiquitous Interactive Systems Group, LMU Munich, Germany
²Max Planck Institute for Informatics, Saarland Informatics Campus, Germany

{mohamed.khamis, axel.hoesl, florian.alt}@ifi.lmu.de, bulling@mpi-inf.mpg.de

ABSTRACT
While gaze holds a lot of promise for hands-free interaction with public displays, remote eye trackers with their confined tracking box restrict users to a single stationary position in front of the display. We present EyeScout, an active eye tracking system that combines an eye tracker mounted on a rail system with a computational method to automatically detect and align the tracker with the user’s lateral movement. EyeScout addresses key limitations of current gaze-enabled large public displays by offering two novel gaze-interaction modes for a single user: In “Walk then Interact” the user can walk up to an arbitrary position in front of the display and interact, while in “Walk and Interact” the user can interact even while on the move. We report on a user study that shows that EyeScout is well perceived by users, extends a public display’s sweet spot into a sweet line, and reduces gaze interaction kick-off time to 3.5 seconds – a 62% improvement over state of the art solutions. We discuss sample applications that demonstrate how EyeScout can enable position and movement-independent gaze interaction with large public displays.

INTRODUCTION
Over the last years we have witnessed a significant increase in the number and size of displays deployed in public. Large displays are now commonly found in public communal spaces, such as shopping malls or transit areas in airports and train stations [13]. At the same time, as sensing technologies are becoming cheaper and more robust, various modalities for interacting with these displays have been explored. A particularly promising interaction modality is gaze, given that it is fast, natural, and intuitive to use [41].

However, in contrast to common desktop settings, public displays afford ad-hoc use over short periods of time by “passersby”, i.e. users who move in front of the display [26, 27]. These characteristics pose three unique challenges that have so far forestalled wider adoption of gaze interaction on large public displays: 1) Gaze-enabled public displays cannot afford time-consuming eye tracker calibration for each user prior to interaction [21], 2) they have to allow passersby to interact from different positions [36] and 3) they have to support interactions while on the move [30]. Previous work mainly addressed the first challenge [22, 34, 40]. To date, addressing the latter two currently requires augmentation of each individual user with head-mounted eye trackers as well as interconnected displays [24].

To address the last two challenges we present EyeScout, a novel active eye tracking system that enables gaze interaction for a single user on large public displays from different lateral positions in front of the display and while on the move. EyeScout consists of a body detection and tracking module using a depth sensor, and an eye tracking module using an eye tracker mounted on a rail system. Our system detects
the user’s position in front of the display and then moves the eye tracker to face and follow the user. EyeScout thereby enables gaze interaction with large public displays that are (1) position-independent: the user can interact from different positions within 90 cm in front of the display and along the display’s full extent, and (2) movement-independent: the user can interact via gaze while passing by the display. EyeScout builds on existing work that employed Pursuits [34], a popular calibration-free gaze interaction technique. It is important to note that EyeScout readily supports other gaze interaction techniques, such as gaze gestures [14] or pupil-canthi-ratio [40].

The specific contributions of this work are three-fold: First, we introduce the design and implementation of EyeScout, a novel active eye tracking system that enables gaze interaction for a single user on large displays. Second, we report on our findings from a controlled laboratory study (N=23) to evaluate the performance of EyeScout. We evaluate EyeScout for scenarios in which users “Walk then Interact” (to test for position independence), as well as “Walk and Interact” (to test for movement independence). Findings from our study show that EyeScout is well-suited for both interaction modes and well-perceived by users. In particular, EyeScout reduces the time required to kick-off gaze interaction (i.e., the time starting from the moment the user appears in front of the display until the time the user interacts) to 3.5 seconds—a 62% improvement over state-of-the-art methods [1, 41]. Finally, we discuss how active eye tracking using EyeScout can enable novel gaze-based applications that are not possible with current systems, such as gaze interaction with non-planar displays, and on escalators and moving walkways.

RELATED WORK

Our work builds on three strands of prior work: (1) Gaze interaction with public displays and (2) active eye tracking.

Gaze Interaction With Public Displays

Gaze holds particular promise for interaction with public displays given that it, for example, overcomes the embarrassment problems associated with mid-air gestures [8], reflects attention [32], and allows at-a-distance interactions [17]. However, gaze-enabled public displays also face several unique challenges. First, although eye tracker calibration may be acceptable in desktop settings where users interact for long periods of time, interaction time on public displays is short [26, 27], which makes time-consuming tasks, in particular calibration, undesirable [21]. Recent works have therefore either tried to improve calibration [22] or employed calibration-free interaction techniques [19, 34, 40].

Second, in contrast to desktop settings, users approach public displays from different directions and want to interact with them from different locations, distances, and relative orientations [36]. However, existing gaze-enabled public displays restrict users’ position; users have to position themselves within the tracking box of the eye tracker for their eyes to be detected [20, 22, 34, 41]. A common approach to address this problem is to guide passersby to the right position in front of the eye tracker, for example, using on-screen mirrored video feeds [41], markers on the floor [20], or on-screen visual cues that are adapted based on the user’s distance to the display [1]. In contrast, EyeScout moves the eye tracker to the user as soon as they approach the display or as they walk along it. An alternative approach is to use head-mounted eye tracking that allows for freedom of movement. However, this approach requires the eye tracker to (1) identify the position and borders of surrounding displays, (2) map gaze estimates to on-screen positions, and (3) communicate gaze data to the display. Prior work attached printed markers on the display [39] or used on-screen visual markers [15] to locate the display and map the gaze estimates onto it. These approaches usually rely on a tethered connection to the display. Lander et al. used visual feature tracking to determine the positions of surrounding displays, and exchanged gaze data over Wifi [24].

Although mobile trackers are starting to become ubiquitous and integrated into eyewear computers and despite the vision of pervasive display networks [13], pervasive integration on such a big scale would require taking concepts from lab settings to the field. In-field application is currently challenging, as participants need to be explicitly hired and asked to wear mobile eye trackers [12]. Until passersby wearing mobile eye trackers becomes the norm, there is a need to study user behavior on gaze-enabled public displays using other means, such as remote eye trackers.

Active Eye Tracking

One way to achieve position-independent gaze interaction with public displays is by using active eye tracking, i.e. systems that adapt to the user’s eye position rather than restricting their head and/or body movements. Active eye tracking is particularly popular in medicine; it is used in eye surgery to account for eye and body movements during lasik operations [25]. A common approach is to use a single [9, 10, 11, 29] or multiple pan-and-tilt cameras [6], or pan-and-tilt mirrors [28] to adapt to the user’s head position. Hennessey and Fiset used a Kinect to detect faces and adjust the angle of an eye tracker mounted on a pan and tilt mechanism accordingly [17]. While all of these methods demonstrated the potential of active eye tracking, EyeScout is first to move the eye tracker rather than only panning and tilting it in a single fixed position. This way, EyeScout actively accommodates for the user’s body movements along large displays.

THE EYESCOUT SYSTEM

When interacting with large, cylindrical or spherical displays, users approach from different directions and do not necessarily interact from a static position in front of the display [1]. Instead, passersby expect to be able to walk-up to the display and interact from any position. We refer to this interaction mode as “Walk then interact”. Similarly, passersby move at different speeds and often interact with displays while moving [30]. We refer to this interaction mode as “Walk and interact”. The key challenges in both interaction modes are that the system needs to detect the user’s eyes at arbitrary stationary positions or while the user moves in front of the display.

We designed EyeScout for single user gaze interaction specifically with these two interaction modes and associated challenges in mind. The design was inspired by camera motion.
The rail system consists of a 4-meter twin track aluminium rail\(^1\) and a carriage\(^2\) to move the eye tracker. Two 3D-printed end pieces were attached at both ends of the rail (see Figures 4A and 4C). The end pieces serve two main purposes: (1) To hold switches that are activated once the carriage reaches any of the ends. The switches are used for a one-time system calibration that determines the bounds of the rail’s range to prevent the carriage from colliding with other components of the end pieces. (2) Each end piece harbors a steel axis that holds a pulley. A motor, whose axis is connected to the steel axis, is mounted on one of the end pieces. Thus, when the motor spins, the axis is spun and moves a tightened timing belt that moves the carriage. In addition to a DC motor\(^3\), we used a 2-phase digital stepper driver\(^4\) to convert digital signals to commands that can control the motor. The entire rail system was mounted above three evenly distributed tripods (height: 113 cm), as depicted in Figure 5A.

**Eye Tracking Module**

The eye tracking module consists of a remote eye tracker (Tobii REX) and a custom 3D-printed mount. The mount is attached to a tripod head that allows adjusting the angle of the eye tracker. The tripod head is in turn attached to another 3D-printed base that is screwed into the carriage (Figure 4B). This module is responsible for tracking the eyes once they are in range. The minimum and maximum range of the eye tracker (40 cm to 90 cm in our case) are predefined in the body tracking module. This allows the body tracking module to detect when users are too close or too far away from the eye tracker. Similarly, the eye tracking module continuously detects whether or not the user’s eyes are detected. This information can then be used to provide feedback to the user.

For gaze interaction with our system we use Pursuits\(^{34}\), which has been widely adopted recently for calibration-free gaze interaction. Pursuits checks for motion correlation\(^{33}\) between user’s eye movements and trajectories of on-screen targets. The strength of the method lies in its ability to determine which object the user is gazing at without the need for calibrating the eye tracker to each user. As public displays require immediate usability and cannot afford the time spent for eye tracker calibration\(^{21}\), Pursuits is well-suited for use

---

\(^{1}\)igus® drylin® Double rail http://www.igus.eu/wpck/2003/drylin_w_doppelschiene

\(^{2}\)igus® drylin® Double rail/carriage http://www.igus.eu/wpck/8980/drylin_w_Slider_Schienen

\(^{3}\)igus® drylin® step motor NEMA23 http://www.igus.eu/wpck/7663/N11_6_14_2_Schrittmotor_NEMA23

in the context of public displays. Moreover, as users will be moving, gaze estimates can be expected to have low accuracy as humans naturally bob up and down while walking. Its robustness to inaccurate gaze data makes Pursuits even more suitable for our deployment.

Our implementation of Pursuits is based on prior work; we used Pearson’s product-moment correlation with a threshold of 0.85 and a window size of 500 ms [34]. This means that the system computes Pearson’s correlation every 0.5 s. This is similar to previous work, some of which used a 0.5 s window size [20, 34], while Orbits [16] and TextPursuits [22] used 1 s and 2 s respectively. The stimulus whose movement correlates the most with the eye movement is deemed to be the one being looked at, if the correlation is higher than 85%.

Control Module
The control module handles the logic of EyeScout and the communication between the body tracking module and the rail system. It consists of a software component written in C# and a microcontroller (Arduino Due). The software component runs on a Microsoft Surface Pro that is connected to the Kinect via USB. It receives the coordinates of the user’s body from the body tracking module. According to the predefined distance between the Kinect and the rail system (4 meters in our implementation), the software computes an optimal position for the carriage at which the user’s eyes would be in the eye tracker’s range. Based on the coordinates that are received from the eye and body tracking modules, text prompts are shown to instruct the user to stand back or come closer to the system if necessary. These coordinates can be sent to the microcontroller via Bluetooth or USB.

Given the current position of the carriage, the Arduino Due maps the new coordinates received from the software component to a number of motor steps in a specific direction. These values are then forwarded to the stepping motor, which moves the motor accordingly. The microcontroller also interacts with the switches that are attached to the end pieces. In the aforementioned calibration process, the microcontroller determines the bounds of the rail’s range and updates them internally if necessary. After successful calibration, the carriage is never instructed to move far enough to touch the switches again (see Figure 4A). For example, if a user moves out of range the carriage will stop right before it touches the switch. For additional security, the microcontroller issues an emergency stop command in case the carriage touches the switches after calibration, and resets the boundary values. Although the tasks of the microcontroller could also be performed by the software module, we opted for separating the component that interacts with the rail system and the one that interacts with the body tracking module to further minimize the dependencies between different modules. Additionally, while the heavy traffic generated by the body tracking module often requires a tethered connection to the computer (in our case a Kinect One is connected via USB), microcontrollers can be communicated with wirelessly (e.g., via Bluetooth), which would not necessitate long cables between the Kinect and the rail system; the computer could stay next to the Kinect, and the microcontroller could stay next to the rail system (see Figure 3).

EVALUATION
We designed a controlled laboratory study to evaluate the performance of EyeScout for both interaction modes: (1) Walk then interact: the user approaches the display then interact while stationary and (2) Walk and interact: the user interacts with the display while moving at different speeds.

Participants
We recruited 26 participants (11 females) aged 19 to 37 years ($M = 26.77, SD = 4.46$). All participants had normal or corrected-to-normal vision. Four had previous experience with body tracking devices such as Kinect, out of which three participants had prior experience with eye tracking. One participant was not detected by the Kinect due to wearing a black outfit, and thus was excluded from the analysis.

Apparatus
We deployed EyeScout in one of our lab spaces (7.15 m × 5.65 m). The system was placed parallel to the wall at a distance of 117 cm (see Figure 5A). We used a short throw projector (1920 × 1080 pixels) positioned 92 cm from the wall. The eye tracker angle was adjusted at 50° to the display.

Study Design and Procedure
The study was split into two experiments, each evaluating one of the two interaction modes. All participants took part in both experiments. Half of the participants started with “Walk then Interact” while the other half started with “Walk and Interact”. Each experiment also followed a within-subjects design in which all participants performed all conditions.

The experimenters started by introducing the study and asking the participant to sign a consent form. In both experiments, the system showed a white vertical rectangle (the “interaction frame”) in which three dots moved in either linear or circular trajectory (see Figure 5). The participant’s task was to select the red moving dot from among the grey ones via Pursuits, i.e., by simply following it with their eyes. We picked two trajectory types that are commonly used in implementations of Pursuits: circular [16, 20] and linear [22, 34] trajectories. All simultaneously shown moving dots were selectable and followed the same trajectory. We predefined 8 arrangements for moving dots (see sample arrangement in Figure 5B), 4 of which followed linear trajectories, while the other 4 followed circular trajectories. Figures 5C and 5D show one example of each. The participant was shown one arrangement (i.e., one set of three selectable dots moving in the same trajectory) at a time. In each selection, participants had to select 1 of 3 targets. Dots disappeared after being selected. After participants had performed all selections in both experiments, they filled in a questionnaire and participated in a semi-structured interview.

Experiment 1: Walk then Interact
To study the interaction mode where passersby approach a random spot in front of the display and then interact while stationary, the interaction frame appeared at a random position on the display in this experiment. The participant was asked to walk to the frame and then select the red dot via Pursuits. The carriage approached the participant as he/she approached the frame, to ultimately position the eye tracker in front of
the user. After a successful selection, the frame reappeared at another random position that was at least 50 cm away from the previous one. This was done to ensure that the participant had to approach a different spot in front of the display, rather than performing the selection from the current position. Each participant performed 3 blocks, each of which covered one selection per trajectory arrangement. Thus, every participant performed 8 trajectory arrangements × 3 blocks = 24 selections. We consider these blocks an additional independent variable, referred to in the following as “repetitions”. Repetitions were studied to investigate learning or fatigue effects. The order of conditions was counter-balanced using a Latin-square.

Experiment 2: Walk and Interact
To study the interaction mode where passersby interact while moving, in this experiment the participant, the interaction-frame, and the carriage were all moving. We focus on scenarios in which users interact with content that moves with them, as typically done in large interactive displays intended for moving users [30]. To evaluate if the carriage’s speed had an impact on detection accuracy we introduced an independent variable “carriage speed” with three conditions: 0.36 m/s (maximum speed of EyeScout), 0.3 m/s, and 0.24 m/s. The interaction frame would follow the participant, but the participant would be able to make a selection only when in range of the eye tracker. Each participant performed 24 selections in this experiment (8 arrangements × 3 speeds). The order of conditions was counter-balanced using a Latin-square.

Limitations
In the current version of our prototype, taller participants are asked by means of the aforementioned textual prompts to step back in order for the eye tracker to detect them. In our study, the angle of the eye tracker to the head was between 35° and 50°. The exact value depends on the user’s height and distance from the tracker. Future systems can adjust the eye tracker’s angle dynamically to be within this range.

Another limitation of gaze interaction while on the move is that it might be affected by motion blur. Although we did not face this problem in our study; the performance of EyeScout was almost similar in “Walk then Interact” (baseline) compared to “Walk and Interact”. However we acknowledge that higher carriage speeds might result in less accurate gaze data.

Like current stationary eye trackers, EyeScout only supports a single user. Multi-user support is one of the most important directions for future work. This can be realized by mounting multiple eye trackers on different belts or by using appearance-based gaze estimation methods [32] that use multiple or a single wide angle RGB camera. In the current implementation of EyeScout, there are three possible scenarios in which a person other than the user appears in range of EyeScout: 1) passersby show up near the user, 2) passersby step between the user and the Kinect, 3) passersby step between the user and the eye tracker. The eye tracker locks onto the user whose eyes are detected even if a passerby occludes the user from the Kinect’s view. This prevents disrupting interaction in case of glitching position tracking, and means that 1) and 2) do not influence the system’s usability. The Kinect can be placed at the top of the interaction area to account for 2) when the user is moving. However 3), while unlikely, would result in the passerby taking over the interaction.

Finally, we can validate EyeScout’s performance with higher walking speeds only after upgrading its motor. However users are likely to slow down to interact when moving [30].

Quantitative Results
Although the Kinect performed fairly well in our setup, it failed to detect one participant wearing black. We further excluded the data of two more participants: One found the task to be overwhelming; he struggled to walk around and look at multiple objects at the same time. The second squinted his eyes too often, resulting in very few collected gaze points. We measured the cruise time in the “Walk then Interact” experiment, i.e., the time it took the carriage from the moment the interaction window appeared till the moment eyes were detected. Because this is the first time a commercial IR-PCR eye tracker is used for active eye tracking, we wanted to investigate whether there is any degradation in the eye tracker performance when it is in motion. This was done by logging the gaze points per second during the Walk and Interact experiment. In both experiments, we additionally measured the error count, which we define as the number of grey dots that were selected before the red dot. There can be 0, 1, or 2 errors before selecting the red dot.

Error Count
As shown in Table 1, errors decreased in the “Walk then Interact” experiment as participants performed more repetitions. This suggests that there could be a learning effect, i.e., participants adapted to the system. In the “Walk and Interact” experiment we found a slight increase in the number of errors as the carriage moved at higher speeds.

Figure 4. (A) and (C) show the 3D-printed end pieces attached at the ends of the rail. Each end piece harbors a pulley that moves the timing belt (D), and a switch (E) that prevents the carriage from accidently colliding with the end piece. One end piece carries the motor (C). A 3D-printed base is screwed into the carriage (B), on which a tripod head is attached. The head allows the adjustment of the eye tracker, held by a 3D-printed holder.
Figure 5. An “interaction window” appeared on the projected display with an arrangement showing three selectable moving dots. The dots moved either in circular trajectories (C) or in linear trajectories (D). The participant's task was to select the red dot. In the “Walk then Interact” experiment, the window appeared at a random place on the display; the participant had to walk to the window then perform the selection. While in the “Walk and Interact” experiment, the window moved along the display; the participant had to walk along the window and perform selections while moving.

Table 1. The table shows the percentage of times a successful selection of a red dot was preceded by 0, 1, or 2 errors (i.e., selection of a gray dot). As participants performed more selections, errors in “Walk then Interact” decreased. This suggests that there could be a learning effect. In “Walk and Interact”, errors increased slightly with higher speeds.

<table>
<thead>
<tr>
<th></th>
<th>Walk then Interact</th>
<th>Walk and Interact</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Repetition 1</td>
<td>Repetition 2</td>
</tr>
<tr>
<td>0</td>
<td>73.9%</td>
<td>83.7%</td>
</tr>
<tr>
<td>1</td>
<td>26.1%</td>
<td>15.8%</td>
</tr>
<tr>
<td>2</td>
<td>0%</td>
<td>0.5%</td>
</tr>
</tbody>
</table>

Table 2. The table shows the mean selection time and standard deviation. Selection times in “Walk and Interact” are longer than in “Walk then Interact” due to tracking distortions caused by walking. While in “Walk and Interact” mean selection time is only slightly more than in previous work (e.g., 1.5 s to 2.0 s [20]). In “Walk and Interact” mean selection time is 4.9 s, which is longer due to tracking distortions caused by walking.

<table>
<thead>
<tr>
<th></th>
<th>Walk then Interact</th>
<th>Walk and Interact</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Repetition 1</td>
<td>Repetition 2</td>
</tr>
<tr>
<td>μ</td>
<td>2.4 s</td>
<td>2.9 s</td>
</tr>
<tr>
<td>σ</td>
<td>2.1 s</td>
<td>2.2 s</td>
</tr>
</tbody>
</table>

Selection time

Table 2 summarizes the selection times. The overall mean selection time in “Walk then Interact” is 2.7 seconds, which is only slightly more than in previous work (e.g., 1.5 s to 2.0 s [20]). In “Walk and Interact” mean selection time is 4.9 s, which is longer due to tracking distortions caused by walking.

Questionnaire

We asked participants about the perceived easiness and precision of selections on a 5-point Likert scale. In the “walk then interact” experiment, participants felt that walking up to the display then making selections was easy ($Mdn = 4$, $SD = 0.59$) and precise ($Mdn = 4$, $SD = 1.08$) (see Figures 6 and 7). They also agreed that the eye tracker was positioned properly in front of them ($Mdn = 4$, $SD = 0.66$). A Friedman test showed statistically significant differences in perceived easiness of selections depending on carriage speed $\chi^2(2) = 9.414, p = 0.009$ in the “walk and interact” experiment. Post hoc analysis with Wilcoxon signed-rank tests was conducted with Bonferroni correction, indicating significant differences ($p < 0.017$). Medium perceived easiness of selection levels for the slow, medium, and fast carriage speeds were 5 (4 to 5), 4 (4 to 5) and 4 (4 to 4), respectively. There were no significant differences between medium and slow carriage speeds ($Z = -0.535, p = 0.593$). However, there was a statistically significant reduction in perceived easiness in fast vs slow carriage speed ($Z = -2.555, p = 0.011$), and in fast carriage speed vs medium carriage speed ($Z = -2.517, p = 0.012$). This means that selections done during slower and medium carriage speeds are perceived to be easier to perform (Figure 6). We performed a Friedman test to determine if there were differences in perceived precision across the different carriage speeds. Perceived precision was consistent ($Mdn = 4$) across the different carriage speeds and there were no statistically significant differences $\chi^2(2) = 3.073, p = 0.215$. This means there is no evidence that perceived precision changes depending on the carriage speed (Figure 7).
Walk then Interact
Walk and Interact
(Slow speed)
Walk and Interact
(Medium speed)
Walk and Interact
(Fast speed)

Perceived Easiness of Selections

Figure 6. Overall, participants found it easy to make selections when using EyeScout. However, making selections on medium and slow speeds is perceived to be easier than on fast speeds.

Perceived Precision of Selections

Figure 7. Participants perceived precision to be generally high. It seems that selections at higher speeds in the “Walk and Interact” experiment are perceived to be slightly less precise, however we found no significant differences to support this.

Observations and Qualitative Feedback

The overall feedback about the experience was very positive. Participants mentioned that they found the system “interesting” and thought it was “working surprisingly well”. One participant found the experience of being followed by the eye tracker to be “futuristic”. Another participant commented that he found the idea “novel” and “sci-fi”. This is in-line with the novelty effect often experienced when interacting via gaze.

Participants also mentioned some aspects of EyeScout that could be improved.

Hardware Improvements

Although carefully placed to avoid tripping, three participants reported that the legs of the tripods sometimes distracted them. This suggests that future versions of EyeScout should be mounted differently. For example, the rail can be engraved into the wall underneath the display. Five participants reported that the cable that connects the eye tracker to the PC distracted them shortly when they saw it the first time. In field deployments of EyeScout, wireless technology (e.g., WiFi or Bluetooth) should be utilized instead. To further reduce distraction, the tracker could be embedded into a case with semi-transparent glass so that the current position of the tracker is not visible to users. Yet, knowledge about the position of the eye tracker might positively influence the position of the user – hence, this needs to be subject to future investigation.

A female participant with long dark hair was not always correctly detected by the Kinect due to the aforementioned problem with detecting dark objects. While she reported that the eye tracker was consequently not always in front of her, she did not report any problems regarding the responsiveness of the system. A possible direction of improvement is to place the motion sensing device at the top of the interaction area, or embed a wide range sensor into the display.

Interaction

Two participants reported they were uncomfortable with performing Pursuits against circular trajectories. Some participants also reported feeling tired after performing 48 selections using Pursuits. This feedback is in line with findings reported from lab studies of Pursuits [22]. Given that interactions in real deployments would not involve as many Pursuit selections as in our study, this fatigue effect can be expected to play a minor effect in real deployments. However, applications that expect multiple selections (e.g., games) should be designed with the fatigue effect in mind.

One participant reported not having noticed the text prompts used to guide the user closer or farther from the eye tracker. Previous work has shown that public display users sometimes miss on-screen content, and are more likely to notice it if it is attached to their user representation [35]. Another participant was not confident that the system recognized his eyes, which led him to look at the eye tracker during his first trials. However that was only before he realized that the projected screen shows feedback when eyes are not detected. This suggests that the system should always provide feedback to indicate that the eyes are detected, rather than only when they are not in range. Furthermore, future work should consider different visual feedback methods, such as continuously showing eye symbols on the screen and adapting them depending on the state of eye detection. Similar to previous work [1], the content can be adapted to subconsciously guide the user by making it completely visible only when the user is at a particular distance from the display.

Walking Strategies

When asked how to move, the experimenters told the participants that they are free to move however they liked. We noticed that participants walked in different ways in the “Walk and Interact” experiment. While the majority walked naturally with their head turned towards the display, some walked sideways with their entire body facing the display. Participants who walked sideways reported that it was uncomfortable, but they walked that way thinking that the system would not detect them otherwise. One participant tried both and eventually settled on the natural walk. User interfaces of commercial eye trackers explicitly tell their users to relax and act naturally; EyeScout can similarly provide such feedback when unnatural moving behavior is detected.
SAMPLE APPLICATIONS
We envision EyeScout to be used in a variety of scenarios. We describe three examples in which the use of EyeScout enlarges the interaction space and opens up novel possibilities for interaction via gaze that are otherwise infeasible.

Gaze Interaction with Scenery
EyeScout could be used for gaze-based interaction with a scenery (see Figure 8A). For example, observation decks and towers offer vantage points for scenic overviews (e.g., of a city or a certain landmark). On these platforms, tourists and visitors overlook a scene and are often provided with audio guides that, in a sequential manner, describe what can be seen from the platform (for example, “To your left, next to the red building, you can see the townhall.”). However, this sequential feeding of information eliminates the exploratory nature of these platforms, thus hindering the tourist’s experience. Even with the presence of a human tour guide, pointing at something (for example, a building or a landmark) and asking for information is not straightforward; communicating a point of interest to others by pointing is also inefficient.

We propose augmenting these platforms with EyeScout to enable eye tracking across the whole platform as shown in Figure 8A. In this scenario, the eye tracker would follow the user as he/she walks on the platform, the system would then be able to detect which buildings/landmarks the user is looking at, or allow the user to select landmarks. In-situ information about the area of interest can then be shown in the form of visual overlays or audio messages. The displayed information could be predefined in the system and loaded based on the positions of the areas of interest relative to the platform, and the position of the user. Since this application requires a precise gaze point, calibration might be required. A direction for future work is to investigate how well calibration-free techniques such as TextPursuits [22] perform in scenarios where users are moving. Another alternative is to use gaze gestures (e.g., right and left) to allow users to select the landmark they want to learn about.

Eye Tracking on Moving Walkways and Escalators
Moving walkways (aka travelators) and escalators can be found in large numbers in airports, supermarkets, ski resorts, museums, and public transport stations. People using them are usually presented with static content at one or both sides of the walkways, such as advertisements. Gaze interaction or attention measurements in walkways is infeasible using current systems and techniques, unless each passerby is augmented with a head-mounted eye tracker.

Walkways and escalators could be augmented with EyeScout, such that the eye tracker would follow the user (see Figure 8B). In static non-interactive contexts, eye tracking enabled through EyeScout could provide information about the passerby’s attention (for example, which advertisements passerby look at). The content displayed on one or two sides can also be interactive, in this case interacting with a UI while standing on a moving walkway via touch would be challenging unless the UI moves with the user. On the other hand, in cases where the user “passes by” the content (for example, moving walkway surrounded by stationary exhibits), interaction via gaze extends the user’s reach, as the user’s gaze vector could reach farther areas compared to interaction via touch or via mid-air gestures.

Gaze-based Interaction with Non-Planar Displays
There has been a recent interest in deploying and studying passerby behavior in front of non-planar displays such as cylindrical [4, 5, 7] and spherical [3, 37, 38] displays. Due to their form factor, the requirement of adapting to users approaching and interacting from different directions and positions becomes even more prominent. To date, the strict positioning constraints imposed by eye trackers make gaze interaction infeasible with such non-planar displays.

Although we evaluated EyeScout only in the context of a large planar display, the same concept is applicable to non-planar ones by using a circular rail system (Figure 8C). A camera mounted on the top of the display could detect surrounding motion and move the eye tracker to intercept passersby as they approach the display. Gaze could then be tracked to understand which content the passersby attend to, or to enable interaction as they move around the display. We believe such displays to be particularly useful in guiding users to less crowded areas of a public space.
DISCUSSION
Findings from our study show that EyeScout successfully overcomes the positioning requirements imposed by classical eye tracking systems, and is flexible to lateral movements in front of the display at different speeds. We also found that it is well perceived by users, who reported finding it generally easy and precise to perform selections using Pursuits.

Improvement over State-of-the-art
62% – 87% Faster in Kickstartering Gaze Interaction
State-of-the-art methods for guiding passersby to the sweet spot – which is, in our case, the area in which the user is detected by the eye tracker – were reported to require 4.5 to 23 seconds [1]. After reaching the sweet spot, users typically need to align their face to the correct position in front of the gaze-enabled display. Recent work reported that users required 4.8 seconds for the face alignment on a gaze-enabled display [41]. By adding these values, we can expect that even for state-of-the-art methods, passersby need 9.3 to 27.8 seconds before they can start interaction via gaze. EyeScout reduces this time to 3.5 seconds, which represents a 62% to 87% improvement. This improvement is due to EyeScout not requiring users to move to the sweet spot, nor to align their faces, but instead “doing the work for them”. EyeScout still requires less time compared to previous approaches although it informs participants if they are too far from or too close to the eye tracker, and asks them to reposition accordingly.

Previous work showed that unless displays are immediately usable, users abandon them [26, 27]. Hence, we expect an increase in conversion rates due to EyeScouts reduction of kick-off time. In future work, this increase can be quantified through a field study.

From “Sweet Spot” to “Sweet Line”
EyeScout maximizes the horizontal flexibility of public displays. While previous work report an optimal interaction spot (the sweet spot [27]), our work extends the sweet spot to a sweet line: an area with the width of the display, and the length of the eye tracker’s range. Future work can further extend the distance to the screen by incorporating 3D vector rig6.

Gaze-based Interaction on the Move
Although the Tobii REX eye tracker that we used is intended for stationary settings, it performed fairly well when in motion. The number of collected gaze points stayed almost the same across the different cruise speeds and was sufficient to perform Pursuits-based selections. Although there is a slight increase in error when using faster cruise speeds compared to slower ones (see Table 1), the accuracy of selections achieved in the “Walk then Interact” experiment do not differ much from those in the “Walk and Interact” experiment. We furthermore found that participants generally felt selections to be easy and precise but results were in favor of slower speeds compared to faster ones in the “Walk and Interact” experiment. The differences between the perception of easiness and precision of both experiments were not significant. Figures 6 and 7 suggest that participants perceived selections positively in all modes.

6http://www.vector-cam.com/services.html

Gaze Interaction Techniques Other than Pursuits
We opted to use Pursuits because it is the state-of-the-art method for calibration-free gaze interaction with public displays and it addresses the first of the three challenges mentioned at the beginning of this paper. However, EyeScout is not limited to this technique. It is straight forward to replace Pursuits by other calibration-free techniques such as eye gestures [14] or pupil-cantith-ratio [40]. Furthermore, future work can experiment with calibrating the eye tracker implicitly while users are interacting as in TextPursuits [22] to collect more accurate gaze points.

However, we cannot claim that all methods can be adapted into active eye tracking scenarios; Pursuits is robust to uncalibrated gaze points, which is likely one of the reasons it performs well while the eye tracker was in motion. It will be interesting to see whether techniques that require accurate gaze estimates, such as dwell time, can be used with EyeScout or whether the applications enabled by our system will remain infeasible if these techniques are used.

Upgrading EyeScout
The gaze-interaction technique is not the only upgradable component of EyeScout. The way EyeScout is designed allows straightforward upgrades and improvements to the different hardware and software modules. Basically all modules are upgradable, including the motor, the eye tracker, the body tracker, and even the control unit. For example, a stronger motor can be used to increase the cruise speed, a wide-angle RGB camera can be used instead of infrared-based eye trackers, and body positions can be detected via on-body sensors.

CONCLUSION
In this work we introduced the design and implementation of EyeScout, a novel active eye tracking system that addresses two challenges that were unsolved in research on gaze-enabled public displays to date. Findings from a user study show that EyeScout is not only well-perceived but also allows passersby to interact with large displays (1) from different positions and (2) while one the move. We furthermore introduced several sample applications that demonstrate how active eye tracking can enable new interactions with gaze that were not possible before. Our detailed description of EyeScout’s implementation is valuable for researchers and practitioners alike who would like to employ active eye tracking into their public displays.

ACKNOWLEDGEMENTS
Thanks to Patrick Mörwald for his help in developing EyeScout. Work on this project was partially funded by the Bavarian State Ministry of Education, Science and the Arts in the framework of the Centre Digitisation.Bavaria (ZD.B).

REFERENCES
   http://dx.doi.org/10.1145/2807442.2807490
9 EyeScout: Active Eye Tracking for Position and Movement Independent Gaze Interaction with Large Public Displays.


9 EyeScout: Active Eye Tracking for Position and Movement Independent Gaze Interaction with Large Public Displays.


Using Text for Pursuits-Based Interaction and Calibration on Public Displays.
TextPursuits: Using Text for Pursuits-Based Interaction and Calibration on Public Displays

Mohamed Khamis\textsuperscript{1}, Ozan Saltuk\textsuperscript{1}, Alina Hang\textsuperscript{2}, Katharina Stolz\textsuperscript{1}, Andreas Bulling\textsuperscript{1}, Florian Alt\textsuperscript{1}

\textsuperscript{1} LMU Munich, Germany
\textsuperscript{2} Sonix Group, Munich, Germany
\textsuperscript{3} Max Planck Institute for Informatics, Saarbrücken, Germany

\{mohamed.khamis, florian.alt\}@ifi.lmu.de, bulling@mpi-inf.mpg.de


text

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure1.png}
\caption{We explore the use of text-based stimuli to enable gaze interaction with public displays using the Pursuits technique \cite{47}. Motivated by the fact that much of the content on large displays is text, we investigate two use cases: (a) Users can spontaneously interact with text-based content without calibration. A sample application could be a survey where answers in the form of text are selected by reading them (left). (b) An eye tracker can be calibrated implicitly as users read text on the screen (right). After calibration, fine-grained information on the user’s gaze point are obtained.}
\end{figure}

\section*{ABSTRACT}

In this paper we show how reading text on large display can be used to enable gaze interaction in public space. Our research is motivated by the fact that much of the content on public displays includes text. Hence, researchers and practitioners could greatly benefit from users being able to spontaneously interact as well as to implicitly calibrate an eye tracker while simply reading this text. In particular, we adapt Pursuits, a technique that correlates users’ eye movements with moving on-screen targets. While prior work used abstract objects or dots as targets, we explore the use of Pursuits with text (read-and-pursue). Thereby we address the challenge that eye movements performed while reading interfere with the pursuit movements. Results from two user studies (N=37) show that Pursuits with text is feasible and can achieve similar accuracy as non-text-based pursuit approaches. While calibration is less accurate, it integrates smoothly with reading and allows areas of the display the user is looking at to be identified.

\section*{INTRODUCTION}

As they are becoming ubiquitous and cheap to deploy, displays can be found in public spaces such as airports \cite{4}, shopping centers \cite{9} and train stations \cite{8}. At the same time, sensing technologies are becoming increasingly available for easy and low cost integration with public displays, supporting different ways of interaction. Common interaction modalities for displays include touch \cite{10}, smart phone interaction \cite{3, 12}, mid-air gestures \cite{34}, and recently also gaze \cite{20, 48, 55}.

Gaze holds particular promise for public displays \cite{22}. It is intuitive \cite{46}, natural to use \cite{47}, indicates visual attention, and usually precedes action \cite{31}. However, a drawback is that eye trackers require calibration, which is time-consuming and cumbersome \cite{31}. While devoting time for calibration is acceptable for desktop settings, public displays require immediate usability \cite{33} as interaction times are usually short \cite{34}. Hence, calibration has been identified as one of core challenges of gaze-enabled public displays \cite{21}. Prior work investigated alternative techniques \cite{48, 55}.

A popular approach is Pursuits \cite{47, 49}, which relies on correlating movements of dynamic objects on the display with the smooth pursuit eye movement performed when following a moving object. Pursuits was successfully deployed for multiple public display installations, where it was used for both gaze interaction \cite{20, 48} and eye tracker calibration \cite{6, 36}.
Meanwhile, one of the most prominent content types on public displays is text. For example, displays are utilized for opinion gathering and sharing in public areas [17, 24]. In many applications passersby read and select from a set of text-based options [15, 16, 35, 42, 44, 50]. And also (pervasive) advertising on public displays often heavily relies on text [1].

Nevertheless, little is known about whether and how Pursuits can be used with text. To date, Pursuits has been studied with moving dot-like stimuli, for which the user gazes at a single, spatially clearly defined target. On the other hand, the use of Pursuits with textual stimuli is not straightforward: reading is not spatially confined and overlays the smooth pursuit movement, which could result in difficulty in correlating eye movements and the trajectory of text-based stimuli. Also, due to the Midas effect, gaze-based systems need to distinguish users reading textual content from interacting with it.

We investigate the use of text as stimulus for Pursuits. We see two main use cases for public displays: (1) It can be used for calibration-free gaze interaction [20, 48]; by displaying moving objects, users can "pursue" the object they want to select. The system then determines which object the user is looking at by correlating eye movements and movements of the objects. (2) Pursuits can be used for easier and less tedious calibration of eye trackers [6, 36]; by following a moving stimulus, mappings between movements of the user’s eyes and the stimulus can be collected and used for calibration.

To provide a proof-of-concept we implemented two applications: EyeVote is a survey application for public displays that enables a user to select an answer from a set of text-based options. Read2Calibrate shows animated text on the screen and feeds the gaze data to an algorithm that gradually enhances the calibration of the gaze tracking. We used both systems in two lab studies with the goal to assess accuracy based on different configurations (text inclination, text speed, text length, and trajectory) and to obtain early insights on the users’ view. The results show that text-based stimuli can be used for Pursuits-based gaze interaction but that designers need to be careful about the text trajectories and text length in order to minimize detection errors. Text-based calibration is in general less precise than state-of-the-art calibration procedures. However, the accuracy is sufficient to identify objects a user is looking at on the screen. We found text inclination to have a strong influence on the calibration quality. Both for interaction and calibration designers may need to make a tradeoff between the configuration leading to the most accurate results and the users’ preferred configuration.

The contribution of this work is threefold: (1) We present a prototype application, EyeVote, that allows text to be selected using Pursuits and report on a user study with 19 participants, assessing the accuracy and error rate based on different configurations (text length, trajectory). (2) We introduce Read2Calibrate, a system for calibrating eye trackers for displays by utilizing smooth pursuit eye movements and a text-based stimulus. Again, we evaluated the system with regard to accuracy based on different configurations (text inclination, text speed). (3) We derive a set of guidelines and recommendations for using text with Pursuits for both interaction and calibration.

**RELATED WORK**

We build on two main strands of previous work: gaze interaction with public displays and interaction with text via gaze.

**Gaze-based Interaction with Public Displays**

Due to the benefits of gaze for public displays, two research directions have emerged to counter the problems associated with calibration on public displays: (1) enabling calibration-free gaze interaction for displays, and (2) making gaze calibration on public displays less tedious.

**Calibration-free Gaze Interaction with Public Displays**

Acknowledging the unsuitability of classic calibration for public displays, multiple systems were built to provide calibration-free gaze-based interaction. SideWays [53] and GazeHorizon [55, 56] use the pupil-canthi-ratio [54] to estimate horizontal gaze direction without calibration.

Pursuits [49, 47] can also be used for calibration-free gaze interaction. The technique requires displaying a dynamic interface [48], where “pursutable” objects move. Eye movements are then correlated to the movements of the objects. The object whose movement correlates the most with that of the eyes is then assumed to be the one the user is looking at. Since its introduction, Pursuits has been used in a variety of applications including text entry [29], PIN code entry [11, 28] and entertainment applications [47, 49]. Pursuits has also been used for interaction with smart watches [13, 14] and interaction in smart environments [45]. The technique was shown to be intuitive and positively perceived by users [20].

**Eye Tracker Calibration for Public Displays**

Previous work aimed to reduce the effort needed for calibration. For example GazeProjector [27] allows gaze-based interaction across multiple displays using one time calibration and a mobile eye tracker. While mobile eye trackers have several advantages for interaction, public display users cannot be expected to wear them, unless trackers integrated with eye wear become commonplace. Hence remote eye trackers are currently more suited for that domain. Xiong et al. [52] used a remote RGB-D camera that requires one-time calibration.

Work by Pfeuffer et al. [36] uses the eye’s smooth pursuit movement to facilitate calibration. The approach relies on showing a moving object, which acts as a stimulus for the eyes to perform the smooth pursuit movement. Mappings between eye movements and positions of the stimulus are then collected and used to calibrate the eye tracker.

Pfeuffer et al. used a floating “Please wait” label to calibrate eye trackers. Rather than reading a label and keeping fixating it, our approach for calibration relies on gradually revealing text, which intrigues the user to fixate at the gradually revealed letters to understand the statement. Moreover, our work on interaction with text via Pursuits investigates a different aspect, namely we study how users can select from a set of text-based options using Pursuits.

**Interacting with Text via Gaze**

In gaze-based systems, the “Midas touch” effect [18] occurs when the system mistakes a user perceiving content for selecting content. This effect is amplified in the case of text...
as reading requires time to perceive and understand the text. This challenge has been traditionally addressed by using dwell times—the system would require fixating the action element for a longer period of time (e.g. 1000 ms [30]).

Another approach to overcome the Midas touch is to use another modality in addition to gaze. Users of EyePoint [25] gaze at text, press and hold a keyboard button to magnify the area, refine the selection, and then release the button to select text. Stellmack et al. [40, 41] employed a similar approach by combining gaze and touch input. Although this approach was not used for text selection in particular, it is deemed suitable for the task. Kishi and Hayashi [23] combined gaze with on-screen buttons to enable users to select text. Chatterjee et al. [7] developed a text editor where users can move a text cursor by using gaze and pinch gestures.

A third approach is to use gaze gestures. In work by Toyama et al. [43], text selection was done either by repeatedly gazing at the beginning and the end of the text to be translated, or by gazing gradually from the beginning till the end of the text.

Sharmin et al. [38] introduced an automatic scrolling technique that is based on the user’s gaze while reading text. Text 2.0 [5] exploits gaze by, for example, revealing content based on the words the user is currently reading.

Pursuits has the potential to cope with the Midas touch effect. Reading overlays the smooth pursuit eye movement, making false selections while reading less likely. Moreover, the Pursuits algorithm requires setting a window size, which is a time frame after which the correlation is calculated. This gives users the chance to perceive and read the text.

**INTERACTING WITH TEXT USING PURSUITS**

The use of text for interaction via Pursuits has not been investigated in detail before. With our work we close this gap and support designers and developers when it comes to creating text-based content suitable for interaction using Pursuits. In particular, the following section introduces a prototype application that allowed us important aspects of using text for pursuit interaction to be investigated.

**Concept and Implementation**

We implemented a voting system called EyeVote, that uses Pursuits as its only input mechanism for selecting one of several floating textual answers (see Figures 1A and 2). Once the system detects a selection, a confirmation message is shown on the screen, telling the user which answer was recognized. The message is kept for some seconds, followed by the next question and its options.

In the following we describe our implementation of Pursuits, and the experimental variables that we used in the study.

**Text Selection via Pursuits**

Pursuits works by correlating eye movements with those of the selectable options. Prior work utilized the Pearson’s product-moment coefficient to calculate the correlation. Based on pilot experiments and previous work [14, 20, 47, 49], we used the same correlation function with a threshold of 0.9 and a window size of 2000 ms. This means that every 2 seconds, the system computes Pearson’s correlation. The floating answer whose movement correlates the most with the eye movement, is deemed to be the object the user is looking at, as long as the correlation is more than 90%.

To account for reading time and overcome the midas effect, the used window size value is higher than those used in other implementations (e.g. previous work used 500 ms [20, 47] and 1000 ms [14]).

**Trajectories and Text Representations**

We investigate how already established trajectory movements perform with respect to text selection. In particular, the following trajectories were used in our experiments.

1. Circular trajectory [13, 14, 20, 47] (Figure 2 top left).
2. Linear trajectory [20, 47, 49] (Figure 2 top right).
3. Rectangular trajectory [36] (Figure 2 bottom left).
4. Zigzag trajectory [47] (Figure 2 bottom right).
5. Mixed trajectory (each object follows one of the above trajectories).

We supported different text representations for the answers:

1. Short answers (<25 characters).
2. Two-lined answers.
3. Long answers (25+ characters).

**Evaluating Text Selection Using Pursuits**

The main goal of this experiment was to understand the influence of different text characteristics on the accuracy of selection via Pursuits. In particular, we compared the effect of the aforementioned trajectory types and text lengths on detection errors. In addition, we assessed the effect of the different trajectory types on the perceived workload. To minimize any external influences, we conducted the study in the lab [2].

**Design**

The study was designed as a repeated measures experiment. Each participant performed five blocks with each block covering one of the five trajectory types. In every block, participants performed 4 selections × 3 text representations = 12 text selections using Pursuits. The order was counter-balanced across participants using a Latin-square.

The theme of the study was a voting application, where participants had to answer questions by selecting one of three possible floating text-based answers via Pursuits (see examples in Figure 2). In total, every participant answered 60 questions: 5 trajectory types × 3 text representations × 4 selections.

**Apparatus**

We deployed the EyeVote system on a 42-inch display (3810 × 2160 pixels) in our lab (see Figure 1A). The display was equipped with a Tobii EyeX Controller (30Hz). Participants stood at a distance of roughly 60 cm from the eye tracker.

**Participants**

We recruited 19 participants (10 females) between 20 and 60 years through mailing lists and social networks. Four of them had previous experience with eye trackers but with Pursuits. All participants had normal or corrected-to-normal vision.
Procedure

The experimenter began by explaining the study and asking the participant to sign a consent form. The experimenter then started the first block of 12 questions. After each successful Pursuit selection and before showing the following question, the system showed the user which answer was recognized. At that point the participant was asked to confirm whether or not the system detected the intended answer. In case of false detection, the participant was further asked to specify whether (a) the system detected a selection prematurely (i.e. the participant was still reading it) or (b) the participant was trying to select a different answer. To assess the perceived workload associated with text-based selection of every trajectory type, participants filled in a Nasa TLX questionnaire after each block.

Results

We logged the time taken to answer each question as well as the false detections by the system. In total, we recorded 1140 selections (19 participants × 60 selections).

We classify errors as (a) early detection errors, that are errors due to a premature recognition of an option while the participant is still reading, and (b) false detection errors, that are errors due to the system recognizing a selection other than the one the participant intended. Out of 1140 selections, there were 124 errors (∼10.9%): 88 of them were early detections (∼7.7%), while 36 were false detections (∼3.2%).

<table>
<thead>
<tr>
<th>Significantly different pair</th>
<th>p-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Circular (10.2%) Linear (36.4%)</td>
<td>(p &lt; 0.005)</td>
</tr>
<tr>
<td>Circular (10.2%) Zigzag (35.2%)</td>
<td>(p &lt; 0.05)</td>
</tr>
<tr>
<td>Rectangular (8.0%) Linear (36.4%)</td>
<td>(p &lt; 0.001)</td>
</tr>
<tr>
<td>Rectangular (8.0%) Zigzag (35.2%)</td>
<td>(p &lt; 0.005)</td>
</tr>
<tr>
<td>Mixed (10.2%) Linear (36.4%)</td>
<td>(p &lt; 0.01)</td>
</tr>
<tr>
<td>Mixed (10.2%) Zigzag (35.2%)</td>
<td>(p &lt; 0.05)</td>
</tr>
</tbody>
</table>

Table 1. We classify errors as (a) early detection errors, that are errors due to a premature recognition of an option while the participant is still reading, and (b) false detection errors, that are errors due to the system recognizing a selection other than the one the participant intended. Out of 1140 selections, there were 124 errors (∼10.9%): 88 of them were early detections (∼7.7%), while the remaining 36 were false detections (∼3.2%). Early Detection Errors

A repeated measures ANOVA showed significant effects for trajectory type on early detection errors $F_{4,72} = 15.353$, $p < 0.001$. Table 1 summarizes the results of post-hoc anal-
uses using Bonferroni correction, which revealed significant differences between multiple pairs. Circular, rectangular and mixed trajectories result in significantly fewer early detection errors compared to linear and zigzag trajectories.

No significant effects were found for text representation on early detection errors. However the results suggest that fewer early detections occurred in the case of short text (Figure 5).

**False Detection Errors**
No significant effects for trajectory type on false detection errors were found ($p > 0.05$). Although a repeated measures ANOVA showed significant effects for text representation on false detection errors $F(3,36) = 3.916$, $p < 0.05$, no significant differences were found between pairs. This is likely due to the low number of false detection errors (36 out of 1140 selections). Figures 3 and 5 indicate a tendency for fewer false detections in the case of Circular trajectories and short text.

**Perceived Workload**
Figure 4 summarises the mean values for each subscale. A repeated measures ANOVA showed significant effects for trajectory type on physical demand $F(3,84) = 4.631$, $p < 0.005$ and effort $F(3,84) = 4.334$, $p < 0.005$. Post-hoc analyses using Bonferroni correction showed that there are significant differences between physical demand ($p < 0.05$) induced by circular ($M = 9.4, SD = 5.1$) and mixed trajectories ($M = 5.7, SD = 3.8$). Significant differences were also found between effort ($p < 0.05$) to select circular ($M = 8.9, SD = 5.5$) and mixed trajectories ($M = 5.3, SD = 3.7$).

In summary, although circular trajectories were perceived to perform similar to mixed trajectories, participants perceived mixed trajectories as less demanding in all other aspects.

**Summary**
The previous study showed that it is feasible to use Pursuits for text selection, with the selectable text itself being the moving stimulus. Our results confirm that by using certain text representations and trajectory types, text can be used as a stimulus for one of the major uses of Pursuits, that is interaction. Next we investigate the second major usage of Pursuits, which is eye tracker calibration, with text used as stimulus.

**PURSUIT CALIBRATION USING TEXTUAL STIMULI**
One of the major strengths of using text for Pursuit calibration is that it allows for seamlessly calibrating an eye tracker on a public display as passersby simply read text. Previous work by Pfueffer et al. [36] on Pursuit calibration also included a text label “Please wait”, floating from one corner of the display to another for 13 seconds. However, this required users to fixate the floating word for a long period even after reading, which might not be intuitive without prior instructions. To address this, we reveal text gradually to intrigue users to pursue the revealing text till its end (see Figure 1B).

In this section we present the implementation and evaluation of our prototype system called Read2Calibrate.

**Concept and Implementation**
In our implementation of Read2Calibrate, we developed a method that reveals the text gradually at an inclined angle. As new parts of the text gradually appear, the preceding parts disappear at the same rate (see Figure 1B). We opted for this representation in order to (1) ensure that users follow the stimulus (i.e. the text) till the end of the calibration session, (2) control the user’s reading speed, and (3) calibrate for as much area of the display as possible. In the following, we explain the rationale behind these three motives.
Revealing Speed
To correlate eye movements to movements of a stimulus, the speed of the moving stimulus needs to be known beforehand. Reading speeds are different across users [26], which makes it difficult to predict which part of the text a user is looking at, in particular in a public display setting with no prior knowledge about the user\(^1\). However, revealing the text gradually ensures a controlled reading speed, that is, if the user is reading any part of the text, we can expect that the user is looking at the letters being revealed at that moment. To ensure the user is performing a smooth pursuit eye movement rather than a series of saccadic jumps, new parts of the text are gradually revealed. As newer parts of the text appear, preceding parts disappear gradually, to reduce the chances of backward saccadic jumps (see Figure 1B).

The speed of revealing text is an important variable. If the revealing speed is too fast, users might not have enough time to comprehend the text. Additionally, because eye trackers are typically limited to a certain sampling frequency, the faster text is revealed, the less mappings between eye movements and points on the display are collected. On the other hand, very slow revealing speed could also result in difficulty in understanding the text; as the time difference between revealing the first and last letters of a word is larger, the more difficult it becomes to understand the word. There is also a risk of the users losing interest and looking at other parts on the screen, which would negatively influence the calibration.

Based on prior work [36] and pilot tests, we introduced pauses of 500 ms, 350 ms and 200 ms in-between revealing each letter, which in visual angles equates to speeds of 1°/s, 2.1°/s, and 4°/s. Higher and lower revealing speeds were found to be very difficult to read. We refer to these speeds as the slow, medium and fast speed, respectively.

Inclination Angle
Read2Calibrate needs to collect mappings between gaze points and points on the display. To cover as large an area of the display as possible, previous work used a stimulus that moved across the display in diagonal, circular, or rectangular trajectories [36, 37]. Circular and rectangular trajectories are unnatural for gradually revealing text, while limiting stimuli to a horizontal line would calibrate with respect to the x-axis only. Hence, we chose to reveal the text in diagonal shapes.

However, since there has been no previous work about reading inclined text that is gradually appearing, we experimented with multiple inclination angles. Latin script is read from left to right, hence the text could be shown in two ways: starting from the upper-left part and ending in the lower-right part of the screen, or starting from the lower-left part and ending in the upper-right part of the screen. This translates to inclination angles between 270° – 360° and between 0° – 90 degrees\(^\circ\).

Taking into consideration the need to move the stimulus with respect to both axes, we experimented with six angles: 15°, 45°, 75°, 285°, 315°, and 345°. Figures 6A and 6B show sample text displayed at inclination angles of 15° and 315°.

Calibration Correction
A prerequisite for calibration is to gaze at the stimulus. To exclude cases where users are not looking at the stimulus, our system calibrates after a certain correlation has been reached. We used the Pearson’s product-moment coefficient with a correlation threshold of 0.6, that is, the user is assumed to follow the stimulus if the correlation between its movement and the users’ eye-movements is ≥ 60%. We selected this value based on pilot testing and experience from prior work [36, 47].

In a calibration session, the letters are placed on the screen according to their angle. As the letters start to appear, pairs of gaze points and the revealed letter’s coordinates are collected. To calculate the correction offset, for every gaze point \(G\) recorded by the eye tracker, we measured the Euclidean distance between \(G\) and the center of the currently revealed letter \(L\). After the calibration phase ends, the sum of these distances is divided by the total number of gaze points \(N\) detected in that time frame. The resulting average distance value is then used as the correction offset (see Equation 1).

\[
Offset = \frac{\sum_{k=1}^{N} L_k - G_k}{N} \tag{1}
\]
Evaluation of Pursuit Calibration Using Text

The goal of this study was to evaluate the effectiveness of Read2Calibrate, as well as to understand the influence of the different variables (inclination angle of the text and revealing speed). We studied the influence of these variables on the calibration quality, in addition to how users perceive them.

Apparatus
A 24 inch display (1920×1080 pixels) was equipped with a Tobii REX eye tracker (30Hz) and deployed in our lab (see Figure 1B). We invited 18 participants (10 females) aged 18 – 42 years (M = 26.2, SD = 5.3) through mailing lists. All participants had normal or corrected to normal vision.

Design
In a repeated measures experiment, every participant performed one calibration session per condition (6 angles × 3 speeds = 18 conditions). Each calibration session was followed by a testing session, where the participant was asked to gaze at a stationary point that appeared at nine positions on the screen (see Figure 7). The point blinked at each of the nine positions for 3 seconds. The order of the conditions was counter balanced across participants using a Latin-square.

Procedure
The experimenter started by explaining the study and the participant filled in a consent-form. The participant was then asked to stand at a distance of 60 cm from the eye tracker. The font-size was set to be easily readable at that distance. In each calibration session, the participant read a piece of a story that was gradually revealing across the display according to the condition’s angle and speed. The participant then proceeded to a testing session, where we logged the gaze points, as detected by the eye tracker, the coordinates of the revealed text, as well as the corrections by our algorithm. After gazing at all nine stimuli, the calibration was reset and the participant proceeded to the next calibration session.

To reduce possible eye fatigue, participants were optionally allowed to take a break after every 3 calibration sessions. No visual feedback was shown during the whole experiment to avoid influencing the participant’s gaze behavior.

We used a fable as a source of the revealing text. To ensure the validity of our analysis, it was crucial to make sure participants paid attention to the text. Hence, we asked participants three questions about the fable at the end. In addition to the compensation for participation, participants were encouraged to pay attention to the story by promising them an additional monetary incentive (50 Euro cents) for each correct answer they provide to the questions. All participants were aware of the rewarding mechanism before taking part in the study.

We concluded the study with a questionnaire and a semi-structured interview.

Results
To evaluate the effectiveness of Read2Calibrate, we based our calculations on

1. the target point (the center of the stimulus which was gazed at during the testing session),
2. the uncalibrated gaze point (the gaze point as detected by the uncalibrated eye tracker),
3. the calibrated gaze point (the gaze point after correction by Read2Calibrate).

For each stimuli shown in the testing sessions, we measured the mean Euclidean distance (1) between the uncalibrated gaze points and the target point and (2) between the calibrated points and the target point. Moreover, we measured the positive correction rate, which we define as the number of times the calibrated gaze point was closer to the target compared to the uncalibrated one.

Quantitative Results
Figure 8 shows that the mean Euclidean distance is shorter when text is revealed at angles of 315° and 45°. Thus, these angles result in better correction compared to others.

A repeated measures ANOVA showed significant main effects for angle $F_{3,251} = 5.2$, $p < 0.005$ and speed $F_{2,34} = 4.8$, $p < 0.05$ on positive correction rate. Post-hoc analysis using Bonferroni correction showed a significant difference ($p < 0.05$) in positive correction rate for an inclination angle of 315° ($M = 65\%, SD = 0.06\%$) compared to 15° ($M = 39.7\%$,
Figure 9. Revealing the text in a 315° inclination resulted in the highest number of positive corrections, e.g. 73% of the corrections by the Read-2-Calibrate brought the gaze point closer to the target.

SD = 0.05%), and also for 315° (M = 65%, SD = 0.06%) compared to 345° (M = 41.4%, SD = 0.06%). There were also significant differences in positive correction rate for fast revealing speed (M = 40.7%, SD = 0.05%) compared to slow revealing speed (M = 53.2%, SD = 0.045%). Figure 9 shows that angles of 315° and 45° resulted in more positive corrections compared to other angles. The figure also shows that fast revealing speeds result in less positive corrections.

**Qualitative Feedback**

When asked how easy it is to read the text at the different angles (5-point Likert scale; 1=Strongly disagree; 5=Strongly agree), participants indicated that most angles were easy to read (see Table 2). As for the revealing speeds, participants found the medium speed (Med = 5, SD = 0.6) to be easier to follow compared to slow (Med = 4, SD = 1.3) and fast speeds (Med = 4, SD = 1.2).

When asked about their preference, participants pointed that they preferred angles that are closer to a horizontal line (i.e. 15° and 345°) as they felt more natural. However as indicated in the questionnaire, other angles are also easy to follow. On the other hand, multiple participants indicated that it felt unnatural to read the slow revealing text, P6 noted that “I felt I was following the letters without really understanding the words”. According to the participants, fast text is easy to follow, but difficult to comprehend.

**DISCUSSION & DESIGN RECOMMENDATIONS**

Overall, the results of both studies suggest that text can be used as a stimulus to support interaction and calibration using Pursuits.

The text selection study showed that when using circular and mixed trajectories, shorter text can be selected with high accuracy using Pursuits while selecting longer pieces of text is more difficult. The text-based Pursuit calibration study showed that text can be an effective stimulus for seamlessly integrating eye tracker calibration as users read text. More specifically, gradually revealing text inclined at 45° or 315° at a speed of 2.1 visual degree angles per second highly improves the accuracy of the gaze point. The results also indicated that although participants preferred flat text, revealing text at inclined angles is easily readable and can be used for calibration.

**Text Selection via Pursuits**

Overall, there was a low number of false detection errors – 36 false detection errors out of 1140 selections. Figure 3 shows that circular trajectories tend to be associated with less false detection errors. The results of the text selection study show that circular, rectangular, and mixed trajectories result in fewer early detection errors compared to linear and zigzag ones (see Figure 5 and Table 1). This means that reading text moving in linear and zigzag trajectories results in high correlation between eye movements and text movement, making the system confuse reading with selection.

Reading involves repeated saccades across a line of text as well as back to the beginning of the next line. Performing these saccades while pursuing text moving in a circular or rectangular trajectory distinguishes the eye movements from the text’s trajectory. This reading overlay makes the gaze trajectory less likely to correlate with that of the moving text, giving the user a chance to read and comprehend the text. On the other hand, reading text moving in linear and zigzag motion can be hardly distinguished from a Pursuit selection, resulting in a high correlation while reading, which in turn results in many early detection errors.

**Selecting Long Pieces of Text**

Our motivation behind the use of different text representations and trajectories was to study how the Pursuits method can cope with a read-and-pursue eye movement. Our main finding is that Pursuits is indeed suitable for text selection, but only with shorter pieces of text. In cases where it is essential to show passersby longer pieces of text to select from, we recommend using a different stimulus.

**R1: Use Pursuits for selection of short pieces of text; for longer pieces of text (25+ letters) use non-textual stimuli.**

In case longer textual descriptions are needed, a display could be divided into two regions: a non-interactive region and an interactive region. In the case of the voting application, the non-interactive region could display the detailed answer options, each with a short but meaningful headline. The interactive region could then display the moving headlines only in the interactive region. In the case of the voting application, the non-interactive region could display the detailed answer options, each with a short but meaningful headline. The interactive region could then display the moving headlines only in the interactive region.

Table 2. The table summarizes responses of participants when asked if it is easy to read text inclined at the corresponding angles (5-point Likert scale; 1=Strongly disagree; 5=Strongly agree). This indicates that the closer the text’s inclination to a horizontal line (i.e. 15° and 345°), the easier it is to read. However, the average scores of other angles as well as feedback from participants indicate that although other angles are less preferred, they are still readable.

<table>
<thead>
<tr>
<th>Text Inclination Angle</th>
<th>15°</th>
<th>45°</th>
<th>75°</th>
<th>285°</th>
<th>315°</th>
<th>345°</th>
</tr>
</thead>
<tbody>
<tr>
<td>Median Score</td>
<td>5</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4.5</td>
</tr>
<tr>
<td>Standard Deviation</td>
<td>0.43</td>
<td>0.67</td>
<td>0.92</td>
<td>0.96</td>
<td>0.83</td>
<td>0.70</td>
</tr>
</tbody>
</table>

Figure 10 TextPursuits: Using Text for Pursuits-Based Interaction and Calibration on Public Displays.
Choosing the Right Trajectory Type

Our analyses of the text selection experiment results show that circular trajectories are safer to use, as they result in significantly fewer errors. However, circular trajectories are perceived to be highly demanding (see Figure 4). Mixed trajectories result in slightly more errors than circular ones. However, mixed trajectories were perceived to be significantly less demanding compared to other trajectories. This indicates a trade-off between user preference and accuracy of the system.

R2: For peak performance when performing text selection using Pursuits, move text in circular trajectories. To increase user experience, a mixture of trajectories can be used together with an undo option.

We conclude that if high accuracy when selecting text using Pursuits is required, designers should opt for circular trajectories. An example could be a situation where users are encountered in a passing-by situation or in a situation where they have only too little time to undo their selections, such as asking a question on customer satisfaction near a cashier. On the other hand, in cases where user experience should be maximized, mixed trajectories may be used that are slightly more error prone. In these cases, a floating “undo” label could be shown after the system detected a selection. An example could be users filling in a longer survey in return for an incentive, such as a gift voucher. Here it may be acceptable to occasionally ‘correct’ an answer while at the same time having a less demanding experience with the system.

Text-based Pursuit Calibration

In general, participants’ feedback indicates that Read2Calibrate is positively perceived. The results of its evaluation show that inclination angles that result in diagonal-like orientation of the revealing text, such as 45° and 315°, significantly improve the accuracy of the gaze point. This is due to the fact that these angles result in the text covering larger areas of the screen. Inclination angles that bring the text closer to a horizontal line are preferred by users (15° and 345°), as they are more similar to flat text which users are acquainted to read. However, at these angles the text covers relatively less area with respect to the y-axis, resulting in poor calibration.

Slow revealing speeds result in users focusing on letters and losing track of the words they read. By analyzing the data, it was found that when using slower speeds participants were more likely to lose interest and look at other parts of the screen, presumably out of boredom. Fast speeds result in less data collected for the correlation check, which in turn results in lower calibration accuracy. Moreover, participants reported that revealing the text too fast makes it harder for them to understand what they read. Medium revealing speed turned out to be a good compromise: it is preferred by users and also results in a good calibration quality.

It should be noted, however, that the accuracy achieved by Read2Calibrate is lower than that of previous approaches that use Pursuits for calibration as well as of explicit calibration methods commonly known from eye tracking in desktop settings. At the same time, the major advantage of text-based Pursuits calibration is the seamless integration with users simply reading content on the public display. As a result, the calibration can be performed and used even in cases where the reader is not being consciously aware of it. Gaze information can then be used to enhance the user interface. For example, one may show a description of different sights next to a map of a city, like museums, churches, or historic buildings. As the system is calibrated, dwell time towards different sights could be used to determine what the reader is most interested in and additional information on how to reach a particular sight together with a discount coupon could be presented.

R3: For moderate eye tracker calibration (accuracy of 5.8° of visual angle), text-based Pursuit calibration is recommended as it results in better user experience. If accuracy is crucial, classical Pursuit calibration should be used.

The trade-off between accuracy and user experience can also be found when determining the angles at which the revealed text is inclined in Read2Calibrate. While bringing the text closer to a horizontal line makes reading feel more natural, revealing the text in a diagonal-like path results in the highest accuracy. Very steep text (e.g. 75° and 285°) result in both low accuracy and worse user experience and should hence be avoided.

R4: Use diagonally-shaped paths, at an inclination of 315° or 45°, when revealing text to achieve highest accuracy with Read2Calibrate. For better user experience at the expense of calibration accuracy, reveal text in flatter shaped paths.

A clear recommendation with regard to revealing speed can be provided. Here, accuracy is highest for medium revealing speed (2.1°/s) and this it in line with the users’ preference.

R5: For text-based Pursuit calibration, an average revealing speeds of about 2.1° of visual angles per second should be used.

The more comfortable a participant is with the revealing speed, the more accurate gaze-to-display mappings are collected and hence the more accurate the calibration is. Faster speeds result in fewer mappings, while slower ones distract the user.

Use Cases and Integration With Interactive Applications

As a sample application that can be explicitly controlled using gaze, we implemented EyeVote, a voting system. Civic discourse is a popular use case for public displays [15, 16, 35, 39], where passersby select from a set of text-based options to express their opinions. Given the advantages of gaze for public displays [22], gaze-based voting systems can utilize Pursuits for selection of textual answers. Similarly, Pursuits can be used to answer on-screen quizzes. Selection of text via Pursuits can be useful in various other contexts, for example, users can select from a set of text-based options displayed at a museum to learn more about particular topics. In train-stations and airports, Pursuits can be employed to set fare preferences or pick products where possible options are displayed as text.

The second major use case is the implicit use of gaze data, either for analysis or for adaptive interfaces. Therefore, text-based Pursuit calibration can be integrated into public display applications in several ways. For example, a common practice
to tackle interaction blindness on public displays is to use call-to-action labels [32]. Such labels could serve as stimuli to calibrate an eye-tracker via Read2Calibrate. Further stimuli could be welcome messages or brief instructions on how to use a system or how to play a game. While the aforementioned examples utilize Read2Calibrate at the beginning of the interaction process, revealing text can also be shown amidst interaction. For example, a short text could hint at hidden or yet undiscovered features. Such a calibration while interacting may be useful for displays featuring multiple applications. Here, a game that is instantly usable may serve as an entry point. As the user finishes playing the game in the course of which the eye-tracker was calibrated using in-game textual content, the display could present further content that could benefit from knowledge about the user’s gaze behavior. Note, that after the calibration, fine-grained gaze points can be collected and, hence, also other types of eye movements, such as fixations and saccades can be detected. As a result, an application may determine interest towards a particular content – this may be of particular interest for advertisers – as well as identify difficulties of users in perceiving content, for example, as they read text over and over again.

Limitations and Future Work
Firstly, our evaluations so far were conducted in the lab. While this controlled setting was necessary to maximize internal validity and obtain comparable results, future work could employ text-based stimuli for Pursuits in an-in-the-wild setting. Apart from verifying the results with regard to accuracy and errors, this may yield further insights on audience behavior and acceptance.

Secondly, participants of the text selection study answered 60 questions using Pursuits, whereas participants of the Read2Calibrate study performed 18 calibration sessions. In a real-world situation, it is unlikely that users would perform such a high number of selections and users would not be required to verify the accuracy of the calibration. As a result, we expect the study to have caused a higher level of eye fatigue as an in-the-wild exposure to any of the systems would have done. Hence, participants may have been overly critical during their assessment of the system. Future work could capture in-situ feedback to verify the impact of our approach on the experience users have during use of our system.

Recent work explored feedback methods for Pursuit selections. Kangas et al. [19] compared different feedback modalities for Pursuits and found that haptic feedback is preferred by users compared to visual and auditory feedback. Špakov et al. [51] compared two smooth pursuit widgets to find that circular widgets exhibit higher performance. An additional direction for future work is to enable feedback methods to improve the user experience when using EyeVote and Read2Calibrate. For example, in text-selection tasks, visual cues can be used to incrementally highlight the text whose trajectory correlates the most with eye movements, depending on the current correlation value.

Another interesting direction for future work would be to try different scripts. For example, Arabic and Hebrew are read from right to left, while Chinese, Japanese and Korean can also be read vertically. In our implementation of Read2Calibrate, text was revealed in a diagonal path. The flexibility of some east Asian scripts makes it possible to experiment with revealing text in different paths (e.g., a rectangular path).

CONCLUSION
In this work we investigated the use of text as a stimulus for Pursuits, to enable gaze-based interaction and eye tracker calibration on public displays. Our results show that text can be a powerful stimulus for both tasks. Shorter pieces of content can be robustly selected using Pursuits, and text-based calibration improves gaze point accuracy. We found that Pursuits-based text selection is less error-prone when text follows circular trajectories. We also found that the use of different trajectories simultaneously (mixed trajectories) is better perceived by users and results in relatively few errors. Read2Calibrate was shown to improve the accuracy of the gaze point, in particular when using text that is gradually revealing at a speed of 2.1\textdegree/s and inclined at a 315° or 45° angle.

REFERENCES


EyeVote in the Wild
Do Users bother Correcting System Errors on Public Displays?
EyeVote in the Wild: Do Users bother Correcting System Errors on Public Displays?

Mohamed Khamis¹, Ludwig Trotter¹, Markus Tessmann¹, Christina Dannhart¹, Andreas Bulling², Florian Alt¹

¹ LMU Munich, Germany
² Max Planck Institute for Informatics, Saarbrücken, Germany
{mohamed.khamis, florian.alt}@ifi.lmu.de, bulling@mpi-inf.mpg.de

ABSTRACT
Although recovering from errors is straightforward on most interfaces, public display systems pose very unique design challenges. Namely, public display users interact for very short amounts of times and are believed to abandon the display when interrupted or forced to deviate from the main task. To date, it is not well understood whether public display designers should enable users to correct errors (e.g. by asking users to confirm or giving them a chance correct their input), or aim for faster interaction and rely on other types of feedback to estimate errors. To close this gap, we conducted a field study where we investigated the users willingness to correct their input on public displays. We report on our findings from an in-the-wild deployment of a public gaze-based voting system where we intentionally evoked system errors to see if users correct them. We found that public display users are willing to correct system errors provided that the correction is fast and straightforward. We discuss how our findings influence the choice of interaction methods for public displays; interaction methods that are highly usable but suffer from low accuracy can still be effective if users can “undo” their interactions.

ACM Classification Keywords
H.5.2. Information Interfaces and Presentation: User Interfaces – Input Devices and Strategies

Author Keywords
Public Displays; Smooth Pursuit; Gaze Interaction; Voting

INTRODUCTION AND BACKGROUND
Designing interfaces for interactive public displays is often associated with unique challenges. Previous work has shown that public display users interact for short amounts of time [2, 15], and usually abandon displays if response time is slow [5] or interaction is interrupted [22].

At the same time, users of public displays often need to deal with errors. This includes situations where users might want to correct typos when providing feedback or undo a selection. This motivated us to study the willingness of users to correct their input on public displays. The question of whether to enable public display users to correct their input, or if they are not motivated to do so and hence designers should rely on other means for detecting errors, has not been addressed by prior research before.

Most relevant to our work is the work by Kukka et al. [13] who found that public display users are willing to dismiss error messages and continue interaction if the messages give users an active role (e.g. Press OK to continue). Our work builds over that by understanding if users bother correcting system errors rather than abandoning the display in frustration.

In this work we study how users behave when correcting errors and how the design of feedback mechanisms can assist in error correction. To do so, we deployed a gaze-based voting application on a public situated display (see Figure 1) in which we occasionally showed intentionally falsified confirmation messages to investigate whether users are willing to correct...
system errors, or if they will abandon the display instead. Using a voting app was motivated by recent research, showing that public displays are a promising medium for collecting votes and opinions [6, 7, 12], encouraging civic discourse [17, 19], and reflecting the population’s opinions [16]. Existing systems used touch [6], mid-air gestures [16, 19], feet [17] and recently also gaze [12] for voting in public.

We deemed gaze as interaction modality to be particularly interesting for our research [11], since it supports fast and natural interaction [23] and is hard to observe by others. The latter property is promising since making voters’ choices less obvious to surrounding users was shown to be a desired quality in public voting systems [19]. At the same time, gaze interaction is prone to input recognition errors, making it a suitable candidate for investigating users’ error correction behavior and the employment of suitable features for error correction.

To obtain a better understanding of how error correction can be implemented and how users react to it, we introduce an undo feature to our voting app that allows users to correct their input by means of different modalities, i.e., gaze and physical touch. We then deployed the system in the real world setting for two days. Findings show that users correct system errors given that the correction method is fast and straightforward.

This paper contributes an investigation of passersbys’ willingness to correct errors caused by faulty system detections.

VOTING SYSTEM

EyeVote is a voting system for gaze-enabled public displays. The system displays questions to the public, with multiple floating textual options to choose from (see Figure 2B).

Collecting accurate gaze points requires time-consuming calibration, which is unaffordable on public displays [9]. It was necessary to either use a calibration-free technique (e.g. [21, 24, 18]) or blend the eye tracker calibration into interaction (e.g. [12]). EyeVote uses Pursuits, a calibration-free gaze-interaction technique that relies on correlating the user’s eye movements with movements of dynamic stimuli on the screen. The moving stimulus that has the highest correlation to the eye movements, is deemed to be the eye movements, is deemed to be what the user is looking at.

Previous work experimented with different types of stimuli for Pursuits, such as images [21], icons [4] abstract objects [20], game elements [8], letters [14] and digits [1]. Only recently, it was found that users are able to pursue moving text while reading it [12], which means that text strings can also be used as stimuli for the Pursuits approach.

Pursuits Implementation

The system uses the Pearson’s product-moment coefficient to calculate the correlation between the eyes movements and the movements of the floating options. Based on pilot experiments and previous work [4, 8, 12, 21], we used the same correlation function with a threshold of 0.7 and a window size of 2000 ms. This means that every 2 seconds, the system computes Pearson’s correlation. The floating answer whose movement correlates the most with the eye movement, is deemed to be the...
System Walkthrough and Undo Feature

By default, a call-to-action label invites passersby to "Look at the screen to start" (Figure 2A). Once gaze is detected, the interface shows the first question (Figure 2B). After the system correlates the users choice, it proceeds to a recap view, where the detected answer is shown to the user. At this point the user is presented which the choice of changing the detected answer, or proceeding to the next question. On the first day we showed the button-condition (Figure 2C), while the gaze-condition was shown on the second day (Figure 2D). Depending on the user’s choice, the system either proceeds to the following question or repeats the last question. After completing 8 questions, the system resets to the default view. If the system loses track of the user’s eyes it shows a warning message indicating that eyes are not detected. If eyes are not detected for 8 continuous seconds, the system restarts. The system showed straight forward questions about the favorite band, the user’s study program, etc.

Falsified answers

Due to the nature of field studies, it was not feasible to have an experimenter ask every participant whether or not the system detected their vote correctly. Hence, to investigate whether users undo system mistakes, we intentionally introduced falsified answers. This was done by showing an answer in the recap view that was not among the options the user had available to choose from. This way we are confident that the system is showing the user a wrong answer, and that the expected behavior is to undo the choice. In every set of 8 questions, fake answers were always shown the first time questions 3 and 6 are answered. For example, the first time a user answers question 3, the system shows “Asics” in the recap view even though it was not among the options. If that user decides to “Change” the answer, the system shows the question again, but this time the system shows the answer that was really detected.

FIELD STUDY

We deployed a 27 inch display (1920×1200 pixels) in large university hall that is expected to be reached by many students, academics, and university staff members. The display was equipped with a Tobii EyeX Controller. A squared marker was placed on the floor with a distance of one meter to the display to guide passersby to standing in the eye tracker’s range.

Design

The study ran for two days and covered two conditions: (1) button-based undo, and (2) gaze-based undo. On the first day, we deployed the button-based undo approach. A red button was placed next to the display, participants were asked to press the button once to proceed to the next question, or twice to repeat the question (see Figures 1 and 2C).

On the second day, we deployed the gaze-based undo approach.

Results

We logged the raw gaze data, all presented and selected answers, as well as button and gaze based undos. The system launched the first question 187 times in total during two days. This means that there were 187 instances where the system detected a user standing within the marked area and facing the screen. We refer to these instances as “interactions”.

On Day 1 there were 106 interactions, that is, gaze was detected at 106 different instances. Out of which, there were 49 instances where at least one question was answered. In total, 220 questions were answered on the first day. On Day 2, at least one question was answered in 30 out of the 81 times in which a user interaction was detected. In total, 243 questions were answered on the second day.

Undos

We distinguish two cases where the “undo” feature was used by users: (1) cases where users corrected falsified answers, and (2) cases where users corrected unaltered answers. The former are cases where we are confident that the system showed a wrong answer that is was not among the choices, while the latter are cases where the system might or might not have shown a wrong answer (see Figure 3).

Out of the 220 questions answered on Day 1, 37 were falsified answers. Out of those, 22 answers (59.5%) were changed using the undo feature. While on Day 2, there were 31 falsified answers out of 243 answered questions. Users corrected 27 out of 31 (87%) falsified answers.

Note that users did not have any motivator to correct their input apart from their intrinsic motivation. Therefore, we interpret the correction rates of 59.5% and 87% as indicators that the users are willing to correct input mistakes.

The undo rate for the gaze method is higher compared to the button method. We attribute this to the use of the same interaction method for selection and correction which is seemingly better accepted by the users.

User dropouts

As this field study was not supervised, users could join and leave at any time. We relied on the number of dropouts as a measure of satisfaction and how users cope with the system. We define a dropout as a situation where the system lost eye contact for more than 8 continuous seconds.

Figure 4 illustrates the number of users who answered each question, as well as the number of users who dropped out after answering each question. It is noticeable that 20 out of 29 users (41%) dropped out after answering the first question in
the first day in which the button-based undo was deployed. While on the second day, relatively few users dropped out after the first question, but relatively more dropped out after questions 3 and 6, in which the falsified answers were shown.

Interestingly, the number of users who answered all eight questions is very similar on both days: 20.4% on Day 1, and 20% on Day 2.

DISCUSSION

To our surprise, many users (59.5% and 87% respectively) corrected their input when we displayed falsified answers. Although prior work reports that people are likely to abandon the display if they find it unresponsive or faulty [2]. Our results indicate that passersby are willing to fix system mistakes as long as the option is available and feasible.

By examining Figure 4, we expect that the dropout rate for the first question on the first day is due to the use of two separate modalities for selection and confirmation; we used gaze for voting, and the button for confirming the answer. On the other hand, although there are less dropouts at the first question on the second day, there are more dropouts at the falsified answers. By the time users reached the falsified questions (questions 3 and 6) they had already answered and confirmed many questions via gaze. Hence we expect that the temporal demand led to fatigue, which discouraged some users from completing the 8 questions. Previous work had reported that performing multiple consecutive gaze-based selections is tiring to the users [12].

The results suggest that users are indeed willing to correct system errors. Figure 4 shows that although relatively more users dropout at system errors, the majority of users corrects them and continues interacting. With this conclusion we encourage the use of highly usable interaction techniques even if this leads to sacrificing some system accuracy, and relying on users to correct the occasional system errors. Taking EyeVote as an example, future systems can offer a dynamic undo function, which can be realized by introducing an additional threshold; if the highest correlation between the eye movements and one of the trajectories is higher than the selection threshold (set to 70% in our implementation), a more conservative threshold is checked (e.g. 90%) and if the correlation is lower than that (i.e. between 70% and 90%) the user is presented with a confirmation message, otherwise the user proceeds to the following question.

Although the use of a button to confirm or revoke an interaction is intuitive, the results suggest that users were more likely to dropout in the case of the button-based undo compared to the gaze-based approach. We believe that the use of the same modality for interaction and confirmation maintains a straightforward flow and is less confusing to the passersby. Hence we recommend future systems to use the same modality for both interaction and confirmation.

While we investigated the willingness to correct input on a gaze-enabled display, the results are also applicable to other modalities. For example, systems can reduce dwell times when selecting via mid-air hand gestures; this increases the responsiveness of the system but makes it more error prone, hence the system should also allow users to undo their actions.

CONCLUSION AND FUTURE WORK

Our results indicate that users will tolerate input errors and correct them if the system allows that. Therefore, we encourage the use of highly usable metrics even if they reduce accuracy provided that correction mechanisms are implemented. In our study, users corrected most of the falsified inputs (59.5% for the button-based undo and 87% for the gaze-based undo). Although the dropout rate was higher when falsified answers were shown during the gaze-based undo condition, the majority of the users corrected them before proceeding to the following question.

In this work we evaluated an undo feature for public displays using Pursuits. One direction for future work is to experiment with other gaze-input methods such as gaze gestures [3, 10]. Additionally, more modalities can be experimented with and compared such as mid-air hand gestures, touch and head gestures (e.g. nodding to confirm interactions).
REFERENCES


EyeVote in the Wild: Do Users bother Correcting System Errors on Public Displays?


[76] Nemanja Memarovic, Marc Langheinrich, and Florian Alt. 2012. Interacting places – A framework for promoting community interaction and place awareness through public displays. In 2012 IEEE International Conference on Pervasive Computing and Communications Workshops. 327–430. DOI:http://dx.doi.org/10.1109/PerComW.2012.6197526


[80] Omar Mubin, Tatiana Lashina, and Evert van Loenen. 2009. How Not to Become a Buffoon in Front of a Shop Window: A Solution Allowing Nat-


hagen, Denmark. ACM, New York, NY, USA, Article 44, 6 pages. DOI:http://dx.doi.org/10.1145/2611009.2611015


Wesson, and Marco Winckler (Eds.). Lecture Notes in Computer Science, Vol. 8118. Springer Berlin Heidelberg, 170–186. DOI:http://dx.doi.org/10.1007/978-3-642-40480-1_11


CURRICULUM VITAE
Mohamed Khamis CV

Address:  
Date of Birth: September 4th, 1989  
Nationality: Egyptian

🔎 Mobile (DE):  
✉️ Email: me@mkhamis.com  
Website: http://www.mkhamis.com/

Education
08/2018  PhD in Media informatics, Ludwig Maximilian University of Munich (LMU).  
( expected)  Dissertation title: Designing Gaze-based Interaction for Public Displays  
First Supervisor: Florian Alt  
Second Supervisor: Andreas Bulling  
External Examiner: Marc Langheinrich
02/2012  M.Sc. in Computer Science & Engineering, German University in Cairo.  
07/2011  B.Sc. with Honors in Media Engineering & Technology, German University in Cairo.

Personal profile

I am a final year PhD student at Ludwig Maximilian University of Munich (LMU) expected to graduate in August 2018. I have worked at four academic institutions throughout my career, in which I participated in teaching and research, as well as administrative tasks (e.g., ERASMUS coordinator).

I have significant research contributions in Human-Computer Interaction (HCI). I published 38 peer reviewed papers (24 since 2017), 5 of which at CHI—the highest ranked conference in HCI. I received 3 Honorable Mention awards (top 5% of submissions) at CHI’18, CHI’17 and MUM’17. I also published at other top HCI venues such as UIST, UbiComp, IMWUT, ICMI, and ETRA. Many of these accomplishments resulted from collaborations with scholars in Germany, UK, USA, Japan, Finland, Italy, and Egypt.

I worked in a large diversity of topics within HCI, privacy and ubiquitous computing. In particular, I am active in the fields of eye tracking, pervasive displays, usable security, and user privacy. I am mainly known for my major contributions that are at the crossroads of user privacy and ubiquitous computing:

- **Designing Gaze-based Systems:** I contributed to comprehensively understanding and addressing challenges of gaze interaction on ubiquitous devices, such as mobile devices (CHI’18) and public displays (UIST’17, UbiComp’16, MUM’16), as well as proposing novel concepts that employ gaze to address problems on public displays (CHI’18, IMWUT, PerDis’17), and mobile devices (ICMI’17).
- **Usable Security and Privacy:** I contributed to both: (1) understanding threats to user privacy that are caused/facilitated by ubiquitous technologies, such as thermal attacks (CHI’17, Honorable Mention), and shoulder surfing (CHI’17), and (2) inventing novel ubiquitous systems for protecting user privacy on mobile devices (ICMI’17), public displays (PerDis’17), and in VR (USEC’17).

I am actively involved in the relevant research communities. I am an **associate chair** for CHI LBW’18, **demo chair** for MUM’18, member of the **program committees** of MUM and PerDis, and **organizer** of PETMEI’16 and ArabHCI’18. I reviewed 50+ papers for esteemed conferences, and received an **Excellent Review** recognition at CHI’17. I also reviewed for **top journals** like ACM Computing Surveys (CSUR), IMWUT, and **ACM Transactions on Privacy and Security** (TOPS).

I taught introductory and advanced courses in HCI, computer science, and security. I significantly contributed to the development of courses such as “**Introduction to Usable Privacy and Security**” at the LMU. I supervised 25+ **bachelor and master theses**. I also helped several students kick off their PhDs.
Employment

Since 10/2014  Research Associate at Ludwig Maximilian University of Munich (LMU) in Munich, Germany

Duties include (1) research in HCI, particularly in gaze-based interaction and pervasive displays, (2) teaching courses, supervising seminars, and supervising thesis projects, (3) Erasmus coordination, and (4) internal organization tasks such as internal doctoral colloquia and open lab days.

04/2014 - 10/2014  PhD Student at the German Research Center for Artificial Intelligence (DFKI) in Kaiserslautern, Germany

03/2012 - 07/2013  Teaching and Research Assistant at the German University in Cairo, Egypt

09/2011 - 02/2012  Student research assistant at the German Research Center for Artificial Intelligence (DFKI) in Kaiserslautern, Germany

07/2011 - 08/2011  Interaction Designer at G-Beehive in Cairo, Egypt

07/2010 - 08/2010  Software development intern at Etisalat in Cairo, Egypt

03/2010 - 05/2010  Student research assistant at University of Würzburg in Würzburg, Germany

08/2009 - 09/2009  Software development intern at Etisalat in Cairo, Egypt

09/2007 - 06/2011  Student teaching assistant at the German University in Cairo, Egypt

Publications List

I published a total of 38 peer reviewed papers. A full list of publications as well as recordings of my talks can be found on my personal website http://www.mkhamis.com/publications.php

Journal papers (3)


Conference papers (20)

2018


2017


2016


2015


Other Peer-reviewed papers (15)


Relevant Honors & Awards

- Honorable mention best paper award (top 5% submissions) at CHI 2018, February 2018
- Honorable mention best paper award (top 5% submissions) at MUM 2017, November 2017
- Honorable mention best paper award (top 5% submissions) at CHI 2017, May 2017
- Excellent Review recognition at CHI 2017 and CHI 2018
- Google IoT Technology Research Award, March 2016
Curriculum Vitae

Mohamed Khamis CV

LMU travel grant for active participation in scientific conferences  
July 2015

GERLS scholarship for PhD studies offered by the DAAD and the MHESR  
April 2013

Excellent with Honors Bachelor of Science  
July 2011

Outstanding Performance in Software Engineering Academic Award  
June 2009

Software Engineering Cup Academic Award  
June 2009

5-year Study Scholarship at the German University in Cairo  
September 2006

Invited Talks


01/2016 Guest Lecture at the Ubiquitous Interactive Systems workshop – University of Stuttgart, Germany.

12/2015 Invited Talk at the IoT research colloquium – German University in Cairo, Egypt.

Professional Services

Erasmus program coordinator at Ludwig Maximilian University of Munich (LMU).

Organizing Committee member:

Proceedings Chair: PerDis 2017
Workshop Organizer: PETMEI 2016
Demos chair: MUM 2018

Program Committee member:

CHI LBW 2018
PerDis 2017, 2018
MUM 2016, 2017
PETMEI 2016
ET4S 2018
Mensch und Computer 2015, 2017
AutoUI 2015

Journal Reviewing:

IMWUT: 2017
ACM Transactions on Privacy and Security (TOPS): 2017
International Journal of Geographical Information: 2017
ACM Computing Surveys: 2016

Conference Reviewing:
CHI (Papers): 2016, 2017, 2018
CHI (Late Breaking Work): 2015, 2016, 2017
CHI (alt.chi): 2016, 2017
UIST (Papers): 2017
HRI (Papers): 2017, 2018
ACM MM (Papers): 2017
CHI PLAY (Papers): 2017
ETRA (Papers): 2018
ICMI (Papers): 2017
DIS (Papers): 2016, 2017
MobileHCI (Papers): 2016, 2017
UbiComp (Papers): 2016
ISWC (Papers): 2016
NordiCHI (Papers): 2016

Teaching Experience and Departmental Talks

Supervised 25+ bachelor and master theses.

Guest lecture at “Introduction to Usable Security” course at the LMU about Research Methods in HCI.

Guest lecture at the “Human Computer Interaction 2” course at the LMU about Eye Tracking.

Guest lecture at the “Open Games Workshop” at the LMU about Eye Tracking.

Taught (in English) introductory and advanced courses in HCI, cryptography, and computer science at Ludwig Maximilian University of Munich (LMU), and the German University in Cairo (GUC), . These included:

1. Usable Privacy and Security (WS17/18, SS18) ~30 students.
4. Advanced Topics in HCI (SS15, SS16, SS17, SS18) ~30 students.
5. Advanced Seminar in Media Informatics (WS15/16, SS16—WS17/18, SS18) ~20 students.
6. Web Technologies and Usability (SS13, SS12) 2 groups × ~25 students
7. Computer and Network Security (SS13) 5 groups × ~25 students
8. Introduction to Media Engineering (WS12/13) 3 groups × ~25 students
9. Introduction to Networks (SS12) 3 groups × ~25 students
10. Information Security (WS12/13) 2 groups × ~25 students

Significantly contributed to the development of several courses at the LMU and the GUC, including “An introduction to Usable Security”, “Computer and Network Security”, and “Information Security”.

I am interested in teaching HCI, computer science, and security courses, as well as introducing new modules (e.g., Usable Security) targeted at interdisciplinary students.
Languages

Arabic Native proficiency
English Full professional proficiency
German Intermediate working proficiency (Level C1)
French Elementary proficiency
Referees

Florian Alt
Professor and head of the Ubiquitous Interactive Systems Group at Ludwig Maximilian University of Munich (LMU).

Email: florian.alt@ifi.lmu.de
Address: Universität München, Frauenlobstr. 7a, 80337, Munich, Germany.

Andreas Bulling
Head of the Perceptual User Interfaces Group at the Max Planck Institute for Informatics and the Cluster of Excellene on Multimodal Computing and Interaction at Saarland University, Germany.

Email: bulling@mpi-inf.mpg.de
Address: Max Planck Institute for Informatics, Campus E1 4, 66123, Saarbrücken, Germany.

Albrecht Schmidt
Professor and head of the Human Centered Ubiquitous Media Group at Ludwig Maximilian University of Munich.

Email: albrecht.schmidt@ifi.lmu.de
Address: Universität München, Frauenlobstr. 7a, 80337, Munich, Germany.
Eidesstattliche Versicherung
(Siehe Promotionsordnung vom 12.07.11, § 8, Abs. 2 Pkt. 5)
Hiermit erkläre ich an Eidesstatt, dass die Dissertation von mir selbständig und ohne unerlaubte Beihilfe angefertigt wurde.
14th of May, 2018

Mohamed Khamis