



Title	Improving Optical-See-Through Experience through Corneal Imaging
Author(s)	Plopski, Alexander
Citation	大阪大学, 2016, 博士論文
Version Type	VoR
URL	<a href="https://doi.org/10.18910/55861">https://doi.org/10.18910/55861</a>
rights	
Note	

*The University of Osaka Institutional Knowledge Archive : OUKA*

<https://ir.library.osaka-u.ac.jp/>

The University of Osaka

# Improving Optical-See-Through Experience through Corneal Imaging

January 2016

Alexander PLOPSKI



# Improving Optical-See-Through Experience through Corneal Imaging

Submitted to  
Graduate School of Information Science and Technology  
Osaka University

January 2016

Alexander PLOPSKI

**Thesis Committee:**

Prof. Haruo Takemura (Osaka University)  
Prof. Koji Nakamae (Osaka University)  
Assoc. Prof. Kiyoshi Kiyokawa (Osaka University)  
Assist. Prof. Christian Nitschke (Kyoto University)

# List of Publications

## Journals

1. A. Plopski, Y. Itoh, C. Nitschke, K. Kiyokawa, G. Klinker, and H. Takemura. Corneal Imaging Calibration for Optical See-Through Head-Mounted Displays. *TVCG (Proceedings IEEE VR)*, 21(4):481–490, 2015. (Chapter 4)
2. T. Mashita, H. Yasuhara, A. Plopski, K. Kiyokawa, and H. Takemura. Parallel Lighting and Reflectance Estimation based on Inverse Rendering, *Journal C of the Institute of Electrical Engineers of Japan*, 134(10):1473–1480, 2014. (in Japanese)

## International Conferences

### Peer-reviewed

1. A. Plopski, C. Nitschke, K. Kiyokawa, D. Schmalstieg, and H. Takemura. Hybrid Eye-Pose Estimation Through Corneal Imaging, *Proceedings ICAT-EGVE*, 183–190, Oct. 2015. (Chapter 5)
2. A. Plopski, K. Kiyokawa, H. Takemura, and C. Nitschke. Corneal Imaging in Localization and HMD interaction, *Proceedings ISMAR*, Doctoral Consortium, 397–400, Sep. 2014. (Chapter 4)
3. D. Kurz, P. G. Meier, A. Plopski, and G. Klinker. Absolute Spatial Context-aware Visual Feature Descriptors for Outdoor Handheld Camera Localization – Overcoming Visual Repetitiveness in Urban Environments, *Proceedings VISAPP*, 56–67, Jan. 2014.
4. T. Mashita, H. Yasuhara, A. Plopski, K. Kiyokawa, and H. Takemura. Parallel Lighting and Reflectance Estimation based on Inverse Rendering, *Proceedings ICAT*, 1–8, Dec. 2013.

### Non-peer-reviewed

1. A. Plopski, T. Mashita, K. Kiyokawa, and H. Takemura. Reflectance and Lightsource Estimation for Indoor AR Applications, *Proceedings KJMR*, 1–1, Apr. 2014.

## Peer-reviewed Posters

1. A. Plopski, K. Moser, K. Kiyokawa, E. J. Swan II, and H. Takemura. Spatial Consistency Perception in Optical and Video See-Through Head-Mounted Augmentations, *Proceedings IEEE VR*, 1–2, Mar. 2016. (Chapter 7)
2. A. Plopski, T. Mashita, K. Kiyokawa, and H. Takemura. Reflectance and Light Source Estimation for Indoor AR Applications, *Proceedings IEEE VR*, 103–104, Mar. 2014.
3. T. Mashita, H. Yasuhara, A. Plopski, K. Kiyokawa and H. Takemura. In-situ Lighting and Reflectance Estimation for Indoor AR systems, *Proceedings ISMAR*, 275–276, Oct. 2013.
4. D. Kurz, P. G. Meier, A. Plopski, and G. Klinker. An Outdoor Ground Truth Evaluation Dataset for Sensor-aided Visual Handheld Camera Localization, *Proceedings ISMAR*, 263–264, Oct. 2013.

## Domestic Conferences

### Non-peer-reviewed

1. T. Furuichi, T. Abe, A. Plopski, T. Mashita, K. Kiyokawa, H. Takemura, and T. Fukuda. Relocalization method of Real-time Wide-Area Reconstruction System Using RGB-D Camera for a Patrol Robot, *IEICE technical report*, 113(403):207–212, Jan. 2014. (in Japanese)
2. T. Abe, T. Furuichi, A. Plopski, T. Mashita, K. Kiyokawa, H. Takemura, and T. Fukuda. Designing a Real-time Wide-Area Reconstruction System Using RGB-D Camera for a Patrol Robot in Factory, *IEICE technical report*, 113(227):129–134, Sep. 2013. (in Japanese)

### Non-peer-reviewed Posters

1. A. Kudo, A. Plopski, T. Höllerer, T. Mashita, K. Kiyokawa, and H. Takemura. Construction of a robust feature database in various light condition, *CVIM*, 2015(65):1–5, No. 58, Jan. 2015. (in Japanese)
2. A. Plopski, Y. Itoh, C. Nitschke, K. Kiyokawa, G. Klinker, and H. Takemura. CIC - Corneal Imaging Calibration for Optical See-Through Head-Mounted Displays, *27th OACIS Symposium*, 1–1, Dec. 2014. (Chapter 4)

## Theses

- A. Plopski. User Distraction through interaction with a wall-sized display, *Master's Thesis, Department of Informatics, Technische Universität München*, May, 2012.
- A. Plopski. Development of a marker-based visual servo control interface for industrial robots, *Bachelor's Thesis, Department of Informatics, Technische Universität München*, May, 2010.



---

## Abstract

Augmented reality (AR) and virtual reality (VR) found their way into various areas of our lives. AR and VR applications greatly benefit from a high degree of immersion that includes consistent visualization and intuitive interaction with the virtual content. Head-mounted displays (HMDs), in particular optical-see-through (OST) HMDs, provide a natural interface for the presentation of AR. However, despite greatly improved design and years of research on OST-HMDs and AR, current systems still suffer from a variety of problems, such as complicated interaction, color inconsistencies and manual calibration. We expect that with further development of OST-HMDs, eye-tracking cameras will become an integral part of the device.

Existing methods in eye detection use either eye features, e.g., the contour of the iris, or reflection of known light sources, commonly infra-red (IR) light emitting diodes (LEDs), to recover the position of the eye. Although the latter allow for high accuracy their use is limited to indoor scenarios and requires an accurate geometric calibration of the LEDs relative to the camera. This also limits their use to headworn or stationary systems. Under natural illumination the extraction of the light sources is a complicated task and user experience suffers from the intrusiveness of the artificial illumination.

We propose to use corneal imaging (CI), the analysis of the corneal reflection of the observed scene under natural illumination, to estimate the pose of the eye. We show how the estimated position can be applied to improve the AR experience in OST-HMDs and enable gaze-based interaction with out-of-reach AR and VR content.

OST-HMD calibration determines the spatial relation between the scene camera of the HMD and a first-person view camera that models the user's perception to correctly align virtual content and the real scene. We introduce Corneal Imaging Calibration (CIC), an automated calibration approach for OST-HMDs. The method does not require user interaction and can detect drift of the HMD. Furthermore, it does not require the detection of the iris contour or the eye pose, a requirement of previous automated methods. This improves the robustness in environments where the iris contour cannot be detected reliably. We present an in-depth evaluation and discuss possible error sources and drift detection strategies.

Interaction with out-of-reach AR and VR content, e.g., projector- and OST-HMD-based AR, requires input through external controllers or voice commands. Eye-gaze based interaction offers a more socially acceptable and natural interaction with such content and has been proposed as part of AR application. We propose a new passive gaze tracking approach based on the estimated position of the eye. Our Hybrid eye-pose estimation does not require IR LEDs commonly used in commercial systems and adopts the approach used by these systems for use with images taken under natural illumination. We show, that our method can estimate the user's iris size and account for the

impact of the illumination on the detected iris size. The proposed method does not require a gaze-mapping calibration and does not suffer from parallax issues as the position of the eye can be estimated in arbitrary scenes, as long as the scene-model is known. We show that our method outperforms standard methods commonly used in passive eye-gaze tracking and achieves an accuracy of about 1.7°.

The proposed applications require feature matches between the scene and the captured image to estimate the position of the eye. However, this approach is unreliable and error prone in CI. We propose a method based on inverse rendering that robustly tracks the position of the eye from the reflection of an arbitrary known scene. We show that the method can deal with various environments and outperforms results from feature matching.

Following our observations of CIC, we present the results of a user study that evaluates the noticeability of spatial misalignment errors of AR shown on an HMD, e.g., as a result of an incorrect calibration or erroneous world model. Existing systems aim for perfect spatial alignment of virtual content with the real scene. In practice, this is not necessary, as users often cannot distinguish small shifts from the ground truth. Answering the question of the noticeability thresholds can help to define realistic goals for future calibration algorithms and improve the understanding of requirements in commercial applications.

The results of this dissertation show that CI can be used to determine the spatial properties of the eye in AR applications where the scene-model is available. In the future, spatial estimation of the eye with the analysis of the content reflected on the cornea may be used to address other aspects of AR, such as color consistency and user experience.

---

## Acknowledgments

This dissertation is the highlight of my academic carrier and the result of multiple years of research. It could not have been completed without the support of my advisors, colleagues, family, and friends.

I would like to thank my supervisors Assoc. Prof. Kiyoshi Kiyokawa and Assist. Prof. Christian Nitschke for their continuous support and advice. Without their guidance this work would not have been possible.

I would also like to express my gratitude to Prof. Haruo Takemura for providing me with the facilities to conduct the research of this dissertation and the Ministry of Education, Culture, Sports, Science and Technology of Japan (MEXT) for providing me with the funds over the duration of the Ph.D. program and the opportunity to conduct my research in Japan. This work is the result of research funded in part by the Grants-in-Aid for Scientific Research(B), #24300048 and #15H02738 from Japan Society for the Promotion of Science.

I want to express my special thanks to Prof. Gudrun Klinker who supported my stay in Japan and provided me with research opportunities and facilities in Germany.

I appreciate the help of Dr. Amit Agrawal for making available a copy of his code for spatial catadioptric camera calibration. I would also like to thank Yuta Itoh for providing me with images taken while wearing an optical-see-through head-mounted display and valuable discussions on various research problems.

My gratitude goes out to Kenneth R. Moser and Prof. Dieter Schmalstieg for their insight and discussions on the research presented in this dissertation.

I want to thank all member of the Takemura Laboratory who supported me during my stay in Japan. In particular, I want to express my gratitude to Dr. Nicholas Katzakis and Jason Orlosky for their support and numerous discussions on various research topics and future directions. I would also like to express my thanks to everyone who participated in the experiments presented in this dissertation.

Many thanks go out to Megumi, who supported me the whole time through the easy and stressful periods and never stopped believing in me.

Finally, I wish to express my thanks to my family and all my friends who supported me from afar and offered spiritual support when I needed it the most. I am grateful to them for always urging me on to pursue my goals and interests throughout the Ph.D. program.

Alexander Plopski  
*Osaka University*  
January 2016



# Contents

<b>List of Tables</b>	<b>xvii</b>
<b>List of Figures</b>	<b>xix</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Head-mounted Displays . . . . .	2
1.2 Corneal Imaging . . . . .	4
1.3 Contribution . . . . .	5
1.3.1 Corneal Imaging Calibration of OST-HMDs . . . . .	5
1.3.2 Hybrid Eye-pose Estimation . . . . .	8
1.3.3 Inverse-rendering-based Cornea Tracking . . . . .	9
1.3.4 User Spatial Consistency Perception in HMD-based AR	9
1.4 Notation . . . . .	10
1.5 Outline . . . . .	10
<b>2 Fundamentals of the Eye</b>	<b>13</b>
2.1 Eye Anatomy . . . . .	13
2.1.1 Composition of the Eye . . . . .	13
2.1.2 Axes of the Eye . . . . .	16
2.2 Eye Model . . . . .	17
2.2.1 Schematic Eye Models . . . . .	17
2.2.2 Spherical Eye Model . . . . .	17
2.3 Eye-pose Estimation . . . . .	19
2.3.1 Methods . . . . .	20
2.3.2 Pupil Center Corneal Reflection (PCCR) . . . . .	24
2.3.3 Limbus Reconstruction . . . . .	28
2.4 Point-of-Regard Estimation . . . . .	32
2.4.1 Regression-based . . . . .	32
2.4.2 Geometric . . . . .	33
<b>3 Fundamentals of OST-AR</b>	<b>35</b>
3.1 Manual OST-HMD Calibration . . . . .	35
3.2 Spatial OST-HMD Calibration . . . . .	39
3.2.1 Extrinsic Camera Calibration . . . . .	40
3.2.2 Screen Reconstruction . . . . .	41
3.3 Automated OST-HMD Calibration . . . . .	42
3.3.1 INDICA Full . . . . .	43
3.3.2 INDICA Recycle . . . . .	43
3.4 Perception of AR Content . . . . .	44
3.4.1 User Perception . . . . .	45

3.4.2	Error impact	45
<b>4</b>	<b>Corneal Imaging Calibration of OST-HMDs</b>	<b>47</b>
4.1	Introduction	47
4.2	Corneal Imaging Calibration	48
4.2.1	Eye Position Estimation	48
4.2.2	Drift Detection	50
4.3	Synthetic-Data Experiment	50
4.4	Real-Data Experiment	52
4.4.1	Quality of Spatial OST-HMD Calibration	53
4.4.2	Implementation	53
4.4.3	Cornea Position Estimation	55
4.4.4	Eye Position Estimation	55
4.4.5	Projection Error	57
4.4.6	Discussion	58
4.5	Conclusion	61
<b>5</b>	<b>Hybrid Eye-pose Estimation</b>	<b>65</b>
5.1	Introduction	65
5.2	Approach	67
5.3	Experiment	69
5.3.1	Environment Calibration	69
5.3.2	Evaluation	70
5.4	Conclusion	75
<b>6</b>	<b>Inverse-rendering-based Cornea Tracking</b>	<b>77</b>
6.1	Introduction	77
6.2	Inverse Rendering Tracking	78
6.3	Experiment	80
6.3.1	Experiment Environment	80
6.3.2	Stability Comparison	82
6.4	Conclusion	85
<b>7</b>	<b>User Spatial Consistency Perception in HMD-based AR</b>	<b>87</b>
7.1	Introduction	87
7.2	Experiment Design	89
7.2.1	Setup	89
7.2.2	Task	90
7.3	Implementation	91
7.3.1	Environment Calibration	91
7.3.2	Visualization and Interaction	94
7.4	Experiment	94
7.4.1	User Response Time	95
7.4.2	User Confidence	95

7.4.3	Alignment Accuracy	96
7.4.4	Discussion	101
7.5	Conclusion	103
<b>8</b>	<b>Conclusion</b>	<b>107</b>
8.1	Summary	107
8.2	Future Directions	109
<b>Bibliography</b>		<b>111</b>



# List of Tables

1.1	Comparison of eye-pose estimation strategies . . . . .	9
2.1	Eye parameters assumed by various eye models (in mm) . . . . .	18
5.1	Estimated personal parameters and eye-pose error . . . . .	72
7.1	VST vs OST statistical significance in accuracy by marker orientation . . . . .	97



# List of Figures

1.1	Different HMD models. . . . .	2
1.2	Augmentation from the perspective of the scene camera . . . . .	6
2.1	The frontal view and the anatomical composition of the eye . . . . .	14
2.2	The geometric eye model used in this dissertation . . . . .	14
2.3	Distribution of rods and cones on the retina of the eye . . . . .	16
2.4	Reflection of a point on the cornea and its projection into the camera . . . . .	25
2.5	Reflection of a backprojected ray on the cornea . . . . .	26
3.1	Concept of OST-HMD calibration . . . . .	36
3.2	Estimation of a point relative to the HMD . . . . .	37
3.3	Results of the SPAAM OST-HMD calibration . . . . .	39
3.4	Instability of the eye-position estimated by SPAAM . . . . .	40
3.5	Setup for spatial calibration of the HMD . . . . .	41
3.6	An OST-HMD equipped with an eye-tracking camera . . . . .	42
4.1	Cross-section view of rotations of the eyeball . . . . .	49
4.2	Estimated distance of the cornea from the camera depending on the assumed cornea size . . . . .	51
4.3	Deviation of estimated eye positions from the ground truth due to data perturbed by noise . . . . .	52
4.4	Estimated planes of the HMD screen for four different views . . . . .	53
4.5	Reflection of the HMD-screen in a mirrorball . . . . .	54
4.6	Reflection of the display on the estimated corneal sphere . . . . .	55
4.7	Convergence rate of evaluated calibration methods . . . . .	56
4.8	Projection error for evaluated calibration approaches . . . . .	57
4.9	Error vector distribution for each evaluated method . . . . .	58
4.10	Reprojection error depending on the cornea radius and distance from the camera . . . . .	59
4.11	Schematic representation of the reflection of a backprojected ray on corneas of different size along the same ray . . . . .	60
5.1	Two steps of the Hybrid eye-pose estimation . . . . .	67
5.2	Visibility of the iris varies depending on the gaze direction . . . . .	68
5.3	Gaze estimation by the Hybrid method . . . . .	69
5.4	Setup of the eye-pose estimation experiment . . . . .	70
5.5	Results of the iris estimation by the compared methods . . . . .	71
5.6	POR estimated by the evaluated methods . . . . .	74
5.7	Eye-pose estimated from the reflection of the HMD-screen . . . . .	75

6.1	Detection and matching of features by different methods . . . . .	78
6.2	Inverse Rendering approach for estimation of the position of the corneal sphere . . . . .	79
6.3	Estimation of the origin of a camera pixel given a known corneal sphere . . . . .	79
6.4	Results of estimating the position of the corneal sphere from an initial guess through inverse rendering . . . . .	81
6.5	Cornea tracking experiment environment . . . . .	81
6.6	Display patterns used in the experiment . . . . .	81
6.7	Estimated position of the corneal sphere for the recorded sequences . . . . .	83
6.8	Images captured by the camera and the overlay of the reflection of the scene on the estimated cornea position . . . . .	84
7.1	Various types of spatial registration error . . . . .	88
7.2	Side-view of the perception experiment setup . . . . .	92
7.3	The wall is illuminated simultaneously by two projectors . . . . .	93
7.4	Schematic experiment setup . . . . .	95
7.5	Display mode set completion times, in minutes . . . . .	96
7.6	Confidence distributions for the VST and OST display sets . . . . .	98
7.7	Confidence values by marker distance . . . . .	99
7.8	Translation error along the x-axis by marker distance . . . . .	100
7.9	Translation error along the y-axis by marker distance . . . . .	100
7.10	Angular error by marker distance . . . . .	101

# CHAPTER 1

## Introduction

---

Presenting stories, ideas and information to others has been an integral part of our society since its beginnings. During the Stone Age, *Homo sapiens* used charcoal and cave walls to write down the stories of their hunt and pray for a fertile year. Ever since then, people came up with new ways of presenting what they have seen or imagined to others. The primary medium shifted over the years from speech to paper, theater, and finally in the past century movies. Today, an increasing portion of the backgrounds and characters presented in the movies is created through computer renderings. This allows the creation of worlds not achievable by other means and further enhances the ability to present an imagined environment. Various applications, e.g., in gaming, remote avatar systems, navigation, and training, use computer rendered content to visualize not easily accessible information.

The content can be presented from the perspective of a third-person where the viewer is an unrelated observer of the depicted scene or as a first-person view where the viewer slips into the body of a participant and share's the view. The third-person view is the choice when a wide view is beneficial, while the first-person view results in a higher immersiveness into the scene. The degree of immersiveness is expressed by the feeling of being part of the depicted scene, in particular the character taken by the user, and depending on the experience can involve one or multiple senses. For example, haptic feedback devices ([Stone \(2001\)](#); [Bau and Poupyrev \(2012\)](#)) provide the sense of touch, body tracking ([Bleiweiss et al. \(2010\)](#); [OptiTrack \(2015\)](#)) enables natural interaction and motion re-targeting to the avatar, and surround sound addresses hearing.

Visual immersiveness can be achieved in two ways—the virtual content can become a part of the user's surroundings or make the user a part of the virtual setting. The transition from the real environment to an entirely virtual scene is described by the Reality-Virtuality Continuum ([Milgram et al. \(1995\)](#)) with environments composed only of real and only virtual objects as the two extremes. An environment that contains only virtual objects is referred to as virtual reality (VR) and an environment that combines real and virtual objects is referred to as mixed reality (MR) or augmented reality (AR). [Milgram et al. \(1995\)](#) define AR in a stricter context of a scene that consists primarily of real objects, while a scene that is predominantly virtual is referred to as augmented virtuality. In the context of this work we use the broader definition of AR as a general combination of virtual content with the real scene.



Figure 1.1: Different HMDs developed over the years. (a) The sword of Domacles — the first HMD ([Sutherland \(1968\)](#)), (b) Occulus Rift DK2 equipped with the Ovrvision Module — a typical VST-HMD ([Oculus Rift \(2015\)](#); [Ovrvision \(2015\)](#)), and Epson Moverio BT 200 — an OST-HMD ([Epson \(2015\)](#)).

## 1.1 Head-mounted Displays

Window-on-the-world (WoW) is a common approach to present the virtual content. Hereby, a remote display, e.g., a stationary monitor or a handheld device, is used to present the content. The view of the content is occupying only a small portion of the user’s field-of-view (FOV) and is mostly unrelated to the user’s actual viewing direction. Some systems incorporate tracking of the user’s perspective to create a more consistent view through the display ([Tomioka et al. \(2013\)](#)). Although the WoW approach is very simple and wide-spread it provides a very limited degree of immersiveness.

For VR experiences CAVE environments can be used to achieve a high degree of immersiveness by surrounding the user with displays ([Freitag et al. \(2015\)](#)) and rendering the virtual content from the user’s perspective. However, such environments are difficult to deploy in everyday applications. Head-mounted displays (HMDs) (Figure 1.1) offer a more generally applicable solution to this problem. HMD research began more than half a century ago with the first HMD by [Sutherland \(1968\)](#). Due to their versatility HMDs found application in stress therapy ([Pair et al. \(2006\)](#)) and pain killer replacement ([Li et al. \(2011\)](#)) through out-of-body-experiences. Recently, these devices have received a lot of public attention due to their application in the gaming industry ([Oculus Rift \(2015\)](#); [Thomas \(2012\)](#)).

AR experiences embed virtual content into the scene surrounding the user. This can be done by either capturing the surroundings on an imaging device, like a camera, or by adding virtual content that is visible to the user, together with the surroundings. Current state-of-the-art (SOTA) technologies use the first way to present AR through a WoW approach on mobile devices and have become a common occurrence in advertisement and media. AR and VR technology is a common occurrence in science fiction movies, however the presented view does not require handheld devices, instead the graphics are commonly displayed directly into the user’s FOV. HMDs that can enable such

interaction are referred to as optical see-through (OST) HMDs. It is possible to adapt existing non-OST HMDs to display the surroundings by attaching a front facing camera—thus the HMD becomes a video-see-through (VST) HMD. As a result, VST-HMDs manipulate the user’s perception, as the scene is presented from the camera’s point of view and is based on the camera’s imaging system. Thus the presented image can include deformations caused by the camera’s lens, and its color perception and light sensitivity settings. Finally, VST systems are not fail safe—if the camera turns off, the user is left in the dark. Although the optics of OST systems slightly distort the user’s perception (Itoh and Klinker (2015a)), the effect is far less than that of VST systems. Furthermore, OST systems are fail-safe—if the device turns off, the user can still see the surroundings. Therefore, OST systems appear to be a natural interface for AR experiences and OST-HMDs are often depicted as the deployment platform of AR in science fiction movies. Nonetheless, despite the benefits, existing OST-HMD technology still is not suitable to provide the envisioned experience. Current devices suffer from various problems (Rolland and Fuchs (2000)) that can be categorized as follows:

**Design issues** address a variety of problems that have to be solved to enable everyday use of OST-HMDs. These include among others, heavy weight, short battery life, small FOV, and complicated interaction.

**Spatial consistency** addresses the stable visualization of virtual content at the intended location. In OST-HMDs, this requires tracking of the user in the scene (Kato and Billinghurst (1999); Klein and Murray (2007)) accounting for the refraction of the incoming light by the HMD optics (Itoh and Klinker (2015a,b)), and the alignment of the virtual content from the user’s perspective (Azuma (1995); Tuceryan and Navab (2000); Owen et al. (2004); Itoh and Klinker (2014a)).

**Temporal delay** refers to the delay between the user’s motion, e.g., head rotation, and the update of the content shown on the screen. This delay is caused by the processing of sensor signals, e.g., capturing the image by a scene camera, or reading of gyroscope data, to detect any motion, the content processing pipeline and finally the rendering of the virtual content on the screen. Delayed visualization has been shown to impact the realism of VR systems and can even lead to simulation sickness (AGARD Conference Proceedings No. CP-433 (1988)). It is well known that the just noticeable difference (JND), the point at which users begin to notice a discrepancy between virtual and real content, lies in the range of 5-20ms (Adelstein et al. (2003); Ellis et al. (2004); Bailey et al. (2004); Oculus Rift (2016)). VR environments that require a high degree of immersion, may even require latency of no more than 3ms (Jerald and Whitton (2009)). Although existing systems fail to prevent

temporal delay, future systems may incorporate specialized hardware, such as built-in sensors and predictive tracking of the scene, for low-latency augmentations. [Zheng et al. \(2014b\)](#) show that a custom-built projector with a partial image update algorithm can show augmentations of a grayscale image with a delay of only 44  $\mu$ s.

**Visual consistency** includes issues such as, transparency ([Gao et al. \(2012\)](#); [Kiyokawa et al. \(2001\)](#)), color consistency ([Itoh et al. \(2015\)](#); [Hincapie-Ramos et al. \(2014\)](#)), depth perception ([Swan II et al. \(2015\)](#)) and occlusion ([Shah et al. \(2012\)](#); [Lieberknecht et al. \(2011\)](#)).

**Social acceptance** includes various user related issues such as safety concerns, user expectations and privacy concerns ([Roesner et al. \(2014\)](#)). One of the reasons previous attempts to introduce OST-HMDs to the consumer market failed was the scene camera, one of the requirements of AR applications. Continuously facing a scene camera and a microphone possibly recording every conversation led to a large number of complaints by bystanders. This led to a variety of institutions banning the use of OST-HMDs, e.g., restaurants ([MyNorthwest.com \(2011\)](#)), which in turn caused a decline in interest in said devices.

It is necessary to address the above problems to allow OST technology, in particular OST-HMDs, to find wide-spread application as a commodity device.

## 1.2 Corneal Imaging

Current OST-HMD systems are designed as a black box environment where no feedback is available on whether the displayed content is spatially correct, or the selected contrast and color correction actually result in the desired experience. To enable analysis of the user's perception it is necessary to develop an approach to model the user's view. The visual information of the real and virtual scenes are perceived by the user's eyes. The anatomical composition of the eye does not only bundles incoming light onto our visual sensor, but also reflects up to 15% ([Kaufman and Alm, 2011](#), p.79) of incoming light. This reflection can be detected by an onlooker, or a camera focused onto the eye. This property is commonly used to detect the reflection of infra-red (IR) LEDs on the corneal surface ([Guestrin and Eizenman \(2006\)](#)). Use of IR cameras however eliminates the reflection of visible light. However, this information can provide essential information about the user's view and their surroundings. [Nishino and Nayar \(2004b\)](#) refer to the analysis of corneal reflections under natural illumination as *corneal imaging* (CI). They show that CI can be applied for scene mapping, face detection ([Nishino and Nayar \(2004b, 2006\)](#)) or scene relighting ([Nishino and Nayar \(2004a\)](#)). Backes et al.

(2009) show that the reflection of letters on a monitor can be detected with a specialized setup, even from a remote location — effectively enabling spying on the user. Nitschke et al. (2009) use CI to detect the reflection of points shown on a monitor and use known eye-poses to reconstruct the pose of the monitor relative to the camera. Schnieders et al. (2010) have extended their method to also estimate the Point-of-Regard (POR) of the user. CI has found application to a variety of problems (Nitschke et al. (2013c)) since it was introduced by Nishino and Nayar (2004b).

As virtual content shown on a screen is also reflected in the eye, CI offers a unique opportunity to understand if the user is perceiving the intended visualization or whether adjustments are necessary. Furthermore, as CI is not restricted by the distance to the tracking camera, it can be applied for non-intrusive eye analysis in arbitrary AR and VR scenarios, such as projector based AR and large-scale CAVE scenarios.

## 1.3 Contribution

In this work we address estimation of the geometric eye-pose in AR and VR through CI. We focus on OST-HMDs, where the eye–camera relation remains relatively stable. Nonetheless, a modified version of the proposed methods can be applied in arbitrary AR and VR applications.

### 1.3.1 Corneal Imaging Calibration of OST-HMDs

Estimating the user’s pose in the world, or relative to the reference object, is a common problem in a variety of computer vision (Se et al. (2001)) and AR applications. Over the years a large number of solutions was developed to achieve stable results in various application scenarios (Kato and Billinghurst (1999); Klein and Murray (2007); Kurz et al. (2014)). In an OST-HMD application, however, it is not enough to determine the pose of the scene camera. Figure 1.2 shows the results of an augmentation from the perspective of the scene camera. As its pose does not coincide with the perspective of the viewing camera (the user) the augmentation is strongly misaligned.

Two approaches exist that model the augmentation of the scene from the user’s perspective, instead of the scene camera.

- Modeling of the eye–OST-HMD-screen relation as a second camera, whose image plane corresponds to the OST-HMD-screen. This approach works well for static setups. Hereby, the user has to perform a manual calibration to determine the intrinsic and extrinsic parameters of said camera (Azuma (1995); Tuceryan and Navab (2000)). Whenever the Eye–OST-HMD-screen relation changes, e.g., the HMD is moved on the head or taken off and put on again, the calibration becomes invalid and

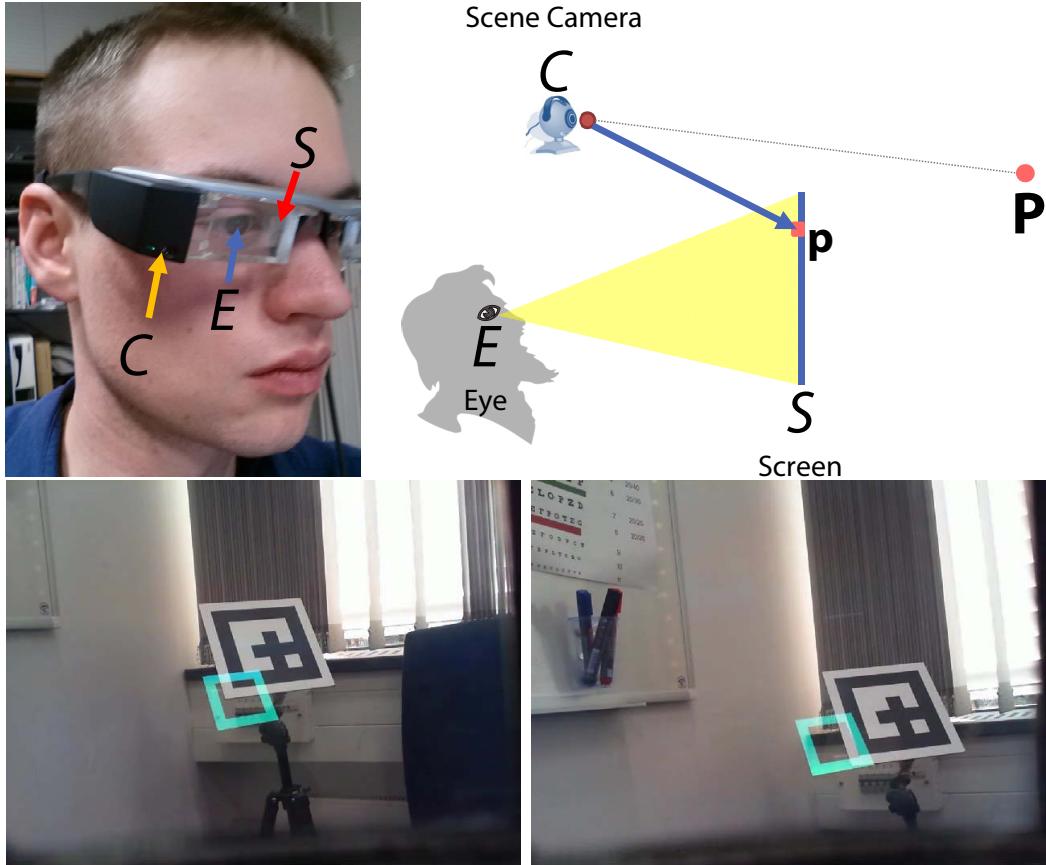


Figure 1.2: Augmentation from the perspective of the scene camera. (Top row) A user is looking through an OST-HMD. The position  $\mathbf{P}$  of the augmentation target and the overlaying pixel  $\mathbf{p}$  is determined by the scene camera  $C$ . (Bottom row) The objective is to show a blue overlay over the marker contour. A camera placed behind the OST-HMD-screen captures the augmentation shown on the screen  $S$ . As the pose of the scene camera  $C$  does not coincide with the view of the camera, the augmentation is displayed incorrectly.

has to be repeated. Although it can achieve very stable results, the manual calibration is a very tedious process and in practice, even though a recalibration is necessary, it is often skipped.

- Alternatively, the OST-HMD-screen can be modeled as a surface in space (Owen et al. (2004); Itoh and Klinker (2014a); Klemm et al. (2014)) or a light-field Itoh and Klinker (2015b). If the screen is modeled as a surface the pixel that augments a 3D point  $\mathbf{P}$  can be determined by intersecting the ray from the center of projection of the human eye towards  $\mathbf{P}$  with the OST-HMD-screen. On the other hand, if the screen is modeled as a light field, the augmented pixel is computed as a function of the eye-position under consideration of the direction of the incoming

light ray.

The manual calibration is the go-to approach in today’s applications. Various researchers have explored its application for stereo OST-HMD calibration (Genc et al. (2000)), how various aspects impact the calibration results (Axholt et al. (2010, 2011)), further improvements to the calibration method (Moser and Swan II (2015)), and simplification of the recalibration (Itoh and Klinker (2014a,b)). Nonetheless, the simple modeling limits its applicability and demands users to be cautious as to not let the OST-HMD slip. By regarding the eye and the OST-HMD-screen as two separate systems, it is possible to develop a more universal calibration approach. Additionally, this allows for more complex modeling of the HMD-screen. While the manual calibration represents the HMD-screen as a plane, it has shown that it is possible to model it as a curved surface (Owen et al. (2004)) or a light-field (Itoh and Klinker (2015b)).

To our knowledge, Interaction Free Display Calibration (INDICA) by Itoh and Klinker (2014a) is the only existing calibration approach for OST-HMDs without the need for user interaction. Their method is based on an eye-pose estimation by a passive eye tracker, due to a complicated spatial calibration of IR-based trackers relative to the HMD-screen and their inapplicability in outdoor environments. As INDICA is based on geometric passive eye-pose estimation even small errors in the extraction of the iris contour, e.g., due to highlights or eye lashes, can lead to large errors in the estimation process. We present *Corneal Imaging Calibration* (CIC) for OST-HMDs (Chapter 4). Our method is based on the observation that the reflection of OST-HMD-screen content on the cornea can be detected by an eye-tracking camera. By analyzing this reflection, we apply the theory presented in Chapters 2 and 3 to estimate the position of the cornea. From at least three non-coplanar cornea positions, we obtain the eye position as their center of rotation. This achieves a practical and lightweight HMD (re-)calibration method.

The main benefits of this method are:

- CIC is based on the accurate detection of the cornea, similar to SOTA commercial eye-gaze trackers.
- We show that the approach has major advantages over SOTA methods: It is more practical as it uses simple and automatable image processing, less depending on correct eye modeling as the error propagates less into the result, and more robust as it does not require iris detection.
- Our Hybrid eye-pose estimation is based on the cornea estimated by the same approach as in CIC. Thus CIC can be extended to incorporate high-quality eye-pose and POR estimation in OST-HMDs (Section 5.3.2.3.)

### 1.3.2 Hybrid Eye-pose Estimation

Current solutions that enable interaction with an OST-HMD require either to voice the desired action, e.g., Google Glass, use an external controller, e.g., Moverio BT200, or detect hand motion by the user (Colaço et al. (2013)). All these solutions however are tedious, tiring, or simply inconvenient. When interacting with an OST-HMD users commonly look at the content they are interested in and as such, POR estimation is a useful alternative to existing pointing solutions. Additionally, binocular POR estimation can be used to determine the plane the user is focused on and blend out the virtual content if the user is focused on the scene behind it. The idea of using eye-gaze trackers with HMDs is not new. Park et al. (2008) use an off-the shelf eye tracker to enable interaction with virtual content shown on the display of an OST-HMD. Tsukada et al. (2011) propose to use an eye-tracker for first-person view applications — a combination with an OST-HMD seems natural. Nilsson et al. (2009) and Orlosky et al. (2015) use VST-HMDs with eye-gaze tracking (EGT) capabilities to trigger interaction with AR content.

Over the years a number of methods have been developed to enable EGT and POR estimation. The POR can either be learned from known matches of the projection of the eye into the camera and known PORs (Kassner et al. (2014)), or the geometric eye-model can be recovered and used to intersect the gaze with the assumed scene-model. Although the first approach is simple and has been used to acquire an accurate eye-pose in a variety of setups, it requires continuous recalibration and as such, the geometric reconstruction offers a more generally applicable solution used in SOTA commercial devices.

Eye-Pose estimation techniques can be divided into active methods that require manipulation of the scene by known light sources in the environment, typically infra-red LEDs, and passive methods that use images taken under natural illumination. Active light techniques require an accurately calibrated environment and controlled illumination to ensure robustness.

The benefits and drawbacks of the different solutions are shown in Table 1.1. We propose to combine the benefits of both strategies through corneal imaging and refer to this approach as *Hybrid* eye-pose estimation (Chapter 5). When wearing an OST-HMD, or interacting with augmented content in the environment, the scene-model is known, e.g., the OST-HMD-screen or the surface of the handheld device. By detecting the reflection of the content shown on this surface, the position of the cornea can be reconstructed. Active methods determine the gaze direction from the easily detectable pupil contour. However, under natural illumination this contour cannot be detected reliably. We therefore propose to fit the iris contour to the tracked/detected position of the cornea instead. This allows us to not only recover the eye-pose under accurate geometric constraints, but also estimate additional personal parameters, in particular the iris size. As a result, the eye-pose estimated by our Hybrid approach is more reliable to outliers and recovers a more accurate

Table 1.1: Comparison of eye-pose estimation strategies.

Features	Active	Passive	Hybrid
Accurate eye-pose	yes	no	yes
Eye-pose from natural images	no	yes	yes
Personal parameter calibration	beneficial	beneficial	beneficial
Eye-model parameter estimation	yes	no	yes
Geometric estimation	yes	no	yes
Restrictions			
ROI required	no	yes	no
IR light: limited outdoor use	yes	no	no
IR light: long-term exposure	yes	no	no
Complex setup	yes	no	no
Scene-model required	no	no	yes
Parallax issues	yes	yes	no

eye-pose than traditional passive estimation. In the future this approach can be used in combination with either a 2.5D external camera or in remote systems where active light trackers cannot be deployed, to recover an accurate POR without extensive training sessions.

### 1.3.3 Inverse-rendering-based Cornea Tracking

Existing methods in CI assume that the cornea has been already detected and use the estimated eye-pose for further analysis (Nishino and Nayar (2004a); Nitschke et al. (2009); Takemura et al. (2014b,a)). In practice it is difficult to detect the cornea from feature matches as the corneal reflection contains strong distortion and noise. Furthermore, other features in the surrounding, e.g., the eye lashes, may lead to false matches and thus impact the overall results. Reliable tracking is a requirement for the practical application of CI. We present a novel approach to track the cornea from an initial guess in an environment with a known scene-model (Chapter 6). This assumption is also viable in see-thruh (ST) AR, as the content shown on the display is known at any point in time. We show that our method can be applied in different scenarios and achieves satisfactory results.

### 1.3.4 User Spatial Consistency Perception in HMD-based AR

Our final contribution is a subjective evaluation of the accuracy of the visualized content in a VST- and an OST-HMD environment. The purpose of OST-HMD calibration and ever improving tracking algorithms is to provide a seamless overlay of virtual content onto the scene. As such, it is generally assumed that perfect alignment is necessary. However, this is not necessarily the case and it is sufficient to achieve a degree of accuracy, where users can

no longer notice that the augmentation is misaligned. Various studies investigated the accuracy required in handheld AR ([Madsen and Stenholz \(2014\)](#); [Tokunaga et al. \(2015\)](#)). HMDs present a different view on the augmentation than a handheld solution. To our knowledge the question of spatial consistency in HMDs has not been addressed so far and our study provides a lower boundary threshold for VST and OST setups and a comparison between the two visualization methods ([Chapter 7](#)).

## 1.4 Notation

In this section we explain the notation used throughout the remainder of the dissertation. We denote an object and its coordinate system by an upper case letter, such as  $S$ . A single 3D point is denoted by a bold upper case letter, such as  $\mathbf{P}$ , and a 2D point by a bold lower case letter, such as  $\mathbf{p}$ . If we refer to a point  $\mathbf{P}$  not in the world coordinate system, we introduce the coordinate system as an upper index to the left of  $\mathbf{P}$ , e.g.,  ${}^A\mathbf{P}$  is used to refer to  $\mathbf{P}$  in the coordinate system  $A$ .

We denote a vector between two points, with a bold lower case letter, such as  $\mathbf{v}$ . The unit vector of a vector  $\mathbf{v}$  is given by  $\hat{\mathbf{v}} = \mathbf{v}/\|\mathbf{v}\|$ . Lower case letters, for example  $d$ , are used to represent scalar values.

We represent a matrix in sans serif font, such as  $\mathbf{P}$ . In particular, we always refer to an  $n \times n$  identity matrix as  $\mathbf{I}_{n \times n}$  and a rotation matrix as  $\mathbf{R}$ . The transformation from coordinate system  $A$  into coordinate system  $B$  is described by a  $4 \times 4$  matrix  $\mathbf{T}$ , denoted as  ${}^A\mathbf{T}$ .  ${}^A\mathbf{T}$  is composed of  $({}^A\mathbf{R}, {}^A\mathbf{t})$  where  ${}^A\mathbf{R}$  and  ${}^A\mathbf{t}$  stand for rotation and translation respectively. Furthermore, explicit transformation of  ${}^A\mathbf{P}$  to  ${}^B\mathbf{P}$  can be written as  ${}^B\mathbf{P} = {}^A\mathbf{R}{}^A\mathbf{P} + {}^A\mathbf{t}$ . We refer to the transpose of a matrix or a vector as  $(\cdot)^T$ .

## 1.5 Outline

The remainder of this dissertation is structured as follows:

[Chapter 2](#) presents work related to the modeling of the eye. We first describe the anatomy of the eye and the applied geometric eye model. This explanation is followed by an overview of eye-pose estimation techniques and the mathematical background of eye-pose estimation from the detected iris contour. We follow this with an explanation on how an accurate position of the cornea can be computed from the detected reflection of known light sources.

[Chapter 3](#) gives an overview of OST-HMD calibration methods and discusses how various aspects of AR impact the user's perception.

[Chapter 4](#) introduces CIC for OST-HMDs. We explain how the detected reflection of content shown on an OST-HMD can be used to estimate the

user's view relative to the scene camera and the HMD-screen and compare the method with existing solutions.

Chapter 5 presents Hybrid eye-pose estimation — a combination of active and passive eye-pose estimation methods. The chapter explains how the estimated position of the cornea can be used to determine a user-specific iris size and refine the estimation of the iris contour. The explanation is followed by an evaluation in a simple scenario and a comparison with the traditional approach that detects the iris contour in the camera image and to recover the eye-pose.

Chapter 6 presents our approach for tracking the cornea from a known scene-model through inverse rendering. We compare the results with estimation from stable 2D–3D correspondences and show that the method is more robust and applicable in general scenarios.

Chapter 7 presents an empirical study of the impact caused by incorrect spatial alignment of virtual and real content on the user. The chapter explains the setup used to simulate the view through an OST and a VST system followed by the results of the user study.

Chapter 8 summarizes the findings presented in this dissertation and presents a number of possible future directions.



# CHAPTER 2

# Fundamentals of the Eye

---

In this chapter we review the modelling of the eye and it's applications in eye-pose and POR estimation.

Section 2.1 explains the anatomical composition of the eye and the schematic eye model used in this work. Section 2.3 gives an overview of existing methods for eye localization and gaze estimation.

## 2.1 Eye Anatomy

To comprehend our surroundings, we use primarily five sensory inputs — touch, hearing, taste, smell and vision. Each sensation is useful at different ranges, e.g., touch can be used as long as we can reach something, while vision interprets what lies ahead of us. Over the course of the evolution the eyes have developed to capture visual information that enables long-range navigation and remote analysis of the surroundings. In this section we first provide an overview of the anatomy of the eye followed by a model representation used in this work.

### 2.1.1 Composition of the Eye

When the eye is looked at from the front (Figure 2.1a) three significant parts of the eye, the pupil that is surrounded by the iris and the sclera, can be seen. A sideways view additionally exposes that the pupil and sclera are covered by an additional, spherical layer, the cornea.

As the photoreceptor organ of the human body the eye functions similar to a camera. The cross section (Figure 2.1b) discloses that it consists of a casing (sclera), the capturing lens (cornea), the focusing lens, the shutter (iris and pupil) and the receptor sensor (retina). Overall, the eye resembles a spherical surface with a diameter of 24 mm. The rotation center **E** of the eye is 13.5 mm behind its frontal surface. The eye is divided into three areas, the anterior chamber between cornea and iris, the posterior chamber between iris and lens, and the vitreous chamber behind the lens. The anterior and posterior chambers are filled with the aqueous humor, a clear watery solution with a refractive power of  $n=1.337$ . The vitreous chamber on the other hand is filled with a jelly-like fluid, the vitreous humor. This fluid can liquefy as the person ages, a mostly harmless process, and has a refractive power of  $n=1.336$ . The main elements of the eye are shown in Figure 2.2.

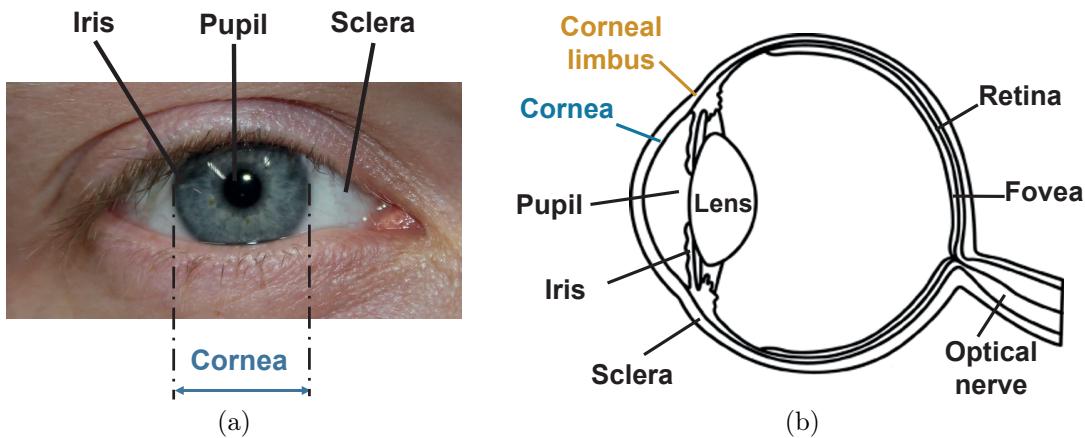


Figure 2.1: (a) The frontal view and (b) the anatomical composition of the eye.

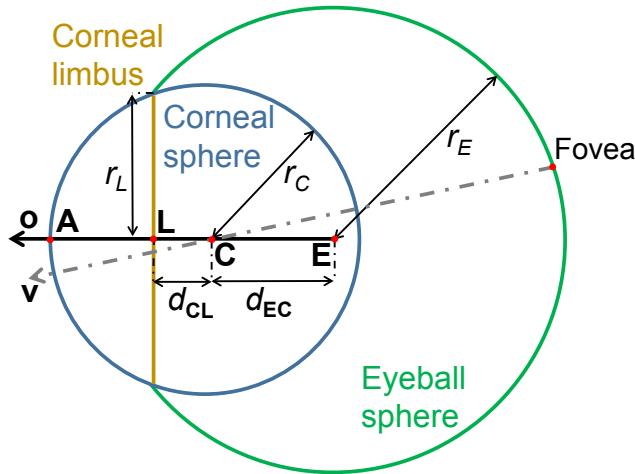


Figure 2.2: The geometric eye model used in this dissertation.

### 2.1.1.1 Cornea

The cornea is a transparent, about 0.5 mm wide spherical layer that covers the pupil and the iris. The corneal surface has a refraction power  $n=1.377$  and is responsible for the majority of the refractive power of the eye. The cornea is transparent due to the absence of blood vessels and relative dehydration. The surface of the cornea is covered by a thin layer of tear fluid for protection. This creates a specular surface that reflects approximately 1-15% of incoming light (Kaufman and Alm, 2011, p.79).

The general anterior of the cornea resembles a spherical shape, centered at **C** with a radius  $r_C=7.8$  mm. The degree of the curvature of the eye varies depending on the distance to the optical axis. Commonly, the curvature flattens as the distance to the optical axis increases, however in some cases it may

become steeper. Additionally, the radius of the vertical curvature is slightly smaller than that of the horizontal curvature. To address the asphericity and toricity of the cornea it can be modelled by an elliptical or conical shape (Mandell and St Helen (1971)). Nonetheless, a spherical approximation is often sufficient for simple applications because a normal cornea usually deviates by no more than a few microns from the spherical shape (Kilic and Roberts (2013)).

#### 2.1.1.2 Sclera

The sclera is the protective outer layer of the eye, it is connected to the oblique and rectus muscles that control the orientation of the eye. Although the sclera displays some reflective properties and texturization, it is primarily a white, diffuse surface. As it covers the majority of the eye, its shape can be approximated by a spherical shape centered at **E**. The actual shape of the cornea is not symmetrical, with the horizontal diameter of 23.5 mm ( $d_H$ ), the vertical diameter of 23mm ( $d_V$ ) and the depth ( $d_{AP}$ ) of 24 mm (Remington (2011)).

#### 2.1.1.3 Limbus

As the cornea transitions into the sclera the curvature undergoes a drastic change due to the different radii of the two spherical surfaces. The transition region of 1-2 mm is referred to as the limbus. It is positioned approximately 2.5 mm behind the apex of the eye **A** and has a diameter of 11.7 mm horizontally and 10.6 mm vertically (Buskirk (1989)). As both, the cornea and the sclera, consist primarily of collagen the transition between the two is not abrupt but gradual. The main difference is that the fibers in the sclera are larger, and less regular, which leads to its opaqueness. In Figure 2.1a the limbus is seen as the diffuse transition from the iris into the sclera.

#### 2.1.1.4 Pupil and iris

The pupil and iris are located in front of the crystalline lens, right behind the corneal limbus. The pupil is the dark area visible from the front of the eye and lets incoming light pass through to the lens. The iris changes the size of the pupil depending on the amount of incoming light. The size of the pupil varies depending on incoming energy between 1.5-8 mm and can account for radiance in the range of  $10^{-6}$ - $10^5$  cd/m. The color of the iris is determined by the amount of pigments contained in it. The most common color being brown, and the least common green. Contrary to the transition from the cornea into the sclera the edge between the pupil and the iris is very distinct and can be observed under controlled illumination.

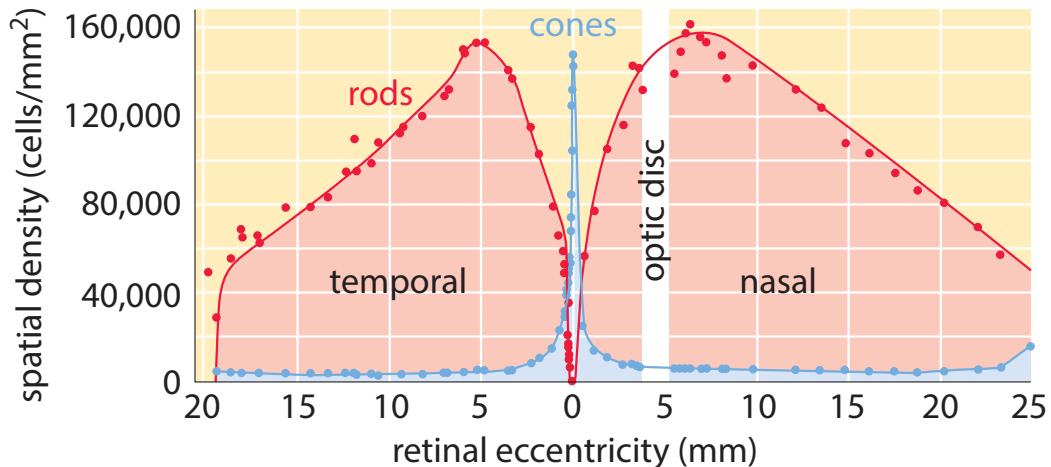


Figure 2.3: Distribution of rods and cones on the retina of the eye. ([Milo and Phillips \(2015\)](#))

### 2.1.1.5 Lens

The lens is responsible for our ability to focus on objects at different distances. Depending on the focused distance, muscles in the ciliary body, the attachment of the lens, contract or relax thus either contracting or relaxing the lens. As the shape of the lens changes, the position of the nodal points changes accordingly. In general, they are located about 1mm behind the lens.

### 2.1.1.6 Retina

The retina is located at the back of the eye and is responsible for the capture of brightness and color. The fovea is a small area of 1.8 mm diameter, located about 2.5 mm towards the temporal region contains approximately 5 million rods that capture incoming light with high resolution and sensitivity to color. The rods located in the remaining retina, especially towards the peripheral regions, capture incoming light at a lower resolution and color sensitivity, but with higher sensitivity to brightness. The distribution of rods and cones is shown in Figure 2.3. The retina contains a blind spot that does not capture any incoming light. This region has a diameter of approximately 1.8 mm and is located above the entry point of the optical nerve.

## 2.1.2 Axes of the Eye

The eye contains multiple axes, with the optical and visual axes as the most relevant. The optical axis  $\mathbf{o}$  is described by the geometric symmetry of the eye. The visual axis of the eye  $\mathbf{v}$ , also referred to as the line of sight (LOS), is described by the gaze point of the eye and the center of the fovea. It is commonly assumed that this axis also goes through the nodal point of the eye

that is located in the vicinity of **C**. The offset between the visual and optical axes is about 5° ([Slater and Findlay \(1972\)](#)).

## 2.2 Eye Model

Eye models represent the optical properties of the eye, rather than its anatomy, and their complexity ranges from very simple representations that represent the eye as a single sphere, to very sophisticated models that model every part of the optical process. An overview of the values assumed by the various models is given in Table [2.1](#).

### 2.2.1 Schematic Eye Models

The first eye models came into conception as early as mid-19th century, however it took until [Gullstrand \(1909\)](#) until a first, widely accepted model was developed. The first models, e.g., Gullstrand's full and simplified eye models ([Gullstrand \(1909\)](#)), and Le Grand's full theoretical eye ([Le Grand and El Hage \(1980\)](#)), represent the eye as a combination of multiple spherical surfaces with fixed refraction parameters. These models assume that all surfaces are parallel to the optical axis of the eye and present an accurate representation of the central region.

The models developed during the second half of the 20th century came to better represent the anatomy of the eye. [Lotmar \(1971\)](#) added aspherical assumptions to previous eye models, and [Kooijman \(1983\)](#) investigated the impact of the curved cornea. These models were motivated by improvements in measurement technologies. Recent research focuses on further personalization of the eye models, e.g., [Navarro et al. \(2006\)](#), [Goncharov et al. \(2008\)](#), and [Polans et al. \(2015\)](#).

Recently, modelling of the topography and the visible parts of the eye, the pupil and the iris, has come into focus within the computer vision community to generate realistic eyes of virtual models. Although the solution by [Bérard et al. \(2014\)](#) can recover a user specific model and determine several topological properties, the approach still requires a very complicated setup to recover the eye model.

### 2.2.2 Spherical Eye Model

Although highly accurate and personalized eye models are required for various applications and treatments in medicine, a simpler model is sufficient for the estimation of the eye position and orientation, as is done in this work. We follow the convention of previous works on eye-pose estimation ([Guestrin and Eizenman \(2006\)](#), [Nishino and Nayar \(2006\)](#), [Villanueva and Cabeza \(2008\)](#)) and use a simplified, spherical eye model. We model the eye as two overlapping

Table 2.1: Eye parameters assumed by various eye models (in mm). (based on [Nitschke \(2011\)](#))

	Eyeball						Posterior				Anterior		
	$d_H$	$d_V$	$d_{AP}$	$d_{AL}$	$d_{IC}$	$d_{CE}$	$r_E$	$r_C$	$r_L$	$r_{LH}$	$r_{LV}$	$r_I$	
Books on eye anatomy													
Snell and Lemp (1997)	23.50	23.00	24.00	—	—	—	12.00	7.70	5.575*	5.85	5.30	6.00	
Crick and Khaw (2003)	—	—	—	—	—	—	—	—	5.75*	6.00	5.50	—	
Kaufman and Alm (2011)	—	—	—	—	—	—	11.7	7.80	5.575*	5.85	5.3	—	
Remington (2011)	23.50	23.00	24.00	—	—	5.70	12.00	7.80	5.75*	6.00	5.50	6.00	
Eye model literature													
Gullstrand (1909) full	—	—	—	24.385	3.60	—	—	—	7.80	—	—	—	
Gullstrand (1909) simplified	—	—	—	23.90	3.70	—	—	—	7.70	—	—	—	
Le Grand and El Hague (1980)	—	—	—	24.197	3.60	—	—	12.30	7.80	—	—	—	
Lotmar (1971)	—	—	—	24.197	3.60	—	—	12.30	7.80	—	—	—	
Kooijman (1983)	—	—	—	24.147	3.55	—	—	—	7.80	—	—	—	
Liou and Brennan (1997)	—	—	—	23.95	3.66	—	—	—	7.77	—	—	—	
Escudero-Sanz and Navarro (1999)	—	—	—	23.92	3.60	—	—	12.00	7.72	—	—	—	
Eye modelling and Application Literature													
Lefohn et al. (2003)	—	—	—	2.50	5.25	4.70	11.50	7.80	5.80	—	—	—	
Morimoto and Minica (2005)	—	—	—	3.53*	4.17	—	—	7.70	—	—	—	—	
Hua et al. (2006)	—	—	—	—	—	—	12.50	7.80	5.50	—	—	—	
Nishino and Nayar (2006)	—	—	—	2.18	—	—	—	7.80	5.50	—	—	—	
Li et al. (2007)	—	—	—	3.05	4.75*	5.70*	12.50*	7.80	6.19*	—	—	—	
Nitschke (2011)	—	—	—	2.27*	5.53*	5.70	—	7.80	5.50	—	—	6.00	
Nakazawa et al. (2015)	—	—	—	—	4.83*	—	—	7.70	6.00	—	—	—	

\* based on given values

spheres, a model that has been used by Nishino and Nayar (2006), Nakazawa and Nitschke (2012), and Itoh and Klinker (2014a). Our model is based on the eye values used in Nitschke (2011).

Given the eye anatomy shown in Figure 2.1b, the curvatures can be modelled as two overlapping spheres, the corneal sphere that approximates the anterior of the cornea and the eyeball sphere that approximates the outer layer of the sclera (Figure 2.2). The size of this sphere is determined so that its intersection with the corneal sphere coincides with the radius of the limbus  $r_L=5.5$  mm. The radius of the corneal sphere is assumed to be  $r_C=7.8$  mm. The corneal sphere is centered at **C** and the eyeball sphere at **E**, the center of rotation of the eye. The points **E**, **C** and **L** lie on the optical axis of the eye. The distance  $d_{EC}$  between **E** and **C** is 5.53 mm. The limbus is modelled as a circular intersection between the corneal and the eyeball sphere. It's distance from **C** is given as

$$d_{CL} = \sqrt{7.8^2 - 5.5^2} = 5.53 \text{ mm} \quad (2.1)$$

and the radius of the eyeball sphere is

$$r_E = \sqrt{5.5^2 + 13.33^2} = 14.42 \text{ mm.} \quad (2.2)$$

This is only a very rough approximation of the shape of the sclera. However, as the reflections of the scene on its surface are not evaluated by our methods, the actual shape is not significant for this work.

Although the described model assumes static parameters previous works, e.g., Tsukada and Kanade (2012); Wu et al. (2007), have shown that adjusting even such a simple model further improves the results of the eye-pose estimation. We follow these observations to estimate user specific model parameters as part of our Hybrid eye-pose estimation method (Chapter 5).

## 2.3 Eye-pose Estimation

Donder and Listing's law (Tweed and Vilis (1990)) state that the gaze direction is uniquely defined for each eye position. Donder's law states that the gaze direction uniquely defines the orientation of the eye, independent of preceding positions. According to Listing's law, the subset of valid eye positions can be obtained by a single rotation of the primary eye position around an axis perpendicular to the gaze direction.

Hansen and Ji (2010) define four calibrations commonly used in eye-gaze estimation:

- **Camera Calibration** determining the intrinsic parameter of the used camera(s);

- **Geometric Calibration** determining the spatial relation of the various elements of the setup, e.g., camera, LEDs, gazed surface;
- **Personal Calibration** determining the intrinsic parameters of the eye, e.g., curvature and offset between the optical and the visual axes;
- **Gaze Mapping Calibration** determining the parameters of the eye-gaze mapping function.

As the requirements of the methods vary, a fully calibrated environment that includes all four calibrations is often not necessary and only a subset is used. Some methods try to omit a calibration in favor of a more simple setup, e.g., Nakazawa et al. (2015), while others exploit scene features, e.g., saliency, for an automated calibration (Yamazoe et al. (2008); Cerf et al. (2008); Sugano et al. (2010)).

### 2.3.1 Methods

Eye-pose estimation methods aim to recover the position and orientation of the eye relative to the eye-tracking camera. Over the years a variety of methods have been developed that address different aspects, such as the visibility, appearance, and shape of the projected eye. As a result, the estimation techniques can be divided into appearance-based methods (Huang and Wechsler (1999); Wang et al. (2005); Hansen and Hansen (2006); Sugano et al. (2014); Wood et al. (2015); Zhang et al. (2015)) that use machine learning to estimate the eye-pose from the similarity to the captured image and shape-based techniques (Kim and Ramakrishna (1999); Guestrin and Eizenman (2006); Wu et al. (2007); Tsukada and Kanade (2012); Nakazawa et al. (2015)) that either look for features that support a hypothesis (voting-based) or reconstruct the used eye-model from detected features (fitting-based). A third option is a combination of both types in a hybrid solution that addresses the weaknesses of the separate methods (Xie et al. (1994)). Further division can be made into active methods that require IR illumination and passive methods that use images taken under natural illumination, or from the positioning of the eye tracking camera into headworn and remote methods. A recent review of eye-pose estimation methods can be found in Hansen and Ji (2010).

#### 2.3.1.1 Appearance-based methods

Appearance-based methods do not model the geometry of the eye, instead they assume that the eye can be detected by the distribution of the color in the captured image. As such, these methods require a large database of images to account for different gaze directions, racial differences and lighting conditions.

Appearance-based methods can be distinguished into intensity-domain methods and filter-response methods. Methods that use intensity-domain assume that the various areas around the eye, i.e., the skin color, the iris, and the sclera, will appear as areas of similar intensity in the image. [Hallinan \(1991\)](#) detects the iris as a dark region surrounded by a bright region of the sclera. Intensity variations are accounted for through statistical measures. [Samaria and Young \(1994\)](#) use Hidden Markov Models (HMDDs) to detect features in normalized images. A general problem of these methods is that they are designed for a very constrained viewing field, as the eye's appearance may change greatly under different head orientations and indecent light. To address this, [Zhang et al. \(2015\)](#) use a notebook camera to accumulate a dataset of more than 200.000 images over an extended period of time. Head-pose invariance is achieved through a head-tracking algorithm followed by a normalization of the eye region. The appearance is then learned through a multimodal convolutional neural network (CNN). [Sugano et al. \(2014\)](#) and [Wood et al. \(2015\)](#) use synthesis to create simulated views of the eye and account for various gaze directions, scale, illumination and racial appearances.

By filtering the input image it is possible to suppress some features, while others are enhanced at the same time. [Huang and Wechsler \(1999\)](#) use a two-staged approach to detect the eye. They represent images as a wavelet in a Radial basis Function classifier and report better localization than intensity-based methods. The fine-alignment is acquired from the contour and region information. Another common approach is to use cascades of Haar-Features ([Hansen and Hansen \(2006\)](#); [Wang et al. \(2005\)](#)).

### 2.3.1.2 Shape-based methods

Contrary to appearance-based methods, shape-based methods try to find distinct features of the eye in the image. Simple methods only model the iris or the pupil as a circular surface that projects onto an ellipse in the camera image ([Kim and Ramakrishna \(1999\)](#); [Itoh and Klinker \(2014a\)](#)). More sophisticated methods also include other features, e.g., the eye lids and the eye corners ([Wu et al. \(2007\)](#); [Nakazawa et al. \(2015\)](#)). Furthermore, many models assume static model parameters, while some use a deformable model to increase the robustness. Higher complexity in general provides better results, however, the more complicated the model, the more computationally expensive the detection becomes.

**Eye features.** Voting methods use the detected features to find support of a given hypothesis. From a variety of hypotheses the one with the highest support is then selected as the detected result. [Kim and Ramakrishna \(1999\)](#) use thresholds in the image to detect the center of the pupil ellipse. The contours of the pupil and the limbus are detected by edge detection. A common approach is to use a circularity constraint to detect the contours by

the Hough transform. However, this limits the applicability of such functions to near-frontal images.

Contrary to the voting methods, model fitting methods assume that the features used in the eye model can be detected robustly in the image, regardless of orientation and without a prior guess.

[Li et al. \(2005\)](#) introduce the *Starburst* algorithm to detect the iris, modelled as an ellipse. Their algorithm detects maxima in the intensity changes along radial rays cast from an initial pose guess. Further rays are cast from the detected maxima. The ellipse is then detected through a random sample consensus (RANSAC) approach.

[Wu et al. \(2007\)](#) suggest a model that includes tracking of both eyes through a particle filter. They use an extensive adaptive model that includes the eye corners, eye lids and the iris contour. Although the model can adjust for a person specific iris size, the method has to track as many as 7 parameters for each frame.

[Tsukada and Kanade \(2012\)](#) propose to model the eye at runtime, however their method assumes that the eye remains static relative to the camera, thus it is sufficient to estimate the contour of the iris. As the number of observations increases, the estimated iris contours and centers are fitted into a consistent guess.

[Pires et al. \(2013a\)](#) extend the results of [Tsukada and Kanade \(2012\)](#) and model the projection of the eye into the camera as a sphere limited by the eye corners. By unwrapping the projection they effectively reduce the number of parameters required to estimate the iris contour to 3 (longitude, latitude, radius). However, as the unwarping has to be learned ahead of application [Tsukada and Kanade \(2012\)](#) and [Pires et al. \(2013a\)](#) require multiple training samples and are restricted to headworn trackers.

[Nakazawa et al. \(2015\)](#) use the reflection of two LEDs attached above and below the camera to estimate the ray towards the cornea center. They use this information to improve the robustness of the ellipse fitting. They use a particle filter to track the iris contour, eye lids and eye corners in consecutive frames and eliminate candidates that disagree with the guess of the cornea center.

**Scene reflection.** Model-based techniques use the detected features, such as eye corners, eye lids and the iris and pupil contour to recover the eye-pose. Current SOTA methods, e.g., [Tobii Technology AB \(2016b\)](#); [Sensor Instruments \(2015\)](#), are based on the Pupil Center Cornea Reflection (PCCR) method ([Guestrin and Eizenman \(2006\)](#)). PCCR detects the reflection of known, accurately calibrated light sources on the cornea as highlights. The combination of the detected highlights and the known origin is then used to accurately estimate the position of the corneal sphere. The result is an accurately estimated position of the corneal sphere. [Villanueva and Cabeza \(2008\)](#) also suggest that the method can be used to estimate the radius of

the user's corneal sphere, thus further improving the results. The orientation of the eye is recovered through reconstruction of the pupil from known or approximated refraction indices of the eye.

### 2.3.1.3 Pupil contour vs. Iris contour detection

Stiefelhagen et al. (1997b,a); Kim and Ramakrishna (1999) have proposed several methods that include thresholding to detect the contour of the pupil under natural illumination, however overall it's detectability remains unreliable, especially in the presence of highlights. To bypass this, passive methods commonly use the iris contour as a more reliable feature. Although detection of the iris contour provides a number of problems, such as

1. gradual transition into the sclera,
2. partial occlusion by eyelids and eyelashes,
3. refraction on the cornea under large viewing angles,
4. varying personal size of the iris and limbus, and
5. ambiguous reconstructed pose

various dedicated algorithms have been developed to address the problems of the detection. The 2-way ambiguity of the 3D origin can be addressed by various assumptions, such as a constrained orientation, or multiple observations if the center of the eye rotation remains fixed. Although the pupil contour is used in methods that work under IR illumination, using its projection poses a more difficult problem than the iris contour.

**Radius.** Depending on the amount of incoming light the pupil's size varies between 1-8mm. Therefore, reconstruction of the elliptical shape results in a conic with an infinite number of possible positions and gaze directions. To resolve this the reflection of at least one known and beforehand calibrated light source is necessary (Villanueva and Cabeza (2007)).

**Refraction.** As the pupil is located right in front of the pupil it's projection undergoes a refraction when the light ray passes from the aqueous humor into the cornea and from the cornea into the air. As this distortion is position and orientation dependent it has to be computed for each point of the ellipse separately. The refraction has been shown to account for  $>1^\circ$  error in the estimated gaze direction (Villanueva and Cabeza (2007)). The pupil can be recovered by estimating the position of the corneal sphere from at least two known light sources (Guestrin and Eizenman (2006); Villanueva and Cabeza (2008)). After the position of the corneal sphere has been estimated, refraction parameters estimated in clinical studies can be combined with the used eye model are used to reconstruct the actual shape of the pupil.

**Shape and location.** Although the pupil is bound by the iris, its shape is not centered along the optical axis and shows decreasing circularity with age (Wyatt (1995); Atchison and Smith (2000); Rakshit and Monro (2007)). Furthermore, the best-fit ellipse describes only up to half of the deformation. An accurate representation can be recovered through circular Fourier series with significant contribution to the shape by the first four to five harmonics. The location and shape of the cornea greatly changes from person to person and is further modified by the amount of incoming light, as the pupil expands or shrinks due to the pupillary light reflex.

Robust pupil extraction and eye-pose estimation requires not only sophisticated algorithms but also IR illumination. However, due to limited range of detectable light by the IR cameras, 780-880 nm, these cameras cannot detect the reflection of content shown on the HMD-screen or the observed scene. Thus we focus on the estimation of the eye-pose from images taken under natural illumination.

### 2.3.2 Pupil Center Corneal Reflection (PCCR)

The estimation of the eye-pose from the detected iris contour is unreliable as occlusion, contrast, highlights, and static model parameters often lead to an incorrect estimation. SOTA systems in eye-pose estimation use corneal reflections of infra-red LEDs, detected as glints, to estimate an accurate position of the eye and the contour of the pupil to determine the orientation of the eye. By modelling the eye as a sphere, the eye-camera system can be described as an off-axial catadioptric system (Nishino and Nayar (2004b)). Computations in such systems are well studied with a long history of theory and applications (Baker and Nayar (1999); Ying and Hu (2004); Lhuillier (2008); Agrawal and Ramalingam (2013); Agrawal (2013)).

The reflection of IR-LEDs can be detected by an IR-light camera as strong highlights on a darker background, in particular the pupil. This allows robust and fast detection and matching of the highlights with the source. In images taken under natural illumination, correspondences can be detected in contrast rich scenes, e.g., a user looking at a checkerboard.

The corneal sphere  $C$  located at  $\mathbf{C}$  can be recovered from  $n \geq 2$  correspondences  $\{\mathbf{p}, \mathbf{P}\}$ , where  $\mathbf{p}$  is a pixel in the image captured by the camera and  $\mathbf{P}$  the corresponding 3D point. This process consists of two steps, first an estimation of the ray  $\mathbf{r}$  from  $\mathbf{T}$ , the position of eye-tracking camera, towards  $\mathbf{C}$ , followed by the estimation of the distance  $d_{\mathbf{TC}}$  along  $\mathbf{r}$ .

**Direction Estimation.** Let  $\mathbf{b}$  be the backprojected ray through the pixel  $\mathbf{p}$ . Then, according to Snell's law,  $\mathbf{b}$  and  $\mathbf{P}$  lie in a plane  $\pi$ . The normal  $\mathbf{n}$  of  $\pi$  is given as  $\mathbf{n} = \mathbf{b} \times \mathbf{v}$ , where  $\mathbf{v}$  is the ray from  $\mathbf{T}$  towards  $\mathbf{P}$ . As shown in Figure 2.4, this plane also contains  $\mathbf{T}$ ,  $\mathbf{C}$ , as well as  $\mathbf{R}$ , the reflection point of  $\mathbf{P}$  on  $C$ . As such, for two planes  $\pi_1$  and  $\pi_2$  the ray  $\mathbf{r}$  from  $\mathbf{T}$  towards  $\mathbf{C}$  can

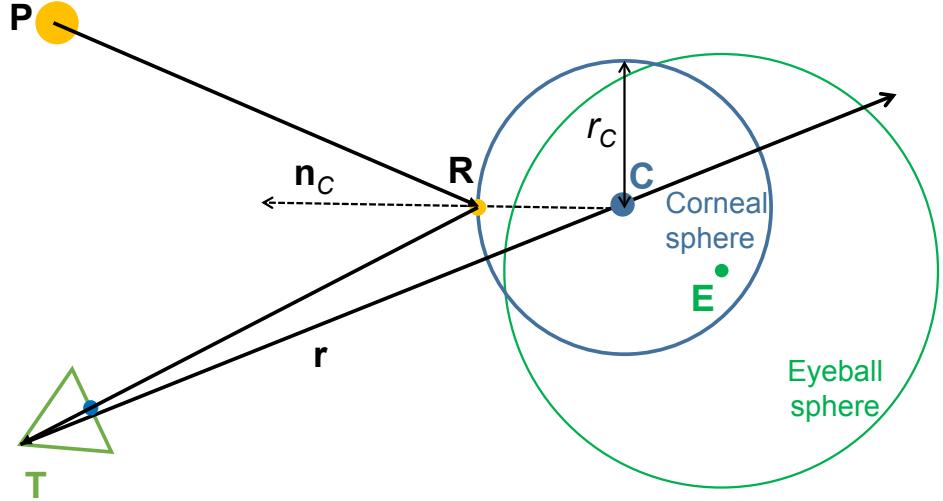


Figure 2.4: The point  $\mathbf{P}$  reflects on the cornea  $\mathbf{C}$  in  $\mathbf{R}$  and intersects the camera located at  $\mathbf{T}$ . According to Snell's law  $\mathbf{P}$ ,  $\mathbf{R}$ ,  $\mathbf{T}$  and  $\mathbf{C}$  lie in the same plane.

be determined as  $\mathbf{r} = \mathbf{n}_1 \times \mathbf{n}_2$ , the intersection of the planes  $\pi_1$ ,  $\pi_2$ , because  $\mathbf{T}$  and  $\mathbf{C}$  are contained in both planes

For more than two correspondences  $\mathbf{r}$  can be recovered through an algebraic approach. For  $n$  correspondences, let

$$\mathbf{A} = \begin{bmatrix} \hat{\mathbf{n}}_1^T \\ \hat{\mathbf{n}}_2^T \\ \vdots \\ \hat{\mathbf{n}}_n^T \end{bmatrix}, \quad (2.3)$$

thus  $\mathbf{A}\mathbf{r} = \mathbf{0}$ . Under the constraint  $\|\mathbf{r}\| = 1$ ,  $\mathbf{r}$  will correspond to the eigenvector of  $\mathbf{A}$  with the smallest eigenvalue. Singular value decomposition (SVD) of  $\mathbf{A}$  results in three matrices  $\mathbf{U}$ ,  $\mathbf{D}$  and  $\mathbf{V}$  with  $\mathbf{A} = \mathbf{U}\mathbf{D}\mathbf{V}^T$ , where  $\hat{\mathbf{r}}$  is the last column of  $\mathbf{V}$ .

If a high number of correspondences is recovered from the corneal image, the matches are likely to contain a number of outliers. Therefore, RANSAC has to be used to improve the robustness of the estimation. This approach can be used if at least four correspondences have been detected.

Given two correspondence pairs  $\{\mathbf{p}_i, \mathbf{P}_i\}$  and  $\{\mathbf{p}_j, \mathbf{P}_j\}$  that describe two non-parallel planes  $\pi_i$  and  $\pi_j$ , the ray  $\mathbf{r}_{ij}$  is obtained as the intersection of  $\pi_i$  and  $\pi_j$ . In general, an erroneous correspondence pair  $\{\mathbf{p}_k, \mathbf{P}_k\}$ , for example due to false matching or measurement errors, will describe a plane  $\pi_k$  whose normal will not be perpendicular to  $\mathbf{r}$ . If  $\{\mathbf{p}_i, \mathbf{P}_i\}$  and  $\{\mathbf{p}_j, \mathbf{P}_j\}$  are correct correspondences, meaning they are inliers of  $\mathbf{r}$ , the ray  $\mathbf{r}_{ij}$  will be perpendicular to the normals of the majority of the planes described by the correspondences

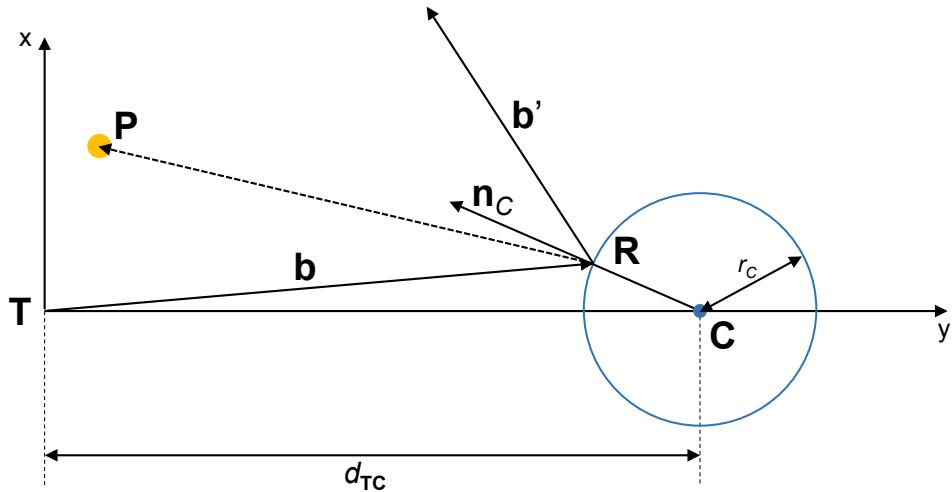


Figure 2.5: The coordinate system of a plane  $\pi$  can be uniquely defined, so that  $\mathbf{r}$  coincides with its y-axis. For a cornea of radius  $r_C$  located  $d_{TC}$  away from  $\mathbf{T}$  the ray  $\mathbf{b}$  will reflect as  $\mathbf{b}'$  on the intersection point  $\mathbf{R}$ . If  $d_{TC}$  is estimated correctly  $\mathbf{b}'$  will intersect with the point  $\mathbf{P}$ .

in the current frame. Inlier correspondence-pairs can be determined with

$$|(\hat{\mathbf{r}}_{ij})^T \hat{\mathbf{n}}_k| \begin{cases} \leq t & \text{if } \{\mathbf{p}_k, \mathbf{P}_k\} \text{ is an inlier,} \\ > t & \text{if } \{\mathbf{p}_k, \mathbf{P}_k\} \text{ is an outlier.} \end{cases} \quad (2.4)$$

We use an empirically estimated threshold  $t=0.0001$  to account for noise and extract the largest subset of  $m$  inliers from the original  $n$  correspondences. For the  $m$  inlier planes the algebraic approach is then used to determine the ray towards  $\mathbf{C}$ .

**Distance estimation.** Given the ray  $\mathbf{r}$ , the distance  $d_{TC}^i$  from the camera center  $\mathbf{T}$  to the cornea center  $\mathbf{C}$  along  $\mathbf{r}$  that is supported by the  $i$ th inlier correspondence pair  $\{\mathbf{p}_i, \mathbf{P}_i\}$  can be computed within  $\pi_i$ . Figure 2.5 shows the geometric relationship within  $\pi_i$ . The ray  $\mathbf{b}$  intersects the corneal sphere in the point  $\mathbf{R}_i$ , where  $\mathbf{P}_i$  reflects at the corneal sphere and projects into the camera. The position of  $\mathbf{R}_i$  is still unknown, as its position varies for different distances  $d_{TC}^i$  and with the radius  $r_C$  of the cornea.

Let the origin of the coordinate-system of the plane  $\pi$  coincide with  $\mathbf{T}$  and the rotation  ${}^T_\pi \mathbf{R}$  convert points on  $\pi$  into the coordinate system of  $T$ . W.l.o.g. assume that  $\pi$  coincides with the  $xy$ -plane, thus  $z=0$ , so that  $\mathbf{r}$  is along the  $y$  axis. As the  $z$ -axis is given by the normal of the plane, the  $x$  axis can be determined as the cross-product of the  $x$  and  $y$  axes. As such, the mapping from  $T$  to  $\pi$  is uniquely defined.

The points  $\mathbf{P}$  and  $\mathbf{C}$  thus transform to  ${}^\pi \mathbf{C} = {}^T_\pi \mathbf{R} \mathbf{C} = (0 \ d_{TC} \ 0)^T$  and  ${}^\pi \mathbf{P} = {}^T_\pi \mathbf{R} \mathbf{P} = (x \ y \ 0)^T$ . Additionally, the corneal sphere maps on a circle of

radius  $r_C$  around  ${}^\pi\mathbf{C}$  and the vector  $\mathbf{b}$  maps onto a vector  ${}^\pi\mathbf{b} = {}_T^\pi\mathbf{R}\mathbf{b} = (u \ v)^T$ . As it is a directional vector, it can be normalized further enforced that  $\|{}^\pi\mathbf{b}\| = 1$ . The reflection point  ${}^\pi\mathbf{R} = d_{\mathbf{TR}}{}^\pi\mathbf{b}$  lies on the surface of the cornea, thus  $\|{}^\pi\mathbf{R} - {}^\pi\mathbf{C}\| = r_C$ . This equation can be reformulated as

$$\|{}^\pi\mathbf{R} - {}^\pi\mathbf{C}\|^2 = (d_{\mathbf{TR}}u)^2 + (d_{\mathbf{TR}}v - d_{\mathbf{TC}})^2 = d_{\mathbf{TR}}^2 - 2d_{\mathbf{TR}}vd_{\mathbf{TC}} + d_{\mathbf{TC}}^2 = r_C^2. \quad (2.5)$$

The normal  ${}^\pi\mathbf{n}_C$  at point  ${}^\pi\mathbf{R}$  is given as  ${}^\pi\mathbf{n}_C = {}^\pi\mathbf{R} - {}^\pi\mathbf{C}$  and the reflection of the ray  ${}^\pi\mathbf{b}$  is according to Snell's law

$${}^\pi\mathbf{b}' = {}^\pi\mathbf{b} - 2{}^\pi\hat{\mathbf{n}}_C({}^\pi\hat{\mathbf{n}}_C^T{}^\pi\mathbf{b}) = \frac{{}^\pi\mathbf{b} - 2{}^\pi\mathbf{n}_C({}^\pi\mathbf{n}_C^T{}^\pi\mathbf{b})}{r_C^2}. \quad (2.6)$$

As the reflected ray  ${}^\pi\mathbf{b}'$  should intersect with  ${}^\pi\mathbf{P}$  it can be expressed as

$${}^\pi\mathbf{b}' \times ({}^\pi\mathbf{P} - {}^\pi\mathbf{R}) = 0. \quad (2.7)$$

By substituting all variables in Equation (2.7) and enforcing the restriction of  $u^2 + v^2 = 1$ , the equation can be written as

$$K_1d_{\mathbf{RT}}^2 + K_2d_{\mathbf{RT}} + K_3 = 0, \quad (2.8)$$

where  $K_1 = 2(du - uy + vx)$ ,  $K_2 = -2d_{\mathbf{RC}}(x + xv^2 + d_{\mathbf{TC}}uv - vxy)$ , and  $K_3 = 2d_{\mathbf{TC}}^2vx + r_C^2uy - r_C^2vx$ . Combining Equations (2.5) and (2.8) results in

$$a_6d_{\mathbf{TC}}^6 + a_5d_{\mathbf{TC}}^5 + a_4d_{\mathbf{TC}}^4 + a_3d_{\mathbf{TC}}^3 + a_2d_{\mathbf{TC}}^2 + a_1d_{\mathbf{TC}} + a_0 = 0 \quad (2.9)$$

with the coefficients being

$$\begin{aligned} a_6 &= -4u^2(v^2 - 1), \\ a_5 &= 8yu^2(v^2 - 1), \\ a_4 &= 4(x^2v^4 - y^2u^2v^2 + r_C^2u^2v^2 - 2x^2v^2 + y^2u^2 - 2r_C^2u^2 + x^2), \\ a_3 &= 4r_C^2u(xv^3 - yuv^2 - xv + 3yu), \\ a_2 &= 4r_C^2(x^2v^2 - y^2u^2 + r_C^2u^2 - x^2), \\ a_1 &= 4r_C^4u(xv - yx), \text{ and} \\ a_0 &= r_C^4(xv - yu)^2. \end{aligned}$$

Solving this equation will result in 6 solutions — two complex, two real negative and two real positive. The negative and complex solutions can be discarded, as the eye is located in front of the camera. Therefore, for each correspondence pair, only two possible solutions remain for  $d_{\mathbf{TC}}$ . The resulting values for all correspondences will create a cluster around the correct position and scatter the remaining estimations, some will be closer to the camera while others will be further away. Thus the ambiguity can be resolved by using the median value. Alternatively, the results from 1 sample can be compared to all

others to find the best cluster. For  $n$  correspondences, the position is further refined by solving

$$\tilde{\mathbf{C}} = \arg \min_{\mathbf{C}} \sum_{i=1 \dots m} \hat{\mathbf{u}}_i \times \hat{\mathbf{w}}_i. \quad (2.10)$$

[Agrawal and Ramalingam \(2013\)](#) note that by combining two observations, Equation (2.9) can be reformulated as a 7th degree polynomial in  $r_C$ . In our tests, we found the results to be highly dependend on accurate estimation of  $\mathbf{p}$  and  $\mathbf{P}$ . Furthermore, as our model naturally contains small inaccuracies because of the simple representation of the eye the results of this estimation depended on what feature pairs were used in the calculation. Therefore, in this dissertation we use the known  $r_C=7.8$  mm for the cornea size.

### 2.3.2.1 Pupil detection

The pupil contour is mostly used in combination with IR illuminated environment ([Guestrin and Eizenman \(2006\)](#); [Villanueva and Cabeza \(2007, 2008\)](#)) where a combination of co-axial and off-axial IR-LEDs generate a bright or a dark highlight on the cornea.

### 2.3.3 Limbus Reconstruction

The limbus varies from person-to-person with different horizontal and vertical radii. Generally, eye-pose estimation and tracking methods assume it to be a circular shape of a static or estimated radius ([Wu et al. \(2007\)](#); [Tsukada and Kanade \(2012\)](#); [Nakazawa and Nitschke \(2012\)](#)). Only few methods either approximate person specific model parameters ([Wu et al. \(2007\)](#); [Tsukada and Kanade \(2012\)](#)) or assume a non-spherical iris ([Nishino and Nayar \(2006\)](#)). The limbus is assumed to coincide with the iris contour because the visible part of the iris is bound by the intersection of the cornea and the sclera, the limbus, and their anatomical proximity. In the following we describe the reconstruction of the limbus from the detected iris contour.

**Ellipse equation.** W.l.o.g. let the ellipse lie in the xy-plane. In this case the boundary of the ellipse is determined by five parameters, the center  $\mathbf{c}$ , its radii  $a, b$  as well as  $\alpha$ , a rotation angle around the z-axis. A point  $\mathbf{p}$  lies on the ellipse if it satisfies

$$A\mathbf{p}_x^2 + 2B\mathbf{p}_x\mathbf{p}_y + 2C\mathbf{p}_y^2 + D\mathbf{p}_x^2 + 2E\mathbf{p}_y + F = 0. \quad (2.11)$$

A detailed explanation on how to derive this equation can be found in [Hartley and Zisserman \(2003\)](#). Equation (2.11) can be reformulated as a matrix multiplication  $\mathbf{p}^T \mathbf{Q} \mathbf{p} = 0$ , where  $\mathbf{p}$  is in homogeneous coordinates and

$$\mathbf{Q} = \begin{bmatrix} A & B & C \\ B & D & E \\ C & E & F \end{bmatrix} \quad (2.12)$$

is a symmetric matrix with at least one element out of  $A$ ,  $B$ , and  $D$  not being equal to zero and  $B^2 - AD < 0$ .

The circle is a special case of the ellipse, where the radii are identical and the rotation angle can be ignored. In this case  $\mathbf{Q}$  is further constrained by  $B=0$  and  $A=D \neq 0$ . Hereby, the origin of the circle is  $\mathbf{c} = (-C - E)^T$  and the radius is  $r_C = \sqrt{F + C^2 + E^2}$ .

**Perspective projection formulation.** The circular limbus  $L$  located at  $\mathbf{L}$  and oriented along the optical axis  $\mathbf{o}$  projects onto an ellipse in the camera image (Hartley and Zisserman (2003)). Although it is possible to recover an approximate pose of the limbus through a weak-perspective reconstruction of the limbus (Nitschke et al. (2009)), the intrinsic parameters of the recording camera are required for an accurate estimation. These parameters can be recovered either through a dedicated calibration, e.g., Zhang (2000), or be estimated from the distortion of the iris contour (Johnson and Farid (2007)).

W.l.o.g. let the origin of the limbus be  ${}^L\mathbf{L} = (0 \ 0 \ 0)^T$  and  ${}^L\mathbf{o} = (0 \ 0 \ 1)^T$ . In this case, all points  ${}^L\mathbf{p} = (x \ y \ 0)^T$  that lie on the contour of the limbus with a radius  $r_L$  satisfy

$$\mathbf{p}^T \mathbf{Q} \mathbf{p} = 0, \quad (2.13)$$

where  $\mathbf{Q}$  is a symmetrical matrix of the form

$$\mathbf{Q} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -r_L^2 \end{bmatrix} \quad (2.14)$$

and  $\mathbf{p} = (x \ y \ 1)^T$  is the homogeneous representation of  ${}^L\mathbf{p}$ .

W.l.o.g. let the image plane be located at the depth  ${}^C\mathbf{L}_z$  along the optical axis of the camera. A point  ${}^C\mathbf{P}$  on this plane maps to homogeneous pixels in the camera image  $I$  as

$${}^I\mathbf{p} = \frac{1}{{}^C\mathbf{L}_z} \mathbf{K} {}^C\mathbf{P}. \quad (2.15)$$

Furthermore, the mapping of homogeneously represented points on the limbus plane onto the image plane can be described by a  $3 \times 3$  matrix  ${}^C\mathbf{T} = [\mathbf{R}_1 \ \mathbf{R}_2 \ {}^C\mathbf{L}]$ , where  $\mathbf{R}$  is a rotation matrix and  ${}^C\mathbf{L}$  is the position of  $\mathbf{L}$  relative to  $C$ . Note that the third column of the rotation matrix  $\mathbf{R}$  can be omitted as the limbus plane corresponds to the xy-plane (Bradski and Kaehler (2008)).

The projection of the limbus onto the image plane describes a general conic  $\mathbf{Q}_C$  as in Equation (2.12).

The conversion between  $\mathbf{Q}$  and  $\mathbf{Q}_C$  can be derived as

$${}^I\mathbf{p}^T \mathbf{Q}_C {}^I\mathbf{p} = 0 \quad (2.16)$$

$$\left(\frac{1}{C}\mathbf{L}_z^C\mathbf{P}\right)^T\mathbf{Q}_C\left(\frac{1}{C}\mathbf{L}_z^C\mathbf{P}\right) = 0 \quad (2.17)$$

$$(\mathbf{K}_L^C\mathbf{T}\mathbf{p})^T\mathbf{Q}_C(\mathbf{K}_L^C\mathbf{T}\mathbf{p}) = 0 \quad (2.18)$$

$$\mathbf{p}^T \underbrace{(\mathbf{K}_L^C\mathbf{T})^T\mathbf{Q}_C(\mathbf{K}_L^C\mathbf{T})}_{\mathbf{Q}}\mathbf{p} = 0. \quad (2.19)$$

If  $\mathbf{Q}$  is known, the conic  $\mathbf{Q}_C$  can be computed as

$$\mathbf{Q} = (\mathbf{K}_L^C\mathbf{T})^T\mathbf{Q}_C(\mathbf{K}_L^C\mathbf{T}) \mid (\mathbf{K}_L^C\mathbf{T})^{-T} \quad (2.20)$$

$$(\mathbf{K}_L^C\mathbf{T})^{-T}\mathbf{Q} = \mathbf{Q}_C(\mathbf{K}_L^C\mathbf{T}) \mid (\mathbf{K}_L^C\mathbf{T})^{-1} \quad (2.21)$$

$$(\mathbf{K}_L^C\mathbf{T})^{-T}\mathbf{Q}(\mathbf{K}_L^C\mathbf{T})^{-1} = \mathbf{Q}_C. \quad (2.22)$$

As  $C\mathbf{L}_z$  can be removed from the equation, the mapping between the conic in the camera image plane and the limbus plane is independent of the actual depth. As such,  $\mathbf{Q}_C$  describes a cone  $\mathbf{C}_C$ . From Equation (2.19) it follows that

$$\mathbf{C}_C = \mathbf{K}^T\mathbf{Q}_C\mathbf{K}. \quad (2.23)$$

Now observe a second camera  $T$  that has the same intrinsic parameters as  $C$  and is located at the same position as  $C$ .  $T$  is oriented so that its optical axis coincides with  $^L\mathbf{o}$ . It follows that  $^T\mathbf{T} = [\mathbf{l}_1 \ \mathbf{l}_2 \ \mathbf{t}]^T$ , where  $\mathbf{l}_1$  and  $\mathbf{l}_2$  are the first two columns of a  $3 \times 3$  matrix. In the image captured by  $T$  the limbus projects onto a circle described by conic  $\mathbf{Q}_T$  of the form

$$\mathbf{Q}_T = \begin{bmatrix} A & 0 & D \\ 0 & A & E \\ D & E & F. \end{bmatrix} \quad (2.24)$$

The conic  $\mathbf{Q}_T$  describes a cone  $\mathbf{C}_T = \mathbf{K}^T\mathbf{Q}_T\mathbf{K}$ . By substituting all parameters it follows that  $\mathbf{C}_T$  has the form

$$\mathbf{C}_T = \begin{bmatrix} \mathbf{l}_{2 \times 2} & -\mathbf{l}_T \\ -\mathbf{l}_T^T & \mathbf{l}_T^T\mathbf{l}_T - r_T^2, \end{bmatrix} \quad (2.25)$$

where

$$\begin{aligned} \mathbf{l}_T &= \begin{pmatrix} \frac{\mathbf{t}_x}{\mathbf{t}_z} \\ \frac{\mathbf{t}_y}{\mathbf{t}_z} \end{pmatrix}, \text{ and} \\ r_T &= \frac{r_L}{\mathbf{t}_z}. \end{aligned} \quad (2.26)$$

It follows that, if  $\mathbf{C}_T$  and  $r_T$  is known, all parameters of  $\mathbf{t}$  can be reconstructed.

The cones  $\mathbf{C}_T$  and  $\mathbf{C}_C$  describe the same cone in space, therefore the transformation from  $\mathbf{C}_T$  to  $\mathbf{C}_C$  is given by the rotation  $\mathbf{R}$  as

$$\mathbf{C}_C = \mathbf{R}\mathbf{C}_T\mathbf{R}^T. \quad (2.27)$$

In the following we describe how the rotation matrix  $\mathbf{R}$  can be obtained.

$\mathbf{C}_C$  is a symmetrical matrix, thus it can be decomposed into  $\mathbf{C}_C = \mathbf{E}\mathbf{D}\mathbf{E}^T$ . Hereby  $\mathbf{E}$  is an orthogonal matrix whose columns are the normalized eigenvectors of  $\mathbf{C}_C$  and  $\mathbf{D}$  is a diagonal matrix of the eigenvalues  $\lambda_1$ ,  $\lambda_2$ , and  $\lambda_3$  of  $\mathbf{C}_C$ .

From equation (2.27) it follows that

$$\mathbf{E}\mathbf{D}\mathbf{E}^T = \mathbf{R}\mathbf{C}_T\mathbf{R}^T \quad (2.28)$$

$$\mathbf{R}^T\mathbf{E}\mathbf{D}\mathbf{E}^T = \mathbf{C}_T\mathbf{R}^T \quad (2.29)$$

$$\mathbf{R}^T\mathbf{E}\mathbf{D}\mathbf{E}^T\mathbf{R} = \mathbf{C}_T \quad (2.30)$$

$$(\mathbf{E}^T\mathbf{R})^T\mathbf{D}(\mathbf{E}^T\mathbf{R}) = \mathbf{C}_T. \quad (2.31)$$

Chen et al. (2004) assume that the elements of  $\mathbf{D}$  satisfy

$$\lambda_1\lambda_2 > 0, \lambda_1\lambda_3 > 0, \text{ and } |\lambda_1| \geq |\lambda_2|. \quad (2.32)$$

This assumption can be satisfied by rearranging the order of the eigenvalues and corresponding eigenelements. They derive a solution for  $\mathbf{E}^T\mathbf{R}$  as

$$\mathbf{E}^T\mathbf{R} = \begin{bmatrix} g \cos \alpha & S_1 g \sin \alpha & S_2 h \\ \sin \alpha & -S_1 \cos \alpha & 0 \\ S_1 S_2 h \cos \alpha & S_2 \sin \alpha & -S_1 g \end{bmatrix}, \quad (2.33)$$

where

$$\begin{aligned} g &= \sqrt{\frac{\lambda_2 - \lambda_3}{\lambda_1 - \lambda_3}}, \\ h &= \sqrt{\frac{\lambda_1 - \lambda_2}{\lambda_1 - \lambda_2}}, \end{aligned} \quad (2.34)$$

$S_1$ ,  $S_2$  are undetermined signs, and  $\alpha$  is the rotation of the limbus around the normal of its plane. As the limbus is a circle  $\alpha$  remains a free parameter. From Equations (2.33) and (2.25) the parameters of the cone  $\mathbf{C}_T$  can be computed as

$$\begin{aligned} \mathbf{l}_T &= \begin{pmatrix} -S_2 \frac{\sqrt{(\lambda_1 - \lambda_2)(\lambda_2 - \lambda_3)} \cos \alpha}{\lambda_2} \\ -S_1 \frac{\sqrt{(\lambda_1 - \lambda_2)(\lambda_2 - \lambda_3)} \sin \alpha}{\lambda_2} \end{pmatrix}, \text{ and} \\ r_T &= S_3 \frac{\sqrt{\lambda_1 \lambda_3}}{\lambda_2}. \end{aligned} \quad (2.35)$$

Here,  $S_3$  is another undetermined sign. As the radius of the limbus  $r_L$  is known it is possible to determine

$$\mathbf{t}_z = S_3 \frac{\lambda_2 r_L}{\sqrt{-\lambda_1 \lambda_3}} \quad (2.36)$$

and therefore

$${}^T \mathbf{L} = \begin{pmatrix} \mathbf{t}_z \mathbf{l}_T \\ \mathbf{t}_z \cdot \end{pmatrix} \quad (2.37)$$

A point  ${}^T \mathbf{P}$  is converted into the coordinate system of  $C$  through  ${}^C \mathbf{P} = \mathbf{R} {}^T \mathbf{P}$ . As  $\mathbf{E}$  is an orthogonal matrix it is possible to formulate  $\mathbf{R} = \mathbf{E}(\mathbf{E}^T \mathbf{R})$ , leading to

$${}^C \mathbf{L} = \mathbf{E}(\mathbf{E}^T \mathbf{R}) {}^T \mathbf{L} = \mathbf{E} \begin{pmatrix} S_2 h \frac{\lambda^3}{\lambda^2} \\ 0 \\ -S_1 g \frac{\lambda^3}{\lambda^2} \end{pmatrix}. \quad (2.38)$$

Applying this transformation to  ${}^T \mathbf{o} = {}^L \mathbf{o}$  results in the gaze direction

$${}^C \mathbf{o} = \mathbf{E}(\mathbf{E}^T \mathbf{R}) {}^T \mathbf{o} = \mathbf{E} \begin{pmatrix} S_2 h \\ 0 \\ -S_1 g \end{pmatrix}. \quad (2.39)$$

The three unknown signs  $S_1$ ,  $S_2$ , and  $S_3$  lead to eight possible solutions. By enforcing that the limbus is in front of the camera and that the gaze direction is towards the camera it is possible to resolve the sign of  $S_1$  and  $S_3$ . This results in a 2-way ambiguity of the computed eye-pose. The last unknown parameter has to be resolved manually or through further constraints, e.g., multiple observations or a known general gaze direction, are necessary.

## 2.4 Point-of-Regard Estimation

The POR is the estimated viewing point of the user. It may be as limited as a single point or describe larger areas, such as a text block or virtual object. In general, the POR can be estimated through two approaches, an association of the estimated eye-pose or eye features with a distinctive POR, or a geometric estimation of the intersection of the estimated gaze direction with the scene-model.

### 2.4.1 Regression-based

Regression-based methods assume that there exists a static association of the eye's appearance in the camera image and the gaze direction (Merchant et al. (1974); Morimoto et al. (2000); Zhu and Ji (2007); Tsukada and Kanade (2012); Pires et al. (2013a,b)). Generalization is acquired through linear mapping, higher order polynomials or neuronal networks. Although these systems report high accuracy, their applicability is limited to static environments, where the head is either fixated by a chinrest or bite-bar, or a head-mount with the camera attached to it is used. Further limitations are the number of

required training data to achieve accurate mapping results and the required continuous recalibration. [Kolakowski and Pelz \(2006\)](#) propose a set of heuristic rules to detect slips of the headworn device and adjust correspondingly. [Pires et al. \(2013b\)](#) track the corners of the eye through template matching and model the movement of the camera as a translational shift and apply this to their learned model.

### 2.4.2 Geometric

Commonly the geometric eye-pose is used to cast a ray that intersects either, a planar surface located at a given distance in front of the user and assumed to be the scene-model, or a 3D scene-model. PCCR methods that model the scene at a predefined distance often suffer from parallax issues, where the accuracy degrades if the actual and assumed gaze planes differ. A 3D scene-model can be acquired through a number of methods, e.g., user calibration, manual reconstruction, single camera Structure-from-Motion, multi-camera stereo, or KinectFusion ([Newcombe et al. \(2011a,b\)](#)).

[Nakazawa and Nitschke \(2012\)](#) have proposed a new method that uses the reconstructed eye-pose to estimate the gaze-reflection point (GRP). The original method required IR illumination, but has since been extended to images taken under natural illumination ([Nitschke et al. \(2013a\)](#)). Through analysis of the corneal reflection around the GRP it is possible to either match the gaze point with the scene-model ([Takemura et al. \(2014a\)](#)) or process the information directly ([Nakazawa et al. \(2015\)](#)). Although it has been shown that estimation of the GRP performs robustly to parallax issues and achieves high accuracy in arbitrary environments, it requires an accurately estimated eye-pose.



# CHAPTER 3

## Fundamentals of OST-AR

---

In this chapter we review the existing calibration methods for spatial and viewpoint OST-HMD calibration and the impact of rendering errors on user perception.

Section 3.1 reviews the manual calibration of an OST-HMD, in particular we introduce the current SOTA method SPAAM and discuss its drawbacks. Section 3.2 explains the spatial calibration of the OST-HMD and how the HMD-screen can be modelled as a plane. Section 3.3 introduces automated OST-HMD calibration, in particular the INDICA Full and Recycle calibration methods. We conclude with a review of the literature on user perception in Section 3.4.

### 3.1 Manual OST-HMD Calibration

If the HMD is positioned rigidly on the users head, the eye-HMD-screen system can be modelled as a pinhole camera  $E$ , where the HMD-screen corresponds to the camera's image plane (Figure 3.1). Although the calibration process of  $E$  resembles the calibration of an off-the-shelf camera or a stereo camera setup, it differs in the following points

- the image plane of the screen is assumed to contain no distortion, and
- the calibration also estimates the pose of  $E$  relative to  $H$ .

Calibration methods for standard cameras, e.g., [Zhang \(2000\)](#), use a high number of 2D–3D correspondences taken from different viewing angles to determine not only the intrinsic parameters of the camera, but also the distortion caused by the camera lens. In an OST-HMD the view seen by the user cannot be recovered similarly to a camera, thus the view to be aligned is shown on the HMD-screen instead. Early methods approximated the intrinsic parameters by aligning a predefined complex object with a targeted projection shown on the HMD-screen ([Azuma \(1995\)](#)). However, this requires complex tracking and extensive hardware preparation that cannot be generally applied. Also, aligning an increasing number of points, e.g., a checkerboard, can easily lead to drift and user errors. As a result, the Single Point Active Alignment Method (SPAAM) of [Tuceryan and Navab \(2000\)](#) emerged as a simplification of this process. In SPAAM the point correspondences are acquired as follows:

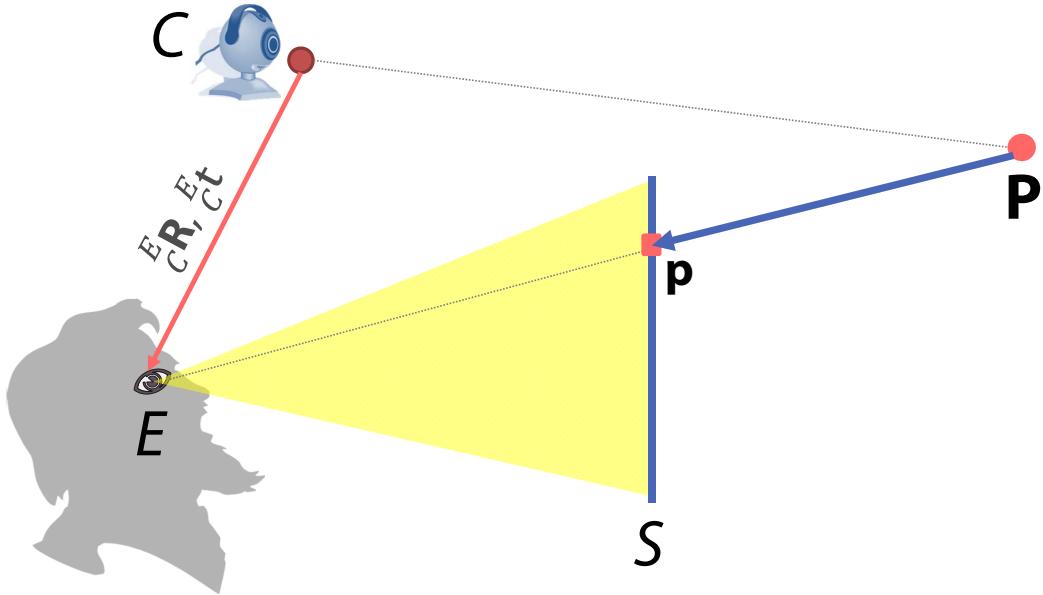


Figure 3.1: Concept of OST-HMD calibration.

1. Select and augment a random pixel  $\mathbf{p}$
2. Align  $\mathbf{p}$  with a 3D point  $\mathbf{P}$
3. Change the user's position relative to  $\mathbf{P}$ , e.g., by moving  $\mathbf{P}$
4. Repeat steps 1-3  $n$  times,  $n \geq 6$

The result of the data acquisition are pairs of homogeneous 2D pixels  $\mathbf{p} = (x \ y \ 1)^T$  and 3D points  $\mathbf{P} = (X \ Y \ Z \ 1)^T$ . As each repetition results in a single correspondence pair, acquiring a sufficient number to also estimate the distortion caused by the HMD-screen would require an extensive period of time and is therefore unviable.

Stereo camera calibration methods are used to calibrate the intrinsic and extrinsic parameters of at least two cameras at the same time, where the coordinate system of one of the cameras is used as the reference. However, these methods commonly apply an iterative calibration approach where first, the intrinsic parameters of each camera are calibrated separately followed by the computation of the extrinsic parameters given the estimated camera poses for each frame. In the SPAAM data acquisition, only the 3D positional information is recovered thus the  $3 \times 4$  projection matrix  $\mathbf{P} = \mathbf{K}_H^E \mathbf{T}$  has to be recovered from the 2D-3D correspondences.

As shown in Figure 3.2 a point  ${}^O\mathbf{P}$  in the object coordinate space  $O$  can be converted to the HMD coordinate space as

$${}^H\mathbf{P} = {}_W^H \mathbf{T}_O^W \mathbf{T} {}^O\mathbf{P}. \quad (3.1)$$

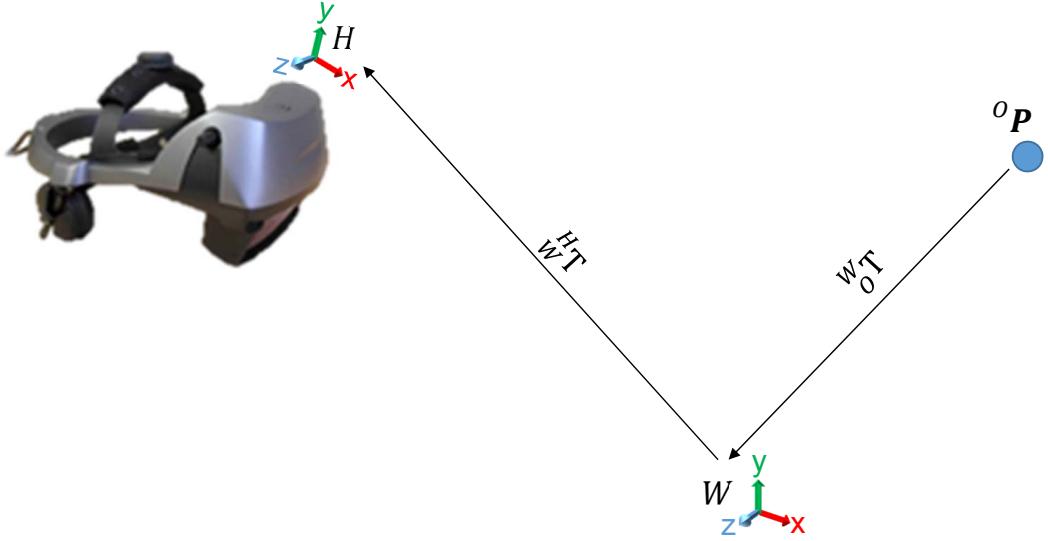


Figure 3.2: Estimation of a point relative to the HMD.

Hereby, the transformations  ${}^H_W T$  and  ${}^W_O T$  are determined by an external tracking system. In a common application scenario where a scene camera is mounted onto the HMD, its coordinate system is assumed to coincide with the world coordinate system and  ${}^H_W T$  is assumed as identity. As this transformation can be applied to all acquired data before processing, in the following explanation we will refer to  ${}^H \mathbf{P}$  as  $\mathbf{P}$ .

For the  $k$ th correspondence the aligned points can be described as follows

$$\begin{pmatrix} u_k \\ v_k \\ w_k \end{pmatrix} = \mathbf{P} \mathbf{P}_k \quad (3.2)$$

with

$$\begin{aligned} x_k &= \frac{u_k}{w_k}, \\ y_k &= \frac{v_k}{w_k}. \end{aligned} \quad (3.3)$$

Let  $\mathbf{P}_{ij}$  refer to elements of  $\mathbf{P}$ , where  $i$  is the row and  $j$  the column of the element. Inserting Equation (3.2) into Equation (3.3) leads to

$$\begin{aligned} x(\mathbf{P}_{31}X_k + \mathbf{P}_{32}Y_k + \mathbf{P}_{33}Z_k + \mathbf{P}_{34}) &= \mathbf{P}_{11}X_k + \mathbf{P}_{12}Y_k + \mathbf{P}_{13}Z_k + \mathbf{P}_{14}, \\ y(\mathbf{P}_{31}X_k + \mathbf{P}_{32}Y_k + \mathbf{P}_{33}Z_k + \mathbf{P}_{34}) &= \mathbf{P}_{21}X_k + \mathbf{P}_{22}Y_k + \mathbf{P}_{23}Z_k + \mathbf{P}_{24}. \end{aligned} \quad (3.4)$$

These equation systems can be rearranged as

$$\mathbf{A}_k \mathbf{b} = \mathbf{0}, \quad (3.5)$$

where  $\mathbf{b} = (P_{11} \ P_{12} \ P_{13} \ P_{14} \ P_{21} \ P_{22} \ P_{23} \ P_{24} \ P_{31} \ P_{32} \ P_{33} \ P_{34})^T$  and

$$\mathbf{A}_k = \begin{bmatrix} X_k & Y_k & Z_k & 1 & 0 & 0 & 0 & -x_k X_k & -x_k Y_k & -x_k Z_k & -x_k \\ 0 & 0 & 0 & 0 & X_k & Y_k & Z_k & 1 & -y_k X_k & -y_k Y_k & -y_k Z_k & -y_k \end{bmatrix}. \quad (3.6)$$

Given  $n$  correspondence pairs, all matrices  $\mathbf{A}_k$  can be accumulated into a single matrix

$$\mathbf{A} = \begin{pmatrix} \mathbf{A}_1 \\ \mathbf{A}_2 \\ \vdots \\ \mathbf{A}_n \end{pmatrix} \quad (3.7)$$

with  $2n$  rows and 12 columns. The equation  $\mathbf{Ab} = \mathbf{0}$  can be solved by minimizing  $\|\mathbf{Ab}\|$  under the constraint of  $\|\mathbf{b}\| = 1$ . The result is the eigenvector of  $\mathbf{A}$  associated with the smallest eigenvalue. The SVD decomposition of  $\mathbf{A}$  results in three matrices  $\mathbf{U}, \mathbf{D}, \mathbf{V}$  with  $\mathbf{UDV}^T = \mathbf{A}$ . The estimated value of  $\mathbf{b}$  is the last row of matrix  $\mathbf{V}$ . A detailed explanation of the SVD decomposition can be found in [Hartley and Zisserman \(2003\)](#). The projection matrix contains 12 unknown parameters, but only 11 parameters are independent as it is defined up to scale. Therefore 6 correspondences are sufficient to determine the projection matrix.

The intrinsic parameters  $\mathbf{K}$  are defined as

$$\mathbf{K} = \begin{bmatrix} f_x & \theta & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}. \quad (3.8)$$

The parameter  $\theta$  defines the skew while  $f_x$  and  $f_y$  are the focal length, and  $c_x$  and  $c_y$  the position of the center in the camera image along the x and y axes, respectively. As the intrinsic parameters form an upper-triangular matrix, QR-decomposition can be applied to the matrix  $\mathbf{M}$ , the first three rows of  $\mathbf{P}$ , to determine  $\mathbf{R}$ , the rotation matrix, and  $\mathbf{K}$ . The translational parameter  $\mathbf{t}$  is defined by  $\mathbf{t} = \mathbf{K}^{-1} \mathbf{P}_4$ , where  $\mathbf{P}_4$  is the 4th column of  $\mathbf{P}$ .

SPAAM has proven to achieve highly satisfactory results (Figure 3.3) and has also been extended to allow stereo calibration of an OST-HMD where the HMD-Screen is calibrated for both eyes at the same time for consistent stereo vision ([Genc et al. \(2000\)](#)). However, the method suffers from a variety of drawbacks, in particular

- need for recalibration when the HMD shifts on the head or is taken off and put on again,
- need for user interaction,
- unstable position estimation,

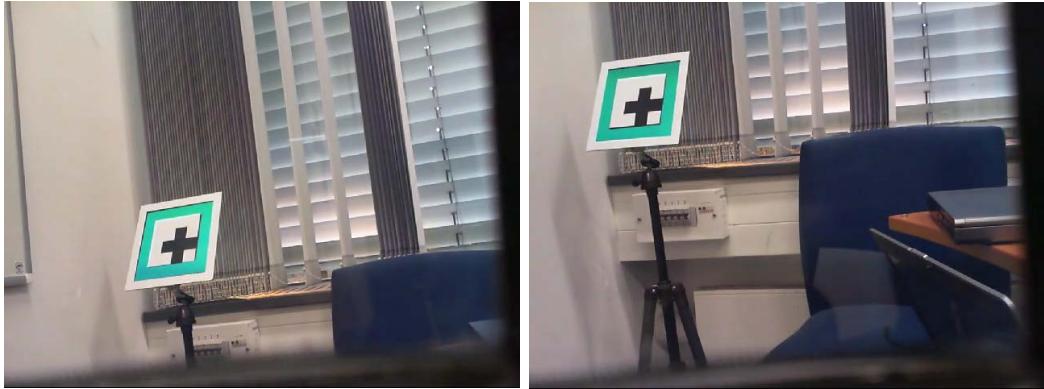


Figure 3.3: As a result of the SPAAM OST-HMD calibration the virtual content is correctly overlaid over the target marker.

- stereo calibration complicated due to simultaneous alignment for both eyes.

As a result of the required user input, the results are highly dependent on correct alignment of the 2D visualization with the 3D target and is also impacted by the distribution of the 3D points (Axholt et al. (2010, 2011)). As a result various methods have been developed to simplify the SPAAM calibration process (Tuceryan and Navab (2000); Navab et al. (2004); Maier et al. (2011)).

The pose of the eye camera estimated by SPAAM is highly unstable—Tuceryan and Navab (2000) suggest at least 12 point correspondences to be collected during the calibration process. Figure 3.4a shows the estimated position from multiple calibrations. The position of the camera  $E$  does not appear consistent with the position of the user’s eye for the majority of the results. Figure 3.4b visualizes how the number correspondences impact the result of the calibration. The ground truth of the calibration was obtained from 20 correspondences. The graph visualizes the offset of the estimated position from a calibration performed with  $n$  pairs selected randomly from the dataset from the ground truth. The graph visualizes the results of 100 random selections of  $n$  correspondences. Even after the recommended number of 12 correspondences has been acquired, the estimation deviates by almost 2 cm and remains above 1 cm even after 16 samples have been selected.

## 3.2 Spatial OST-HMD Calibration

Due to the drawbacks of the SPAAM method, in particular the required user interaction, researchers are exploring how combinations of online- and offline-process can lead to automated calibration of the OST-HMD. The most prominent is the Display Relative Calibration (DRC) approach proposed by Owen

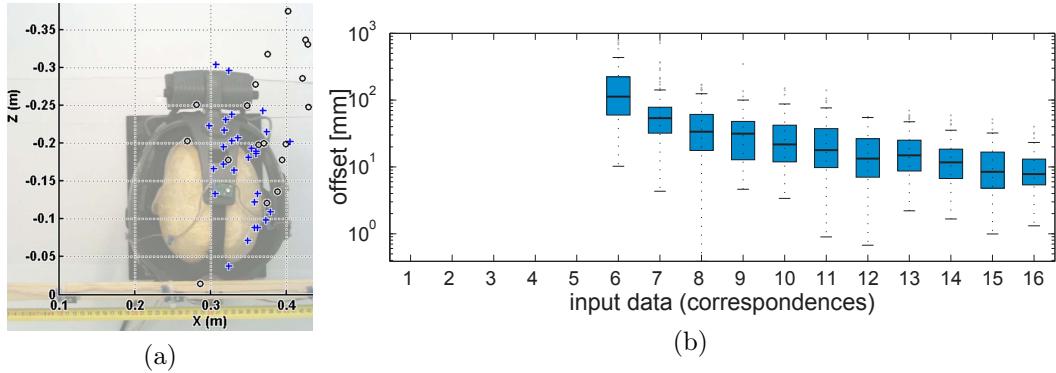


Figure 3.4: Instability of the eye position estimated by SPAAM. (a) Eye position estimated for multiple SPAAM calibrations greatly deviate from the actual position of the right eye (Axholt et al. (2011)). (b) Estimated position is highly dependent on the selected samples and remains noisy even after 16 samples have been collected.

et al. (2004). DRC separates the calibration process into an offline modelling of the HMD-Screen and an online estimation of the eye-pose. Given the surface of the HMD screen and the position of the eye, the projection can be computed either directly as the intersection of the ray from the point  $\mathbf{P}$  towards the eye position  $\mathbf{E}$ , or by computing the projection matrix from the spatial model of the screen and the eye center.

A variety of models of the HMD-screen has been developed over the years. Tuceryan and Navab (2000), and Itoh and Klinker (2014b) represent the HMD-screen as a plane, Owen et al. (2004) model it as a spherical mirror, and Klemm et al. (2014) use a non-parametric approach. Itoh and Klinker (2015b) propose to model the HMD as a light field instead of a 3D surface. We represent the HMD-screen as a plane due to the simplicity of this model.

The idea of employing an eye-tracking camera with OST-HMDs has been proposed in the past (Park et al. (2008); Itoh and Klinker (2014a)). To enable gaze-based interaction and estimation of the eye–HMD-screen relation the pose of the tracking camera relative to the scene camera and the HMD-screen is necessary.

In this dissertation we use the setup shown in Figures 3.5 and 3.6 and determine the spatial calibration of the OST-HMD as described in Itoh and Klinker (2014b). The calibration process consists of a spatial calibration of the cameras and a reconstruction of the screen.

### 3.2.1 Extrinsic Camera Calibration

To determine the spatial transformation between the scene and eye camera we use a rigid system of spatial markers shown in Figure 3.5. The markers are arranged so that when the HMD is placed in between, at least one of

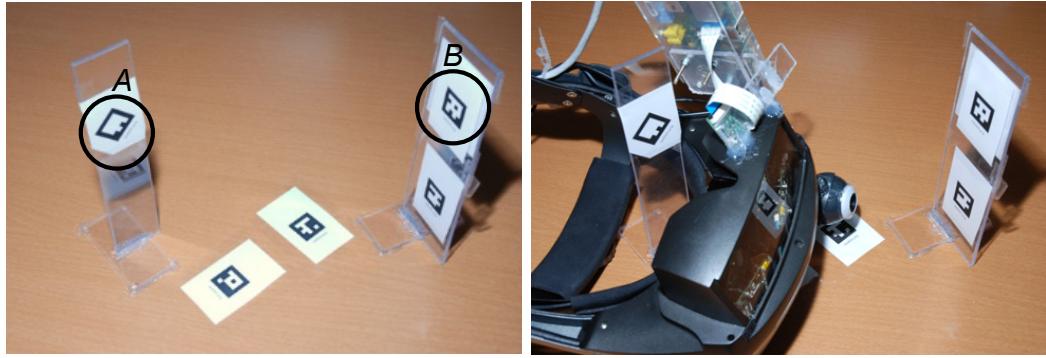


Figure 3.5: Setup for spatial calibration of the HMD. (left) The marker are arranged so that images taken by the camera can detect markers in different planes. (right) When the HMD is placed between the markers the cameras can detect the markers in the environment. We use the same device as was used by [Itoh and Klinker \(2014a\)](#).

the markers can be detected by the cameras  $T$  and  $C$ . The relation between the markers is calibrated from multiple images of the arrangement with the Ubitrak library ([Huber et al. \(2007\)](#)). Let the markers  $A$  and  $B$  be seen by the cameras  $T$  and  $C$  respectively, then the transformation  ${}^T_C\mathbf{T}$  is given by

$${}^T_C\mathbf{T} = {}^T_A\mathbf{T}_B^A {}^T_C\mathbf{T}_C^B \mathbf{T}. \quad (3.9)$$

The transformation is further refined by minimizing the alignment error from multiple images, where the HMD pose is changed between each image.

### 3.2.2 Screen Reconstruction

After the transformation  ${}^T_C\mathbf{T}$  has been estimated, we determine the pose of the HMD-screen  $S$  relative to  $C$ . We attach an opaque cover to the HMD-Screen from the outside and performed the following three steps:

1. A user-perspective camera  $U$  is placed onto a tripod and captures an image of a checkerboard  $C$  placed in front of it.
2. Without touching  $U$ , the HMD is placed so that  $U$  can capture an image of the content shown on the HMD screen.
3.  $U$  captures an image of a checkerboard pattern shown on the HMD screen and the scene camera  $T$  captures an image of the checkerboard  $B$ .

These steps are repeated  $n$  times, and the pose of the camera  $U$  is changed for each iteration. For each iteration the transformation from the camera  $U$  to  $C$  can be computed as

$${}^U_C\mathbf{T} = {}^U_B\mathbf{T}_C^B \mathbf{T}. \quad (3.10)$$



Figure 3.6: An OST-HMD equipped with an eye-tracking camera.

W.l.o.g we assume that the height and width of all pixel is  $s$ , the number of pixel in each patch. Therefore, the pose of  $B$  can be estimated for each frame. As the screen is rigidly attached to the HMD, the rotation  ${}^T_S\tilde{\mathbf{R}}$  can be recovered as the mean of all  ${}^T_S\mathbf{R}$  in quaternion space (Gramkow (2001)). The translation offset for frame  $k$  is described by  ${}^T_S\mathbf{t}_k(s) = {}^S_U\mathbf{R}_S^U\mathbf{t}_k + {}^U_U\mathbf{t}_k$ , where  ${}^U_U\mathbf{t}_k$  is the position of  $U$  relative to  $T$  and  ${}^S_U\mathbf{t}$  the position of  $S$  relative to  $U$  for a predefined scaling parameter for the  $k$ th frame. As such, it is possible to declare an error function

$$e = \frac{1}{n} \sum_{k=1}^n \| {}^T_S\bar{\mathbf{t}}(s) - {}^T_S\mathbf{t}_k(s) \|^2. \quad (3.11)$$

The factor  $s$  can be estimated as  $\tilde{s} = \arg \min_s e$ .

### 3.3 Automated OST-HMD Calibration

Automated calibration methods estimate the intrinsic and extrinsic parameters of the user-perspective camera from the reconstructed surface of the HMD-screen and the estimated eye position. Owen et al. (2004) propose a variety of solutions to determine the eye position, however the proposed solutions still require extensive hardware or user interaction. Furthermore, the authors do not provide an evaluation of the proposed methods and focus on the estimation of the HMD-screen surface.

Itoh and Klinker (2014a) assume that the transformations  ${}^T_S\mathbf{T}$  and  ${}^C_S\mathbf{T}$  between the eye tracking camera  $T$ , the scene camera  $C$  and the HMD-Screen  $S$  are known beforehand, e.g., through a factory calibration or as described in Section 3.2. Therefore, an estimated eye position relative to  $T$  can be used to determine all parameters necessary for spatially correct augmentation. The authors propose two methods for Interaction-free Display Calibration (INDICA). A full calibration that uses the modeled screen to determine the projection matrix (INDICA Full), and a recalibration solution that reuses previously estimated intrinsic and extrinsic parameters (INDICA Recycle), e.g., obtained through a one-time SPAAM calibration. The INDICA method

follows the common notion that the eye-OST-HMD-screen relation can be modelled as a pinhole camera  $E$ , where the HMD-screen covers the frustum of  $E$ .

### 3.3.1 INDICA Full

In the described scenario, let the origin of the screen be at  ${}^E\mathbf{S} = (x \ y \ z)^T$  and the optical axis of  $E$  be perpendicular to  $S$ .  ${}^E\mathbf{S}$  projects onto the origin of the camera image. This projection can be modelled by a  $3 \times 3$  intrinsic matrix  $\mathbf{K}_E$  defined as

$$\mathbf{K}_E = \underbrace{\begin{bmatrix} \alpha_x & 0 & 0 \\ 0 & \alpha_y & 0 \\ 0 & 0 & 1 \end{bmatrix}}_A \underbrace{\begin{bmatrix} z & 0 & -x \\ 0 & z & -y \\ 0 & 0 & 1 \end{bmatrix}}_{\mathbf{S}({}^E\mathbf{S})}. \quad (3.12)$$

The parameter  $\alpha$  is the scaling factor that transforms points on the image plane into screen pixels on the HMD screen and the matrix  $\mathbf{S}(\mathbf{S})$  determines the projection, so that the point  $\mathbf{t}$  is projected at the origin of the image pixel plane. In particular,  $A$  is independent of the position of the eye, while  $\mathbf{S}({}^E\mathbf{S})$  changes depending on the position of the eye.

The projection matrix  ${}^E\mathbf{P}$  that projects a point detected by the scene camera onto the image taken by the  $E$  is composed of the intrinsic matrix  $\mathbf{K}$  and a transformation of points from the coordinate system of the scene camera  $C$  into that of the eye camera,

$${}^E\mathbf{P} = \mathbf{K}_E \begin{bmatrix} {}^E\mathbf{R} & {}^E\mathbf{t} \end{bmatrix}. \quad (3.13)$$

As the optical axis of the camera is perpendicular to the HMD-screen, the rotation matrix coincides with the rotation matrix from the scene camera to the screen  ${}^E\mathbf{R} = {}^S\mathbf{R}$ . The position of the eye is estimated by the eye tracking camera  $T$ , therefore  ${}^E\mathbf{t}$  can be obtained as

$${}^E\mathbf{t} = {}^T\mathbf{R}({}^T\mathbf{E} - {}^T\mathbf{C}) = {}^T\mathbf{R}({}^T\mathbf{E} - {}^T\mathbf{C}). \quad (3.14)$$

The proposed calibration method requires the parameters  $\alpha_x$ ,  $\alpha_y$  and the position of the origin of the screen  ${}^E\mathbf{S} = -{}^T\mathbf{T}{}^T\mathbf{E}$ .

### 3.3.2 INDICA Recycle

As the calibration of the HMD-Screen relative to the scene camera requires sophisticated setup, the authors also propose an alternative approach, INDICA Recycle, that uses known intrinsic parameters and an approximation of the depth of the HMD screen to determine the new projection matrix.

Assume that the eye has moved to a new position  $\mathbf{E}_1 = \mathbf{E} + \mathbf{t} = \mathbf{E} + (\mathbf{t}_x \ \mathbf{t}_y \ \mathbf{t}_z)^T$ . The projection matrix for the new camera  $E_1$  is thus defined as

$$\mathbf{C}_E \mathbf{P} = \mathbf{K}_{E_1} \begin{bmatrix} \mathbf{E}_1 & \mathbf{C} \\ \mathbf{C} & \mathbf{t} \end{bmatrix} = \mathbf{A} \mathbf{S} \begin{bmatrix} \mathbf{E}_1 & \mathbf{C} \\ \mathbf{C} & \mathbf{t} \end{bmatrix} \quad (3.15)$$

From the definition of the matrix  $\mathbf{S}$  it follows that

$$\mathbf{S} \begin{bmatrix} \mathbf{E}_1 & \mathbf{C} \\ \mathbf{C} & \mathbf{t} \end{bmatrix} = \begin{bmatrix} z + \mathbf{t}_z & 0 & -(x + \mathbf{t}_x) \\ 0 & z + \mathbf{t}_z & -(y + \mathbf{t}_y) \\ 0 & 0 & 1 \end{bmatrix} \quad (3.16)$$

$$= \begin{bmatrix} z & 0 & -x \\ 0 & z & -y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 + \frac{\mathbf{t}_z}{z} & 0 & -\frac{\mathbf{t}_x}{z} \\ 0 & 1 + \frac{\mathbf{t}_z}{z} & -\frac{\mathbf{t}_y}{z} \\ 0 & 0 & 1 \end{bmatrix} \quad (3.17)$$

$$\mathbf{S} \begin{bmatrix} \mathbf{E}_1 & \mathbf{C} \\ \mathbf{C} & \mathbf{t} \end{bmatrix} = \mathbf{S} \begin{bmatrix} \mathbf{E} & \mathbf{C} \\ \mathbf{C} & \mathbf{t} \end{bmatrix} \mathbf{S} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \frac{1}{z} \end{bmatrix}. \quad (3.18)$$

From Equations (3.15) and (3.18) the intrinsic parameters  $\mathbf{K}_{E_1}$  can be recovered from the previous intrinsic parameters  $\mathbf{K}_E$  and the depth of the screen  $S$  relative to  $E$ .

Consecutively, all parameters of the Recycle approach can be recovered without explicitly modelling the HMD screen. An initial SPAAM calibration can be used to recover  $\mathbf{K}_E$  and  $\mathbf{C}_E \mathbf{R}$ . The depth  $d_{SC}$  of the screen relative to the scene camera  $C$  can be recovered by a manual focus camera  $F$  that is placed behind the HMD-screen. By focusing the camera  $F$  onto content displayed on the HMD screen and deducting the offset between the cameras  $F$  and  $T$  from the estimated depth. The value  $z$  can then be recovered from the estimated position  ${}^T \mathbf{E}$  and the transformation  $T_C \mathbf{T}$ . One can expect Recycle to perform worse than Full, as the estimated depth and orientation of the screen are only an approximation of the real values.

In their work the authors use 2D–3D correspondences acquired during a SPAAM calibration to evaluate the INDICA Recycle approach. They show that although it does not achieve the accuracy of SPAAM, it does outperform the case of reusing the previous SPAAM calibration without any adjustment, something commonly done when the HMD has to be taken off and put on multiple times for a demo or experiment. Additionally, the variance of the 2D point projection and the estimated eye position was significantly smaller than that of SPAAM. An investigation on noise sensitivity by [Itoh and Klinker \(2014b\)](#) showed that both, Full and Recycle modes are more robust than SPAAM calibration, and are preferable, especially compared to a reuse of a previous calibration. They also note that the estimation of the eye position by their method is likely a primary factor in the incorrect rendering.

## 3.4 Perception of AR Content

Depending on how critical visualization errors are for the user performance, their perception can be seen under the aspect of error impact and noticeability.

Although we focus on the noticeability of errors, we describe how noticeable errors can impact the user’s performance, followed by an overview on how various errors are perceived by users.

### 3.4.1 User Perception

Perception of virtual content greatly impacts not only its utility, but also its acceptability by users. [Gkioulekas et al. \(2013\)](#) investigated how the phase function of translucent objects impact their perception by users, [Křivánek et al. \(2010\)](#) and [Jarabo et al. \(2012\)](#) focused on global illumination, and [Tokunaga et al. \(2015\)](#) investigated how errors in the input data used in P-3-P localization impact the perception of the virtual object’s pose compared to the ground truth. [Kishishita et al. \(2014\)](#) investigated how the appearance of object labels shown on an OST-HMD impacts their noticeability.

[Madsen and Stenholz \(2014\)](#) investigated the noticeability of a static rotational misalignment of virtual content in a controlled hand-held environment. In their study users were asked to perform an alignment task, where they correct a rotational modelling error in direct and indirect AR settings, with the aim of discovering the minimum angular error before users notice a misalignment. They found that users are more perceptive to rotational errors in classical AR, where an existing scene is augmented, than indirect AR, a virtual view of a physical scene from a given viewpoint. There has yet to be a formal study examining the perceptibility, and acceptability thresholds, of similar alignment error in OST HMD systems.

### 3.4.2 Error impact

[Livingston and Ai \(2008\)](#) investigated the influences of different error sources, such as latency, positional errors, orientation errors, and weather, and attempted to quantify the impact of these factors on user’s situational awareness. Surprisingly, they discovered that neither orientation errors, nor system noise presented a problem for users. [Robertson et al. \(2009\)](#) studied how the consistency of misalignment impacts users. They found that consistent error throughout the session, e.g., falsely registered scene-model, resulted in users self-calibrating to the error over time. However, users could not adjust to randomly distributed errors, e.g., as a result of inconsistent tracking, which significantly impacted overall task performance. A study by [Khuong et al. \(2014\)](#) adopted the premise that perfect spatial alignment of virtual and real content could not be practically achieved and investigated how this presumption impacts the usability of assembly guidelines. Their examination found that if ideal registration is not possible, an alternative side-by-side visualization yields better performance results compared to the inaccurate overlay.

[Moser et al. \(2015\)](#) compared the impact of manual SPAAM ([Tuceryan and Navab \(2000\)](#)) and automated INDICA Recycle ([Itoh and Klinker \(2014a\)](#))

OST-HMD calibration on a simple AR localization task. Their findings showed that user accuracy using the automatic calibration improved compared to performance using the manual method. More strikingly though, user indicated quality measures, provided by subjects to indicate how well they felt the virtual content was aligned to the world, were quite high even in instances of noticeable registration error. This perceptual acceptability of registration error has been more closely examined in various studies.

We desire to build upon this body of work by identifying just noticeable translational and rotational accuracy levels for AR registration. We conducted a user study investigation designed to not only measure user perception of registration error, but also determine if the detectability thresholds for misalignment differ between VST and OST presentation methods.

# CHAPTER 4

# Corneal Imaging Calibration of OST-HMDs

---

This chapter focuses on the calibration of an OST-HMD, a step performed to achieve consistent spatial overlay of virtual and real content. After an introduction and review of the problem we explain how estimation of the corneal sphere through CI can be applied to automatically calibrate the OST-HMD.

Section 4.1 explains why the calibration step is necessary and describes the different approaches that can be taken to achieve this objective. In Section 4.2 we introduce Corneal Imaging Calibration (CIC), our improvement of the automated calibration proposed by [Itoh and Klinker \(2014a\)](#). We follow this with a simulated evaluation in Section 4.3 and present our experiment conducted on an actual HMD in Section 4.4. We conclude with a summary and outlook in Section 4.5.

## 4.1 Introduction

Contrary to VST-HMDs and handheld devices that show augmentations on an image captured by the external camera, the screen of an OST device does not occlude the view of the scene. As such, only the virtual graphics are displayed on the screen. Visualizing the virtual content from the estimated pose of the screen results in a spatially incorrect augmentation. Therefore, correct spatial visualization has to be rendered from the user's perspective. In a remote scenario, where the user's pose to the screen can change continuously, sophisticated tracking algorithms have to be used to estimate the user's perspective through the screen, similar to [Tomioka et al. \(2013\)](#). When the user is wearing an OST-HMD, it is necessary to determine an accurate overlay of the virtual and real content, as the distance to the screen is relatively small.

OST-HMD calibration determines what pixel  $\mathbf{p}$  overlaps a 3D point  ${}^H\mathbf{P}$  from the user's perspective. Hereby, the estimation of  ${}^H\mathbf{P}$ , the point  $\mathbf{P}$  relative to the HMD  $H$ , is conducted by an external tracking system, for example infrared markers attached to the HMD and the target object, or a camera that is rigidly attached to the HMD and is tracking the environment.

Commonly it is assumed that the HMD is fixed on the head, thus a one-time calibration is sufficient to provide satisfactory results. Although manual calibration solutions are suitable for this scenario, ideally the calibration

should be an automated online process to account for noise and HMD shifts on the head.

## 4.2 Corneal Imaging Calibration

As was pointed out by [Itoh and Klinker \(2014b\)](#) the main weakness of INDICA is the estimated projection position, assumed to be the center of the rotation of the eye  $\mathbf{E}$ . The original design of INDICA proposes to estimate  $\mathbf{E}$  from the detected elliptical iris contour of the eye. Naturally commercial eye-trackers based on infra-red cameras and LEDs, e.g., Tobii Glasses ([Tobii Technology AB \(2016a\)](#)) or SMI wearable trackers ([Sensor Instruments \(2015\)](#)), can estimate an accurate eye pose. Including an eye-tracker based on infra-red illumination requires extensive additional hardware with accurate, complicated, calibration. Infra-red light trackers are difficult to deploy in outdoor environments, where strong illumination makes accurate eye-pose estimation difficult. Finally, although short-term use of IR-trackers does not pose health problems, the long-term impacts are still unknown.

Our objective is to develop a method to improve the results of the calibration without the reliance on infra-red light. We propose to use corneal imaging to acquire information about the users view.

We follow [Itoh and Klinker \(2014a\)](#) and assume that the HMD  $H$  is equipped with an eye-tracking camera  $T$  and a scene camera  $C$ . The camera  $C$  is used to track the scene and coincides with the coordinate system of the HMD. Additionally, the poses of the cameras  $T$ ,  $C$  and the screen  $S$  relative to each other have been calibrated as described in Section 3.2. The 3D position of every pixel shown on an OST-HMD screen is available as a result of the spatial HMD calibration. By matching pixels shown on the screen with the detection in the corneal reflection it is thus possible to acquire 2D and 3D correspondences, and to estimate the position of the corneal sphere as described in Section 2.3.2.

### 4.2.1 Eye Position Estimation

As described in Section 3.1, for a spatially calibrated OST-HMD, only the position of the projection center of the eye is required to determine the projection matrix. As the eye rotates around its center of rotation  $\mathbf{E}$  the point of projection changes accordingly and should be estimated for every frame. However, as the offset from the center of the rotation is relatively small, compared to the distance to the HMD-screen, we follow the assumption that  $\mathbf{E}$  is the center of projection. This assumption is also beneficial for scenarios, where the position of the corneal sphere cannot be tracked continuously, e.g., if no correspondences between the corneal reflection and the HMD-screen can be found.

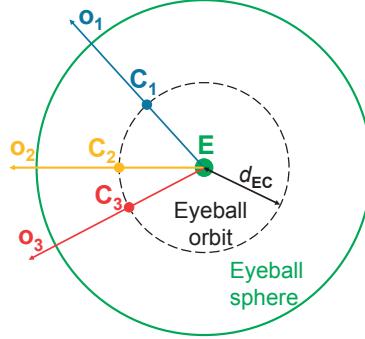


Figure 4.1: Cross-section view of rotations of the eyeball. The corneal spheres centered at  $\mathbf{C}_1$ ,  $\mathbf{C}_2$  and  $\mathbf{C}_3$  result from three different gaze directions. The centers lie on a spherical orbit of radius  $d_{\mathbf{EC}}$  around  $\mathbf{E}$  (shown in green).

$\mathbf{E}$  can be estimated from the cornea position and the gaze direction, in which case  $\mathbf{E} = \mathbf{C} - d_{\mathbf{EC}}\hat{\mathbf{o}}$ , where  $d_{\mathbf{EC}}$  is the distance between the eye center and the cornea center and  $\mathbf{o}$  is the optical axis of the eye. However, estimation of  $\mathbf{o}$  from the iris contour can suffer from the reflection of the environment in the eye, occlusion by eyelids and eyelashes, generally bad lighting conditions, or low contrast between the iris and the sclera. Therefore, we use an alternative approach.

While the user is looking at different areas of the screen, the center of the corneal sphere moves on an orbit with a radius of  $d_{\mathbf{EC}}$  around  $\mathbf{E}$ . Therefore, it is not necessary to estimate the gaze direction to recover the eye center, as  $\mathbf{E}$  can be recovered from multiple cornea centers. For a known  $d_{\mathbf{EC}}$ ,  $\mathbf{E}$  can be estimated from at least three cornea centers. Three cornea centers and the radius  $d_{\mathbf{EC}}$  describe two possible eye centers. However, the solution located closer to the camera can be eliminated, as it is anatomically not plausible. As the estimation from the minimal number of observations is very noise sensitive, it is recommended to include all observations. This includes the removal of outliers. Given an estimated eye center  $\mathbf{E}$ , a cornea located at  $\mathbf{C}$  is an inlier of the estimation if

$$|\|\mathbf{E} - \mathbf{C}\| - d_{\mathbf{EC}}| < d, \quad (4.1)$$

where  $d$  is an inlier threshold. In our experiments we found that  $d = 0.3$  mm provided the best results. To determine a stable  $\tilde{\mathbf{E}}$ , we use a RANSAC approach. From  $n$  estimated cornea centers, we randomly select 3 centers and fit the eye orbit with the known size to determine an initial guess  $\mathbf{E}_0$ . For this initial guess we determine count the number of inliers according to Equation 4.1. Let the best estimation be supported by  $k$  observations, then the final position  $\tilde{\mathbf{E}}$  is determined as

$$\tilde{\mathbf{E}} = \arg \min_{\mathbf{E}} \sum_{i=1 \dots k} |\|\mathbf{C}_i - \mathbf{E}\| - d_{\mathbf{EC}}|. \quad (4.2)$$

### 4.2.2 Drift Detection

As the HMD calibration and the gaze estimation require a good estimation of  $\mathbf{E}$  it is necessary to determine when the HMD has shifted and re-estimate  $\mathbf{E}$ . Therefore, let  $\mathbf{E}_0$  be the current eye position and  $\mathbf{C}_j$  the cornea position for the current frame. If the HMD-screen has not moved, it follows that  $d_{\mathbf{EC}} = \|\mathbf{E}_0 - \mathbf{C}_j\|$ . This does not hold if the HMD has moved or the position of the cornea has been estimated incorrectly. Therefore, we observe subsequent frames: If the majority of these frames supports  $\mathbf{E}_0$  we conclude that  $\mathbf{C}_j$  is the result of an erroneous estimation. On the other hand, if the majority suggests that the HMD has moved we estimate a new  $\mathbf{E}$ . We use a sliding window to determine a stable  $\mathbf{E}$  that continuously accounts for HMD movement. The size of the sliding window depends on the desired stability, while three frames are enough to obtain a guess for a new eye center position, a larger number ensures more stable results.

## 4.3 Synthetic-Data Experiment

To verify the applicability of the proposed method we generate artificial data according to our eye model. The simulated environment was generated to resemble the view through an actual OST-HMD.

We configured the captured image size and the intrinsic parameters of the virtual camera identically to the camera used in our OST-HMD setup. The HMD-screen plane was positioned 700 mm behind the camera and the center of the rotation of the eye,  $\mathbf{E}$ , about 40 mm in front of the camera. Given the fixed eye center we selected 16 random gaze directions and determined the corresponding cornea centers  $\mathbf{C}_i$ ,  $i = 1 \dots 16$ . Each  $\mathbf{C}_i$  was projected into the virtual camera and points arranged in a grid pattern were selected above the projection point.

This corresponds to point distribution observed during data collection with an actual HMD. We discard all points on the grid whose back-projected ray does not intersect the corneal sphere. For the remaining points we reflect the back-projected ray on the corneal sphere and determine the intersection point with the HMD-screen plane. This intersection is the point's origin. For each gaze direction this results in a set of ground-truth 2D–3D correspondences. If no noise is added to these observations our method computes the correct position of the corneal sphere and the eye center.

Eye-related studies in computer vision, such as eye-pose and POR estimation, often assume different values for the eye-model parameters. To investigate the impact caused by an incorrectly assumed radius of the corneal sphere we estimate the position of the corneal spheres with a radius size of  $r_e = r_C + \sigma_e$ , where  $\sigma_e = \{-2, -1, 0, 1, 2\}$  is a static modelling error. As shown in Figure 4.2, the respective errors result in an offset along the vector towards the center of the corneal sphere. The induced errors behave in a linear

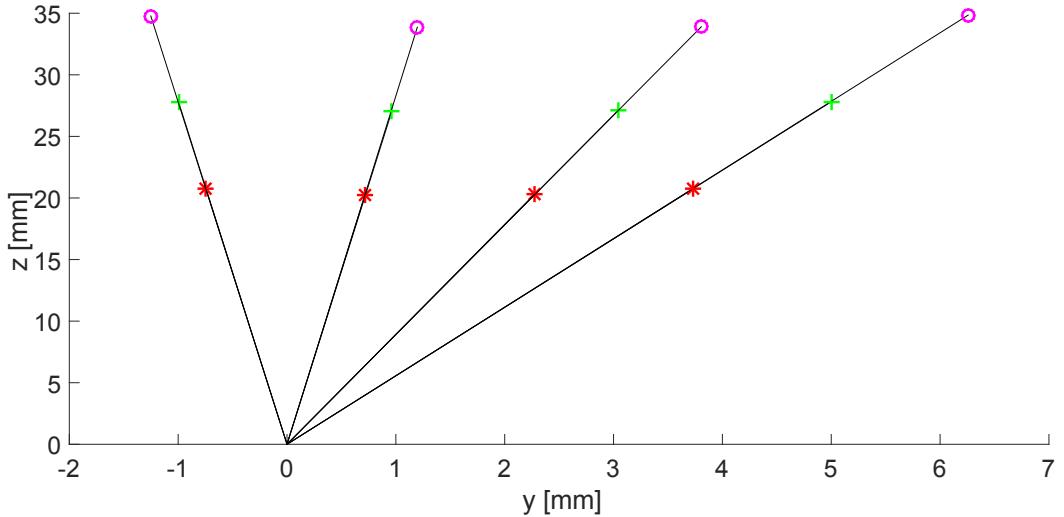


Figure 4.2: Given small noise levels in the 2D–3D correspondences, incorrectly estimated cornea size results in a linear scaling of the distance to the cornea position.

manner and suggest that if the noise in the 2D–3D correspondences is small, an estimation of the radius of the corneal sphere is possible ([Villanueva and Cabeza \(2008\)](#)).

To determine how sensitive the eye center estimation is to various noise sources, we have perturbed the 2D and 3D values with zero-mean noise with a standard deviation of  $\sigma_p$  and  $\sigma_P$ , respectively. We observe the impact of three different noise levels: small, average, and large noise levels. We used  $\sigma_p = \{0.2, 0.5, 1\}$  pixel and  $\sigma_P = \{0.5, 2, 10\}$  mm to represent the noise levels and the respective values are expected results of the calibration and detection process. We assume that the estimated values for  $r_C$  and  $d_{EC}$  are known and evaluate the following scenarios:

- Only the pixels  $\mathbf{p}$  contain noise (2D noise),
- Only the 3D points  $\mathbf{P}$  contain noise (3D noise), and
- Both  $\mathbf{p}$  and  $\mathbf{P}$  contain noise (2D and 3D noise).

Before estimating the eye center, we remove all  $\mathbf{C}$  that we assume to be outliers. Hereby, an estimation  $\mathbf{C}$  is an outlier if its mean reprojection error  $e > 2$  pixel. For each pair  $\{\mathbf{p}, \mathbf{P}\}$ , let  $\mathbf{p}'$  be the projection of the reflection of  $\mathbf{P}$  on the cornea located at  $\mathbf{C}$ . Then the reprojection error is defined as  $e(\{\mathbf{p}, \mathbf{P}\}) = \|\mathbf{p} - \mathbf{p}'\|$ , and for  $n$  pairs of  $\{\mathbf{p}, \mathbf{P}\}$  it follows that

$$e = \frac{1}{n} \sum_{i=1}^n e(\{\mathbf{p}_i, \mathbf{P}_i\}). \quad (4.3)$$

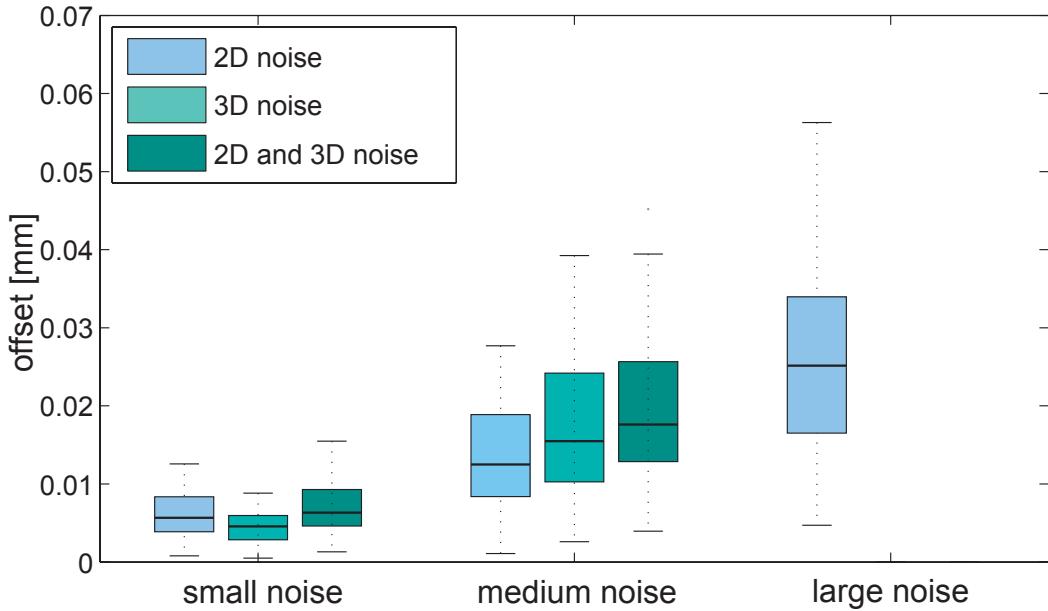


Figure 4.3: Deviation of estimated eye positions from the ground truth due to data perturbed by three levels of noise: Small noise, which is likely to occur; medium noise, which will occur from small inconsistencies or calibration errors; and large noise, which is unlikely to occur. The noise is applied to the detected pixels (2D noise), the 3D point position (3D noise) and a combination of both errors.

For each constellation we repeat the process of adding noise, estimating the cornea centers, followed by the estimation of the eye center, 100 times. The results of the estimation are shown in Figure 4.3. The expected noise values of points on the image plane do not impact the estimation of the eye center, with an offset between 0.005 and 0.027 mm from the ground truth. Adding noise to the 3D points had similar effect for small and medium noise levels. However, for the large noise level the estimation of the eye failed. This signifies that a good calibration of the OST-HMD Screen is necessary to produce reliable estimation results.

## 4.4 Real-Data Experiment

In this section we evaluate various aspects of the CIC method. We first evaluate the applicability of the spatial OST-HMD calibration for the estimation of the corneal sphere's position. We follow with an explanation of the implementation. We evaluate the results of the estimated cornea position, the stability of the eye center estimation and compare the results of the OST-HMD calibration by various methods. We conclude with a discussion of the estimation of the cornea size as a part of CIC.

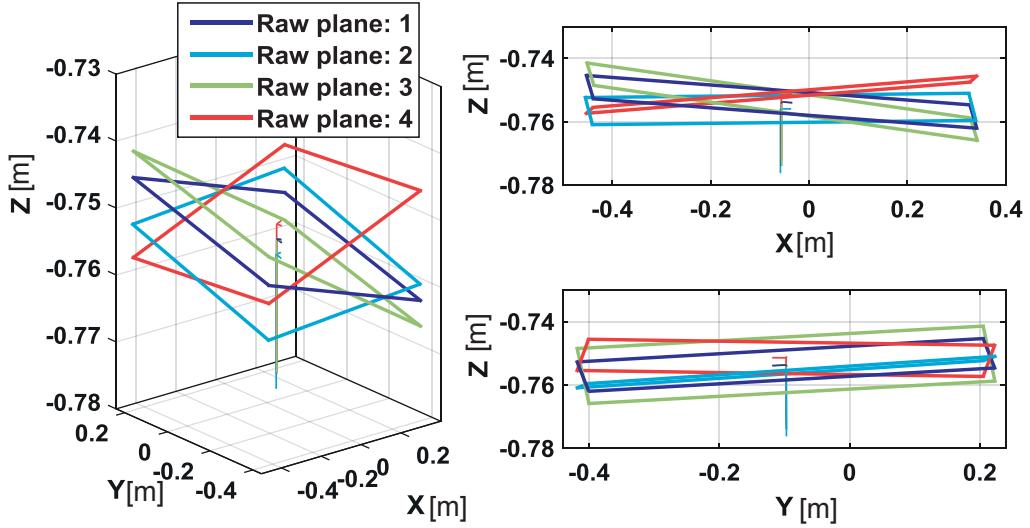


Figure 4.4: Estimated planes of the HMD screen for four different views.

#### 4.4.1 Quality of Spatial OST-HMD Calibration

The results of the spatial OST-HMD calibration is not a perfect alignment of the estimated screen planes. As shown in Figure 4.4 the screen planes are slightly misaligned, in particular the rotation of the screen.

To determine if the estimated HMD is accurate enough to estimate the position of the cornea, we replace the eye with a mirrorball with a radius of 10 mm shown in Figure 4.5. To acquire the ground truth we select multiple points on the contour of the projected mirrorball. The position of the mirrorball is reconstructed as explained in Section 2.3.3. Additionally, we detect 2D-3D correspondences from a checkerboard show on the HMD-screen and estimate the position as described in Section 2.3.2. In our experiments the mean offset between the two positions was 0.3 mm. We believe this accuracy is sufficient for the proposed approach.

#### 4.4.2 Implementation

We have implemented our method in C++. Image capturing and processing used OpenCV 3.0 (Bradski (2000)) and mathematical computations were conducted with Eigen3 (Guennebaud et al. (2010)). In the following we describe how we obtain the reflection of the pattern shown on the HMD-screen in the eye.

**Feature matching.** We use a C++ implementation of the LibCBDetect (Geiger et al. (2012)) library, as the original MATLAB implementation cannot be used for real-time calibration, the objective of automated calibration methods. Our naïve implementation on an Intel i7-7000 with 32 GB RAM

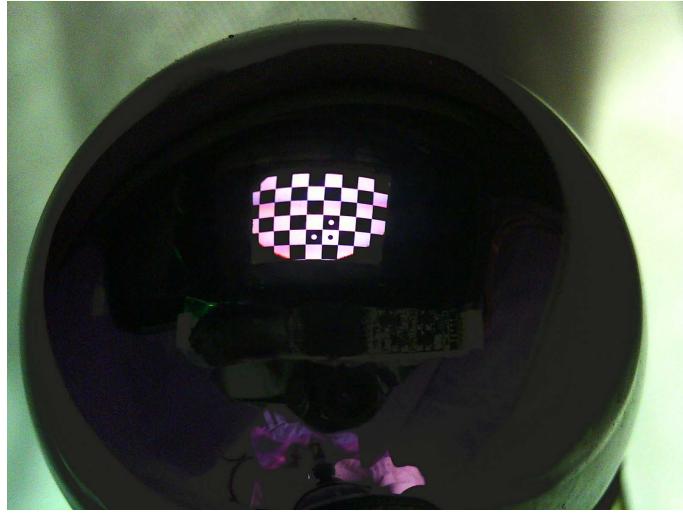


Figure 4.5: Reflection of the HMD-screen in a mirrorball placed instead of the eye.

performs the detection step in under a second on a  $1600 \times 1200$  image. An optimized implementation (e.g., on a GPU) and cropping of the examined area will further improve the processing speed. LibCBDetect detects the inner corners of a checkerboard, even if reflected on a sphere, with sub-pixel accuracy, and arranges them into a grid of at least 9 points, thus further improving the robustness to outliers. The library returns multiple disconnected grids if one or more points of the inner grid are not detected.

We align each grid separately and collect the aligned points afterwards into correspondences for each image. To determine the location of the grids, we have printed a number of dots into the squares of the checkerboard (Figures 4.5, 4.6). Their location is static, and a detected imprinted square allows to align the origin of the coordinate systems. Here, we employ pattern matching to detect imprinted squares, as follows: As the corners of each square are known, we can reproject the enclosed area onto a squared image and compare the result with each possible template using an SAD similarity measure. The pattern with the smallest difference is chosen.

Assuming the orientations of image plane and HMD-screen are aligned and the captured image is a reflection, allows to align the orientations of the displayed and detected grids. Given the orientation and location, each point on the detected grid can be matched with its 3D coordinate on the screen. After correspondence matches have been computed, the estimation of the cornea position and the subsequent (re-)estimation of the eye center can be conducted in real-time.

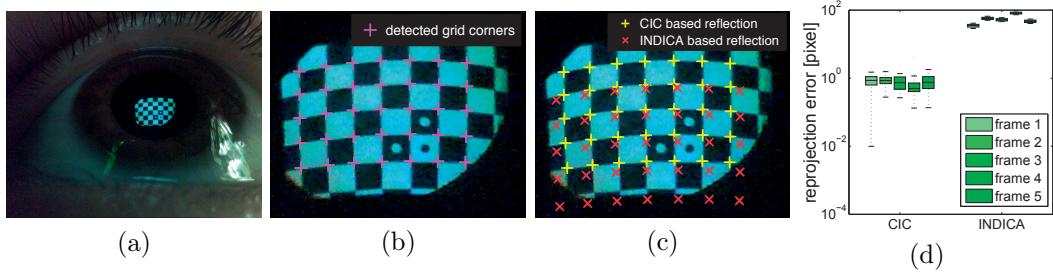


Figure 4.6: Cornea position estimation evaluation from reprojection errors of the reflected checkerboard corners into the camera. (a) Image of the reflected checkerboard shown on the HMD-screen. (b) Detected checkerboard corners in the image. (c) Reprojected corners using corneal sphere positions estimated by INDICA and CIC. (d) The large reprojection error for INDICA persists throughout the sequence (Note the logarithmic scale of the error).

#### 4.4.3 Cornea Position Estimation

The first step of an automated calibration method estimates the position of the cornea. In this section, we compare the results of the HMD-screen reflection on the cornea, estimated by CIC and INDICA. Both methods assume the same two-sphere eye-model and should obtain the correct cornea position. Therefore, the reflection of the checkerboard on the estimated cornea should project onto the detected points, for example those shown in Figure 4.6b. As can be seen in Figure 4.6c, the reflection on the cornea estimated by CIC almost coincides with the expected points, while the reflection for INDICA greatly differs. The large error is consistently obtained throughout the experimental evaluation (Figure 4.6d) with an average error of 54.936 pixel. This indicates that the eye position estimated by INDICA is only a rough approximation of the actual eye center.

#### 4.4.4 Eye Position Estimation

The minimal solution for the calibration is error prone due to outliers and ill-posed data. Although it would be ideal to collect a large dataset for calibration, this is not possible in most application scenarios. Also, an automated calibration should require as little time as possible before the user can experience the correct augmentation whenever the application is started or the HMD has moved on the head. Therefore, it is also necessary to determine the minimum dataset size that allows a reliable calibration.

In this section we evaluate the required dataset size for the SPAAM, INDICA and CIC calibrations. With an increasing number of frames each method will converge towards a stable position. We assume that our recorded datasets are large enough for each method to successfully converge onto a position that we define as ground truth. We then select 100 random com-

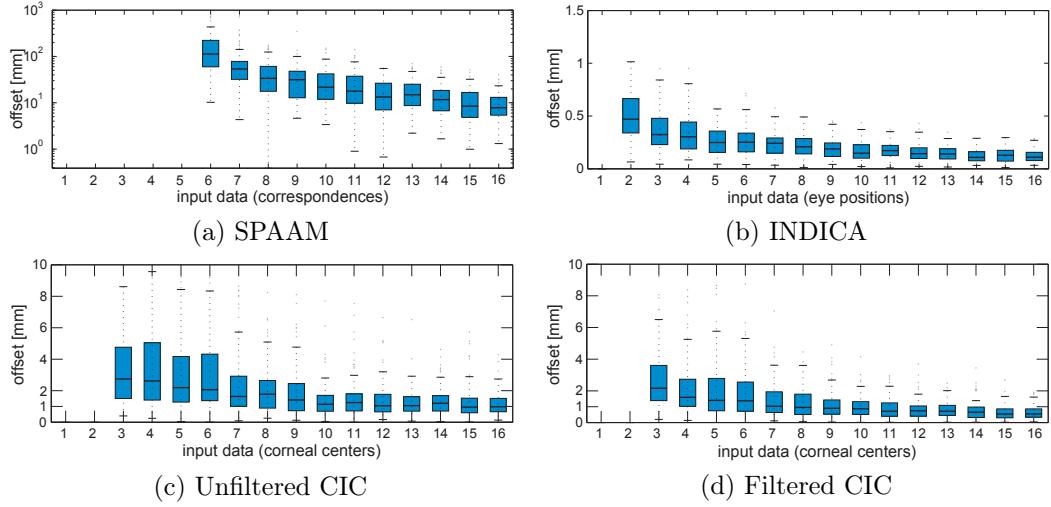


Figure 4.7: The convergence rate to a stable eye position determines the time required for the calibration, i.e., how fast a usable result can be obtained. We show the convergence rate for (a) SPAAM (in logarithmic scale), (b) INDICA, (c) CIC with unfiltered data and (d) CIC with a preprocessing step, where we discard probable outliers. We remove all cornea centers, which result in a screen reflection reprojection error  $> 2$  pixel. Although CIC converges slower than INDICA, it converges to an expected error of  $< 1$ mm within 7 frames in (d) and 16 frames in (c). SPAAM calibration fails to converge to an acceptable value even after 16 correspondences are used.

binations of  $n$  input data (2D–3D correspondences for SPAAM, frames for INDICA and CIC) from each dataset and evaluate the deviation from the ground truth. The results are shown in Figure 4.7. We use at least 6 point pairs for SPAAM, 2 frames for INDICA and 3 frames for CIC. In contrast to any of the three methods, CIC allows us to evaluate the quality of the input data, as the reprojection error  $e$  of the screen reflection on the estimated cornea position is known. Therefore, we assume that all frames with  $e > 2$  pixel are likely to be outliers and remove them from the estimation to further improve the results. Stricter thresholds will naturally speed up the convergence rate at the cost of an increasing number of discarded frames. For CIC we show the results of both, the filtered and unfiltered data. Since the reprojection error for approximately 80% of the recorded frames amounts below the threshold, we believe that it is viable to employ filtering in an application scenario. Our observations show that the converged position of the eye-center estimation for the unfiltered dataset deviates by approximately 0.5mm from  $\tilde{\mathbf{E}}_F$ , the result of the filtered scenario. In some cases, it may not be possible to use filtering due to a generally large reprojection error. Therefore, we observe the deviation of the estimated eye position from  $\tilde{\mathbf{E}}_F$  for the general case in Figure 4.7c.

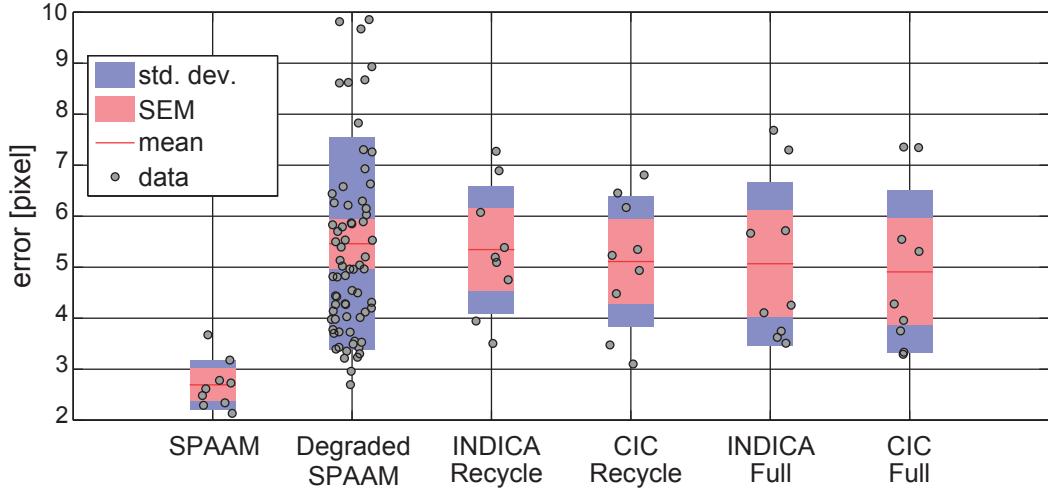


Figure 4.8: Projection error for various calibration approaches. CIC performs slightly better than INDICA for both, Recycle and Full calibration approaches. SEM stands for the standard error of the mean.

The position estimated by SPAAM remains unstable even with 16 samples, with an error of approximately 10 mm. INDICA immediately converges to an error of less than 1mm, which is faster than our method that requires 7 and 16 frames for the filtered and unfiltered datasets, respectively. The high quality of INDICA results from a similar gaze direction for the frames used to estimate the eye center, as the iris detection requires the contour of the iris to be visible. Our method, on the other hand, does not restrict the gaze direction and performs similarly. Furthermore, as is shown in [Schnieders et al. \(2010\)](#), the approach taken by INDICA is very sensitive to erroneous measurements and converges to an incorrect eye center position.

#### 4.4.5 Projection Error

In this section we use point correspondences from the second part of the recording session to evaluate the projection error for various setups and calibration methods. We compare the SPAAM, INDICA and CIC calibration methods. For the automated calibration methods we evaluate the Recycle and Full calibration approaches that were discussed in Section 3.3.

The results of the projection are shown in Figure 4.8. Similar to results reported in [Itoh and Klinker \(2014a,b\)](#), the SPAAM calibration has the smallest projection error among the compared methods. This result is expected, as SPAAM incorporates inaccuracies resulting from, e.g., user errors or distortion of the screen into the projection matrix. Our method performs better than INDICA for both, the Recycle and Full setup. The similar results achieved by INDICA and CIC can be explained by the small deviation of the estimated eye centers. The estimated positions are on average 2.03 mm apart. The eye

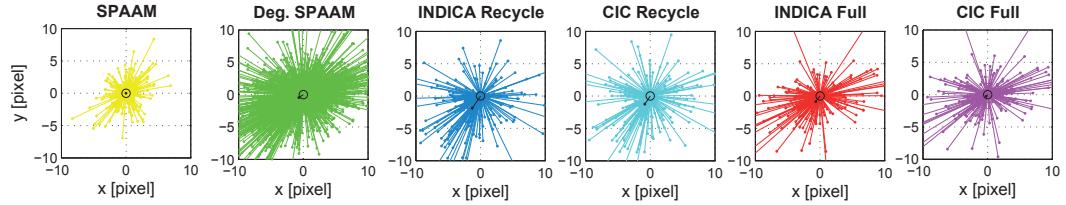


Figure 4.9: Error vector distribution for each evaluated method. Each projection of a 3D point describes a vector from the estimated position to the position aligned by the user. The mean error vectors are shown in black. The position estimation by our method improves the error distribution for both, the Recycle and the Full calibration.

is modeled as a pinhole camera into which we project points at a distance of 2-3 m. In this model, an offset of 2.03 mm does not introduce a substantial error. Additionally, while the shift in the position, compared to CIC, will degrade the results for parts of the dataset (e.g., points projected to the left of the correct position) it will reduce the error for other parts (e.g., points projected to the right of the correct position), thus disguising the degraded performance. Such an erroneous shift can be observed in the error distribution of the point projections. For each 3D point  $\mathbf{P}$  we compute an error vector  $\mathbf{e} = \mathbf{p} - \mathbf{p}_u$ , where  $\mathbf{p}$  is the projection of  $\mathbf{P}$  onto the screen, after the calibration, and  $\mathbf{p}_u$  is the point aligned by the user. For each calibration method we use  $n$  such projections to compute the mean vector

$$\mathbf{e}_0 = \frac{1}{n} \sum_{i=1 \dots n} \mathbf{e}_i, \quad (4.4)$$

where  $\mathbf{e}_i$  is the error vector for point  $i$ . We show the error vectors for each calibration approach in Figure 4.9. As expected, the SPAAM algorithm computes an ideal distribution of the errors. INDICA shows a strong error tendency, while CIC shows a much more uniform result. The remaining vector  $\mathbf{e}_0$  may result from inaccuracies in the eye model, user errors while aligning the screen with the 3D points, or sub-optimal HMD-screen calibration.

#### 4.4.6 Discussion

According to [Villanueva and Cabeza \(2008\)](#) it is possible to estimate the size of the corneal sphere from two point correspondences. As shown in our simulation we expected that under the assumption of a correct eye model and a small error in the detection and screen calibration, it is possible to estimate the radius of the corneal sphere as part of the OST-HMD calibration. However, as shown in Figure 4.10, the global minimal reprojection error is independent of the radius. In particular, for the same eye but different sequences a different minimum can be found. In the following we explain the observed phenomenon.

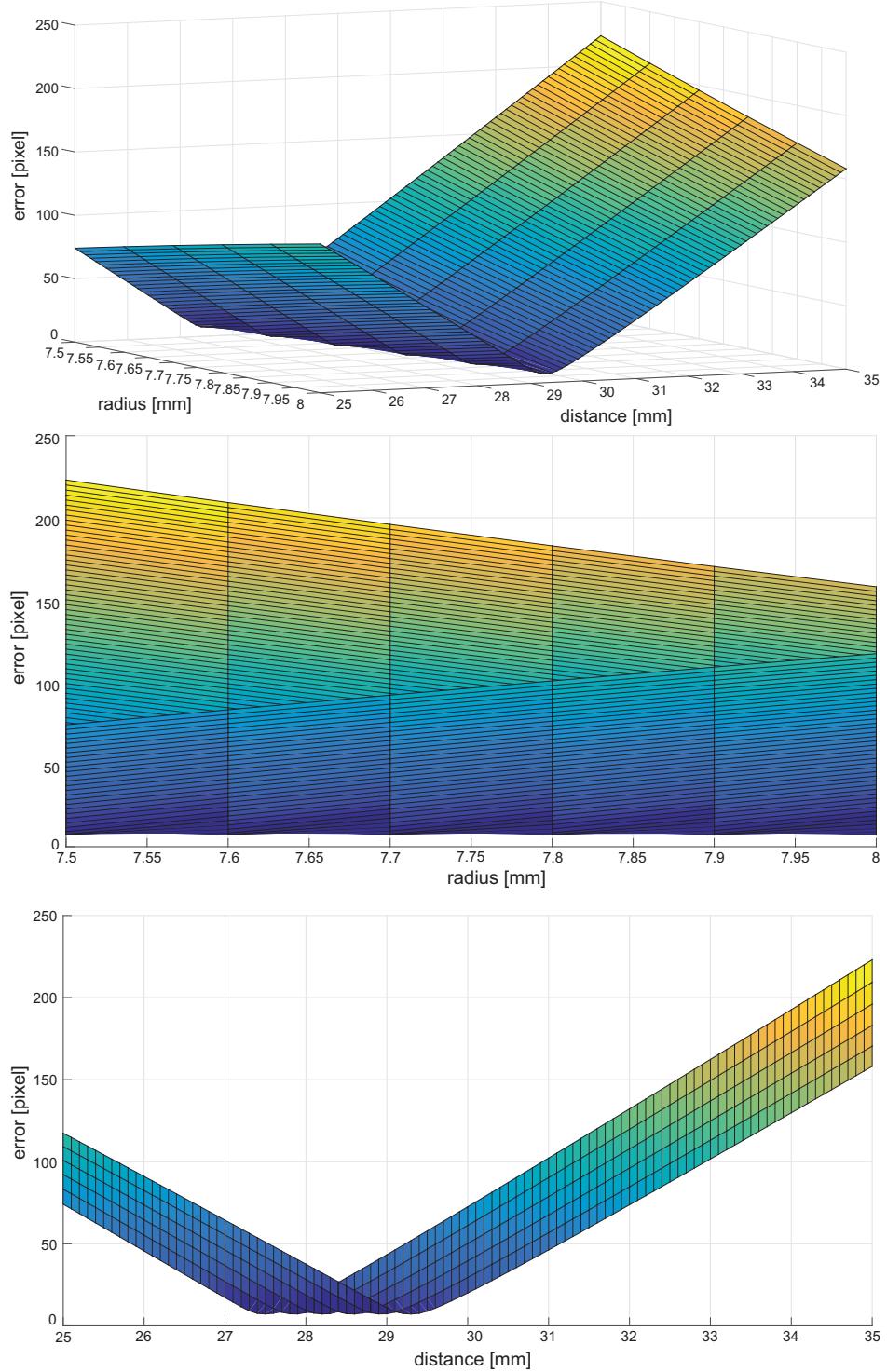


Figure 4.10: (Top) The alignment error as a function of  $d_{TC}$  and  $r_C$  does not display a global minimum. (middle) A view in dependence on  $r_C$  and (bottom) in dependence on  $d_{TC}$ .

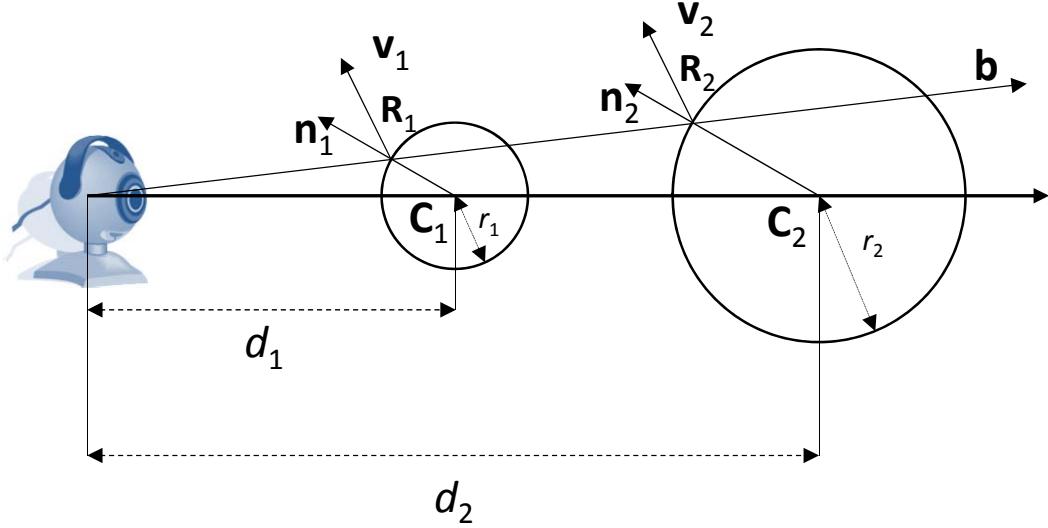


Figure 4.11: Schematic representation of the reflection of a backprojected ray on corneas of different size along the same ray.

W.l.o.g. let the camera  $T$  be positioned at  $\mathbf{T} = (0, 0, 0)^T$  observing a sphere  $C_1$  with a center  $\mathbf{C}_1 = (0, 0, d_1)^T$  and a radius  $r_1$ .

A ray  $\mathbf{b}$  originating at  $\mathbf{T}$  will reflect on  $C_1$  in a point  $\mathbf{R}_1$  as a ray  $\mathbf{v}_1$  (Figure 4.11). To determine  $\mathbf{R}_1$  and  $\mathbf{v}_1$  we compute the following:

$$s = \mathbf{b}^T \mathbf{C}_1 \quad (4.5)$$

$$s_r = s - \sqrt{s^2 - \mathbf{C}_1^T \mathbf{C}_1 + r_1^2} \quad (4.6)$$

$$\mathbf{R}_1 = s_r \mathbf{b} \quad (4.7)$$

$$\mathbf{n}_1 = \frac{(\mathbf{R}_1 - \mathbf{C}_1)}{\sqrt{(\mathbf{R}_1 - \mathbf{C}_1)^T (\mathbf{R}_1 - \mathbf{C}_1)}} \quad (4.8)$$

$$\mathbf{v}_1 = \mathbf{b} - (2\mathbf{n}_1^T \mathbf{b})\mathbf{n}_1 \quad (4.9)$$

Now let us observe a second sphere  $C_2$  located at  $\mathbf{C}_2 = (0, 0, d_2)^T = (0, 0, \frac{r_2}{r_1} d_1)^T$ , where  $r_2$  is the radius of  $C_2$ .

Given the Equations (4.5)–(4.9), we can determine the reflection of  $\mathbf{b}$  on  $C_2$  with

$$s_2 = \mathbf{b}^T \mathbf{C}_2 = \mathbf{b}^T \mathbf{C}_1 \frac{r_2}{r_1} = s \frac{r_2}{r_1} \quad (4.10)$$

$$s_{r_2} = s_2 - \sqrt{s_2^2 - \mathbf{C}_2^T \mathbf{C}_2 + r_2^2} \quad (4.11)$$

$$= s \frac{r_2}{r_1} - \sqrt{s^2 \frac{r_2^2}{r_1^2} - \mathbf{C}_1^T \mathbf{C}_1 \frac{r_2^2}{r_1^2} + r_2^2 \frac{r_2^2}{r_1^2}} \quad (4.12)$$

$$s_{r_2} = s \frac{r_2}{r_1} - \sqrt{s_2^2 - \mathbf{C}_1^T \mathbf{C}_1 + r_1^2 \frac{r_2^2}{r_1^2}} = s_r \frac{r_2}{r_1} \quad (4.13)$$

$$\mathbf{R}_2 = s_{r_2} \mathbf{b} = s_r \mathbf{b} \frac{r_2}{r_1} = \mathbf{R}_1 \frac{r_2}{r_1} \quad (4.14)$$

$$\mathbf{n}_2 = \frac{(\mathbf{R}_2 - \mathbf{C}_2)}{\sqrt{(\mathbf{R}_2 - \mathbf{C}_2)^T (\mathbf{R}_2 - \mathbf{C}_2)}} \quad (4.15)$$

$$= \frac{(\mathbf{R}_1 \frac{r_2}{r_1} - \mathbf{C}_1 \frac{r_2}{r_1})}{\sqrt{(\mathbf{R}_1 \frac{r_2}{r_1} - \mathbf{C}_1 \frac{r_2}{r_1})^T (\mathbf{R}_1 \frac{r_2}{r_1} - \mathbf{C}_1 \frac{r_2}{r_1})}} \quad (4.16)$$

$$= \frac{(\mathbf{R}_1 - \mathbf{C}_1) \frac{r_2}{r_1}}{\sqrt{\frac{r_2}{r_1} (\mathbf{R}_1 - \mathbf{C}_1)^T (\mathbf{R}_1 - \mathbf{C}_1) \frac{r_2}{r_1}}} \quad (4.17)$$

$$= \frac{(\mathbf{R}_1 - \mathbf{C}_1) \frac{r_2}{r_1}}{\frac{r_2}{r_1} \sqrt{(\mathbf{R}_1 - \mathbf{C}_1)^T (\mathbf{R}_1 - \mathbf{C}_1)}} = \mathbf{n}_1 \quad (4.18)$$

$$\mathbf{v}_2 = \mathbf{b} - (2\mathbf{n}_2^T \mathbf{b}) \mathbf{n}_2 = \mathbf{b} - (2\mathbf{n}_1^T \mathbf{b}) \mathbf{n}_1 = \mathbf{v}_1 \quad (4.19)$$

From Equation (4.19) it follows that the reflected rays are parallel. Let  $\mathbf{P}$  be the 3D point used in the error function defined in Equation (2.7). As this point is located very far away from estimated corneal sphere, the displacement of the corneal sphere due to the incorrectly estimated radius of the corneal sphere produces only a relatively small error that is outweighed by other error sources, such as the assumption of a planar HMD-screen.

## 4.5 Conclusion

We have presented CIC, a novel approach for automated spatial calibration of an OST-HMD. The method employs corneal imaging instead of iris detection to determine the position of the user's eye. We use an HMD with pre-calibrated camera and screen positions to establish correspondences of points displayed on the screen and their reflections on the cornea, as captured by an eye-tracking camera. The correspondences allow to compute the position of the user's cornea, and at least three frames with a moving cornea allow to compute the position of the user's eye center. We showed that the position estimated by CIC is closer to the real position, which improves the calibration results. The proposed method is suitable for accurate online calibration of an OST-HMD and can be used to address the spatial consistency problem of such devices.

## Limitations and future work.

CIC requires an unobstructed light-path for the HMD-screen reflecting at the user's eye into the camera. Our observations show that large contact lenses, that cover the cornea entirely, or almost entirely, do not impact the estimation process. However, smaller contact lenses result in incorrectly detected reflection of the HMD screen, due to the curvature of the lens. This also

prevents the application of this method in HMDs that use a lens to provide a large FOV, e.g., [Oculus Rift \(2015\)](#). In the future modelling of the light refraction by the focusing lens may enable the application of the method in scenarios where the scene is distorted by optical surfaces, such as the lenses or the optics of the OST device.

CIC requires multiple observations of the eye rotating around the same eye center, which limits the applicability to HMDs. As such, the estimation is not possible if the user is looking in the same direction and requires a short period before a stable eye center has been recovered. Furthermore, if the shift of the OST-HMD is only very small, CIC may fail to detect it. In order to use the method in more general scenarios, such as user perspective rendering in WoW scenarios ([Tomioka et al. \(2013\)](#)) a per-frame estimation is required. The center of the projection of the eye is the nodal point of the eye and can be assumed to coincide with the center of the corneal sphere. In the future it is necessary to evaluate if estimation of the corneal sphere, rather than the center of the eye results in a smaller reprojection error.

The results of the proposed method are still not ideal, as CIC does not outperform the manual calibration by SPAAM. In our simulation, we determined that errors in the 3D location of the correspondences and deviation of the cornea size are major sources of the error. Another reason may be that the acquired ground truth information was affected by the optical distortion of the light paths through the HMD optics. Accounting for this non-planar deformation ([Itoh and Klinker \(2015a,b\)](#)) may significantly improve the results. Finally, the method should be evaluated in combination with different models of the screen, e.g., [Oike et al. \(2004\)](#); [Klemm et al. \(2014\)](#); [Itoh and Klinker \(2015b\)](#), to determine which produces the best results.

The following chapters of this dissertation address some of the issues left unanswered by our evaluation.

The proposed calibration method can theoretically be applied to estimate the gaze direction, thus enabling automated, light weight gaze estimation as  $\mathbf{o} = \mathbf{C} - \mathbf{E}$ . In practice, this approach requires a very accurately estimated position of  $\mathbf{E}$  and  $\mathbf{C}$ , thus even slight errors in the recovery of the eye center lead to large errors in the estimated gaze direction. As the iris-based approach is unreliable we have developed a new method to recover the eye-pose in environments that are not illuminated by IR light. The proposed approach is presented in Chapter 5.

In our evaluation we used a predetermined pattern, which would have to be displayed on a high frequency display for it not to be noticeable to the users. This is an impractical requirement and a more general method that does not require specialized hardware and can account for varying content shown on the OST-HMD is desirable. We present a solution suitable for continuous tracking of the corneal sphere from the reflection of the content shown on the OST-HMD screen in Chapter 6.

Finally, although CIC is performing worse than SPAAM, a subjective

study of the user's performance has shown that the quantitatively worse INDICA method, outperformed SPAAM in both the user's preference and the qualitative evaluation of the results (Moser et al. (2015)). As such, it is important to answer, how accurate the alignment of virtual and real content has to be before users can no longer distinguish between correctly aligned and misaligned content. We evaluate this in Chapter 7.



## CHAPTER 5

# Hybrid Eye-pose Estimation

---

This chapter covers our proposed estimation of the eye-pose in images taken under natural illumination, called Hybrid eye-pose estimation.

Section 5.1 introduces the topic of eye-pose and point-of-regard estimation. Section 5.2 describes the proposed hybrid method that uses corneal imaging to combine the benefits of active and passive estimation. Section 5.3 explains the evaluation of our method. We conclude with a summary and outlook in Section 5.4.

## 5.1 Introduction

Estimation of the eye-gaze can tell if the user is looking in a particular direction or at an object. This information can be explored for better understanding of the user’s interest and intentions (Jacob (1990); Bulling and Gellersen (2010)), or evaluate the focus and attention during a task (Lee et al. (2007)). In combination with interactive surfaces, e.g., displays, the point-of-regard (POR), the estimated point gazed upon, can be used for interaction guidance, e.g., for impaired participants, or even to trigger interaction through predetermined gaze motion (Toyama et al. (2014)).

**Eye-gaze in AR and VR.** POR estimation is also of interest for interaction with content in AR and VR environments (Duchowski et al. (2000); Nilsson et al. (2009); Ajanki et al. (2011); Ishiguro and Rekimoto (2011)). Environments with a large number of virtual content may suffer from the clutter problem (Ferrer et al. (2013)). In such environments eye-gaze tracking can be used to reduce the number of augmented objects, e.g., the virtual content is shown only if the user is looking closely to its location. Alternatively, the estimated POR can be used to influence the virtual content, e.g., virtual characters reacting to the user’s gaze, or use resources to render the focused area in higher quality and reduce the rendering quality of content in the peripheral vision. Although not applied in current designs, more resources can be spent to address issues such as transparency, color inconsistency, and occlusion at the focused area. Finally, by combining eye trackers, the outwards facing camera and the information shown on the display it is possible to acquire data on personal experience, e.g., in form of a life log (Nakazawa et al. (2015)).

Various designs envision eye-tracking capabilities for OST- (Ishiguro and Rekimoto (2011); Hua et al. (2013)) and VST-HMDs (SensoMotoric Instru-

ments GmbH (SMI) (2015); Fove Inc. (2015)). Current devices require a touchpad, either attached to the device or remove, to control the interaction point. Eye-gaze-based pointing may replace these in various application scenarios (Park et al. (2008); Ishiguro and Rekimoto (2011); Orlosky et al. (2015)), although some drawbacks still remain, e.g., the Midas touch problem (Jacob (1995)). Midas touch problem describes the problem of switching between passive content modification through eye-gaze tracking and active interaction with the content gazed upon, to prevent unintentional interaction.

**Eye-pose estimation methods.** Eye-pose and the POR can be estimated by a large variety of approaches the selected approach often depends on the targeted application, e.g., is highly accurate gaze direction estimation required or is a more coarse estimation sufficient, does the application require the geometric eye-pose or the POR, is the environment indoors or outdoors, is the system wearable or remote, active or passive approach. A pupil illuminated by an IR LED positioned closely to the camera (on-axis) will appear as a bright area in the captured image. If the IR LED is positioned far from the camera’s axis, the pupil will appear dark instead. Systems that exploit IR illumination for robust segmentation of the eye are referred to as active. These methods have been shown to provide highly accurate and robust results in indoor environments and are the SOTA approach for commercial systems. However, as the methods require controlled illumination they have difficulties under visible light conditions. Others argue that the long-term impact of the IR LEDs still remains unknown (Mulvey et al. (2008)). Passive methods have been developed to address these issues. These methods do not require IR light and process images taken under natural illumination.

**Proposed method.** Most proposed HMD systems use IR trackers for user gaze estimation. This limits their applicability to indoor environments and also imposes a high cost. Spatially calibrating the IR tracker relative to the HMD screen is a complicated process and it is not viable for users to do so on their own. Some trackers bypass this through session and user specific calibrations that associate the location of detected eye features with the gaze direction and POR. However this calibration has to be repeated every time the HMD slips or is being taken off and put on again.

Passive methods, use a simpler setup with standard cameras. A tracker that utilizes passive eye tracking can be used outdoors and it’s calibration relative to the screen is a much simpler process (see Section 3.2). By focusing the camera onto the corneal reflection in the eye, it is also possible to detect the reflection of the screen and the scene. In Section 4.4 we have verified that the reflection of a modelled HMD-screen can be used to estimate an accurate position of the corneal sphere. The proposed CIC method assumes that the position of the eye relative to the tracker remains stable and uses multiple

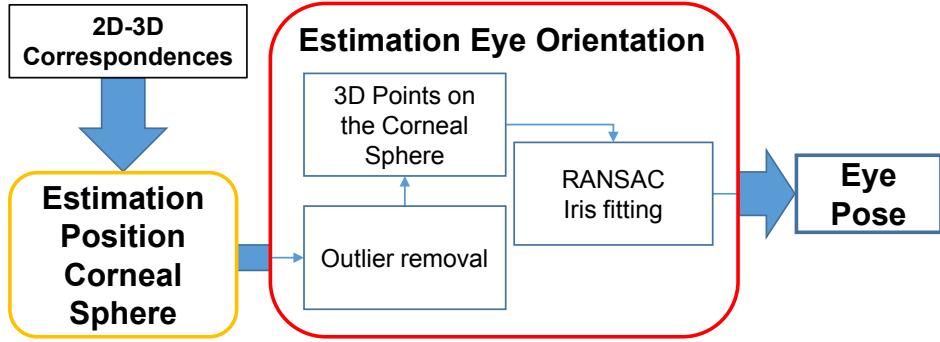


Figure 5.1: Our method estimates the eye-pose in two steps. First we use detected 2D–3D correspondences to estimate the position of the corneal sphere. With this information we can recover the orientation of the eye. The result is an accurate eye-pose.

observations of the corneal sphere to recover the center of the eye. Although in the discussed scenario the optical axis of the eye can be recovered as a ray originating in the eye center towards the corneal center, small errors in the estimation will result in large gaze estimation errors. In this chapter we extend the proposed method to recover the eye-pose from a single image, as a combination of the estimated corneal sphere and the optical axis of the eye that is estimated from the detected iris contour. The proposed method is not limited to static scenarios and can thus be used in any applications where the scene model relative to the eye-tracking camera is known.

## 5.2 Approach

Passive methods recover the eye-pose by fitting the eye model to the extracted iris contour. We use the inverse approach (Figure 5.1) that is similar to the estimation pipeline used in PCCR. We first recover an accurate position of the corneal sphere as described in Chapter 6. The result of this estimation provides the translational parameters of the eye. The accurately estimated position of the corneal sphere is used to determine the rotational parameters of the eye. As we assume a known scene model, the recovered eye-pose can be used to compute the POR by intersecting the estimated gaze with the scene model.

In PCCR, the orientation of the eye is computed as the ray through the centers of the cornea and the pupil. In images captured under visible light, the pupil cannot be detected reliably. The reflection of the scene occludes the pupil in Figure 5.2a, but the contour of the pupil remains visible in Figure 5.2b. Therefore, we estimate  $\mathbf{L}$  as the center of the iris. Iris detection in the camera image suffers from erroneously detected edges. While the iris contour is clearly visible in Figure 5.2a, reflections on the cornea occlude a

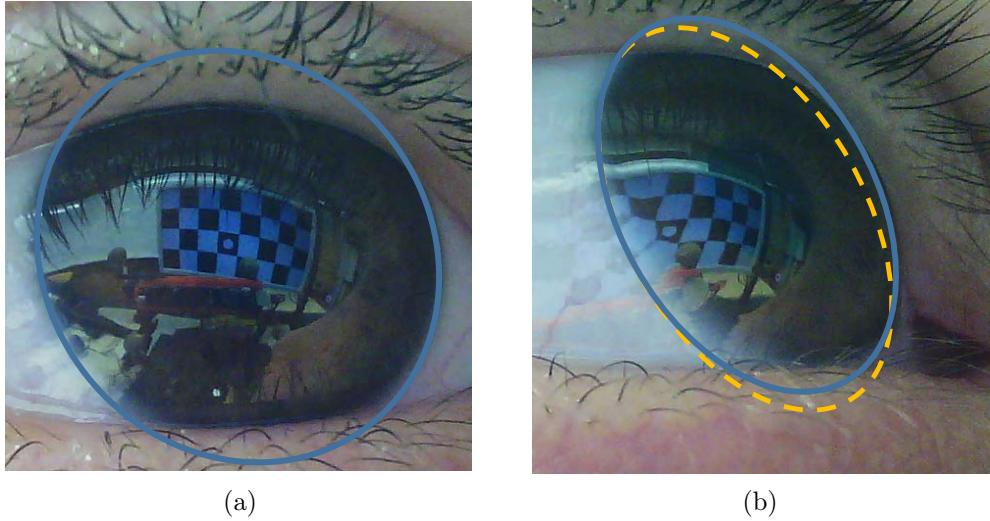


Figure 5.2: (a) When the user looks towards the camera the contour of the iris is clearly visible and the correct iris contour is easily recovered. We show the estimated iris contour in blue. (b) As the eye rotates sideways, reflections on the corneal sphere occlude the iris contour shown in orange. Naïve ellipse fitting assumes that the occluding contour of the cornea is part of the iris.

portion of it in Figure 5.2b. These situations are indistinguishable without 3D constraints. We use the 3D corneal sphere to improve the fitting results and recover a closer representation, which accounts for both cases. Additionally, our approach is robust against other detected edges, such as eye-lids, eye-lashes, sclera and iris patterns, and reflections on the corneal surface. In this section, we describe how we use the accurately estimated corneal sphere to determine the orientation of the eye from edge points detected in the captured image.

Given  $n$  edge points  $\mathbf{p}_i$ ,  $i = 1 \dots n$ , detected in the image, we remove all obvious outliers by intersecting the backprojected rays  $\mathbf{u}_i$  with the corneal sphere. For an inlier point  $\mathbf{p}_i$ ,  $\mathbf{u}_i$  intersects the corneal sphere in  $\mathbf{R}_i$ .

For  $m$  points on the 3D sphere, we determine  $\mathbf{o}$  and  $d_{\text{CL}}$  through a RANSAC approach. From the  $m$  3D points, we select  $l \geq 3$  candidate points and fit the limbal plane to them. The estimated limbal plane intersects the corneal sphere in the corneal limbus. Therefore, the normal of the limbal plane will correspond to  $\mathbf{o}$  and  $d_{\text{CL}} = \mathbf{o}^T(\mathbf{R}_k - \mathbf{C})$ , where  $\mathbf{R}_k$  is one of the candidate points. We determine the support of the estimated limbal plane by counting the number of inlier points  $\mathbf{R}_i$ ,  $i \in m$ . An inlier of the fitted limbal plane satisfies one of the following conditions:

$$\|\mathbf{R}_i - \mathcal{P}_L\| < t_1, \text{ or} \quad (5.1)$$

$$\left| \mathbf{u}_i^T \frac{\mathbf{R}_i - \mathbf{C}}{\|\mathbf{R}_i - \mathbf{C}\|} \right| < t_2, \quad (5.2)$$

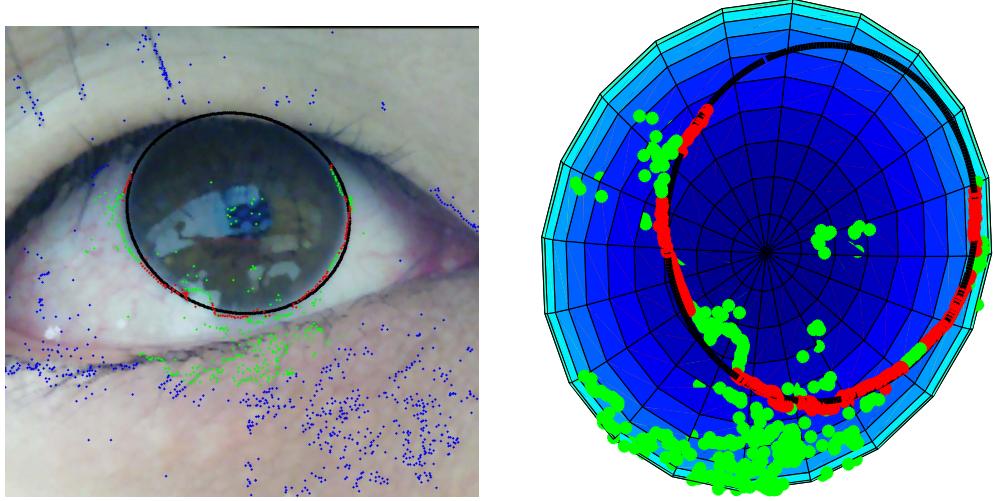


Figure 5.3: Points which do not lie within the projection of the corneal sphere into the image are removed as outliers (blue). From all points on the corneal sphere (green), the corneal limbus (black) is the ring which is supported by the highest number of points on the corneal sphere surface (red). The fitting results are shown in the image on the left, and the corresponding 3D sphere on the right.

where  $t_1$  and  $t_2$  are user-defined inlier thresholds, and  $\mathcal{P}_L$  is the 3D limbus. If  $\mathbf{R}_i$  satisfies Equation (5.1), the point is lying at most  $t_1$  away from the limbus contour.  $\mathbf{R}_i$  will satisfy Equation (5.2), if the eye is oriented so that the cornea is occluding a portion of the iris, as in Figure 5.2b. In this case,  $\mathbf{p}_i$  lies at the edge of the projection of the corneal sphere into the image. After the best inlier subset has been selected, we perform the fitting step again with all inlier points. We use the following empirically selected thresholds:  $t_1 = 0.3$  mm, and  $t_2 = 5^\circ$ . We show a sample result of the fitting process in Figure 5.3.

## 5.3 Experiment

We have implemented our Hybrid method in C++ on an Intel i7-7000 with 32 GB RAM. Our implementation recovers the eye-pose in less than 0.6 s/frame (0.1-0.3 s for checkerboard detection and matching, 0.1-0.25 s for estimation of the position of the corneal sphere and 0.05 s for estimation of the orientation).

### 5.3.1 Environment Calibration

We evaluate the accuracy of the eye-pose estimated by our method in a simple environment shown in Figure 5.4a. The users were shown a  $8 \times 4$  checkerboard

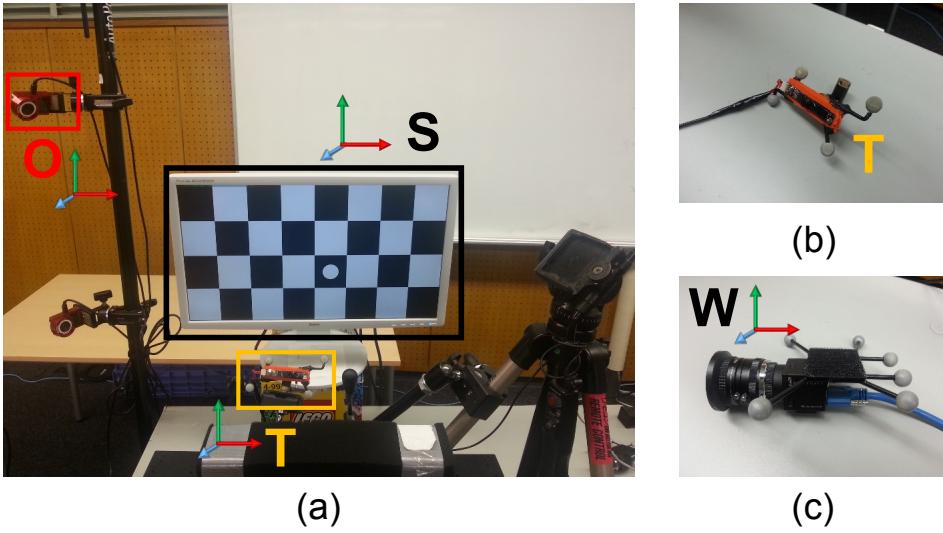


Figure 5.4: (a) Our experiment environment consists of an OptiTrack system, an eye-tracking camera and an LCD monitor. (b) The camera  $T$  is mounted onto a tripod, and its position can be continuously adjusted. IR markers attached to  $T$  allow us to continuously track the pose of the environment  $S$  relative to  $T$ . (c) We compute an accurate position of  $S$  with a second camera, which is tracked by the OptiTrack system.

pattern on an LCD monitor  $S$  ( $293.2 \times 521.3$  mm) that was positioned at a variable distance in front of the user. We use a Delock USB 2.0 camera with a  $64^\circ$  lens focused at a 5-7 cm distance as the eye-tracking camera  $T$  (Figure 5.4b). We mount the camera onto an adjustable mount and adjust its position for each user. To track the camera pose, we have attached IR-reflective markers that can be tracked by an OptiTrack tracking system to  $T$ . We use Ubitrack (Huber et al. (2007)) to calibrate the transformation  ${}^T_W T$  which transforms a point  ${}^W P$  in the OptiTrack coordinate systems to  ${}^T P = {}^T_W T {}^W P$ , the point  $P$  in the coordinate system of  $T$ .

We reconstruct  ${}^W P_S$ , the position of the checkerboard corners relative to  $W$ , with a PointGrey FL3-U3-13S2C-CS camera  $C$  with IR-markers attached to it (Figure 5.4c). We use Ubitrack again to compute  ${}^W T$ . We show the checkerboard on the monitor screen and detect the corners in images taken by  $C$ . We repeat this step for different camera poses.  ${}^W P_S$  is the intersection of the backprojected rays from all images taken by  $C$ .

### 5.3.2 Evaluation

We compare our Hybrid method with the method presented by Itoh and Klinker (2014a) (IK), as both methods are designed to recover the eye-pose from extracted ellipse edge points. We acquire 2D–3D correspondences of points in the camera image and the scene as described in Section 4.4. The iris

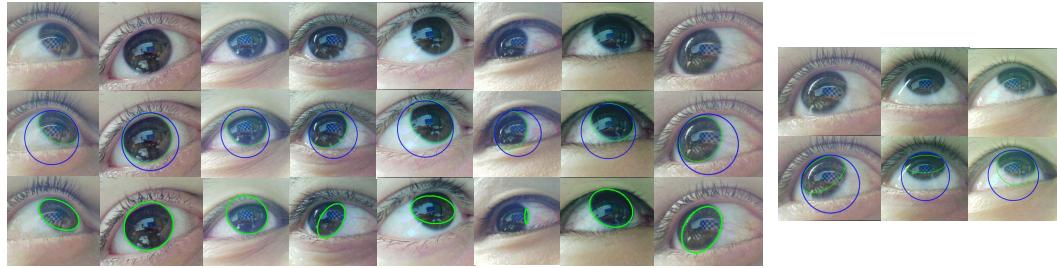


Figure 5.5: Results of the iris estimation. (top row) We show the cropped eye region within the captured images. We show the recovered iris contour with our method (middle row) and by Itoh and Klinker (2014a) (bottom row). Our method successfully recovers the iris boundary for most cases. We show some of failure cases to the right.

contour is recovered from edge points detected by IK. Our method recovers the eye-pose from all detected edge points. For IK, we manually select a ROI slightly larger than the iris contour.

We conduct our evaluation on four male participants (two Asians, two Europeans; 22-31 years old with no vision impairments (participant 2 underwent a laser surgery)). The participants were asked to look at each inner corner of the checkerboard. For each participant, we recorded two sessions. Between the sessions, we changed the distance to the monitor, the position of the user’s head and the eye-tracking camera. The distance to the monitor was 40 cm and 90 cm. Some estimation results are shown in Figure 5.5.

### 5.3.2.1 Personal parameter estimation

Although our method can estimate all relevant parameters of the model, we found that imprecisions in the corner detection and the fact that the cornea is not an ideal sphere, resulted in ambiguous solutions for  $r_C$ . Additionally, as the eye is located very closely to the camera  $T$ , changes in the cornea size did not impact the results of our method. Therefore, we use  $r_C = 7.8$  mm and estimate  $d_{CL}$ . Note, that the estimated  $d_{CL}$  is up to scale of  $r_C$ . We show the results of the estimation in Table 5.1, where  $r_L = \sqrt{r_C^2 - d_{CL}^2}$ . For all participants, our method estimates that the size of the iris is as large or larger than the values assumed in Nitschke (2011). We believe that this is a result of the gradual transition of the cornea into the sclera and the assumption that the iris and limbus are identical. This signifies the importance of the estimation of personal parameters. Our method estimates a stable radius  $d_{CL}$  for each recorded session. However, in the case of participants 1 and 4, this distance varied by more than 0.2 mm between the sessions. This suggests that the size has to be re-estimated for the conditions present during the eye-pose estimation.

Table 5.1: Estimated personal parameters and eye-pose error.

Sample eye images Participant	1	1	2	2	3	3	4	4
Distance to the monitor[cm]	90	40	90	40	90	40	90	40
Personal Parameters								
$d_{\text{CCL}}, r_L$ [mm]	(4.60, 6.29)	(5.06, 5.92)	(5.03, 5.96)	(5.11, 5.87)	(5.14, 5.85)	(5.08, 5.91)	(4.97, 6.00)	(5.45, 5.56)
stddev $d_{\text{CCL}}, r_L$	(0.11, 0.8)	(0.37, 0.34)	(0.14, 0.12)	(0.38, 0.35)	(0.29, 0.27)	(0.26, 0.23)	(0.21, 0.18)	(0.19, 0.18)
$\text{IK}(\alpha, \beta)$	(0.09, 10.35)	(1.66, 7.47)	(6.31, 4.48)	(1.38, 4.73)	(0.64, 1.23)	(2.65, 12.93)	(1.88, 4.91)	(0.57, 3.53)
$\text{HF}(\alpha, \beta)$	(1.98, 3.84)	(3.04, 2.07)	(2.42, 1.99)	(2.49, 0.29)	(1.32, 0.79)	(0.72, 0.83)	(2.65, 1.77)	(2.40, 0.10)
Error optical axis								
IK (mean, std)	(11.65, 3.67)	(8.97, 3.55)	(8.01, 1.40)	(6.89, 3.80)	(2.98, 1.44)	(12.57, 5.02)	(8.32, 3.45)	(6.05, 4.01)
HA (mean, std)	(4.24, 1.06)	(4.64, 1.21)	(3.46, 0.78)	(3.55, 1.50)	(1.89, 1.04)	(3.09, 1.83)	(3.39, 0.75)	(3.88, 1.40)
HF (mean, std)	(4.43, 1.03)	(4.32, 0.68)	(3.72, 1.19)	(3.23, 1.03)	(1.80, 0.69)	(2.68, 1.37)	(3.48, 0.66)	(3.21, 1.42)
Error visual axis								
IK (mean, std)	(4.03, 2.39)	(5.87, 4.30)	(2.76, 1.15)	(4.31, 3.07)	(2.70, 1.41)	(6.14, 5.07)	(7.29, 3.47)	(5.10, 3.48)
HA (mean, std)	(2.07, 0.79)	(3.07, 1.42)	(3.06, 1.25)	(2.24, 1.37)	(1.45, 0.67)	(2.69, 1.66)	(1.65, 0.69)	(3.23, 1.55)
HF (mean, std)	(1.42, 0.71)	(1.65, 1.08)	(2.00, 1.54)	(1.84, 1.06)	(1.28, 0.62)	(2.22, 1.02)	(1.20, 0.72)	(1.43, 0.88)

best performing method  
second best performing method

### 5.3.2.2 Eye-pose estimation

We compare three different methods to estimate the eye-pose: IK, our Hybrid approach with a per-frame estimated size of the iris (HC) and a fixed iris size estimated for each session separately (HF).

We distinguish between HF and HC, because it may be necessary to re-estimate the size of the iris to account for illumination changes. Our method achieves an accuracy of  $3.63^\circ$  with a standard deviation (stddev) of  $1.37^\circ$  for HC and  $3.44^\circ$  (stddev  $1.23^\circ$ ) for HF. IK performs worse with an accuracy of  $9.57^\circ$  (stddev  $6.16^\circ$ ).

For each session, we perform a calibration of  $(\alpha, \beta)$  to determine the accuracy after alignment with the visual axis. We perform outlier removal for each session. Given the gaze errors  $e_i$ ,  $i = 1 \dots n$  for  $n$  frames, we determine the first quartile  $Q_1$  and the third quartile  $Q_3$ . The eye-pose estimated for frame  $i$  is an outlier, if  $e_i < Q_1 - 1.5(Q_3 - Q_1)$  or  $e_i > Q_3 + 1.5(Q_3 - Q_1)$ . Out of 160 evaluated frames, three were removed as outliers for HF and six for IK. We estimate  $(\alpha, \beta)$  for each session and user separately for HF and IK, and apply the values computed for HF to HC as well.

For the estimated visual axis, the eye-pose error is reduced for IK to  $6.73^\circ$  (stddev  $8.15^\circ$ ), HA to  $2.09^\circ$  (stddev  $1.49^\circ$ ), and HF to  $1.74^\circ$  (stddev  $1.35^\circ$ ). We show the results for each session after outlier removal in Table 5.1 and display some of the estimation results in Figure 5.6. Overall, HF performs the best, followed by HA. KI falls short for all, but one sequence.

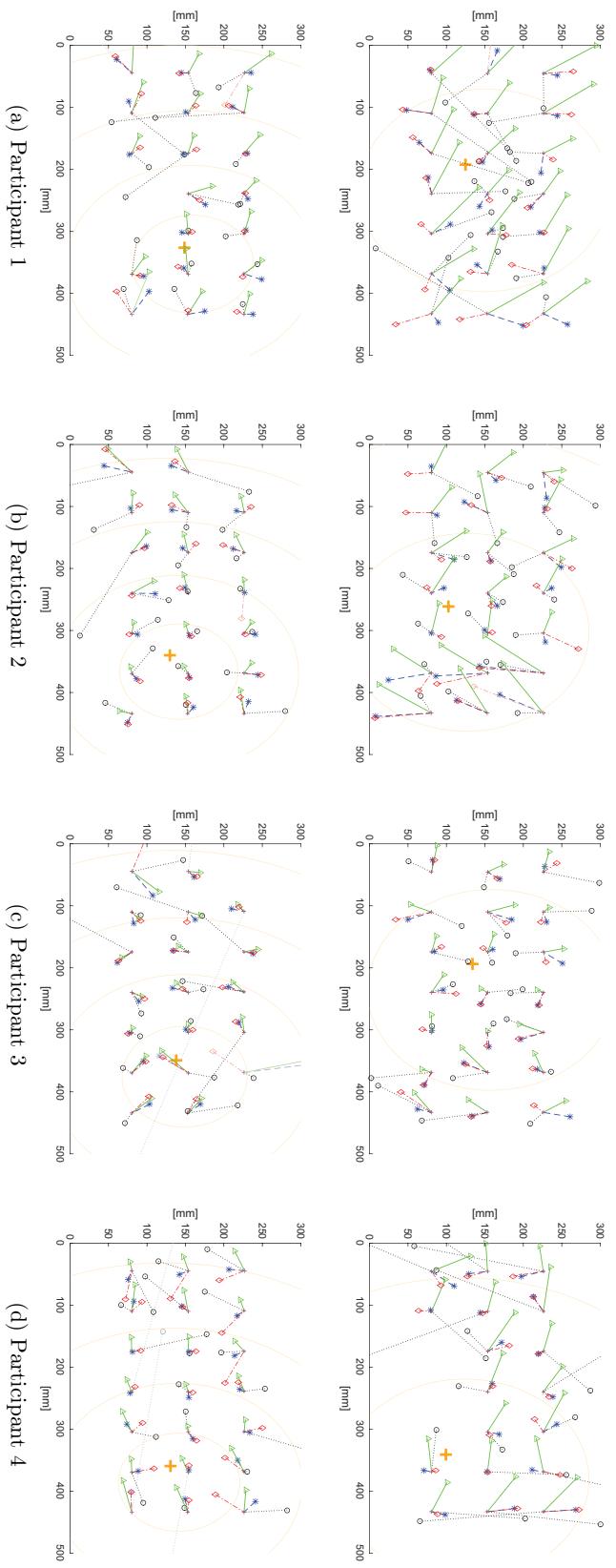
We have estimated a different offset of the visual and optical axes for the two session for each user. According to the two-sphere model, this value should be similar or identical. We suspect that the difference is caused by our eye model, which does not perfectly represent the human eye. Another explanation could be that the camera had to be positioned at a much steeper angle, when the display was at a 40cm distance to prevent it from occluding the screen. This is supported by the fact that for participants 2 and 4, the difference of the estimated angles is primarily along the vertical axis.

### 5.3.2.3 Hybrid eye-pose estimation in OST-HMD

Although the developed Hybrid eye-pose estimation method can be applied with arbitrary environments, e.g., a remote screen was used in the formal evaluation, it is also necessary to determine if images taken while wearing an OST-HMD are sufficient for the estimation of the eye-pose. We perform an informal evaluation of our method on the dataset taken during the CIC experiment.

We show the results of the estimated eye-pose (corneal sphere and iris contour) in Figure 5.7. We observe similar results to the evaluation of the Hybrid method. In some, few cases the estimation failed due to relatively few features extracted at the contour of the iris and a high number of features

Figure 5.6: The POR on the screen estimated by HF (green triangles), after applying the calibrated offset angles ( $\alpha, \beta$ ) to the estimation of HF (blue stars), and HA (red diamonds), as well as KI with the corresponding correction angles (black circle). Values assumed to be outliers are grayed out. The ground truth is shown as magenta crosses. Additionally, we show the POR when the user is looking straight forward as the orange cross and draw contours around it in  $10^\circ$  increments as grayed out orange lines. (top row) The monitor is positioned 90cm and (bottom row) 40cm away from the participant.



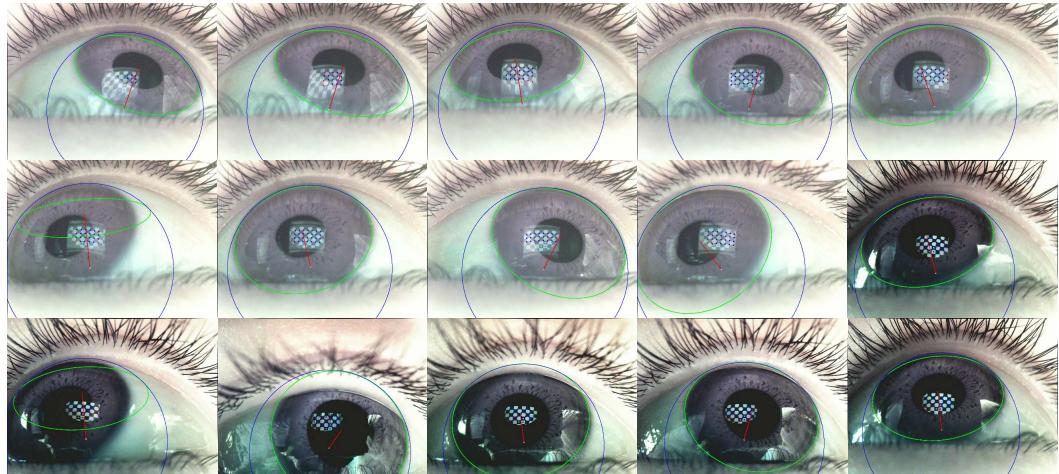


Figure 5.7: Eye-pose estimated from the reflection of the HMD-screen.

extracted on the corneal surface. Adjustments of the method that enable tracking in consecutive frames and further exploitation of the knowledge of the area the scene is reflecting at can help address this case in the future.

## 5.4 Conclusion

We have introduced the Hybrid eye-pose estimation approach. Our method requires a full calibration of the environment, this means camera intrinsic parameters and the geometric calibration in form of scene tracking. Furthermore, similar to other eye-gaze estimation methods, a one-time offline calibration of the user-specific offset between the optical and the visual axes is required. The developed method can adjust other parameters of the eye during runtime to account for illumination changes.

The eye is detected in the image taken by the camera through corneal imaging and the position of the cornea is estimated from matches of points in the scene and the corresponding pixels in the corneal reflection. The 3D point associated with the detected iris contour candidates is recovered through intersection of backprojected rays with the corneal sphere. Points whose back-projected ray does not intersect with the estimated sphere are removed as outliers and the remaining points are used to recover the circular 3D limbus.

We show that the estimated POR recovered through intersection of the estimated gaze with the scene model results in an accuracy of about  $1.7^\circ$ . Although our method is not directly compared against methods that use a complex and adaptable eye model to estimate the geometric eye-pose, the results of our experiments suggest that it is likely to outperform these methods as well, e.g., even in the uncalibrated scenario, we achieve results comparable to those of [Wu et al. \(2007\)](#).

## Limitations and future work.

A major limitation of the proposed method is the required scene model. Although this model can be recovered by existing reconstruction methods during runtime (Newcombe et al. (2011b,a)), it is desirable to remove the requirement of the accurate scene model to improve the general applicability of the proposed method.

The current system, does not exploit the known reflection of the scene model to eliminate edge candidates and uses only the iris contour candidates detected by the edge detection step. Therefore, in the failure cases in shown in Figures 5.5 and 5.7 a small number of edge candidates detected along the actual iris contour and a large number of erroneous candidates in the corneal reflection and the eye lid led to a false estimate. Future improvements of the proposed method should include further exploration of the projection of the iris contour into the image. One possible solution could involve a remapping of the image-based on the estimated cornea position, similar to the approach of Pires et al. (2013a).

Our evaluation of the Hybrid eye-pose estimation was conducted in a very simple environment. Future evaluations should include more complex environments and verify the applicability of the Hybrid eye-pose estimation in environments reconstructed at runtime.

In the presented evaluation the inner corners of the checkerboard pattern were used to estimate the position of the corneal sphere. In practice, such a pattern is unlikely to be located in the observed scene and thus a more general solution is necessary. We present one possible solution in Chapter 6.

# CHAPTER 6

# Inverse-rendering-based Cornea Tracking

---

In this chapter we present our proposed tracking method of the corneal sphere. The proposed method does not require feature matching between the corneal reflection and the scene, and estimates the position from a known scene model. The chapter is structured as follows:

Section 6.1 introduces the problem of cornea tracking. Section 6.2 explains the proposed approach and Section 6.3 evaluates the accuracy of the method in various scenarios. Section 6.4 reviews the findings and discusses future directions.

## 6.1 Introduction

IR LEDs reflect as distinctive glints on the corneal surface and can therefore be easily detected and matched to their origin. Active eye-pose estimation methods, e.g., [Guestrin and Eizenman \(2006\)](#), exploit this to robustly estimate the position of the corneal sphere as described in Section 2.3.2. In Chapters 4 and 5 the correspondences were recovered from images taken under natural illumination through corneal imaging. Hereby, the reflection of a distinctive pattern on the cornea was detected by a dedicated algorithm. Such a pattern cannot be assumed to be present in the observed scene and therefore the features through corneal imaging. Existing corneal imaging applications assume that the eye-pose has been already recovered, e.g., through iris contour detection ([Nishino and Nayar \(2004b\)](#); [Nitschke and Nakazawa \(2012\)](#); [Nitschke et al. \(2013b\)](#); [Takemura et al. \(2014b\)](#)) or a combination with an IR-light based tracker ([Nakazawa and Nitschke \(2012\)](#)). [Takemura et al. \(2014b\)](#) report success using Scale Invariant Feature Transform (SIFT) ([Lowe \(2004\)](#)) to match detected keypoints with a stored database after the corneal reflection has been rectified ([Nitschke and Nakazawa \(2012\)](#)). However, SIFT feature matching cannot be used reliably in the scenario where the eye-pose is still unknown. This is due to low contrast in corneal images, as only a fraction of the incoming light is reflected on the corneal surface ([Nishino and Nayar \(2006\)](#)), a high amount of noise from the eye features, and the distortion of the scene on the corneal surface. An example of feature matching results is shown in Figure 6.1.

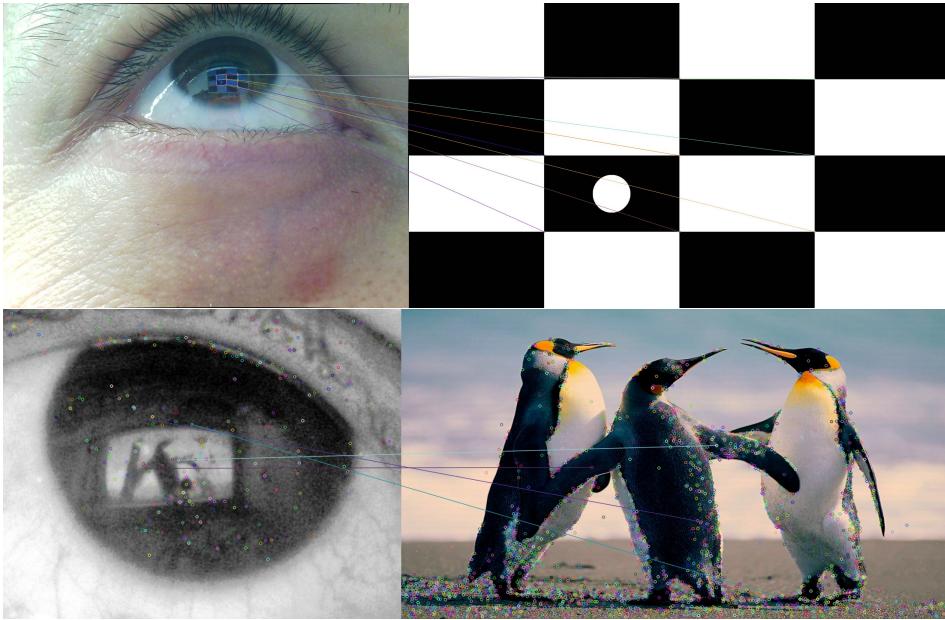


Figure 6.1: Detection and matching of features (top) by a dedicated algorithm in a controlled environment and (bottom) SIFT matches from a natural scene. Despite a selected ROI, SIFT matching detects only few correct matches.

Instead of using sparse matches, we propose to use inverse rendering to generate a dense representation of the scene’s reflection to track the position of the cornea. Inverse rendering has been used successfully in a variety of AR applications, such as scene lighting estimation (Patow and Pueyo (2003)) and color correction Tsukamoto et al. (2015). Zheng et al. (2014a) have applied this concept to camera pose estimation in a process they refer to as *closed-loop tracking*. Our solution is based on the same concept. If the observed scene is known, then it’s reflection on the surface of a known cornea can be reliably predicted. As the shape of the cornea creates a distinctive reflection, the correct position can be found by comparing the captured image with the prediction. In the following we explain how the predicted scene reflection can be generated and the cornea be tracked from an initial guess.

## 6.2 Inverse Rendering Tracking

The image captured by  $T$  is a reflection of the scene  $S$  on the cornea  $C$ . For a given position  $\mathbf{C}$  a predicted reflection of the known scene can be generated according to the used eye model (Figure 6.3). In the following explanations, we assume that the scene is completely reflected on the cornea. The approach can easily be extended to account for the partial reflection of the scene due to the eye pose.

For a pixel  $\mathbf{p}$  on the camera image plane let  $\mathbf{b}$  be the ray from  $\mathbf{T}$  through

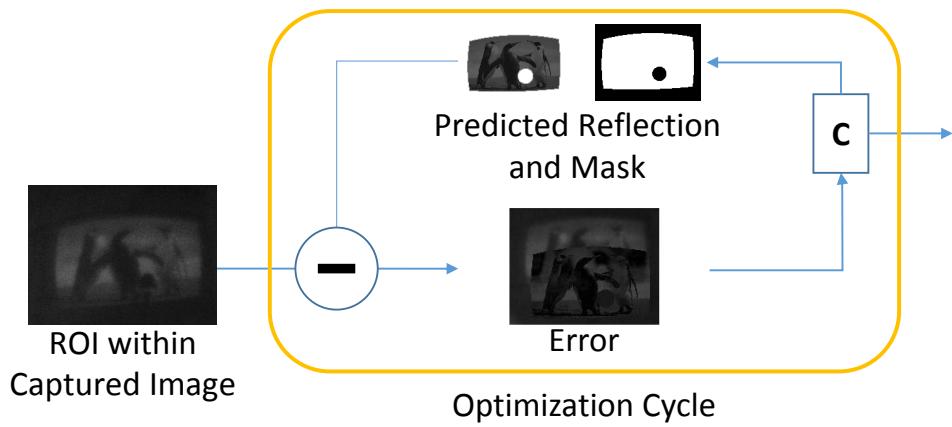


Figure 6.2: The position of the corneal sphere **C** is optimized by minimizing the error between the predicted and captured images through inverse rendering.

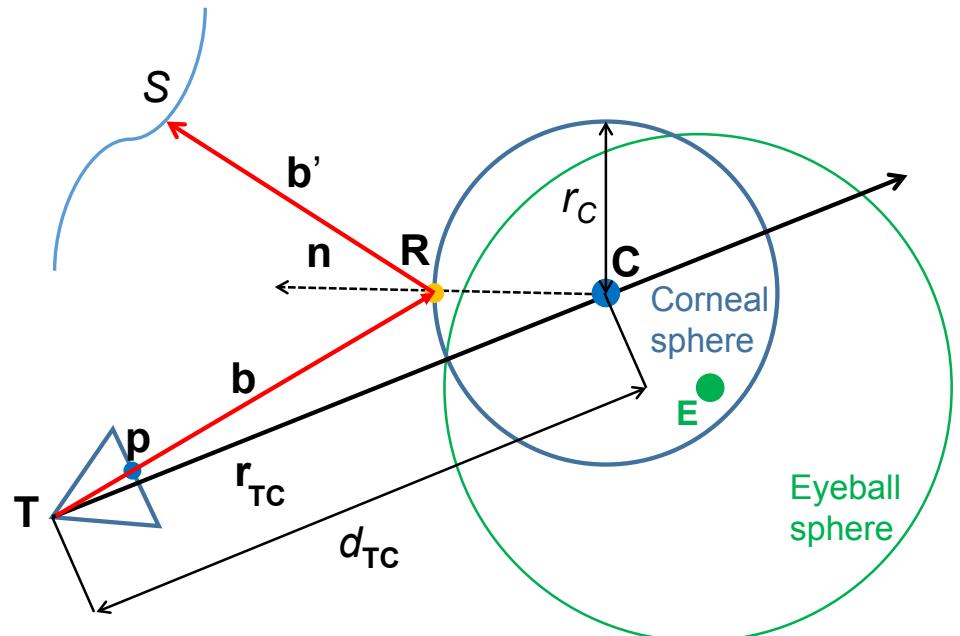


Figure 6.3: The pixel observed by the camera is the intersection of the ray reflected on the corneal surface with the scene.

**p.**  $\mathbf{b}$  intersects  $C$ , if  $\mathbf{b}^T \mathbf{C} < r_C$ .  $\mathbf{b}$  reflects on  $C$  in  $\mathbf{R}$ .  $\mathbf{R}$  is given as

$$\mathbf{R} = k\mathbf{b}, \quad (6.1)$$

$$k = d - \sqrt{d^2 - \mathbf{C}^T \mathbf{C} + r_C^2}, \quad (6.2)$$

$$d = \mathbf{b}^T \mathbf{C}. \quad (6.3)$$

Given the normal  $\mathbf{n}$  of the corneal sphere at  $\mathbf{R}$ ,  $\mathbf{b}$  is reflected on the cornea as  $\mathbf{b}'$ , with  $\mathbf{b}' = \hat{\mathbf{b}} - 2(\hat{\mathbf{b}}^T \hat{\mathbf{n}})\hat{\mathbf{n}}$ . The value observed at  $\mathbf{p}$  corresponds to the intersection of  $\mathbf{b}'$  with  $S$ .

The spherical shape of the cornea generates a distinctive reflection of the environment, thus even a slight shift in its position results in an image which shows a clear distinction. An initial guess  $\mathbf{C}_0$  can be acquired from a sparse distribution of likely cornea positions  $\mathbf{C}_{1\dots n}$  for which the predicted reflection image  $P(\mathbf{C}_i)$  and the corresponding reflections mask  $M(\mathbf{C}_i)$  is generated. The corneal sphere is not a perfect mirror, thus the captured image will have lower contrast than the predicted images. For each image  $P(\mathbf{C}_i)$  we determine  $s_{min}$  and  $s_{max}$ , the minimal and maximal values of all pixels of  $I$ , given  $M(\mathbf{C}_i)$  and rescale  $P(\mathbf{C}_i)$  to correspond to the range  $\{s_{min}, s_{max}\}$ . We determine  $\mathbf{C}_0$  as

$$\mathbf{C}_0 = \arg \min_{\mathbf{C}_{i=1\dots n}} \sum_{p \in M(\mathbf{C}_i)} |\mathbf{p}_I - \mathbf{p}_{P(\mathbf{C}_i)}|, \quad (6.4)$$

and refine it through inverse rendering by minimizing the error function

$$e = \arg \min_C \sum_{p \in Q} |\mathbf{p}_P - \mathbf{p}_I|, \quad (6.5)$$

where  $Q$  is the delated mask  $M$ . Figure 6.4 shows the results of the localization from two different patterns displayed on the wall. Alternatively, the initial guess can be obtained by detecting a predefined pattern, as was done in previous solutions. In a tracking environment, the previously estimated location can be used as an initial guess for the concurrent frame.

## 6.3 Experiment

We present an evaluation of the proposed tracking method in an environment that resembles the view through an HMD and examine its stability depending on the displayed content. We present a comparison of the proposed method and an estimation from stable 2D–3D correspondences.

### 6.3.1 Experiment Environment

To evaluate our method we use a planar surface in front of the user (Figure 6.5). This setup resembles the screen of an OST-HMD or an augmented

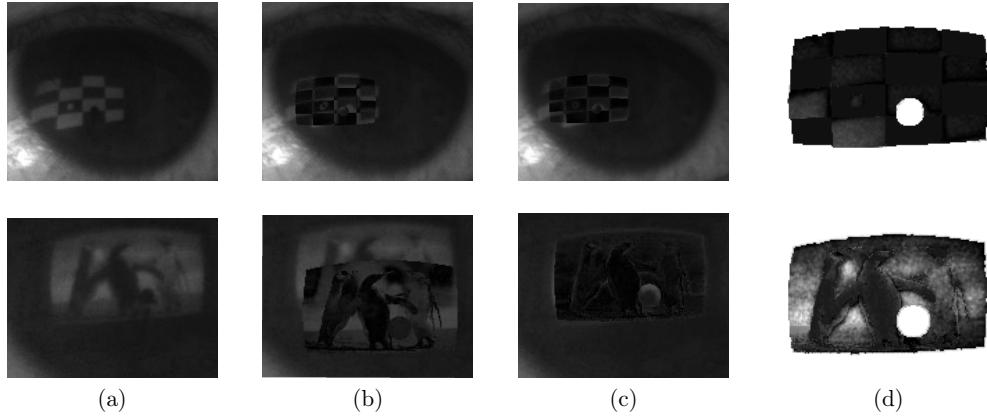


Figure 6.4: The position of the corneal sphere is updated for successive frames through inverse rendering. (a) Image captured by the eye-tracking camera. (b) Reflection of the screen on the corneal sphere located at the position estimated in the previous frame and (c) after re-estimating the position of the corneal sphere (c) overlaid onto the original image. (d) Rescaled residual

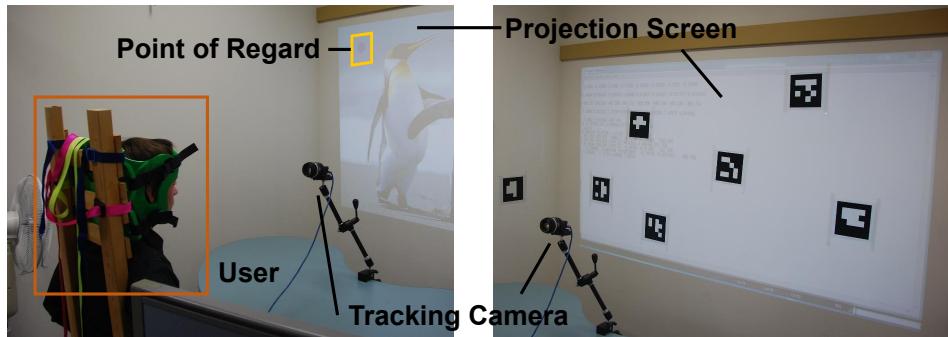


Figure 6.5: A user taking part in the experiment (left) is asked to look at a moving augmented point on the screen. The environment was then calibrated by placing multiple markers onto the augmented surface and the surroundings (right). For visualization purposes the markers are shown together with the projector illumination.

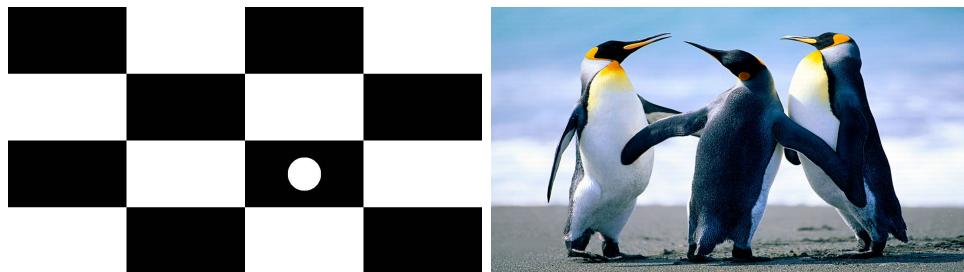


Figure 6.6: The two patterns used in the evaluation: (left) a checkerboard pattern and (right) a natural scene.

planar surface. The scene  $S$  was illuminated by a SANYO close-range projector with a resolution of  $1280 \times 720$ . The remainder of the surroundings remained unknown. The user was seated about 2 m away from the wall with the head fixated to prevent out-of-focus blur or unintentional head movement. A PointGrey FL3-U3-13S2C-CS camera with a 50 mm Nikon lens mounted on it was used as the eye-tracking camera  $T$ .  $T$  captured  $1324 \times 1048$  grayscale images at 45 fps and was placed about 50-60 cm away from the user.

Our method requires a known relation of the observed scene and the eye-tracking camera. In the following we describe how we calibrated the position of the illuminated area relative to  $T$ .

We calibrated the surroundings of  $T$  with a Nikon D60 camera  $C$  with an 18 mm lens attached to it, which captured images of  $3872 \times 2592$  pixels. We placed multiple markers onto surfaces surrounding  $T$ , so that one marker  $M_1$  could be detected in the image captured by  $T$ . Additionally multiple markers  $M_2$  were placed onto the illuminated surface (Figure 6.5). The spatial relation of all markers to an origin  $W_0$  was calibrated by Ubitrack (Huber et al. (2007)). We modeled  $S$  by fitting a plane to the markers  $M_2$ .

$C$  was placed onto a tripod, so that it could capture the illuminated surface and the markers  $M_2$ .

Given the observed transformations, the transformation  ${}^T_C T$  was recovered as  ${}^T_C T = {}^T_M T {}^{M_1}_W T {}^W_{M_2} T {}^{M_2}_C T$ .

After removing all markers the wall was illuminated with coded patterns, which were captured by  $C$ . We used the method of Yamazaki et al. (2011) to acquire pixel-wise correspondences  $\{\mathbf{p}_P, \mathbf{p}_C\}$  for the projector and  $C$ . The 3D position  ${}^C \mathbf{P}$  of every pixel  $\mathbf{p}$  displayed by the projector was computed as the intersection of the backprojected ray from  $C$  through  $\mathbf{p}_C$  and  ${}^C S$ .

Finally,  ${}^T S = {}^T_C T {}^C S$  is the model of the environment aligned with  $T$ .

### 6.3.2 Stability Comparison

In this section we discuss the stability of the closed-loop tracking approach compared to a point-correspondence based re-estimation of the corneal position for each frame. We evaluate the tracking results by showing the users a  $4 \times 4$  checkerboard pattern, and a natural image, shown in Figure 6.6, and ask them to follow a dot shown on the screen. For both scenarios we expect the estimated positions to be closely clustered, as the user's head remained stable for the duration of the experiment. For our tracking solution we manually set the position of the cornea in the first frame. Manual matching of features was conducted for the checkerboard image.

In Figure 6.7 we show the estimated position of the corneal sphere relative to the tracking camera. The estimation from point correspondences fails to estimate the position of the corneal sphere whenever the 2D–3D matching step fails due to the orientation of the eye. On the other hand, our proposed solution computes stable results throughout the session and remains stable even

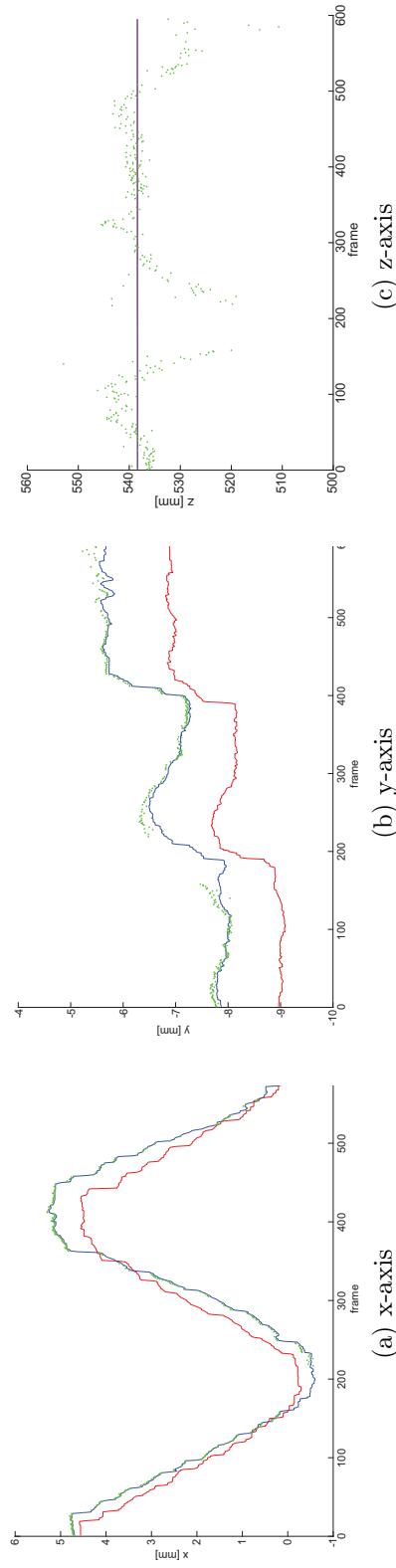


Figure 6.7: An example of the estimated position of the corneal sphere for each frame of the recorded sequence. Estimation from accurately matched 2D-3D correspondences from a checkerboard pattern is shown in green. A number of frames could not be estimated due to too few features detected in the respective frame. This was mostly the case when the cornea moved out of focus or under large orientation from the camera. The tracking results by our method for the same sequence are shown in blue. The method did not fail even under extreme orientations and displays similar behavior to the estimation from correspondences. In red we show the results of the tracking when a natural scene was displayed (penguins). The tracking shows similar behavior as for the checkerboard pattern. The displacement between the two patterns is likely due to small shift in the user's position.

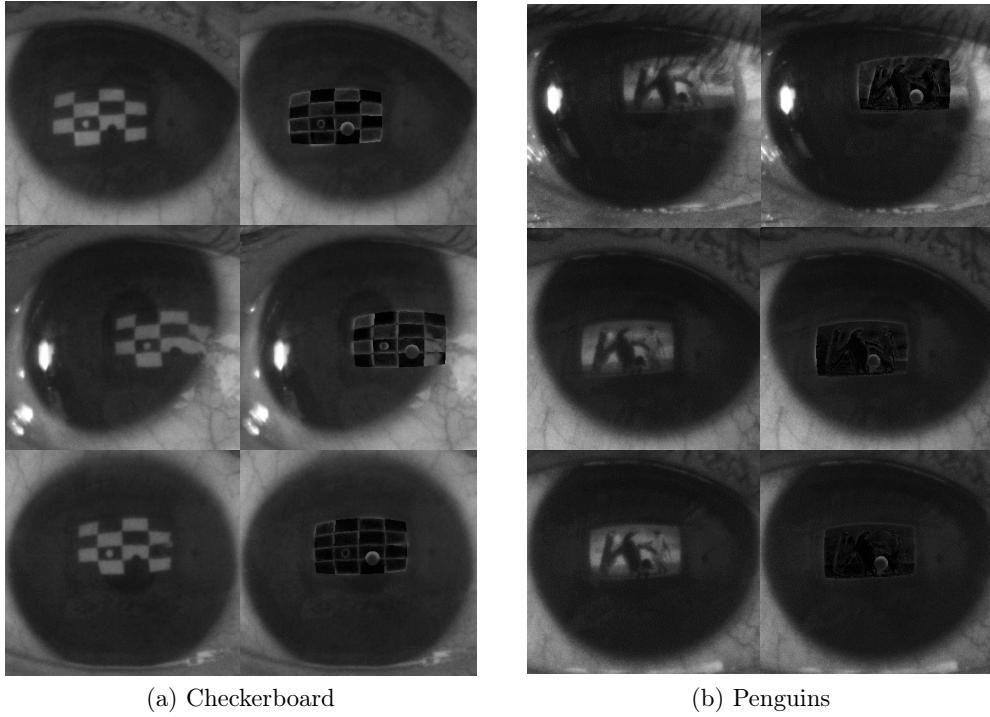


Figure 6.8: Images captured by the camera (left) and the overlay of the reflection of the scene on the estimated cornea position (right) for (a) the checkerboard pattern and (b) the penguins image.

when the reflection is strongly distorted. Naïve SIFT feature matching has failed to provide stable matches between the natural scene and the reflection in the user’s cornea, thus we omit the visualization of these results. On the other hand, our solution successfully determines the position of the corneal sphere throughout the sequence and displays a movement pattern similar to that of the checkerboard setup.

As can be seen in Figure 6.7c position estimation from correspondences computes a stable position, with a stddev of 3.3 mm along the z-axis. However, the results show a tendency for a closer than actual estimation. The inaccurate measurements are primarily located in areas where the corneal surface is seen under a steep angle. At such angles the model errors are most vivid. On the other hand the estimated depth value remains stable for all frames estimated by our method. The lack of minor diversity is likely due to the minimal impact the displacement along the z-axis had on the projection of the content into the camera. A detailed explanation of this observation is provided in Section 4.4.6.

Besides the stability issue, it is important to determine whether the estimated position is correct. We verify this observation by reflecting the screen at the position estimated by our method. Some of the results are shown

in Figure 6.8. As can be seen the reflections on the estimated corneal sphere correctly overlays the reflection of the screen captured by the tracking camera.

## 6.4 Conclusion

We have presented an approach to track the position of the cornea in images captured by an eye-tracking camera, without the use of IR illumination. The method uses the known scene model to generate a prediction of its reflection on the corneal sphere. The correct position is found through minimization of the error between the captured camera image and the predicted reflection. We show that the method can deal with different environments and accurately track the cornea under strong deformations due to the cornea shape.

### Limitations and future work.

The proposed method utilizes the available 3D and color information to generate predictions of the scene reflection on the corneal sphere. As such, the method assumes that the observed scene occupies a sufficiently large FOV. If only a very small surface of the cornea is occupied by the reflection the optimization algorithm could drift towards areas that show similar intensity distribution. Additionally, a small reflection area limits the benefit of the dense tracking solution over a sparse point cloud.

Another difficult scenario is similar intensity of the scene and the iris pattern, as the error function is more likely to find an incorrect minimum. A potential solution could be a stabilization term based on the distance of the projection of the estimated cornea center into the cameras or features tracked in consecutive frame.

The proposed solution requires an initial position that is close to the ground truth, e.g., obtained from feature matches or a predetermined tracking pattern. As the eye's movement varies between periods of fixation and saccades (Purves et al. (2001)) a high-speed camera is required to provide sufficient tracking results.

The proposed method was evaluated in an environment with a simple scene model that resembles the modelled screen of an OST display. However, the evaluated setup did not include noise from the unknown background. The impact of the background noise and separation of the different reflections has to be studied in future applications. Furthermore, the method has to be tested in setups with complex, non-static scenes.

In the discussed evaluation the cornea was modelled as a spherical surface without any aberrations. The asphericity of the corneal surfaces causes distortion of the reflected scene (Figure 6.8) that are not modelled by the current method. Future applications of inverse rendering combined with CI could include the reconstruction of the corneal surface without complicated hardware

setups used in existing methods (Halstead et al. (1996); Wood et al. (2015)). Additionally, the detected shape depends on the gaze directions, thus if the shape of the eye has been acquired beforehand, this could be used to estimate the user's gaze direction without the need of iris or pupil contour detection.

# CHAPTER 7

# User Spatial Consistency Perception in HMD-based AR

---

As tracking and calibration methods improve it is necessary to determine a threshold, at which the virtual overlay appears spatially consistent with the reference real object. This chapter is structured as follows:

Section 7.1 introduces the research objective and describes the approach. Section 7.2 explains the experiment design, followed by the implementation in Section 7.3. Section 7.4 presents the results of the experiment and discusses their implications. The chapter is concluded with a summary and outlook in Section 7.5.

## 7.1 Introduction

AR found application in a large variety of fields, such as medicine, entertainment, training, and guidance to name a few. The visualization of the virtual content can have a big impact on the user's acceptance of the content (Kishishita et al. (2014); Rolland and Fuchs (2000)). Depending on the target application this may require consistent relighting of the scene (Gruber et al. (2012); Jachnik et al. (2012)), consistent spatial overlay (Kurz et al. (2014)), or variable coloring (Kishishita et al. (2014)). Current AR applications use handheld devices, such as smart phones and tablets, equipped with an outwards facing camera. The outwards facing camera is used to analyze the scene to correctly render the virtual content. By overlaying the virtual content over the image captured by the camera the user can be presented with an augmented experience.

**Head-mounted displays.** In recent years HMDs have gained a lot of attention. VST-HMDs function similar to handheld devices, however the user's view is of a much wider angle, e.g., the Oculus Rift has a viewing angle of more than 110°. OST-HMDs on the other hand show only the virtual content on the HMD-screen, commonly a half-mirror. Due to technical limitations these devices can augment only a very limited FOV, e.g., 20° on a Moverio BT200, although various concepts of large FOV HMDs exist, e.g., Kiyokawa (2007)'s HHMPD display, Orlosky et al. (2014)'s Fisheye vision, or Ardouin et al. (2012)'s FlyVIZ. Although various aspects of the visualization on an

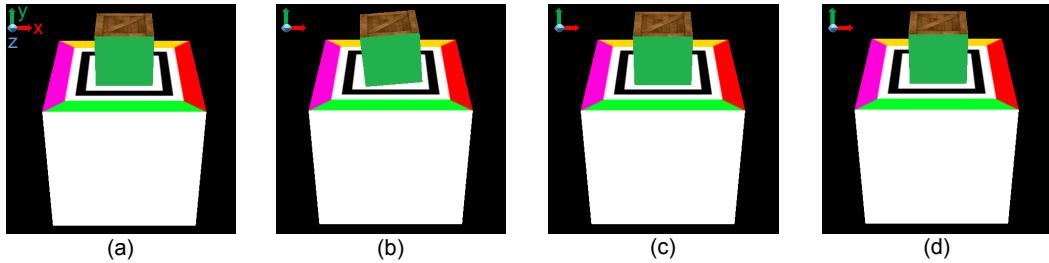


Figure 7.1: Various types of spatial registration error between the user alignment cube and the target marker. (a) Exact alignment in both position and orientation. (b) Clearly visible error due to a large rotational misalignment around the z-axis. (c) A smaller, far less visible, rotational error and (d) Subtle translational error along the x-axis result in misalignments which cannot be distinguished from the ground truth by most users.

OST-HMD coincide with that of a VST-HMD and handheld devices, it also incorporates an additional calibration step, as explained in Section 3.1.

Entirely eliminating the projection error is difficult, if not impossible, as even a perfect calibration will degrade over time. Therefore, it is necessary to determine when the calibration results are satisfactory and users no longer perceive a misalignment between the augmentation and the reference object. An example of alignment errors above and below the JND is shown in Figure 7.1.

**Spatial error noticeability.** Various studies have investigated how AR impacts the user’s depth perception (Altenhoff et al. (2012); Swan II et al. (2015)). Others focused on the acceptance of AR (McDonnell et al. (2012)) and user error noticeability in handheld scenarios (Madsen and Stenholt (2014); Tokunaga et al. (2015)). For head-mounted displays this question remains unanswered. Hereby, the results from handheld studies cannot be transferred without further evaluation, as the perceived FOV in HMDs is larger than in handheld devices, they are more immersive and in case of OST-HMDs, transparency and color inconsistencies also impact the user’s perception. Furthermore, empiric results may not reflect the user’s impressions. Moser et al. (2015) found that although the INDICA OST-HMD calibration (Itoh and Klinker (2014a)) approach empirically performed worse than the manual SPAAM calibration (Tuceryan and Navab (2000)), user’s performed better in setups calibrated by INDICA. Users also preferred the INDICA method over SPAAM.

**Contribution.** The CIC approach (Chapter 4) outperforms INDICA in terms of calibration results and is therefore likely to also be preferred over SPAAM, although the empirical data suggests that users are more likely to notice misalignments in setups calibrated by CIC than those calibrated by SPAAM. It is necessary to determine the user’s JND to understand the accept-

able projection error of OST-HMD calibration methods for a better empirical comparison of different solutions.

Although the evaluation targets headworn devices, the use of OST-HMDs would lead to a number of problems

- Incorrect alignment of content on the OST-HMD due to an incorrect calibration that varies for every user (Chapter 4).
- Refraction of the incoming light rays on the OST-HMD screen (Itoh and Klinker (2015a))
- Visualization has to be adjusted as the user's pose changes relative to the target.

We therefore choose a controlled environment, where the visualized content is displayed in front of the user through a two-projector system that models the dual-optics system of an OST-HMD. The use of the two-projector system offers the following benefits:

- It can present a correctly aligned view of virtual and the target object.
- It can be used to present augmentations at different distances without physical modification of the environment.
- Even though the system is spatially calibrated, the visualization still contains artifacts as the centers of the pixels do not perfectly align.
- By using only one projector the view through a VST-HMD can be simulated. Hereby, the target object can be displayed at the same location for both modes to ensure that the participants' experience is consistent.

Through understanding of the JND through this experiment we can better understand how experiences presented on the various devices differ from each other and what technical requirements have to be fulfilled to present a satisfactory experience. These results benefit not only future researchers but also system designers and manufacturers.

## 7.2 Experiment Design

### 7.2.1 Setup

Current OST-HMD calibration algorithms fail to perfectly align the virtual and real content. Consecutively, a study performed on an OST-HMD will inevitably include unmodeled errors. The objective of our research was to determine the noticeability thresholds for content misalignment in an HMD-worn scenario. Therefore, we require a suitable representation of the view

observed through a VST- and an OST-HMD. The view can be modeled by an HUD, a half-mirror that visualizes graphics projected onto it, without occluding the background. As the view through the HUD depends on the user's position it becomes also necessary to model and track the user's view. This makes it more difficult to ensure that the view through the HUD coincides with the intended presentation. Displaying the content by a single projector, with a predefined transparency, can ensure that the view corresponds to the intended representation. However it does not allow to modify the representation of the virtual content, e.g. through antialiasing effects, varying resolution of the HMD-screen, as well as how different visualization systems impact the perception. We propose to use a two-projector system to create the impression of looking through an HMD. The benefit of the two-projector system, is that, even in a calibrated setup, the projectors generate slightly displaced images, as the pixels do not perfectly overlap, and create varying transparency effects, similar to content seen through an OST-HMD. By modifying the resolution of each projector we can generate different experiences, e.g., to determine if pixelation has any impact on the user's perception.

### 7.2.2 Task

We use a simple alignment task, as it allows us to easily determine the accuracy thresholds. AR used in demos and other simple applications requires markers that the content is displayed upon. The most common is a black-and-white color-coded marker. This kind of marker is used in a variety of applications and is the basis of ARToolkit ([Kato and Billinghurst \(1999\)](#)). Additionally, it is more difficult for users to find the center of a pattern that is used as a marker than a black-white marker. Therefore, a pattern would naturally allow for higher displacement.

Our target object is a white cube shown in Figure 7.4 and resembles a fiducial marker. The top side of a cube, with a size of 16 cm, facing the user contains a black-and white pattern. The size of this marker is 10 cm, the size of a marker in ARToolkit. Additionally the border is colored in a unique color. We choose a simple object, a cube, as the virtual object. Each side of the virtual cube is 6 cm long, and if placed correctly the center of the bottom of this cube will coincide with the center of the target object's top side. Additionally, each side of the virtual cube is colored in the same manner as the target object. Correctly aligned, the colored sides will face in the same direction. The representation as two cubes was chosen, due to participants in the trial study complaining about the ambiguous representation of the rotation of the target marker. We have decided to use a cube as the virtual object, as its rotation is not ambiguous, its simple shape does not distract the user and allows them to focus on the task, and is thus suitable to determine the lower threshold of the visualization.

As our projector system is displaying monocular vision, users cannot adjust

the depth, however they can modify the position and rotation of the virtual cube until they are satisfied with the alignment. We display the marker in the center of the user’s vision, as users experiencing AR commonly focus their attention and vision onto it. We use three different depth values,  $d = \{0.6 \text{ m}, 1 \text{ m}, 2 \text{ m}\}$ . If the marker is placed onto the top of a table, user’s will likely not come closer than 0.6 m to it, as this causes tracking issues and already covers a 10° FOV. Existing OST-HMD devices, cover a similar FOV. For example, Moverio BT200 covers a 20° horizontal view. Additionally, if the user is standing in front of the table, the distance to the marker will be around 1 m, and as the users move away, it is unlikely that they will move more than 2 m from the marker, due to tracking issues and a decreasing size of the virtual object. When looking at an augmented marker, users commonly move around it or rotate it in their hand to look at the varying presentation. We represent this by a subset of marker orientations  $\{\theta_x, \theta_y\}$ , where  $\theta_x = \{0^\circ, 30^\circ, 60^\circ\}$  and  $\theta_y = \{-60^\circ, -30^\circ, 0^\circ, 30^\circ, 60^\circ\}$  are the rotation angles around the x and y axes, respectively.

## 7.3 Implementation

### 7.3.1 Environment Calibration

As previously explained, we use a 2-projector system to simulate the view seen through a VST- and an OST-HMD. In our setup, shown in Figure 7.2, we assume that the content is shown on a quasi-planar wall W and the user U is sitting approximately 2 m away from the wall. We chose this distance, as it allows us to present the augmentations at the desired distances and is within the distance to the virtual screen plane of various HMDs, e.g., the virtual screen of an Nvisor ST60 is at 0.7 m and that of a BT200 at 2.5 m. Furthermore, we assume that the user is looking at the center of the projection and the eyes are at a height of 1.2 m above the floor. This height was measured after seating one of the participants on a chair. The objective of the calibration is to determine, the transformation of a pixel  ${}^i\mathbf{p}$  shown by a projector, where  $i \in \{A, B\}$  and A, B are the used projectors, to a pixel  ${}^U\mathbf{p}$  and  $\mathbf{p}_U$  in the users viewing frustum.

The projectors used in our setup were two SANYO PDG-DWL2500J close-range projectors, with a maximum contrast ratio of 2000:1 and a brightness of 2500 lumens. The projector’s native resolution is 1280×800 pixels and they were set to display images at a resolution of 1920×1080 pixels. As shown in Figure 7.2 the projectors were placed side-by-side. Each projector illuminated an area of approximately 3×2 m and the majority of the illuminated surface of both projectors overlapped (Figure 7.3).

A Nikon 60D that captured images of 3200×2500 pixels was placed onto a tripod so that it did not contain any in-plane rotation and could capture the

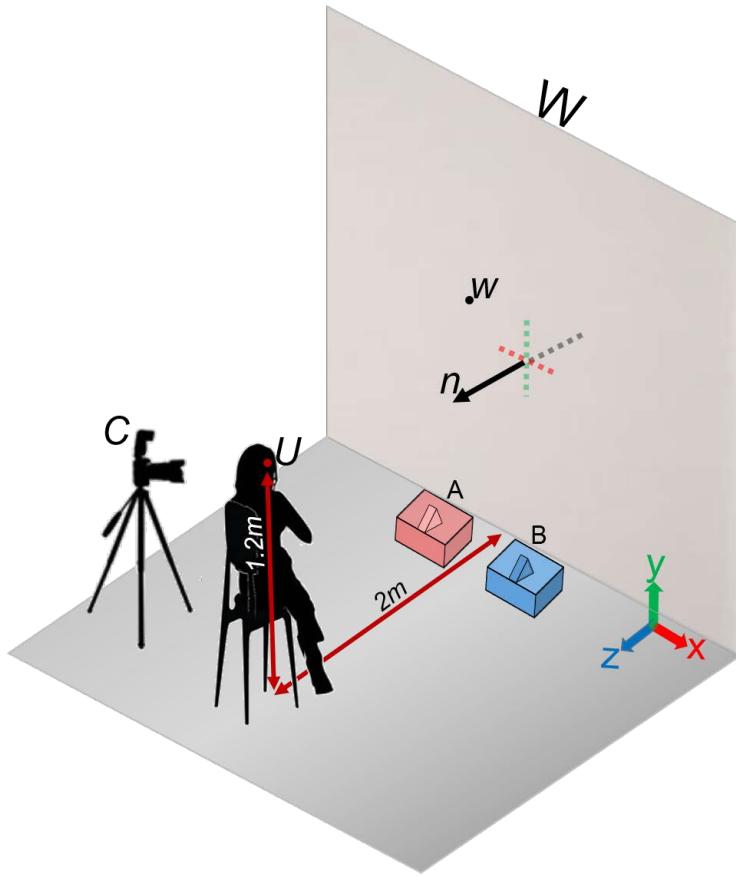


Figure 7.2: Side-view of the experiment. The wall  $W$  is illuminated by two projectors  $A$  and  $B$ . Camera  $C$  is positioned so that it can capture the illuminated area. Its position differs from the user's viewpoint  $U$ .  $W$  is modeled as a plane by  $\mathbf{W}$ , a 3D point on  $W$ , and  $\mathbf{n}$ , the normal of  $W$ .  $W$  is reconstructed relative to  $U$  from images captured by  $C$ .

entire illuminated area. We denote this camera as  $C$ .

To reconstruct the illuminated surface  $W$ , we have attached multiple markers to the wall and took multiple images with the camera  $C$ . By fitting a plane to the markers, we recover  ${}^C\mathbf{W}$ , a point on the illuminated area, and  ${}^C\mathbf{n}$ , the normal of the plane relative to  $C$ . After removing the markers, we determine matches  $\{{}^i\mathbf{p}, {}^C\mathbf{p}\}$  between pixels illuminated by each projector and the images taken by  $C$  through the camera-projector calibration toolbox by Yamazaki et al. (2011). Although this method provides subpixel accurate matches for the majority of the projected pixels, for some pixels a match is not found. For these points we determine the match by computing a homography in their region. Hereby, for a pixel  ${}^i\mathbf{p}^0$ , we determine the 10 closest pixels  ${}^i\mathbf{p}^j$ , that have been matched successfully, and do not lie in a single line. We compute the homography  $\mathbf{H}$  that maps the pixels from the projector to the camera,  ${}^C\mathbf{p}^j = \mathbf{H}{}^i\mathbf{p}^j$ , and determine the match by applying it to  ${}^i\mathbf{p}^0$ , thus  ${}^C\mathbf{p}^0 = \mathbf{H}{}^i\mathbf{p}^0$ .

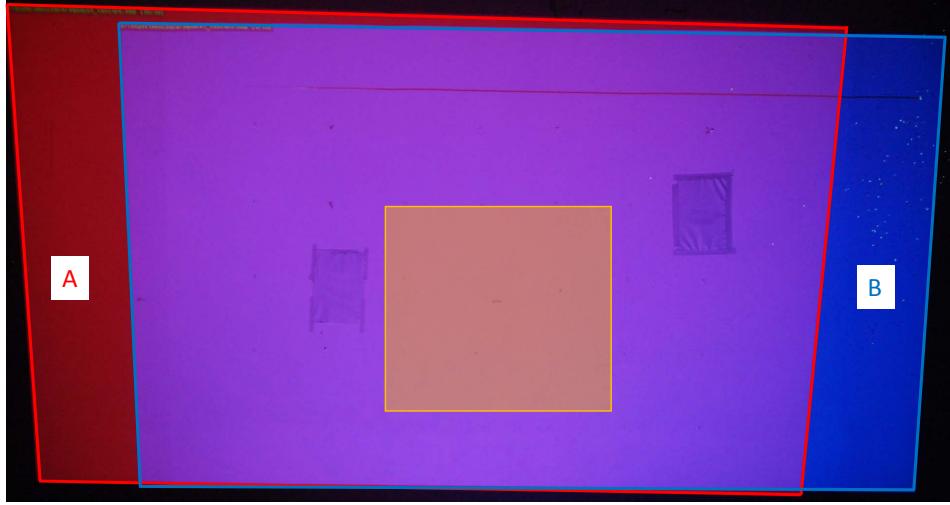


Figure 7.3: Projectors  $A$  and  $B$  are illuminating slightly displayed surfaces that largely overlap. Consistent rendering has to account for the distortion of the illuminated surface. The area within the orange square is augmented during the experiment. The visible defects in the wall were outside the augmentation region and thus did not impact the experiment results.

The 3D location  ${}^C\mathbf{P}$  of each pixel  ${}^i\mathbf{p}$  is determined by intersecting the back-projected ray through  ${}^C\mathbf{p}$  with  $W$ .

However, the content has to be visualized not from the perspective of  $C$ , but the user's perspective. To determine  ${}^U\mathbf{P}$  we recover  ${}^U\mathbf{R}$  and  ${}^U\mathbf{t}$ , the rotation and translation from the camera coordinate system to the user's coordinate system. The rotation  ${}^U\mathbf{R}$  aligns  ${}^C\mathbf{n}$  with  ${}^U\mathbf{n}$ . As we assume that the user's gaze direction corresponds to the normal of the wall,  ${}^U\mathbf{n} = (0 \ 0 \ -1)^T$  and that there is no in-plane rotation of  $C$ , a simple angle-axis representation  $\langle \alpha, \hat{\mathbf{v}} \rangle$  of the transformation can be recovered, where  $\alpha = {}^C\mathbf{n}^T {}^U\mathbf{n}$  and  $\mathbf{v} = {}^C\mathbf{n} \times {}^U\mathbf{n}$ .

The translation is given as  ${}^U\mathbf{t} = -{}^U\mathbf{R} {}^C\bar{\mathbf{P}} + (0, -y, -2)^T$ . After aligning the center of the projection with  $(0 \ 0 \ 0)^T$  we translate all points along the y-axis, so that the offset of the bottom of the projection is  $y$  m above the floor, and translate them to a depth of 2 m.

Let the points that result from the area illuminated by projector  $i$  be denoted by  $\mathbf{P}^i = (\mathbf{P}_x^i \ \mathbf{P}_y^i \ \mathbf{P}_z^i)^T$ . All points  ${}^U\mathbf{P}$  are in a plane parallel to the xy-plane, thus the frustum border for each projector can be determined as  $left_i = \min({}^C\mathbf{P}_x^i)$ ,  $right_i = \max({}^C\mathbf{P}_x^i)$ ,  $bot_i = \min({}^C\mathbf{P}_y^i)$ , and  $top_i = \max({}^C\mathbf{P}_y^i)$ . The determined frustum contains areas that are not illuminated by the projectors, and does not reflect possible skewing due to the projectors' alignment with the wall. To determine the mapping between  ${}^i\mathbf{p}$ , and the intended image  $I_G$  we determine the size  $(w, h)$  of each pixel in  $I_G$  given the estimated frustum. For each pixel  ${}^i\mathbf{p}$  we can compute the corresponding pixel in the generated

$$\text{frustum view } x = \frac{^{\text{U}}\mathbf{P}_x^i - \text{left}_i}{w}, y = \frac{^{\text{U}}\mathbf{P}_y^i - \text{bot}_i}{h}.$$

The resulting mapping generates spatially consistent images from the user's perspective, for both projectors.

### 7.3.2 Visualization and Interaction

We create the user's view in OpenGL (OpenGL (2015)) given the computed frustums and use OpenCV 3.0 (Bradski (2000)) to map  $I_G$  for each projector. The users can control the virtual cube with a 3Dconnexion SpaceNavigator. Although this device allows users to control all six DOFs at the same time and the controls feel intuitive, we do not allow users to manipulate the translation and rotation at the same time to prevent unintended displacement, due to unfamiliarity with the input device. To switch between the different modes users were asked to press a key on the input device. Additionally, we displayed what mode the input device was in to the users. By pressing a second key, users could switch into an accuracy mode with reduced sensitivity for minute adjustments. Whenever users were satisfied with the alignment, they were asked to score their confidence that the alignment was correct on a scale from 1-10, where 1 represented no confidence and 10 absolute confidence.

## 7.4 Experiment

We conduct our evaluation on 16 participants (11 male, 5 female), between the ages of 21 and 33 (mean age of 24.8 years, stddev 3.6 years), with seven subjects claiming to have little to no prior experience with AR. Each subject was confirmed to have normal, or corrected to normal, vision and were monetarily compensated for their time.

The objective of the experiment was explained to each participant before the trial and users were asked to align the objects as accurately as possible. Each user participated in an unrecorded trial session that consisted of 10 random tasks each for the VST and OST modes. Users could use this time to get accustomed to the controls and the different visualization modes. After the trial session, users completed one session for the VST and OST-mode, where the order was selected randomly. Each session consisted of 45 unique combinations of  $\{\theta_x, \theta_y, d\}$  presented in random order. For each alignment we recorded the user's view as they performed the task, the alignment time, the final pose and the user's confidence in the alignment. After the experiment users were asked to fill out an anonymized online questionnaire. An example of a user taking the experiment is shown in Figure 7.4.

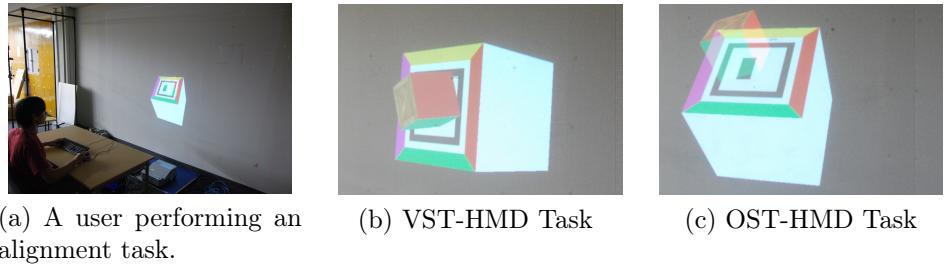


Figure 7.4: Users were seated approximately 2 m away from the wall while taking the experiment. They were asked to perform a docking task for two visualization modes VST- and OST-HMD for targets of varying orientations and distances to the user.

#### 7.4.1 User Response Time

A boxplot of total set time, completion of all 45 trials per display type, across all 16 subjects is provided in Figure 7.5. Mean completion time for VST sets is 29.17 min with standard deviation of 17.18 min. OST trial sets were completed with a mean of 31.17 min and standard deviation of 15.85 min. One-way analysis of variance (ANOVA) comparing the times between display modes reveals no statistical difference between the two groups ( $F < 1$ ). An examination of response times across marker pose groups, both within and between display modes, produced a similar result ( $F < 1$ ). While completion times varied greatly between participants, trial response times for each subject remained consistent, regardless of display mode or marker pose. The relatively long performance time is a result of users aiming for high confidence values. As such, we often observed that users would align the virtual cube with the marker, then add some error and correct it again. We believe that this is a result of the user’s intention to verify that slight changes do not visually improve the alignment results.

#### 7.4.2 User Confidence

Figure 7.6 displays the distribution of confidence values for the VST and OST display sets. The occurrence rate, as a percentage of all subject values, is shown for each confidence level. The occurrence rates for confidence levels 10, 5, 4, 3, 2, and 1 are all below 10% for OST trials; as are the 5, 4, 3, 2, and 1 confidence levels for VST trials. Users selected confidence level 10, 9, 8, 7, and 6 at rates above 10% during VST trials, at 10.14%, 23.2%, 23.33%, 20.83%, and 13.2% respectively. Only user confidence values of 9, 8, 7, and 6 occur at rates above 10% during OST trials, at 15.8%, 26.4%, 24.6%, and 16.5% respectively. The highest three confidence levels, 8-10, combined, yield over half, 56.67%, of the total responses for the VST mode compared to only 47.6% of the total responses for the OST trial sets.

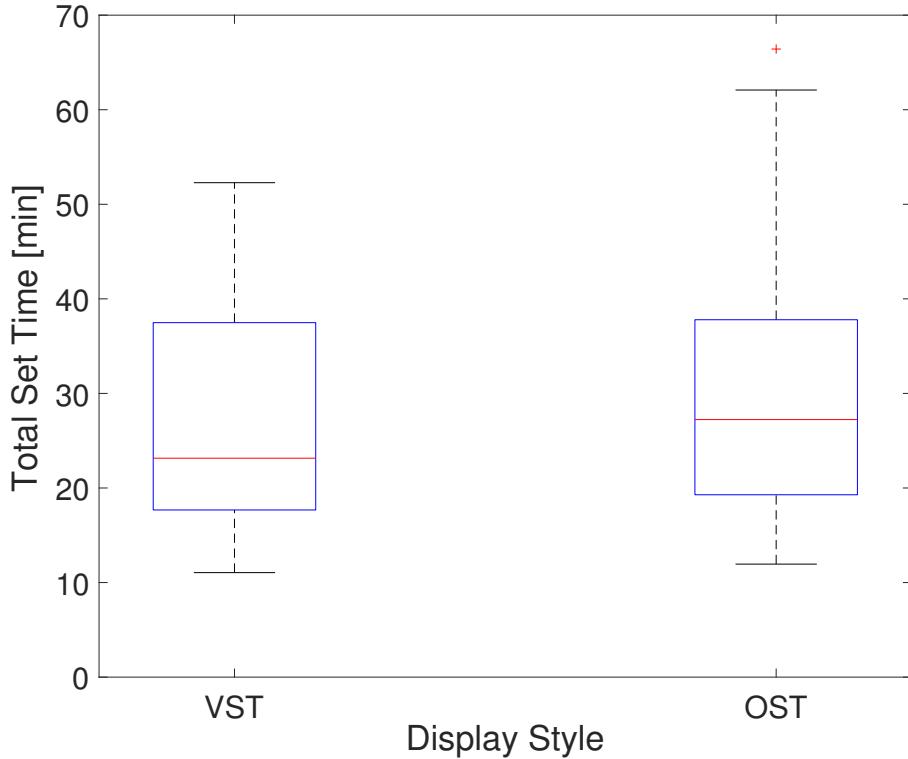


Figure 7.5: Display mode set completion times, in minutes.

The mean confidence and standard deviation for all VST and OST trials are mean = 7.62, stddev = 1.6 and mean = 7.28, stddev = 1.54 respectively. ANOVA results show a statistical significance between the two groups, ( $F(1, 1438) = 16.128, p < 0.001$ ). A Kruskal-Wallis test also reveals a statistical significance between display style sets, yielding ( $\chi^2(1) = 19.89, p < 0.001$ ).

Additional comparisons were performed for confidence responses across marker orientation and position groups. Statistical significance was found between values for sets at orientation (0, 0) and all other orientation groups within the VST and OST display types themselves, ( $F(14, 705) = 3.64, p < 0.001$ ) and ( $F(14, 705) = 4.14, p < 0.001$ ) respectively. No further significance was discovered through comparison between display mode groups at each orientation. Similar analysis of confidence grouped by target marker position revealed no significance between display groups at .6 m and 1 m marker distances. Significance was discovered between the VST and OST confidence values at 2 m, ( $F(1, 478) = 14.86, p < 0.001$ ). Figure 7.7 provides boxplots of the confidence values for each marker distance and display mode.

### 7.4.3 Alignment Accuracy

Accuracy metrics for user alignment include both x and y translational as well as absolute orientation error. While each error type could be categorized by

Table 7.1: VST vs OST Statistical Significance in Accuracy by Marker Orientation. Row 1 provides images of the marker and cube by orientation configuration. Rows 2, 3, and 4 provide ANOVA, means, and stddev results for absolute orientation and x- and y-axis translation error. Bold values indicate statistical significance between display modes.

		(X,Y)																		
		(-60,-60)			(-60,-30)			(-60,0)			(-60,30)			(-30,-60)			(-30,0)			(-30,30)
O	F(1, 94)=	1.89	0.66	1.19	0.39	1.96	2.18	2.31	2.09	0.37	2.60	1.09	0.13	1.16	1.21	0.63				
	p =	0.1725	0.4187	0.2777	0.536	0.165	0.1432	0.1319	0.1511	0.5433	0.11	0.2984	0.7144	0.2833	0.2751	0.4293				
VST	mean <sup>o</sup>	3.4837	4.114518	4.697888	4.581095	4.54568	3.943852	4.038162	3.95848	4.765658	4.837247	3.038949	5.351697	3.202456	5.661742	8.599831				
	stddev <sup>o</sup>	6.620599	4.755278	3.801509	4.462169	3.525571	3.186917	2.398861	2.218512	3.003582	12.4581	2.090438	12.851296	1.832061	4.95654	12.90922				
OST	mean <sup>o</sup>	2.28452	3.189042	4.021029	4.202069	8.488677	4.339988	5.114964	6.905399	6.702964	3.688887	5.580499	3.791801	3.954709	10.825231	6.387261				
	stddev <sup>o</sup>	3.928528	3.390997	4.522003	5.57562	25.885065	3.1749	3.952583	13.262453	8.582166	2.970994	12.408988	3.434984	3.319623	24.214267	5.522806				
X	F(1, 94)=	12.41	7.62	34.84	21.97	4.42	2.1	0.61	44.16	5.23	0.32	0.68	0.02	24.33	4.8	3.69				
	p =	<b>0.0007*</b>	<b>0.007*</b>	<b>0.001*</b>	<b>0.001**</b>	<b>0.0381*</b>	<b>0.1504</b>	<b>0.437</b>	<b>0.001*</b>	<b>0.0245*</b>	<b>0.5711</b>	<b>0.412</b>	<b>0.8771</b>	<b>0.001**</b>	<b>0.0309*</b>	0.0582				
VST	mean <sup>o</sup>	0.051473	0.173823	0.115329	0.208384	0.130121	0.093149	0.095478	0.112088	0.128635	0.060019	0.126252	0.071594	0.156171	0.029517	0.04141				
	stddev <sup>o</sup>	0.044137	0.164506	0.093486	0.179266	0.118465	0.086505	0.098832	0.127799	0.127439	0.054523	0.134188	0.066003	0.129309	0.027465	0.028594				
OST	mean <sup>o</sup>	4.4628	0.72678	0.695956	0.62223	4.0301	3.8185	1.0766	0.59766	0.89187	2.7705	3.644	1.8568	0.85187	1.7289	3.4375				
	stddev <sup>o</sup>	3.2268	0.55948	0.41679	0.50025	4.8141	3.2569	1.0441	0.42856	0.69411	2.73	2.7487	1.2224	0.61904	1.3533	2.4777				
Y	F(1, 94)=	49.48	38.87	0.35	33.77	14.17	3.63	1.66	1.12	9.56	9.51	62.9	17.81	18.17	69.76	79.61				
	p =	<b>0.001*</b>	<b>0.001**</b>	<b>0.5551</b>	<b>0.001**</b>	<b>0.0003**</b>	<b>0.0599</b>	<b>0.2003</b>	<b>0.2916</b>	<b>0.0026**</b>	<b>0.0027**</b>	<b>0.001**</b>	<b>0.001**</b>	<b>0.001**</b>	<b>0.001**</b>	<b>0.001**</b>				
VST	mean <sup>o</sup>	0.085214	0.059609	0.050742	0.05798	0.061684	0.10418	0.093753	0.119901	0.094007	0.069442	0.07923	0.079366	0.060546	0.15556	0.155582				
	stddev <sup>o</sup>	0.155083	0.061169	0.04205	0.053107	0.074036	0.104702	0.093483	0.115471	0.105418	0.061628	0.107914	0.056125	0.044826	0.179913	0.116964				
OST	mean <sup>o</sup>	0.210656	0.119046	0.185628	0.261924	0.199311	0.181281	0.160835	0.156256	0.196992	0.162763	0.157093	0.173914	0.169245	0.205166	0.172159				
	stddev <sup>o</sup>	0.103369	0.082693	0.092607	0.29664	0.08006	0.109644	0.09154	0.1052	0.321298	0.079947	0.096357	0.09724	0.091533	0.233826	0.119773				
	mean <sup>o</sup>	6.5639	6.6148	5.9635	6.1031	5.8571	2.2806	1.8484	2.2053	2.156	1.9574	4.3359	5.1384	4.4439	4.1685	3.958				
	stddev <sup>o</sup>	5.6113	5.0833	4.3859	4.591	2.3821	1.2762	2.0137	1.5197	1.4524	2.7642	4.0271	3.1702	2.8552	2.8646					

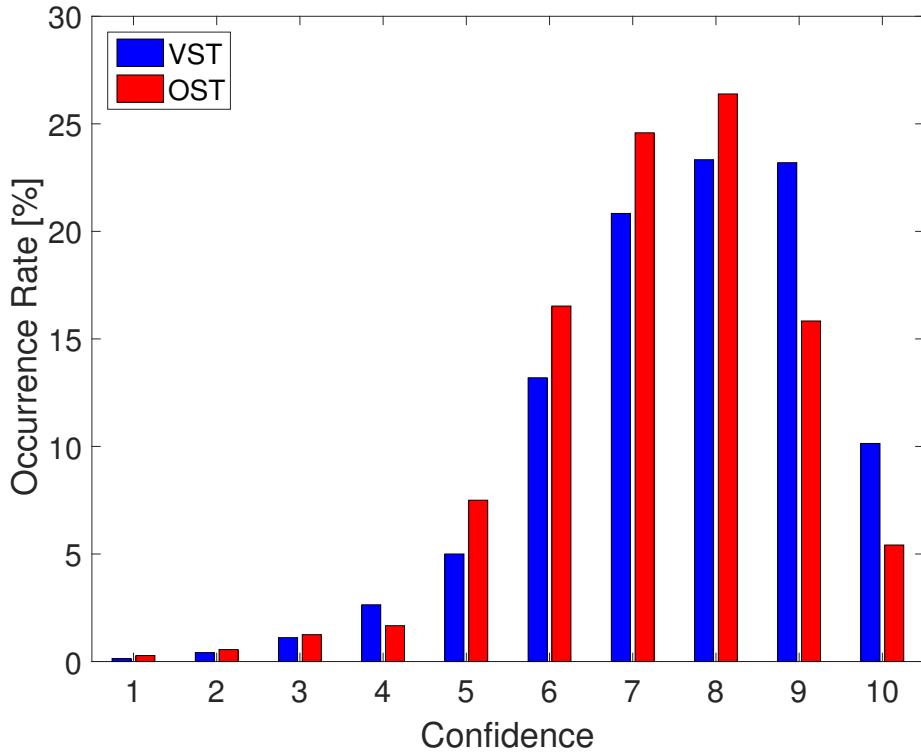


Figure 7.6: Confidence distributions for the VST and OST display sets, across all participants. x-axis values represent each of the 10 possible confidence levels. y-axis values represent occurrence rate, as a percentage of all confidence responses.

positive or negative, higher or lower, value relative to the ground truth, we instead analyze the alignment results as magnitudes, or absolute error. By doing so, we remove false detections of significant effects due to tendencies of error along a particular direction. Even though certain target marker positions may influence the direction of error, the primary focus of our study is in the noticeability of alignment error by the user, and therefore, the direction of the error itself is not considered.

#### 7.4.3.1 Translational error

Translational error, for our experiment, refers to offset between the center of the alignment cube and the center of the marker, along either the x- or y-axis relative to the user's view. In addition, we convert the translational error into visual angle error, to facilitate comparison and applicability of the results across a wider range of systems. In the VST-mode the visual angle decreased as the object was moved further away from the user, as we expected, and was similar for both the translation along the x-axis and the y-axis. In the OST visualization we observed a similar, but less prominent, decrease in the error along the x-axis. However, along the y-axis the error did not decrease. We

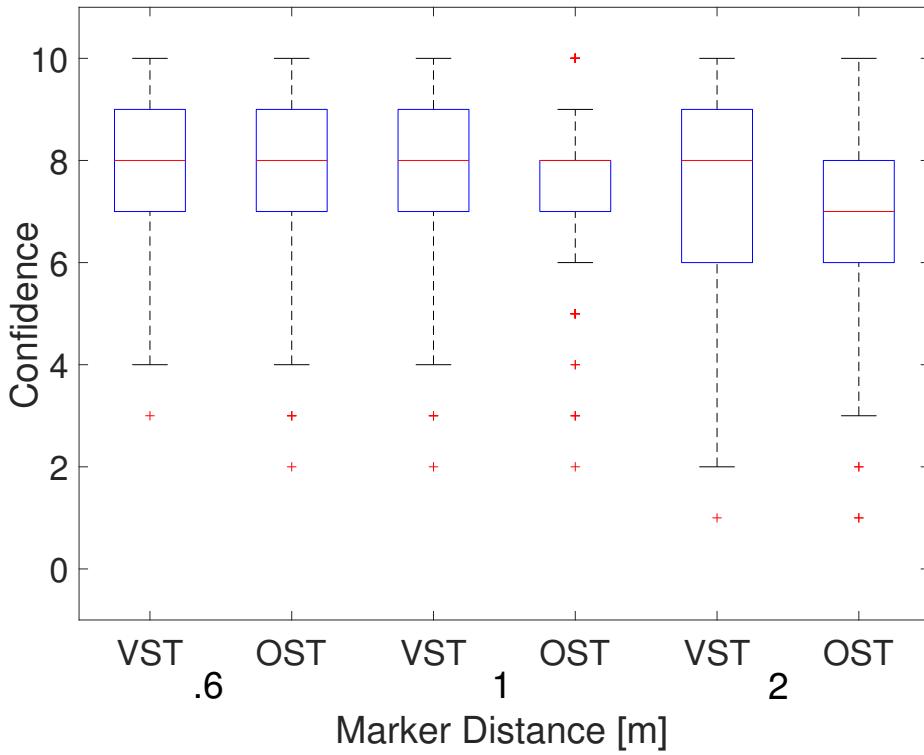


Figure 7.7: Confidence values by marker distance.

assume that the visualization played an essential part in this.

Figures 7.8 and 7.9 provide boxplots of the translation error along the x- and y-axes, respectively, by marker distance. ANOVA testing between x translation error groups revealed no statistical significance between the display mode types at .6 m, ( $F(1, 478) = 2.09, p = 0.15$ ). Significant effects between display styles were determined at the 1 m, ( $F(1, 478) = 17.36, p < 0.001$ ), and 2 m, ( $F(1, 478) = 98.01, p < 0.001$ ), marker distances. Comparison between y translation error groups shows statistical significance between display modes at .6 m, ( $F(1, 478) = 11.83, p < 0.001$ ), 1 m, ( $F(1, 478) = 142.46, p < 0.001$ ), and 2 m, ( $F(1, 478) = 419.93, p < 0.001$ ). Additional statistical tests were performed to identify significance between display types at all 15 marker orientations for x and y error. Table 7.1, rows 3 and 4, respectively, provide the results of these tests. Nearly all y orientations produced significant results between display modes.

#### 7.4.3.2 Orientation error

Orientation error denotes the absolute rotational difference between the marker and the alignment cube as determined by the difference in quaternion orientations. Boxplots of the orientation error separated by marker distance and display mode are provided in Figure 7.10. ANOVA tests show no significant difference between errors at the .6 m marker distance ( $F(1, 478) = 2.48, p = 0.1156$ ),

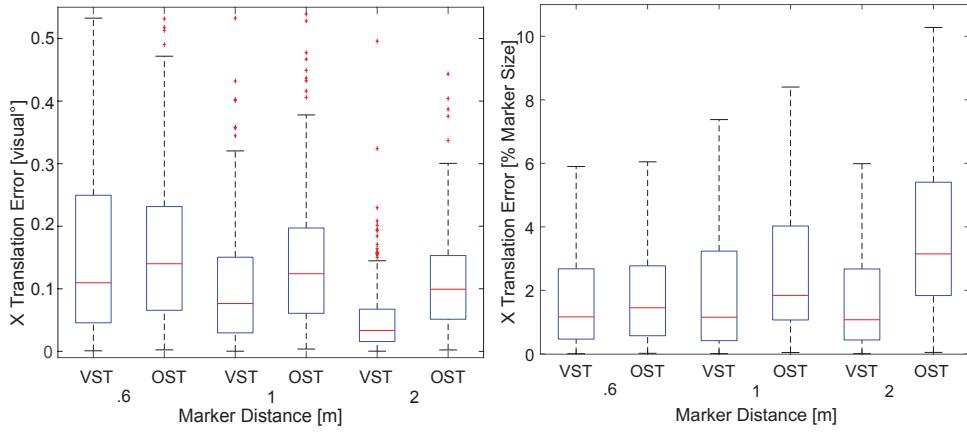


Figure 7.8: Translation error along the x-axis by marker distance. (left) Values along the y-axis represent error magnitudes in terms of visual angle and (right) relative to the marker size at the displayed distance.

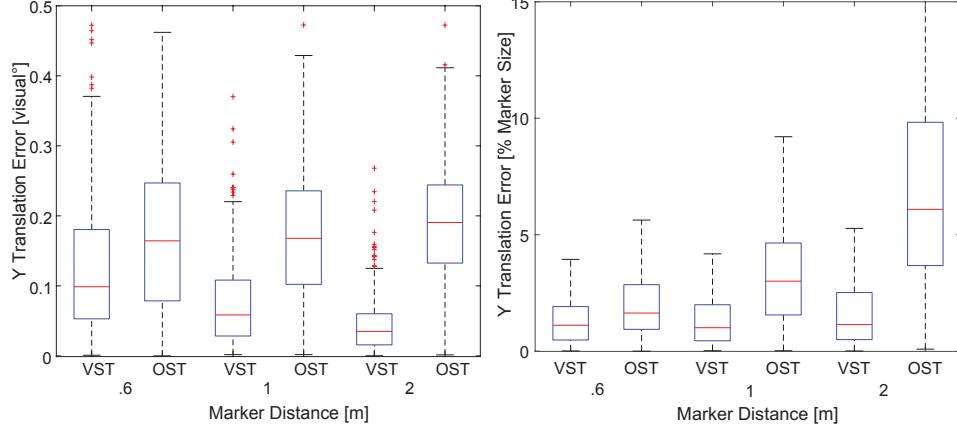


Figure 7.9: Translation error along the y-axis by marker distance. (left) Values along the y-axis represent error magnitudes in terms of visual angle and (right) relative to the marker size at the displayed distance.

no significance between the two groups at 1 m, ( $F(1, 478) = 0.05$ ,  $p = 0.8267$ ), and no significance between the errors at 2 m ( $F(1, 478) = 1.57$ ,  $p = 0.211$ ). We also did not find any significant error between the OST and VST modes after separating the rotational error into its roll, pitch and yaw components. This finding is contrary to that found by [Madsen and Stenholz \(2014\)](#). Results from their investigation yielded significant differences in accuracy levels between all three major rotation axes. The magnitude of noticeable rotation errors measured in indirect AR was similar to our findings. Compared to the results measured in the direct AR scenario, our findings indicate that when wearing an HMD device, it is not necessary to achieve the same degree of accuracy and that users are generally more forgiving towards rotational misalignments. Additional statistical tests were performed across the 15 marker orientation types. Row 2 of Table 7.1 provides the results of these tests. Only

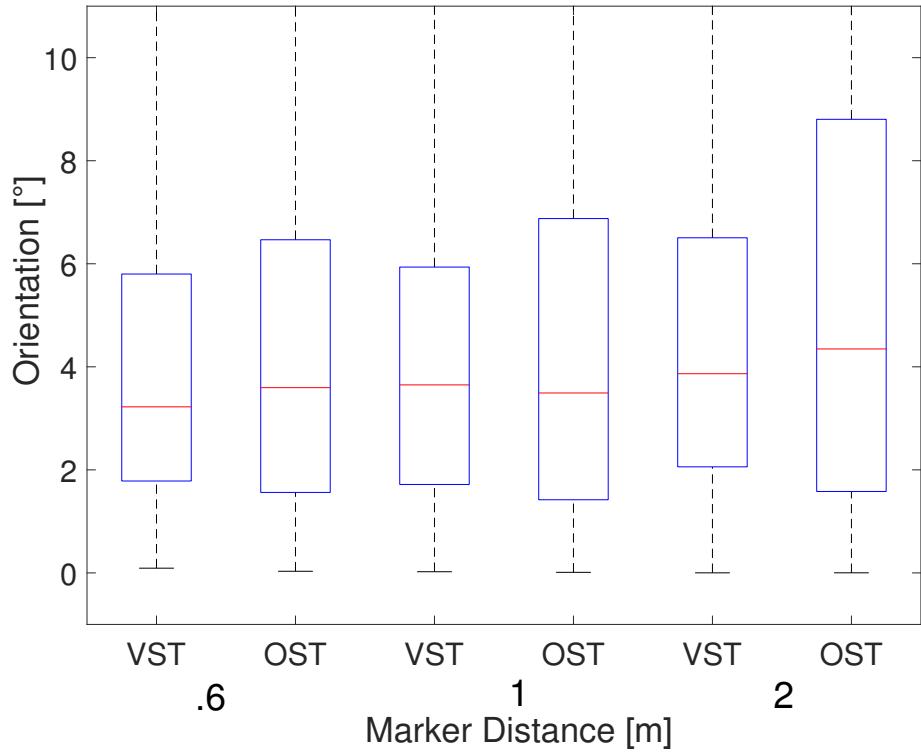


Figure 7.10: Angular error by marker distance. Values along the y-axis represent the rotational difference between quaternion orientations.

one orientation, (-30, 60), was found to exhibit any statistical effect due to display mode.

#### 7.4.3.3 User experience

We compared the users with and without previous AR experience with each other and found that the ANOVA test revealed that both groups performed equally well. Additionally, we found that the 9 participants with gaming experience ( $\geq 5$  hours per week) performed statistically better than those without. There were statistically significant differences in the translation along the x-axis ( $F(1, 808) = 18.46, p < 0.0001$ ), y-axis ( $F(1, 808) = 11.5, p = 0.0007$ ), as well as the rotational alignment ( $F(1, 808) = 25.93, p < 0.0001$ ).

#### 7.4.4 Discussion

All factors within our experimental design were held constant across display modes, except one. We theorized that this defining factor, transparency of AR content, would play the biggest role in effecting user perception. Therefore, we will perform our discussion within the context of the following three hypotheses:

- User sensitivity to alignment errors will be highest in the VST mode than in the OST mode, due to transparency effects.
- The transparency of the OST display mode will result in more ambiguity of alignment position, resulting in lower user confidence responses.
- The VST mode will be easier to perform allowing users to finish the trial set faster.

#### **7.4.4.1 Lower OST error sensitivity due to transparency.**

Alignment errors within the OST mode, overall, were consistently higher than those in the VST mode. Significant effects were found across both marker distance and marker orientations. It is possible that systematic error due to the two projector system created a bias toward OST error. While this is possible, it is unlikely that any significant systemic influence were present, simply due to the inconsistent levels across conditions. Effects due to marker distance were seen, primarily in translational alignment. It is likely, that by changing the marker distances, we unintentionally decreased the sensitivity of translation error due to depth scaling, i.e. a noticeable 1 cm error at .6 m will be considerably smaller at 2 m. This is the primary reason for expressing positional errors as visual angles and percentages of the marker size. Nonetheless, OST error tolerance remained statistically higher than VST at each distance level. It is worth noting, that orientation errors were not significantly different between display modes. Although, rotational alignment perception is highly dependent upon context. Since we neither allowed the user to move about in the physical room nor the virtual experiment space, the rotational misalignment exhibited is potentially a by-product of viewpoint and rendering style. Transparency of the AR content, remains the most viable explanation for our registration sensitivity results. While we did ensure that a constant level of transparency was maintained throughout the entirety of the experiment, we did not specifically attempt to control it. Post experiment surveys did reveal that nearly every user rated the difficulty of the OST set higher than that of the VST. Additionally, users stated that it was difficult to clearly distinguish between the virtual cube and the target marker at the maximum distance.

#### **7.4.4.2 Lower OST confidence due to transparency.**

There is a natural correlation between accuracy and confidence. Our driving hypothesis for this study is that the perceptual differences between OST and VST display modes will result in different thresholds for registration error in each system type. Lower sensitivity to alignment error should be evidenced by lower confidence rates, longer task times, and of course, larger errors. An examination of our experimental results seem to support these assertions.

While set completion time did not show a significant difference between display modes, the VST mean and general tendency was lower than that of the OST condition. A logical cause of longer response time is an increased difficulty associated with the OST trials. Support for this hypothesis is provided by the confidence metric. Participants' responses in VST trials were found to be statistically more confident than those within the OST set. Of course, the mean confidences of each display set, 7.62 for VST and 7.28, are still quite close. While, transparency, and in turn difficulty, no doubt played a role in user's confidence, the novelty of the task itself, user unfamiliarity with AR content, or simply poor spatial reasoning skills are also potential influences worthy of mention.

#### 7.4.4.3 Slower OST performance due to transparency.

It is, of course, natural, that if the OST trial set was more difficult to perform that the overall completion time across the set would be significantly greater compared to the VST. Surprisingly, though completion times remained nearly identical between the two display modes. It is likely, that the novelty of the task, to most users, required users to take longer. Also, before each trial set, users were instructed to take their time and to try to be as accurate as possible. Nonetheless, there were no statistical influences from time found on any factor. Therefore, the hypothesis that difficulty would impede completion of an OST AR task, is the only one of our three that fails.

## 7.5 Conclusion

We have presented the results of an evaluation of subjectively perceived spatial consistency in a marker-based AR scene viewed through an HMD. A simulated environment was used instead of an actual device to ensure consistency between the presentations and prevent unmodeled errors as a result of the OST-HMD calibration. Our results indicate that although both types of HMDs display similar tendencies, there were significant differences in the JND of the spatial misalignment.

We found that the rotational tolerance is within 3-4° and the translational error remains less than 0.1-0.2°. Overall the JND error was around 1% of the marker size for the VST mode and 1-6% for the OST mode. Hereby the tolerance level in VST mode is not impacted by the distance to the object. On the other hand, in the OST mode users accepted larger misalignments as the distance to the target object increased. This is mostly due to the transparency difference between the two visualization modes. As the model used in our experiment was very simple, with strongly defined edges and orientations, it is likely that larger errors will remain unnoticeable with more complex models or less prominent targets, e.g., a template image instead of a binary marker.

The misalignment of CIC observed in Chapter 4 corresponds to an error of approximately  $0.24^\circ$ . This value is larger than the thresholds observed in our experiment, however further improvements of the method are likely to result in misalignment that can no longer be perceived by most users.

## Limitations and future work.

Our experimental design can be improved upon, for future studies, in a number of ways. The inclusion of fewer marker orientation states, for example, would greatly reduce the number of trial combinations required for comparison and analysis. While we chose the orientation levels to ensure a representative coverage of possible viewing angles, it is also reasonable to presume that errors will occur equally around each axis. Therefore, mirrored angles, such as  $(0, 30)$  and  $(0, -30)$  for example, provide redundant measures. Additionally, showing the marker at three distance levels artificially decreases the sensitivity to translational error at the higher levels. A correction to this method is to maintain the relative distance and modify the scale instead. This will ensure that relative distances between target and marker remain constant and do not shrink or expand with depth.

Although the two-projector system successfully presented the view visible through an OST-HMD, it also imposed a number of limitations. First, as the projectors displayed the scene as well as the virtual content on the same plane it remains unclear how varying focal planes in actual HMD devices would impact the perception of virtual content. Furthermore, the dual-projector system presented the user with a mono-view, which did not allow to evaluate how estimation errors in the depth of the target object influence the user's perception.

As our investigation did not differentiate between the various error sources, but focused on the overall error instead, future investigations have to answer how the magnitude of the different registration error sources—modeling, localization and OST-HMD calibration—impacts the overall experience.

In the OST mode the overlays produced by the two-projector setup contained a very high degree of transparency, something users would try to avoid in a standard OST-HMD. This higher transparency significantly increased the difficulty for users to resolve the alignment, artificially inflating the response time and deflating the confidence levels. It is essential for future experimental iterations to include a transparency control, if only to prevent unnecessarily high difficulty. Also, as the transparency proved to be the main factor in the different performance of the OST and VST setups, in future experiments the impact of the degree of transparency on user spatial error perception should be investigated.

Finally, in our experiment the augmentation was shown in the region focused by the users, where the perceived image has the highest quality. Images in the peripheral vision are perceived in lower contrast and resolution,

therefore it is necessary to investigate if higher thresholds are acceptable for augmentations shown in the peripheral vision.



# CHAPTER 8

# Conclusion

---

AR can create an intuitive presentation of information for a variety of fields, including, but not limited, to entertainment, training, surgery, and navigation. Current systems use a VST approach to present augmentations, however OST systems offer a more natural interface. Current systems still suffer from a large variety of problems. Through analysis of the corneal reflection under natural illumination it is possible to understand how the surroundings are perceived by the user. In this dissertation we have explored how corneal reflections can be used to recover spatial information of the eye and shown its application in OST environments, in particular in OST-HMDs. The results indicate that CI can be used to improve the accuracy and robustness compared to SOTA methods. It follows a summary of the findings and future applications.

## 8.1 Summary

**Corneal imaging calibration of OST-HMDs.** We have presented a novel approach for calibration of OST-HMDs. Our method exploits corneal imaging to detect the reflection of the spatially calibrated HMD-screen on the cornea surface. The proposed method uses the reflection of the HMD-screen on the corneal surface to estimate the position of the cornea. We recover the center of projection, estimated as the center of rotation of the eye. Our method does not require to detect the orientation of the eye and recovers the center of rotation from multiple observations. CIC presents the following benefits:

- An automated calibration of the OST-HMD that does not rely on accurate eye-pose estimation and uses only the detected reflection of the HMD-screen.
- Automated detection of the need for recalibration if multiple successive estimations of the position of the cornea do not support the current calibration.
- CIC displays a smaller bias in the error distribution than SOTA automated calibration methods, thus further reducing the noticeability of misalignment errors.

**Hybrid eye-pose estimation.** Our Hybrid eye-pose estimation adopts the SOTA methods for accurate eye-pose estimation under IR illumination for images taken under natural light. The resulting method displays the following benefits:

- The proposed method does not require the detection of face features, it can automatically segment the region of interest from feature matches detected in the corneal reflection and the scene model. Alternatively, the proposed dense tracking approach can be used to achieve robust tracking results.
- The iris contour detected under visible light is not subject to refraction on the corneal surface, thus it does not require complicated modeling of the off-axis pupil and refraction of the light on different mediums.
- Our method does not require extended learning sessions and can be applied in headworn and remote scenarios. The accurately estimated cornea position is also used to remove a number of false iris-contour candidates.
- The recovery of the iris on the corneal surface improves the robustness under extreme orientations, where the refraction of the light on the corneal surface is usually detected as part of the iris contour. Furthermore, the estimation complexity is reduced to only two parameters, the latitude and longitude.
- The constraint of the estimated position also enables estimation of the radius of the user’s iris without tracking and estimating a large number of parameters for every frame.
- We show that the developed Hybrid eye-pose estimation methods can be applied in conjunction with an OST-HMD for eye-gaze-based interaction in outdoor environments.

**Cornea tracking.** Existing methods for cornea position estimation use IR illumination or visible light sources to detect the respective glints in the corneal reflection. We have proposed a new solution that utilizes the reflection of a densely reconstructed scene. The proposed solution has the following benefits:

- The proposed method does not require active illumination of the scene thus it can be used in outdoor and indoor environments. It does not require extensive geometric calibration of the environment as the scene model can be acquired during runtime through reconstruction algorithms.

- A dense mapping area is used instead of sparse features from IR-LEDs or scene feature matching. As a result the proposed method is more robust against possible false matches or missing glints or reflections.
- The densely rendered reflection of the scene on the cornea improves the robustness of the tracking against the unmodeled surface properties of the cornea, e.g., violation of the spherical assumption. The dense mapping can also be used to reconstruct a more detailed cornea model without the requirement for highly complicated and specialized setups.

**User spatial consistency perception in HMD-based AR.** Finally, we have also conducted an empirical study, to improve our understanding of spatial consistency requirements in OST-HMD environments. Our study was conducted in a controlled simulation of the view through an HMD and we provide a comparison of VST and OST-Systems. Our results indicate that VST systems behave similar for handheld and headworn systems. Contrary to that, the OST system shows a much larger tolerance to translational misalignment but requires the same degree of rotational accuracy. The determined threshold can be used in future studies on OST-HMD calibration and to investigate other manifestations of the spatial error, e.g., jitter. Finally, our results suggest that transparency seems to have the largest impact on the user’s ability to notice spatial inconsistency. In the future it will be necessary to determine how different levels of transparency, e.g., due to different lighting conditions, impact the perception.

Our results indicate that the performance of current automated methods is on the border of noticeability and with further improvements an augmentation where users no longer notice spatial misalignment is within reach. It remains to verify if our analytical results apply in tests with an OST-HMD system.

## 8.2 Future Directions

In this work corneal imaging was applied to determine the spatial pose of the eye and thus improve the interaction and rendering of the content on the HMD screen. However, corneal imaging may also be used to address various other issues that arise when an OST-HMD is used for extended periods of time. Some potential future application scenarios are discussed in the following.

**Adaptive camera–display setups.** The current design of CIC assumes that the eye-tracking camera is rigidly attached to the HMD screen. However, such setups may be unviable for all users and some adjustments of the camera pose may be necessary. As such corneal imaging could be incorporated as part of an autonomous recalibration procedure, similar to [Nitschke et al. \(2009\)](#) and [Agrawal \(2013\)](#).

**Modification of visual properties.** Existing OST-HMD AR applications are developed as a black-box system, where no feedback on the user’s experience is available. As such, even if the experience degenerates, e.g., bad contrast between the virtual content and the scene, or changing lighting conditions, the content does not adjust for this change. Through analysis of the corneal reflection it is possible to develop algorithms that account for temporal variance of the content’s reflection and modify the content to improve the experience.

**Automatic user recognition.** An HMD that is being used by different persons may require a personalized interface, e.g., workers in maintenance. Corneal imaging systems can be used to recover user specific properties, i.e., the iris pattern, to differentiate between users and present personalized content.

**Scene-camera-less OST-HMD.** The requirement of scene cameras in OST-HMDs limits their acceptance in everyday scenarios. Through corneal imaging it is possible to extract vital scene information without the need of a scene camera. Although some results have reported successful localization of user’s through corneal imaging, it is still unclear, how this could work in a more general scenario or in scenes without a prior model.

**Life logging.** An OST-HMD that is used as a commodity tool is an ideal candidate to record information about the user’s everyday life, understand the social contacts and also help remember important information, e.g., to assist Alzheimer patients. Contrary to existing life-log methods, CI does not require a scene facing camera, thus the life log can be recorded without inconveniencing bystanders ([Nakazawa et al. \(2015\)](#)).

**Object recognition.** In a number of applications it is not necessary to recover the correct pose or the exact object a user is looking at. For life logging solutions or Human–Computer Interaction it is often sufficient to determine the general category of the gazed object, e.g., is the user looking at a chair or a bed? Corneal imaging can be used to extract this information even if the gazed object is not visible in the image taken by the observing camera, e.g., the user only rotates the eyes to look at the object but is facing in a different direction.

# Bibliography

Adelstein, B. D., Lee, T. G., and Ellis, S. R. (2003). Head tracking latency in virtual environments: psychophysics and a model. In *Proceedings Annual Meeting of the Human Factors and Ergonomics Society (HFES)*, volume 47, pages 2083–2087.

AGARD Conference Proceedings No. CP-433 (1988). *Motion Cues in Flight Simulation and Simulator Induced Sickness*. NATO.

Agrawal, A. (2013). Extrinsic Camera Calibration Without a Direct View Using Spherical Mirror. In *Proceedings IEEE International Conference on Computer Vision (ICCV)*, pages 2368–2375.

Agrawal, A. and Ramalingam, S. (2013). Single Image Calibration of Multi-Axial Imaging Systems. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1399–1406.

Ajanki, A., Billinghurst, M., Gamper, H., Järvenpää, T., Kandemir, M., Kaski, S., Koskela, M., Kurimo, M., Laaksonen, J., Puolämäki, K., Ruokolainen, T., and Tossavainen, T. (2011). An Augmented Reality Interface to Contextual Information. *Virtual Reality*, 15(2-3):161–173.

Altenhoff, B. M., Napieralski, P. E., Long, L. O., Bertrand, J. W., Pagano, C. C., Babu, S. V., and Davis, T. A. (2012). Effects of Calibration to Visual and Haptic Feedback on Near-Field Depth Perception in an Immersive Virtual Environment. In *Proceedings ACM Symposium on Applied Perception (SAP)*, pages 71–78.

Ardouin, J., Lécuyer, A., Marchal, M., Riant, C., and Marchand, E. (2012). FlyVIZ: A Novel Display Device to Provide Humans with 360° Vision by Coupling Catadioptric Camera with HMD. In *Proceedings ACM Symposium on Virtual Reality Software and Technology (VRST)*, pages 41–44.

Atchison, D. A. and Smith, G. (2000). *Optics of the Human Eye*. Butterworth-Heinemann.

Axholt, M., Skoglund, M., Peterson, S., Cooper, M., Schön, T., Gustafsson, F., Ynnerman, A., and Ellis, S. (2010). Optical See-Through Head Mounted Display: Direct Linear Transformation Calibration Robustness in the Presence of User Alignment Noise. In *Proceedings Annual Meeting of the Human Factors and Ergonomics Society (HFES)*, volume 54, pages 2427–2431.

Axholt, M., Skoglund, M. A., O’Connell, S. D., Cooper, M. D., Ellis, S. R., and Ynnerman, A. (2011). Parameter Estimation Variance of the Single Point

Active Alignment Method in Optical See-Through Head Mounted Display Calibration. In *Proceedings IEEE Virtual Reality (VR)*, pages 27–34.

Azuma, R. T. (1995). *Predictive Tracking for Augmented Reality*. PhD thesis, University of North Carolina.

Backes, M., Chen, T., Dürmuth, M., Lensch, H. P. A., and Welk, M. (2009). Tmpest in a Teapot: Compromising Reflections Revisited. In *Proceedings IEEE Symposium on Security and Privacy (SP)*, pages 315–327.

Bailey, R. E., Arthur III, J. J., and Williams, S. P. (2004). Latency Requirements for Head-Worn Display S/EVS Applications. In *Proceedings Society for Optics and Photonics (SPIE)*, pages 98–109.

Baker, S. and Nayar, S. K. (1999). A Theory of Single-Viewpoint Catadioptric Image Formation. *International Journal of Computer Vision (IJCV)*, 35(2):175–196.

Bau, O. and Poupyrev, I. (2012). REVEL: Tactile Feedback Technology for Augmented Reality. *ACM Transactions on Graphics (TOG)*, 31(4):89.

Bérard, P., Bradley, D., Nitti, M., Beeler, T., and Gross, M. (2014). High-Quality Capture of Eyes. *ACM Transactions on Graphics (Proceedings ACM SIGGRAPH Asia)*, 33(6):223.

Bleiweiss, A., Eshar, D., Kutliroff, G., Lerner, A., Oshrat, Y., and Yanai, Y. (2010). Enhanced Interactive Gaming by Blending Full-Body Tracking and Gesture Animation. In *ACM SIGGRAPH ASIA Sketches*, pages 34:1–34:2.

Bradski, G. (2000). The OpenCV Library. *Dr. Dobb's Journal of Software Tools*.

Bradski, G. and Kaehler, A. (2008). *Learning OpenCV*. O'Reilly Media, Inc., Sebastopol, CA.

Bulling, A. and Gellersen, H. (2010). Toward Mobile Eye-Based Human-Computer Interaction. *IEEE Pervasive Computing*, 9(4):8–12.

Buskirk, E. M. V. (1989). The Anatomy of the Limbus. *Eye*, 3(2):101 – 108.

Cerf, M., Harel, J., Einhäuser, W., and Koch, C. (2008). Predicting human gaze using low-level saliency combined with face detection. *Advances in Neural Information Processing Systems*, 20:241–248.

Chen, Q., Wu, H., and Wada, T. (2004). Camera Calibration with Two Arbitrary Coplanar Circles. In *Proceedings European Conference on Computer Vision (ECCV)*, pages 521–532.

Colaço, A., Kirmani, A., Yang, H. S., Gong, N.-W., Schmandt, C., and Goyal, V. K. (2013). Mime: Compact, Low-Power 3D Gesture Sensing for Interaction with Head Mounted Displays. In *Proceedings ACM Symposium on User Interface Software and Technology (UIST)*, pages 227–236.

Crick, R. P. and Khaw, P. T. (2003). *A Textbook of Clinical Ophthalmology*. World Scientific, Singapore, 3rd edition.

Duchowski, A. T., Shivashankaraiah, V., Rawls, T., Gramopadhye, A. K., Melloy, B. J., and Kanki, B. (2000). Binocular Eye Tracking in Virtual Reality for Inspection Training. In *Proceedings ACM Symposium on Eye Tracking Research & Applications (ETRA)*, pages 89–96.

Ellis, S. R., Mania, K., Adelstein, B. D., and Hill, M. I. (2004). Generalizability of latency detection in a variety of virtual environments. In *Proceedings Annual Meeting of the Human Factors and Ergonomics Society (HFES)*, volume 48, pages 2632–2636.

Epson (2015). Moverio BT 200. [http://www.epson.com/alf\\_upload/pdfs/brochure\\_moverio.pdf](http://www.epson.com/alf_upload/pdfs/brochure_moverio.pdf). Last accessed on January 2, 2016.

Escudero-Sanz, I. and Navarro, R. (1999). Off-axis aberrations of a wide-angle schematic eye model. *Journal of the Optical Society of America (JOSA)*, 16(8):1881–1891.

Ferrer, V., Yang, Y., Perdomo, A., and Quarles, J. (2013). Consider Your Clutter: Perception of Virtual Object Motion in AR. In *IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 1–6.

Fove Inc. (2015). FOVE. <http://www.getfove.com/>. Last accessed on January 2, 2016.

Freitag, S., Weyers, B., Bönsch, A., and Kuhlen, T. W. (2015). Comparison and Evaluation of Viewpoint Quality Estimation Algorithms for Immersive Virtual Environments. In *Proceedings International Conference on Artificial Reality and Telexistence and Eurographics Symposium on Virtual Environments (ICAT-EGVE)*.

Gao, C., Lin, Y., and Hua, H. (2012). Occlusion Capable Optical See-through Head-Mounted Display Using Freeform Optics. In *IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 281–282.

Geiger, A., Moosmann, F., Car, O., and Schuster, B. (2012). Automatic Calibration of Range and Camera Sensors using a Single Shot. In *Proceedings International Conference on Robotics and Automation (ICRA)*, pages 3936–3943.

Genc, Y., Sauer, F., Wenzel, F., Tuceryan, M., and Navab, N. (2000). Optical See-Through HMD Calibration: A Stereo Method Validated with a Video See-Through System. In *Proceedings IEEE/ACM International Symposium on Augmented Reality (ISAR)*, pages 165–174.

Gkioulekas, I., Xiao, B., Zhao, S., Adelson, E. H., Zickler, T., and Bala, K. (2013). Understanding the Role of Phase Function in Translucent Appearance. *ACM Transaction on Graphics*, 32(5):147:1–147:19.

Goncharov, A. V., Nowakowski, M., Sheehan, M. T., and Dainty, C. (2008). Reconstruction of the Optical System of the Human Eye with Reverse Ray-Tracing. *Optics Express*, 16(3):1692–1703.

Gramkow, C. (2001). On Averaging Rotations. *Journal of Mathematical Imaging and Vision*, 15(1-2):7–16.

Gruber, L., Richter-Trummer, T., and Schmalstieg, D. (2012). Real-Time Photometric Registration from Arbitrary Geometry. In *Proceedings IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 119–128.

Guennebaud, G., Jacob, B., et al. (2010). Eigen v3. <http://eigen.tuxfamily.org>. Last accessed on January 2, 2016.

Guestrin, E. D. and Eizenman, M. (2006). General Theory of Remote Gaze Estimation Using the Pupil Center and Corneal Reflections. *IEEE Transactions on Biomedical Engineering (TBME)*, 53(6):1124–1133.

Gullstrand, A. (1909). *Helmholtz's Handbuch der Physiologischen Optik*, volume 1, chapter Appendices II and IV, pages 301–358, 382–415. Voss Hamburg.

Hallinan, P. W. (1991). Recognizing human eyes. In *Proceedings Society for Optics and Photonics (SPIE)*, volume 1570, pages 214–226.

Halstead, M. A., Barsky, B. A., Klein, S. A., and Mandell, R. B. (1996). Reconstructing Curved Surfaces From Specular Reflection Patterns Using Spline Surface Fitting of Normals. In *Proceedings ACM SIGGRAPH*, pages 335–342.

Hansen, D. W. and Hansen, J. P. (2006). Robustifying Eye Interaction. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 152–152.

Hansen, D. W. and Ji, Q. (2010). In the Eye of the Beholder: A Survey of Models for Eyes and Gaze. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 32(3):478–500.

Hartley, R. and Zisserman, A. (2003). *Multiple View Geometry in Computer Vision*. Cambridge University Press, New York, NY, 2nd edition.

Hincapie-Ramos, J. D., Ivanchuk, L., Sridharan, S. K., and Irani, P. (2014). SmartColor: Real-Time Color Correction and Contrast for Optical See-Through Head-Mounted Displays. In *IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 187–194.

Hua, H., Hu, X., and Gao, C. (2013). A high-resolution optical see-through head-mounted display with eyetracking capability. *Optics Express*, 21(25):30993–30998.

Hua, H., Krishnaswamy, P., and Rolland, J. P. (2006). Video-based eyetracking methods and algorithms in head-mounted displays. *Optics Express*, 14(10):4328–4350.

Huang, J. and Wechsler, H. (1999). Eye Detection Using Optimal Wavelet Packets and Radial Basis Functions (RBFs). *International Journal of Pattern Recognition and Artificial Intelligence (IJPRAI)*, 13(07):1009–1025.

Huber, M., Pustka, D., Keitler, P., Echtler, F., and Klinker, G. (2007). A System Architecture for Ubiquitous Tracking Environments. In *Proceedings IEEE/ACM International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 211–214.

Ishiguro, Y. and Rekimoto, J. (2011). Peripheral Vision Annotation: Noninterference Information Presentation Method for Mobile Augmented Reality. In *Proceedings ACM Augmented Human International Conference (AH)*, pages 8:1–8:5.

Itoh, Y., Dzitsiuk, M., Amano, T., and Klinker, G. (2015). Semi-Parametric Color Reproduction Method for Optical See-Through Head-Mounted Displays. *IEEE Transactions on Visualization and Computer Graphics (TVCG) (Proceedings IEEE International Symposium on Mixed and Augmented Reality (ISMAR))*, 21(11):1269–1278.

Itoh, Y. and Klinker, G. (2014a). Interaction-Free Calibration for Optical See-Through Head-Mounted Displays based on 3D Eye Localization. In *Proceedings IEEE Symposium on 3D User Interfaces (3DUI)*, pages 75–82.

Itoh, Y. and Klinker, G. (2014b). Performance and Sensitivity Analysis of INDICA: INteraction-Free DIspaly CALibration for Optical See-Through Head-Mounted Displays. In *Proceedings IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 171–176.

Itoh, Y. and Klinker, G. (2015a). Light-Field Correction for Spatial Calibration of Optical See-Through Head-Mounted Displays. *IEEE Transactions*

on Visualization and Computer Graphics (TVCG) (Proceedings IEEE Virtual Reality (VR)), 21(4):471–480.

Itoh, Y. and Klinker, G. (2015b). Simultaneous Direct and Augmented View Distortion Calibration of Optical See-Through Head-Mounted Displays. In *Proceedings IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*.

Jachnik, J., Newcombe, R. A., and Davison, A. J. (2012). Real-Time Surface Light-field Capture for Augmentation of Planar Specular Surfaces. In *Proceedings IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*.

Jacob, R. J. (1990). What You Look at is What You Get: Eye Movement-based Interaction Techniques. In *Proceedings SIGCHI Conference on Human Factors in Computing Systems (CHI)*, pages 11–18.

Jacob, R. J. (1995). Eye Tracking in Advanced Interface Design. *Virtual Environments and Advanced Interface Design*, pages 258–288.

Jarabo, A., Eyck, T. V., Sundstedt, V., Bala, K., Gutierrez, D., and O’Sullivan, C. (2012). Crowd Light: Evaluating the Perceived Fidelity of Illuminated Dynamic Scenes. In *Computer Graphics Forum*, volume 31, pages 565–574. Wiley Online Library.

Jerald, J. and Whitton, M. (2009). Relating Scene-Motion Thresholds to Latency Thresholds for Head-Mounted Displays. In *Proceedings IEEE Virtual Reality (VR)*, pages 211–218.

Johnson, M. K. and Farid, H. (2007). Exposing Digital Forgeries Through Specular Highlights on the Eye. In *Proceedings International Workshop on Information Hiding (IH)*, pages 311–325.

Kassner, M., Patera, W., and Bulling, A. (2014). Pupil: An Open Source Platform for Pervasive Eye Tracking and Mobile Gaze-based Interaction. In *Proceedings ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp): Adjunct Publication*, pages 1151–1160.

Kato, H. and Billinghurst, M. (1999). Marker Tracking and HMD Calibration for a Video-based Augmented Reality Conferencing System. In *Proceedings IEEE/ACM International Workshop on Augmented Reality (IWAR)*, pages 85–94.

Kaufman, P. L. and Alm, A. (2011). *Adler’s Physiology of the Eye*. Mosby, Inc., St. Louis, MO, 11th edition.

Khuong, B. M., Kiyokawa, K., Miller, A., La Viola, J., Mashita, T., and Takemura, H. (2014). The Effectiveness of an AR-based Context-Aware Assembly Support System in Object Assembly. In *Proceedings IEEE Virtual Reality (VR)*, pages 57–62.

Kilic, A. and Roberts, C. (2013). *Corneal Topography: From Theory to Practice*. Kugler Publications.

Kim, K.-N. and Ramakrishna, R. S. (1999). Vision-Based Eye-Gaze tracking for Human Computer Interface. In *Proceedings IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 324–329.

Kishishita, N., Kiyokawa, K., Orlosky, J., Mashita, T., Takemura, H., and Kruijff, E. (2014). Analysing the Effects of a Wide Field of View Augmented Reality Display on Search Performance in Divided Attention Tasks. In *Proceedings IEEE/ACM International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 177–186.

Kiyokawa, K. (2007). A wide field-of-view head mounted projective display using hyperbolic half-silvered mirrors. In *Proceedings IEEE/ACM International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 207–210.

Kiyokawa, K., Kurata, Y., and Ohno, H. (2001). An optical see-through display for mutual occlusion with a real-time stereovision system. *Computers & Graphics*, 25(5):765–779.

Klein, G. and Murray, D. (2007). Parallel Tracking and Mapping for Small AR Workspaces. In *Proceedings IEEE/ACM International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 225–234.

Klemm, M., Hoppe, H., and Seebacher, F. (2014). Non-Parametric Camera-Based Calibration of Optical See-Through Glasses for Augmented Reality Applications. In *Proceedings IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 273–274.

Kolakowski, S. M. and Pelz, J. B. (2006). Compensating for Eye Tracker Camera Movement. In *Proceedings ACM Symposium on Eye Tracking Research & Applications (ETRA)*, pages 79–85.

Kooijman, A. C. (1983). Light distribution on the retina of a wide-angle theoretical eye. *Journal of the Optical Society of America (JOSA)*, 73(11):1544–1550.

Kurz, D., Meier, P. G., Plopski, A., and Klinker, G. (2014). Absolute Spatial Context-Aware Visual Feature Descriptors for Outdoor Handheld Camera Localization. In *Proceedings International Conference on Computer Vision Theory and Applications (VISAPP)*, pages 56–67.

Křivánek, J., Ferwerda, J. A., and Bala, K. (2010). Effects of Global Illumination Approximations on Material Appearance. *ACM Transactions on Graphics (TOG) (Proceedings ACM SIGGRAPH)*, 29(4):112:1–112:10.

Le Grand, Y. and El Hage, S. G. (1980). *Physiological Optics*. Springer Berlin.

Lee, Y.-C., Lee, J. D., and Boyle, L. N. (2007). Visual Attention in Driving: The Effects of Cognitive Load and Visual Disruption. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 49(4):721–733.

Lefohn, A., Budge, B., Shirley, P., Caruso, R., and Reinhard, E. (2003). An Ocularist’s Approach to Human Iris Synthesis. *IEEE Computer Graphics and Applications (CG&A)*, 23(6):70–75.

Lhuillier, M. (2008). Automatic scene structure and camera motion using a catadioptric system. *Computer Vision and Image Understanding (CVIU)*, 109(2):186 – 203.

Li, A., Montaño, Z., Chen, V. J., and Gold, J. I. (2011). Virtual reality and pain management: current trends and future directions. *Pain Management*, 1(2):147–157.

Li, D., Winfield, D., and Parkhurst, D. J. (2005). Starburst: A hybrid algorithm for video-based eye tracking combining feature-based and model-based approaches. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 79–86.

Li, F., Kolakowski, S., and Pelz, J. (2007). Using Structured Illumination to Enhance Video-Based Eye Tracking. In *Proceedings IEEE International Conference on Image Processing (ICIP)*, pages 373–376.

Lieberknecht, S., Huber, A., Ilic, S., and Benhimane, S. (2011). RGB-D Camera-Based Parallel Tracking and Meshing. In *Proceedings IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 147–155.

Liou, H. L. and Brennan, N. A. (1997). Anatomically accurate, finite model eye for optical modeling. *Journal of the Optical Society of America (JOSA)*, 14(8):1684–1695.

Livingston, M. A. and Ai, Z. (2008). The Effect of Registration Error on Tracking Distant Augmented Objects. In *Proceedings IEEE/ACM International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 77–86.

Lotmar, W. (1971). Theoretical Eye Model with Aspherics. *Journal of the Optical Society of America (JOSA)*, 61(11):1522–1529.

Lowe, D. (2004). Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision (IJCV)*, 60(2):91–110.

Madsen, J. and Stenholz, R. (2014). How Wrong Can You Be: Perception of Static Orientation Errors in Mixed Reality. In *Proceedings IEEE Symposium on 3D User Interfaces (3DUI)*, pages 83–90.

Maier, P., Dey, A., Waechter, C., Sandor, C., Tönnis, M., and Klinker, G. (2011). An Empiric Evaluation of Confirmation Methods for Optical See-Through Head-Mounted Display Calibration. In *Proceedings IEEE/ACM International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 267–268.

Mandell, R. B. and St Helen, R. (1971). Mathematical model of the corneal contour. *British Journal of Physiological Optics*, 26:183–197.

McDonnell, R., Breidt, M., and Bülthoff, H. H. (2012). Render me Real?: Investigating the Effect of Render Style on the Perception of Animated Virtual Humans. *ACM Transactions on Graphics (TOG)*, 31(4):91.

Merchant, J., Morrisette, R., and Porterfield, J. L. (1974). Remote measurement of eye direction allowing subject motion over one cubic foot of space. *IEEE Transactions on Biomedical Engineering (TBME)*, 21(4):309–317.

Milgram, P., Takemura, H., Utsumi, A., and Kishino, F. (1995). Augmented Reality: A class of displays on the reality-virtuality continuum. In *Photonics for Industrial Applications*, pages 282–292. International Society for Optics and Photonics (SPIE).

Milo, R. and Phillips, R. (2015). *Cell Biology by the Numbers*. Garland Science.

Morimoto, C. H., Koons, D., Amir, A., and Flickner, M. (2000). Pupil Detection and Tracking Using Multiple Light Sources. *Image and Vision Computing*, 18(4):331–335.

Morimoto, C. H. and Mimica, M. R. M. (2005). Eye gaze tracking techniques for interactive applications. *Computer Vision and Image Understanding (CVIU)*, 98(1):4–24.

Moser, K. R., Itoh, Y., Oshima, K., Swan II, J. E., Klinker, G., and Sandor, C. (2015). Subjective Evaluation of a Semi-Automatic Optical See-Through Head-Mounted Display Calibration Technique. *IEEE Transactions on Visualization and Computer Graphics (TVCG) (Proceedings IEEE Virtual Reality (VR))*, 21(4):491–500.

Moser, K. R. and Swan II, J. E. (2015). Improved SPAAM Robustness Through Stereo Calibration. In *Proceedings IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 200–201.

Mulvey, F., Villanueva, A., Sliney, D., Lange, R., Cotmore, S., and Donegan, M. (2008). Exploration of safety issues in Eyetracking. *Communication by Gaze Interaction (COGAIN)*, EU IST-2003-511598.

MyNorthwest.com (2011). Seattle bar steps up as first to ban Google glasses. <http://mynorthwest.com/926/2222088/Google-Glasses-Banned>. Last accessed on January 2, 2016.

Nakazawa, A. and Nitschke, C. (2012). Point of Gaze Estimation through Corneal Surface Reflection in an Active Illumination Environment. In *Proceedings European Conference on Computer Vision (ECCV)*, pages 159–172.

Nakazawa, A., Nitschke, C., and Nishida, T. (2015). Non-calibrated and real-time human view estimation using a mobile corneal imaging camera. In *Proceedings International Conference on Multimedia Expo Workshops (ICMEW)*, pages 1–6.

Navab, N., Zokai, S., Genc, Y., and Coelho, E. M. (2004). An On-line Evaluation System for Optical See-through Augmented Reality. In *Proceedings IEEE Virtual Reality (VR)*, pages 245–246.

Navarro, R., González, L., and Hernández-Matamoros, J. L. (2006). On the Prediction of Optical Aberrations by Personalized Eye Models. *Optometry & Vision Science*, 83(6):371–381.

Newcombe, R. A., Izadi, S., Hilliges, O., Molyneaux, D., Kim, D., Davison, A. J., Kohi, P., Shotton, J., Hodges, S., and Fitzgibbon, A. (2011a). KinectFusion: Real-Time Dense Surface Mapping and Tracking. In *IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 127–136.

Newcombe, R. A., Lovegrove, S. J., and Davison, A. J. (2011b). DTAM: Dense Tracking and Mapping in Real-Time. In *IEEE International Conference on Computer Vision (ICCV)*, pages 2320–2327.

Nilsson, S., Gustafsson, T., and Carleberg, P. (2009). Hands Free Interaction with Virtual Information in a Real Environment: Eye gaze as an Interaction Tool in an Augmented Reality System. *PsychNology Journal*, 7(2):175–196.

Nishino, K. and Nayar, S. K. (2004a). Eyes for Relighting. *ACM Transactions on Graphics (Proceedings ACM SIGGRAPH)*, 23(3):704–711.

Nishino, K. and Nayar, S. K. (2004b). The World in an Eye. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Nishino, K. and Nayar, S. K. (2006). Corneal Imaging System: Environment from Eyes. *International Journal of Computer Vision (IJCV)*, 70(1):23–40.

Nitschke, C. (2011). *Image-based Eye Pose and Reflection Analysis for Advanced Interaction Techniques and Scene Understanding*. PhD thesis, Graduate School of Information Science and Technology, Osaka University, Japan.

Nitschke, C. and Nakazawa, A. (2012). Super-Resolution from Corneal Images. In *Proceedings British Machine Vision Conference (BMVC)*, pages 22.1–22.12.

Nitschke, C., Nakazawa, A., and Nishida, T. (2013a). I see what you see: Point of Gaze Estimation from Corneal Images. In *Proceedings IAPR Asian Conference on Pattern Recognition (ACPR)*, pages 298–304.

Nitschke, C., Nakazawa, A., and Nishida, T. (2013b). I see what you see: Point of gaze estimation from corneal images. In *Proceedings 2nd IAPR Asian Conference on Pattern Recognition (ACPR)*, pages 298–304.

Nitschke, C., Nakazawa, A., and Takemura, H. (2009). Display-Camera Calibration from Eye Reflections. In *Proceedings IEEE International Conference on Computer Vision (ICCV)*, pages 1226–1233.

Nitschke, C., Nakazawa, A., and Takemura, H. (2013c). Corneal Imaging Revisited: An Overview of Corneal Reflection Analysis and Applications. *IPSJ Transactions on Computer Vision and Applications (TCVA)*, 5:1–18.

Oculus Rift (2015). Oculus Rift DK2. <https://www.oculus.com/en-us/dk2/>. Last accessed on January 2, 2016.

Oculus Rift (2016). Oculus Introduction to Best Practices. [https://developer.oculus.com/documentation/intro-vr/latest/concepts/bp\\_intro/](https://developer.oculus.com/documentation/intro-vr/latest/concepts/bp_intro/). Last accessed on January 2, 2016.

Oike, H., Kato, T., Wada, T., and Wu, H. (2004). A High-Performance Active Camera System for Taking Clear Images (in japanese). In *Proceedings CVIM-144*, volume 2004, pages 71–78.

OpenGL (2015). OpenGL. <http://www.opengl.org>. Last accessed on January 2, 2016.

OptiTrack (2015). Motive:Body. <http://www.optitrack.com/products/motive/body/indepth.html#subject-calibration>. Last accessed on January 2, 2016.

Orlosky, J., Toyama, T., Kiyokawa, K., and Sonntag, D. (2015). Modular: Eye-controlled Vision Augmentations for Head Mounted Displays. *IEEE Transactions on Visualization and Computer Graphics (TVCG) (Proceedings IEEE International Symposium on Mixed and Augmented Reality (ISMAR))*, 21(11):1259–1268.

Orlosky, J., Wu, Q., Kiyokawa, K., Takemura, H., and Nitschke, C. (2014). Fisheye Vision: Peripheral Spatial Compression for Improved Field of View in Head Mounted Displays. In *Proceedings of ACM Symposium on Spatial User Interaction (SUI)*, pages 54–61.

Ovrvision (2015). Ovrvision PRO. <http://ovrvision.com/buy-en/>. Last accessed on January 2, 2016.

Owen, C. B., Zhou, J., Tang, A., and Xiao, F. (2004). Display-Relative Calibration for Optical See-Through Head-Mounted Displays. In *Proceedings IEEE/ACM International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 70–78.

Pair, J., Allen, B., Dautricourt, M., Treskunov, A., Liewer, M., Graap, K., and Reger, G. (2006). A Virtual Reality Exposure Therapy Application for Iraq War Post Traumatic Stress Disorder. In *Proceedings IEEE Virtual Reality (VR)*, pages 67–72.

Park, H. M., Lee, S. H., and Choi, J. S. (2008). Wearable Augmented Reality System using Gaze Interaction. In *Proceedings IEEE/ACM International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 175–176.

Patow, G. and Pueyo, X. (2003). A Survey of Inverse Rendering Problems. *Computer Graphics Forum*, 22:663–688.

Pires, B., Devyver, M., Tsukada, A., and Kanade, T. (2013a). Unwrapping the Eye for Visible-Spectrum Gaze Tracking on Wearable Devices. In *IEEE Workshop on Applications of Computer Vision (WACV)*, pages 369–376.

Pires, B. R., Hwangbo, M., Devyver, M., and Kanade, T. (2013b). Visible-Spectrum Gaze Tracking for Sports. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1005–1010.

Polans, J., Jaeken, B., McNabb, R. P., Artal, P., and Izatt, J. A. (2015). Wide-field optical model of the human eye with asymmetrically tilted and decentred lens that reproduces measured ocular aberrations. *Optica*, 2(2):124–134.

Purves, D., Augustine, G. J., Fitzpatrick, D., Katz, L. C., LaMantia, A.-S., McNamara, J. O., Williams, S. M., et al. (2001). Types of Eye Movements and Their Functions. In *Neuroscience 2nd edition*. Sinauer Associates.

Rakshit, S. and Monro, D. M. (2007). Pupil Shape Description Using Fourier Series. In *Proceedings IEEE Workshop on Signal Processing Applications for Public Security and Forensics (SAFE)*, pages 1–4.

Remington, L. A. (2011). *Clinical Anatomy of the Visual System*. Elsevier Health Sciences, 3rd edition.

Robertson, C. M., MacIntyre, B., and Walker, B. (2009). An Evaluation of Graphical Context as a Means for Ameliorating the Effects of Registration Error. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 15(2):179–192.

Roesner, F., Kohno, T., Denning, T., Calo, R., and Newell, B. C. (2014). Augmented Reality: Hard Problems of Law and Policy. In *Proceedings ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp): Adjunct Publication*, pages 1283–1288.

Rolland, J. P. and Fuchs, H. (2000). Optical Versus Video See-Through Head-Mounted Displays in Medical Visualization. *Presence: Teleoperators and Virtual Environments*, 9(3):287–309.

Samaria, F. and Young, S. (1994). HMM-based architecture for face identification. *Image and Vision Computing*, 12(8):537 – 543.

Schnieders, D., Fu, X., and Wong, K.-Y. K. (2010). Reconstruction of Display and Eyes from a Single Image. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1442–1449.

Se, S., Lowe, D., and Little, J. (2001). Vision-based Mobile Robot Localization and Mapping using Scale-Invariant Features. In *IEEE International Conference on Robotics and Automation (ICRA)*, volume 2, pages 2051–2058.

SensoMotoric Instruments GmbH (SMI) (2015). SMI Eye Tracking HMD Upgrade Package for the Oculus Rift DK2. [http://www.smivision.com/fileadmin/user\\_upload/downloads/product\\_flyer/prod\\_smi\\_eyetracking\\_hmd\\_oculusDK2\\_screen.pdf](http://www.smivision.com/fileadmin/user_upload/downloads/product_flyer/prod_smi_eyetracking_hmd_oculusDK2_screen.pdf). Last accessed on January 2, 2016.

Sensor Instruments (2015). SMI Eye Tracking Glasses 2 Wireless Product Description. [http://www.eyetracking-glasses.com/fileadmin/user\\_upload/documents/smi\\_etg2w\\_flyer\\_naturalgaze.pdf](http://www.eyetracking-glasses.com/fileadmin/user_upload/documents/smi_etg2w_flyer_naturalgaze.pdf). Last accessed on January 2, 2016.

Shah, M. M., Arshad, H., and Sulaiman, R. (2012). Occlusion in augmented reality. In *Proceedings International Conference on Information Science and Digital Content Technology (ICIDT)*, volume 2, pages 372–378.

Slater, A. and Findlay, J. (1972). The Measurement of Fixation Position in the Newborn Baby. *Journal of Experimental Child Psychology (JECP)*, 14(3):349 – 364.

Snell, R. S. and Lemp, M. A. (1997). *Clinical Anatomy of the Eye*. Blackwell Publishing, Malden, 2nd edition.

Stiefelhagen, R., Yang, J., and Waibel, A. (1997a). A Model-Based Gaze Tracking System. *International Journal on Artificial Intelligence Tools (IJAIT)*, 6(02):193–209.

Stiefelhagen, R., Yang, J., and Waibel, A. (1997b). Tracking Eyes and Monitoring Eye Gaze. In *Proceedings Workshop on Perceptual User Interfaces (PUI)*, pages 98–100.

Stone, R. J. (2001). Haptic Feedback: A Brief History from Telepresence to Virtual Reality. In *Proceedings International Workshop on Haptic Human-Computer Interaction*, pages 1–16.

Sugano, Y., Matsushita, Y., and Sato, Y. (2010). Calibration-free gaze sensing using saliency maps. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2667–2674.

Sugano, Y., Matsushita, Y., and Sato, Y. (2014). Learning-by-Synthesis for Appearance-based 3D Gaze Estimation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1821–1828.

Sutherland, I. E. (1968). A head-mounted three dimensional display. In *Proceedings Fall Joint Computer Conference, Part I (AFIPS)*, pages 757–764.

Swan II, J. E., Singh, G., and Ellis, S. R. (2015). Matching and Reaching Depth Judgments with Real and Augmented Reality Targets. *IEEE Transactions on Visualization and Computer Graphics (TVCG) (Proceedings IEEE International Symposium on Mixed and Augmented Reality (ISMAR))*, 21(11):1289–1298.

Takemura, K., Kimura, S., and Suda, S. (2014a). Estimating point-of-regard using corneal surface image. In *Proceedings ACM Symposium on Eye Tracking Research and Applications (ETRA)*, pages 251–254.

Takemura, K., Yamakawa, T., Takamatsu, J., and Ogasawara, T. (2014b). Estimation of a focused object using a corneal surface image for eye-based interaction. *Journal of Eye Movement Research*, 7(3):4.

Thomas, B. (2012). A Survey of Visual, Mixed and Augmented Reality Gaming. *ACM Transactions on Computers in Entertainment*, 10(3):3:1–3:33.

Tobii Technology AB (2016a). Tobii Pro Glasses 2 Product Description. <http://www.tobiipro.com/siteassets/tobii-pro/product-descriptions/tobii-pro-glasses-2-product-description.pdf>. Last accessed on January 2, 2016.

Tobii Technology AB (2016b). Tobii T/X Series Eye Trackers Product Description. <http://www.tobiipro.com/siteassets/tobii-pro/product-descriptions/tobii-pro-tx-product-description.pdf>. Last accessed on January 2, 2016.

Tokunaga, D. M., Corrêa, C. G., Bernardo, F. M., Jr., J. L. B., Ranzini, E., Nunes, F. L. S., and Tori, R. (2015). Registration System Errors Perception in Augmented Reality Based on RGB-D Cameras. In *Proceedings International Conference on Virtual, Augmented and Mixed Reality (VAMR)*, pages 119–129.

Tomioka, M., Ikeda, S., and Sato, K. (2013). Approximated User-Perspective Rendering in Tablet-Based Augmented Reality. In *IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 21–28.

Toyama, T., Sonntag, D., Dengel, A., Matsuda, T., Iwamura, M., and Kise, K. (2014). A Mixed Reality Head-Mounted Text Translation System Using Eye Gaze Input. In *Proceedings International Conference on Intelligent User Interfaces (IUI)*, pages 329–334.

Tsukada, A. and Kanade, T. (2012). Automatic Acquisition of a 3D Eye Model for a Wearable First-Person Vision Device. In *Proceedings ACM Symposium on Eye Tracking Research and Applications (ETRA)*, pages 213–216.

Tsukada, A., Shino, M., Devyver, M. S., and Kanade, T. (2011). Illumination-Free Gaze Estimation Method for First-Person Vision Wearable Device. In *Proceedings IEEE International Conference on Computer Vision Workshops (ICCVW)*, pages 2084–2091.

Tsukamoto, J., Iwai, D., and Kashima, K. (2015). Radiometric Compensation for Cooperative Distributed Multi-Projection System Through 2-DOF Distributed Control. *IEEE Transactions on Visualization and Computer Graphics (TVCG) (Proceedings IEEE International Symposium on Mixed and Augmented Reality (ISMAR))*, 21(11):1221–1229.

Tuceryan, M. and Navab, N. (2000). Single point active alignment method (SPAAM) for optical see-through HMD calibration for AR. In *Proceedings IEEE/ACM International Symposium on Augmented Reality (ISAR)*, pages 149–158.

Tweed, D. and Vilis, T. (1990). Geometric relations of eye position and velocity vectors during saccades. *Vision Research*, 30(1):111–127.

Villanueva, A. and Cabeza, R. (2007). Models for Gaze Tracking Systems. *EURASIP Journal on Image and Video Processing*, 2007(3):1–16.

Villanueva, A. and Cabeza, R. (2008). A Novel Gaze Estimation System With One Calibration Point. *IEEE Transactions on Systems, Man, and Cybernetics (SMC), Part B: Cybernetics*, 38(4):1123–1138.

Wang, P., Green, M. B., Ji, Q., and Wayman, J. (2005). Automatic Eye Detection and Its Validation. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 164–164.

Wood, E., Baltrusaitis, T., Zhang, X., Sugano, Y., Robinson, P., and Bulling, A. (2015). Rendering of Eyes for Eye-Shape Registration and Gaze Estimation. *arXiv preprint arXiv:1505.05916*, pages 1–9.

Wu, H., Kitagawa, Y., Wada, T., Kato, T., and Chen, Q. (2007). Tracking Iris Contour with a 3D Eye-Model for Gaze Estimation. In *Proceedings Asian Conference on Computer Vision (ACCV)*, pages 688–697.

Wyatt, H. J. (1995). The Form of the Human Pupil. *Vision Research*, 35(14):2021–2036.

Xie, X., Sudhakar, R., and Zhuang, H. (1994). On improving eye feature extraction using deformable templates. *Pattern Recognition*, 27(6):791–799.

Yamazaki, S., Mochimaru, M., and Kanade, T. (2011). Simultaneous Self-Calibration of a Projector and a Camera Using Structured Light. In *Proceedings IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 60–67.

Yamazoe, H., Utsumi, A., Yonezawa, T., and Abe, S. (2008). Remote and Head-Motion-Free Gaze Tracking for Real Environments with Automated Head-Eye Model Calibrations. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1–6.

Ying, X. and Hu, Z. (2004). Catadioptric Camera Calibration Using Geometric Invariants. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 26(10):1260–1271.

Zhang, X., Sugano, Y., Fritz, M., and Bulling, A. (2015). Appearance-Based Gaze Estimation in the Wild. In *Proceedings IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4511–4520.

Zhang, Z. (2000). A Flexible New Technique for Camera Calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 22(11):1330–1334.

Zheng, F., Schmalstieg, D., and Welch, G. (2014a). Pixel-Wise Closed-Loop Registration in Video-Based Augmented Reality. In *Proceedings IEEE/ACM International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 135–143.

Zheng, F., Whitted, T., Lastra, A., Lincoln, P., State, A., Maimone, A., and Fuchs, H. (2014b). Minimizing Latency for Augmented Reality Displays: Frames Considered Harmful. In *IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 195–200. IEEE.

Zhu, Z. and Ji, Q. (2007). Novel Eye Gaze Tracking Techniques Under Natural Head Movement. *IEEE Transactions on Biomedical Engineering (TBME)*, 54(12):2246–2260.