Technische Universität München

Technical University of Munich
Dissertation

Calibration and Use of Optical See-Through Head-Mounted Displays towards **Indistinguishable Augmented Reality**

Yuta Itoh

Fakultät Für Informatik
Lehrstuhl für Informatikanwendungen in der Medizin & Augmented Reality
Fachgebiet Augmented Reality (FAR)

Garching, 2015

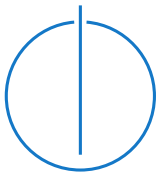
FAKULTÄT FÜR INFORMATIK

LEHRSTUHL FÜR INFORMATIKANWENDUNGEN IN DER MEDIZIN & AUGMENTED REALITY,
FACHGEBIET AUGMENTED REALITY (FAR)

TECHNISCHE UNIVERSITÄT MÜNCHEN

Calibration and Use of Optical See-Through Head-Mounted Displays towards Indistinguishable Augmented Reality

Yuta Itoh



Vollständiger Abdruck der von der Fakultät für Informatik der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften (Dr. rer. nat.)

genehmigten Dissertation.

Vorsitzender:	Univ.-Prof. Dr. Daniel Cremers
Prüfer der Dissertation:	1. Univ.-Prof. Gudrun J. Klinker, Ph.D.
	2. Assoc. Prof. Kiyoshi Kiyokawa, Ph.D., Osaka University, Japan

Die Dissertation wurde am 21.12.2015 bei der Technischen Universität München eingereicht und durch die Fakultät für Informatik am 22.02.2016 angenommen.

To my parents

Acknowledgments

I sincerely thank my supervisor Prof. Gudrun Klinker for guiding my doctor study. Her talent of seeing valuable aspects of things always helped me to think of new ideas and to deepen the ideas in my study.

Besides my advisor, I would like to thank Prof. Kiyoshi Kiyokawa both as my thesis reviewer and my research collaborator. His enthusiasm in research has stimulated me to drive my study.

My gratitude also goes to my colleagues, especially to Dr. Manuel Huber, Christian Waechter, and Frieder Pankratz for helping me in using our Ubitrack framework. I also appreciate them and Dr. Marcus Toennis and Patrick Meier for discussing with me in various topics.

As a friend and colleague, my grateful thanks goes to Andreas Dippon, who supported me getting being settled and living in Germany. His logical thinking in research discussions and his sense of humor made study life all the more enjoyable.

I thank to all my research collaborators particularly Alexander Plopski and Kenneth Moser. It will be my lifetime memory that we had the trilogy session together in IEEE VR 2015.

This dissertation would not have been possible without a funding from the European Union, a Marie Curie Initial Training Network Fellowship of the European Commission FP7 Programme (FP7- PEOPLE-2012-ITN) under contract number: PITN-GA-2012- 316919 (EDUSAFE).

Last but not the least, I would like to thank my parents for their continuous mental and material support.

Abstract

This dissertation explores how to improve the realism of Augmented Reality (AR) in applications using Optical See-Through Head-Mounted Displays (OST-HMDs). In particular, I focus on rendering spatially-aligned AR contents with respect to physical space. This leads us to investigate spatial calibration problems between the displays and the world.

In spatial calibration, we have to estimate the pose of an HMD on the user's head relative to his or her eyes. In current methods, users have to manually calibrate the system through manual interaction. This is a key limitation in practice because users have to recalibrate the system whenever the HMD shifts on their head, which happens frequently, similar to wearing eyeglasses.

Our first contribution is automated calibration method for spatial calibration (Contribution 1). Our method tracks the 3D position of the user's eyeball with respect to the OST-HMD, and estimates calibration parameters automatically without the need of unnecessary human interaction.

Following this method for spatial calibration, I investigate the influence of the calibration error on overall spatial alignment quality (Contribution 2). Through numerical simulation I found that the orientation of the display's image screen has the highest impact on alignment quality.

I further break down the eye-HMD system to improve the alignment quality. I tackle the problem of optical aberration in the display caused by its complex optics (Contribution 3). The aberration is a nuisance since it is non-linear and, more importantly, is dependent on the user's viewpoint. We propose a light field model to naturally handle this 3D aberration. Using this model in our calibration method reduces alignment error by 80 percent.

I then apply this methodology to correct image distortions in OST-HMDs (Contribution 4). The results indicate that the quality can only be improved by simultaneously correcting both the optical and the image aberration.

Finally, I present a vision enhancement concept for OST-HMDs (Contribution 5). I aim to overcome eye aberration by augmenting human vision. The concept is to insert a filter image in the user's field of vision and cancel eye aberration.

In summary, this work aims to take a step towards realizing AR that is indistinguishable from reality. Therefore, I investigate the spatial calibration issues of OST-HMDs, and demonstrate a vision enhancement concept as a potential application for the ultimate OST-HMD of the future.

Preface

For Indistinguishable Augmented Reality

Augmented Reality (AR) technology overwrites our reality by inserting synthesized information between the world and humans. Before the term even appeared in the community in 1992 [CM92], researchers and engineers have worked on increasing the realism of superimposed virtual stimuli. Making AR contents *indistinguishable* to the real world is indeed one of the long-dreamed visions in the community since 60's:

> *"The ultimate display would, of course, be a room within which the computer can control the existence of matter. A chair displayed in such a room would be good enough to sit in. Handcuffs displayed in such a room would be confining, and a bullet displayed in such a room would be fatal. With appropriate programming such a display could literally be the Wonderland into which Alice walked."*
>
> *–Ivan Sutherland, "The Ultimate Display"* [Sut65]

Over the past half century, the mobile display and computing technologies – they did not even exist in the 60's, have developed dramatically. While this rapid growth has brought us steps closer to the realization of indistinguishable AR, there are still numerous obstacles lying in the way to reach this ultimate goal.

For taking an additional step toward this ambitious vision, this dissertation explores how to make *consistent* AR experiences, especially, with the current Optical See-Through Head-Mounted Display (OST-HMD) technology.

Content overview

Part I: We first introduce the basics of AR with focus on displays. We then explain the consistency problems to realize the indistinguishable AR with the display technology.

Part II: We introduce the spatial calibration problem from the basics to our novel automated calibration method.

Part III: We elaborate optical distortion problems in OST-HMDs to improve the calibration quality even further.

Part IV: We explore what would be possible if we achieve the indistinguishable AR experience with OST-HMDs. We demonstrate our proof-of-concept vision enhancement technique.

Part V: We conclude the dissertation with future works and a remark

Contents

Part I

Introduction

This part serves as a holistic introduction to get into the overview of Augmented Reality (AR) and Optical See-Through Head-Mounted Displays (OST-HMDs). As mentioned in the preface, this dissertation explores how to increase the realism of AR experiences given by OST-HMDs. We first introduce the AR concept briefly (Chapter 1). We then describe the HMD technology (Chapter 2). Finally, we analyze the consistency issues in AR with OST-HMDs that affect the realism of AR contents (Chapter 3). Among the issues, we emphasize the spatial consistency, which is one of the biggest hindrances to achieve the indistinguishable AR. This leads us to the calibration of OST-HMDs, which we elaborate in the next part (Part II). We also provide technical preliminaries (Chapter 5) required for the following parts.

1 Augmented Reality

This chapter introduces the concept of AR with focus in vision-based applications.

1.1 What is Augmented Reality?

Augmented Reality technology aims to support human tasks and/or to provide new experiences by augmenting our sensory perceptions with virtual information by computers. For example, an AR system can affect our visual perception by inserting virtual images into our field of view. Fig. 1.1 showcases some of such AR applications from pioneering works published in the '90s. Although common AR applications including the above focus on overwriting our sense of vision, the AR technology may augment other sensory systems [KP10; LN07; Sig+13] including auditory [Här+04; Bed95; Myn+97], haptics [VB99; Noj+02], gustatory [Iwa+04; Has+06], olfactory [Has+06], vestibular [Yen+11], and cross-modal perception, i.e., mixtures of them [Ina+00; Nar+11].

The term AR itself refers to a real-world environment where computers alter our sense of reality in some ways. This is in contrast to Virtual Reality (VR) environments where the stimuli we receive from the environment are mostly synthesized. Milgram et al. suggest that the two environments are conceptually continuous under the view of the Reality-Virtuality Continuum [Mil+95] (Fig. 1.2). The AR technology modifies our real environment towards the virtual environment as a smooth transition. The amount of the transition is based on how much the technology replaces the real stimuli to the synthesized.

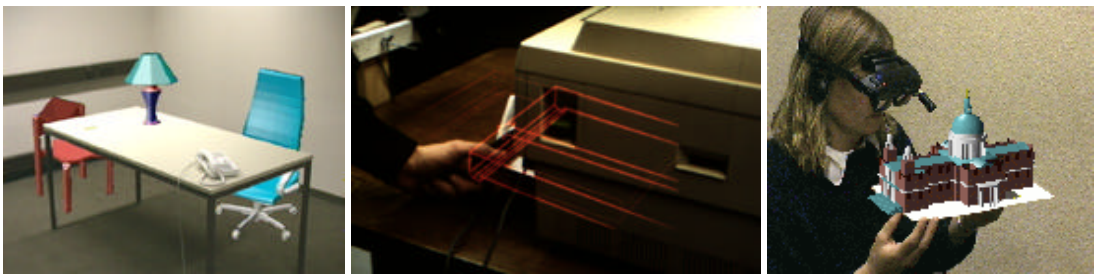


Figure 1.1: Pioneering AR applications from early days. From left to right: a virtual furniture arrangement [Bre+96], a maintenance assistance [FMS93], and a model presentation [KSR99]. They add virtual images into our field of view to facilitate our tasks and/or assist our understanding of the real world.

Figure 1.2: The Reality-Virtuality Continuum [Mil+95]. Augmenting our reality with synthesized information shifts our real environment towards the virtual environment.

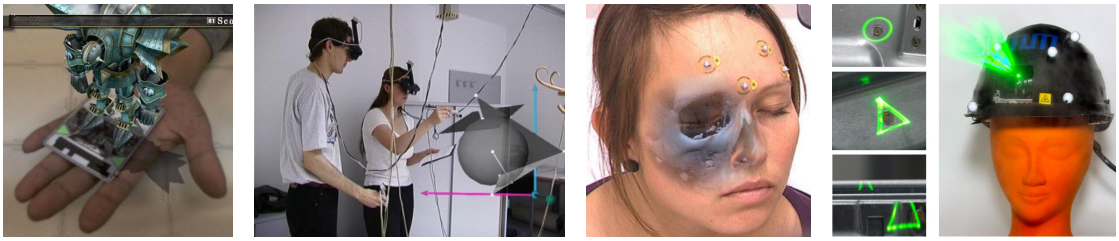


Figure 1.3: AR applications in various domains. From left to right: entertainment[1], education [KS03], medical [Bic+07], and industry [Sch+08].

1.2 Indistinguishable AR

Why we making AR *indistinguishable* to the real world is important? In other words, why we expect AR systems to create virtual stimuli that are consistent and coherent with respect to the real environment (True AR [San+15])? One possible answer from us is: "so that people can rely on the AR information without doubts". The core motivation of AR is to modify/overwrite/convert the real world around us to support any human activities. Since perceived AR information would affect user's judgments, inconsistent information would hinder user's seamless actions, or even might lead to unexpected wrong judgments. In our view, therefore, the indistinguishable AR appears as a consequence after we become able to create consistent AR stimuli that users can believe in to engage in their tasks.

It is worth while mentioning that, in some applications, AR does not necessarily need to be fully coherent to the real world. Haller poses a question in using photorealistic AR visualization (e.g., casting a virtual shadow and reflecting scene illumination. See Fig. 1.5 middle) over non-photorealistic rendering [Hal04]. The author finds that both renderings have their own values in an AR application depending on user's expectations, and suggests that AR information should appear *convincing*.

Therefore, understanding what part of the reality should be kept consistent in what use cases is important. In the next section, we briefly introduce existing AR applications in various domains to give the idea of the consistency requirement.

(a) (b) (c)

Figure 1.4: A maintenance use case in an extreme environment. (a) A 3D model of the ATLAS detector. (b) A front view of the detector. (c) Tile calorimeters and their front-end electronics. (Image a and b: ATLAS Experiment copyright 2014 CERN)

1.3 AR Applications

AR has been used in various domains [Car+11] (Fig. 1.3) including advertisement, entertainment [PT02], education [KS03], medical and especially industry. Navab reports the increasing interest of AR in industrial applications such as design, commissioning, and manufacturing [Nav04].

As an example use case, we describe a maintenance scenario in the EDUSAFE project – a Marie Curie ITN project of the European Union led by the European Organization for Nuclear Research (CERN).

CERN conducts particle physics experiments by using the Large Hadron Collider (LHC). LHC is the world's largest particle accelerator[2] which consists of a 27-kilometer ring of superconducting magnets, and is installed in a tunnel at a depth ranging from 50 to 175 meters underground. LHC accelerates particle beams travel at close to the speed of light, and makes them to collide inside one of seven detectors installed around the ring.

The ATLAS (A Toroidal LHC ApparatuS, Fig. 1.4 a and b) detector is one of those detectors. The detector is gigantic – 46 meters long, 25 meters in diameter, and 7,000 tonnes. The detector consists of several units to measure various properties of particles created by collisions. The tile calorimeters are detector units occupying the most central region of the ATLAS detector (Fig. 1.4 b and c). The calorimeters measure the hadron's energy produced in proton-proton collisions in the LHC. Technicians of the ATLAS experiment have to maintain this complex units.

The maintenance task of tile front-end electronics of a calorimeter may require more than 100 steps. Due to the limited working space around the construction with even scaffolding, it is unrealistic for a technician to carry a thick paper manual or a tablet at hand. Therefore, apprentice technicians have to go through intensive training on the ground before actually going to the underground. In this use case, near-eye displays provide such instructions ideally hands-free, and can even augment the real working area with 3D annotations, which is not possible with paper media.

[1] THE EYE OF JUDGMENT, http://www.jp.playstation.com/software/title/bcjs30007.html
[2] http://home.cern/topics/large-hadron-collider

Figure 1.5: A vision-based AR application from a pioneering work by State et al. presented in 1996 [Sta+96]. The system integrates virtual models in the real world. Its tracking is based on prediction of concentric circular dots and and a magnetic tracker.

1.4 AR for Vision Sensory

The most common type of AR applications are vision-based, which modulate our field of view by adding computer-generated imageries.

The seminal work by State et al. demonstrates a video-based AR which can superimpose virtual contents in the real world in real time [Sta+96] (Fig. 1.5). The system employs a hybrid tracking which fuses head pose data from a vision-based tracker and a magnetic tracker. It also tracks a real light source in the scene to cast a virtual shadow. The shadow and the virtual content can even occlude a real object. Kato and Billinghurst et al. [KB99; Kat+00] distribute ARToolKit, an open-source software for video-based AR, which allows people to make their own AR applications easily.

In principle, these vision-based AR applications require a screen to overlay a virtual image between the user's eye and the real world. According to the Spatial AR concept suggested by Raskar et al., our AR experience changes based on where a display screen is located [RWF98; BR05]. Figure 1.6 [BR06; KP10] illustrates this Spatial AR concept (which is somewhat analogical to the Reality-Virtuality Continuum).

For example, projectors turn a physical object into a *screen* by projecting an image onto it. Projector-based AR systems have the benefit of providing the same AR contents to many users in a physical space simultaneously. This property is suitable for collaborative AR scenarios [Luk+15] such as education [Coo+01].

Raskar et al. develops a Spatial AR system which projects information on packets in a shelf based on the information retrieved from their embedded wireless radio frequency identifier tags [Ras+04]. Another advantage of the projector-based AR is that we can directly overwrite the appearance of the world. Examples include an appearance control of objects for alleviating color blindness [AK10]. On the other hand, the projector-based AR has a limited working space for augmentation unless we move the projectors.

Now we consider other display types that are placed closer to a user: we can place a screen between the user and a real object; a screen can be also held by a user's hand like a smartphone; and even we can show a screen in front of user's view by head-attached displays.

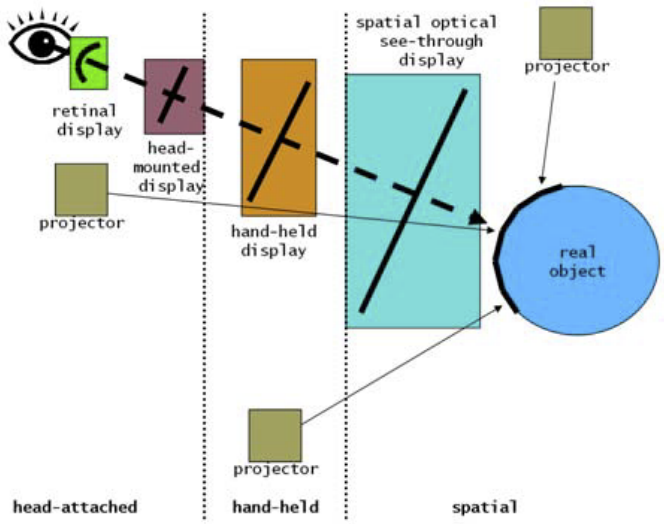


Figure 1.6: AR display categories based on how they are installed in the working space. This figure is taken from work by Bimber and Raskar [BR06].

Head-attached displays are in contrast to other spatial displays in terms of their higher flexibility in working area. They also provide customized AR contents for each individual users. Such contents can be rendered differently based on the user's viewpoint. The displays can also provide 3D contents by using stereo imageries. A typical implementation of such display is Head-Mounted Displays (HMDs).

Opaque HMDs are commonly used to create video-based AR applications, where a user sees the real world through a video shown in front of the user. While those video-based AR applications can offer various AR experiences, such experiences are indirect. We see the real world merely through a video, and AR contents are composed into the video by post-processing. This type of indirect HMDs is often called Video See-Through (VST).

Contrary to this indirect displays, there are direct displays known as Optical See-Through (OST) HMDs. OST-HMDs provide an image in front of the user's view while keeping the real world directly visible. Figure 1.8 demonstrates a simplistic AR rendering with an OST-HMD. On the other hand, a head-attached display normally requires the current viewpoint of a user in the space for AR rendering. We thus need a tracking system to compute the viewpoint. People often use an image-based tracking with a scene camera [KB99; Kat+00].

Note that the taxonomy in Fig. 1.6 defines two types of head-attached displays: retinal displays and the OST-HMDs. The retinal displays directly renders image onto the retina of the eyes by scanning the retina with modulated light [Kol93]. OST-HMDs create screens floating in mid air and are mounted on the users' heads. While the retinal displays are intriguing devices, we focus on the OST-HMDs as it is more commonly used for now.

Head-attached (HMD) Hand-held Spatial (Projector)

Figure 1.7: Example AR applications representing one of the Spatial AR categories in Fig. 1.6. From left to right: an HMD system for AR visualization on laparoscopic surgery [Fuc+98], a handheld system which annotates a bookshelf [RN95], and a projector system which paste a virtual texture on a real object [Ras+01].

Figure 1.8: An example of an OST AR application with our OST-HMD. The system tracks the position of a physical square marker in the scene, and it renders a virtual green frame aligned on the marker from the user-perspective view. We used a see-through head-mounted display for the rendering.

1.5 Summary

AR can change our perception of the real world in various ways. Among them, we are interested in vision-based AR with HMDs, which can superimpose the field of view of individual users. The next chapter (Chapter 2) elaborates HMD technologies.

2 Head-Mounted Displays

As we introduced in the previous section, HMDs present images in front of the user's field of view. Given this feature, AR with HMDs has gained particular interest in professional applications such as medical assistance [Azu97; Nic+11] and assembly/maintenance tasks [Rei+98; Tan+03; HF09; HF11] including our maintenance use case in the previous section. To build a practical system, however, each applications must consider various specifications such as image resolution, field of view of the screen, depth of the field, dynamic range, monocular or stereo, and design factors.

Figure 2.1 illustrates common display designs of both VST- and OST-HMDs. Roland et al. provide thorough comparisons between VST- and OST-HMDs in three key aspects: technological, perceptual, and human-factor issues[RHF94; RF00]. In Table 2.1, we also describe characteristics of the two display types. Figure 2.2 presents various existing HMDs.

In the following sections, section, We elaborate the two types of the HMDs for AR applications: VST-HMDs and OST-HMDs.

2.1 Video See-Through HMD

VST-HMDs are opaque, and gives an indirect view of the real world. A VST-HMD uses a video feed from a camera to its video screen. The screen becomes the background image(stream) to augmented with virtual information. A user does not see the real world directly the display. Users can not see the real world directly while wearing them. These displays are often available as personal multimedia displays, e.g., Vuzix AV920 and SONY HMZ-T3D, or as Virtual Reality (VR) such as Oculus Rift DK2. For AR-oriented applications, some VST-HMDs come with built-in scene cameras that can feed a live stream of the actual scene to the users. Examples are Trivisio SXGA61 3D and Vuzix Wrap1200DX.

The advantage of VST-HMDs is in its ease of augmentation. Since they have direct access to a digital copy (image) of the real scene, which will be shown to the user. We can track the scene by using the image, and we then overwrite a part of the image for AR visualization. This feature is particularly favorable for medial AR applications where misalignment of virtual contents on the real world might lead to fatal misjudgments.

On the other hand, the disadvantage of VST-HMDs is that they may not correspond completely to the user's real field of view, depending on the position and sensing qualities of the camera (Table 2.1). For instance, in the maintenance application we mentioned above, technicians engage their tasks in spatially limited work areas, such as even on a scaffolding. Thus their prime request on an AR system is to maximize the awareness of dangers.

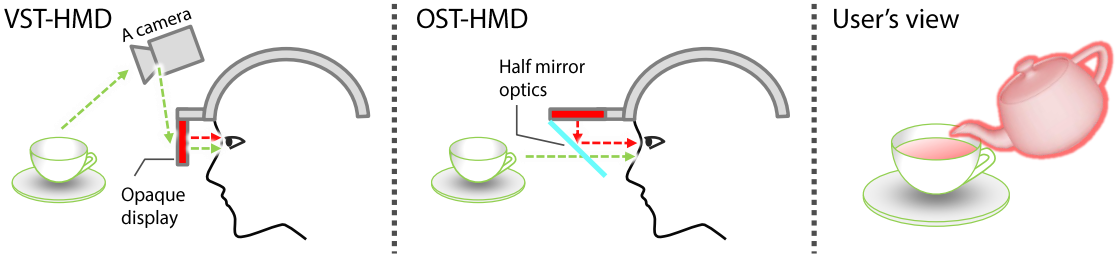


Figure 2.1: Conceptual drawings of the basic design of VST- and OST-HMDs. In this example, a VST-HMD and an OST-HMD augment a virtual teapot in the real world in two different ways. See also Table 2.1 for their different characteristics.

Table 2.1: Comparison of VST-HMDs and OST-HMDs.

	Pros (+)	**Cons (-)**
VST-HMD	- Easy to occlude (superimpose) the world image by AR contents - Typically has a larger field of view than OST-HMDs - Calibration between eye and the display is not required	- **Indirect** view of the physical space through video images • Delay in the world image • Low resolution of the world image • Misaligned user's view • Shifted hand-eye coordination
OST-HMD	- **Direct** view of the physical space • No delay in the world image • Unlimited resolution of the world image • Less view distortion	- Images appear semitransparent - Typically has a narrower field of view thanVST-HMDs - Calibration between eye and the display is required

2.2 Optical See-Through HMD (OST-HMD)

In contrast to VST-HMDs, OST-HMDs keep a user's direct view. A common OST-HMD merges an image into the user's field of view via half mirror optics, which results in a semi-transparent image floating mid air from user's perspective [CR06]. The user thus can see through the physical world while seeing the image. In a typical optics design, a light from a microdisplay of an OST-HMD is reflected to the user's view, thus those images perceived by a user are 2D planar image [CR06].

The direct view of OST-HMDs is probably the greatest advantage of OST-HMDs against VST-HMDs (Table 2.1). For example, the technicians in the EDUSAFE project do not accept VST-HMDs for safety reasons. In driving assistance applications, drivers prefer information appear in the direct field of view [LW04].

However, even though OST-HMDs were part of the settings in the early days [FMS93; Rei+98; CM92; Azu95; Sut65; RHF94], subsequently they have been superseded by video-based AR solutions (using VST-HMDs, smartphones or tablets) in the history. There are many reasons for this: the limited field of view, contrast issues, and especially misaligned view of the real and

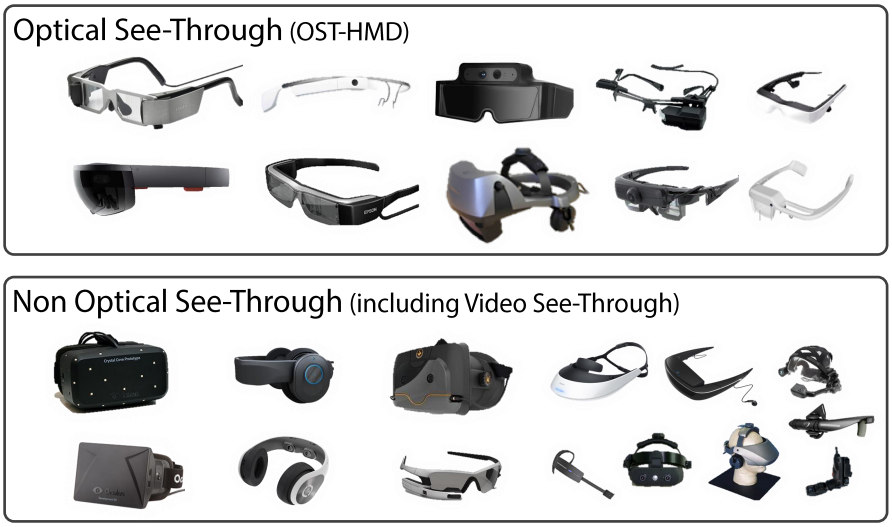


Figure 2.2: Examples of commercially available HMDs in two different categories. (top) Modern OST-HMDs are getting smaller. (bottom) Non-OST HMDs. Some VR HMDs are getting popular for the game industries.

virtual contents.

For correct registration of AR contents in the real world, we need to understand the spatial relationship among this 2D screen, the 3D world, and the eye positions. As we introduce in the next chapter, the misalignment between AR contents and the real world in the user-perspective view decreases the user experience, and may even be dangerous to users. For example, if a user drives a car based on an AR navigation which is misaligned to a real road, the user might cause an accident.

2.3 Trends in Commercial HMDs

Before closing this chapter by a summary section, we briefly introduce commercial HMDs available in the market (Fig. 2.2).

HMD products have been commonly designed as external displays: they had display input like VGA and DVI. Those products were mostly for academic or industrial use, and the form factor was unacceptable for continuously wearing them in daily life. In military and aviation applications, Helmet-mounted displays are also common hardware for AR visualization [Ras99].

Recently, OST-HMDs have penetrated the market as typified by Google Glass, EPSON BT-200, Vuzix M100 Smart Glasses, and Microsoft HoloLens. It is getting more common to have OST-HMDs being integrated as a mobile system typically as an Android OS system. These modern OST-HMDs, tend to have a scene camera and a 9DoF sensor as common sensing system.

Given the recent growth of commercial OST-HMD markets just as commodity 3D printers have boosted the activities of do-it-yourself makers, we believe that the penetration of OST-HMDs will attract increasing number of public developers and users.

However, the increasing interest and inflating expectation among people might backfire. Current commercials of OST-HMDs typically do not convey their still limited capabilities well, showing zero latency, unlimited frame rate, perfect occlusion, exact color replication, and perfectly-aligned virtual objects in the real world. The farther the currently achievable state of the art displays is apart from such videos, the more users get disappointed when they use the displays. This gap might cause people not to use such displays. These excessive expectations in their capabilities suggest issues that we have to solve for realizing ultimate OST-HMDs. Most of such issues concern assuring the consistencies that we mention in the next chapter.

2.4 Summary

Having explained HMDs for AR, we believe that the OST-HMD's direct-view capability is the key requirement for immersive AR experiences. We now shift our discussion more into OST-HMDs. We consider how to make AR applications with the displays more immersive or even indistinguishable. The next chapter (Chapter 3) thus introduces various consistency issues in OST-HMDs.

3 Consistency Issues with HMDs

Realizing immersive AR experiences requires an OST-HMD system to keep the virtual and physical objects consistent in various aspects – temporally [Zhe+14; DRM05; AB95], visually [KKO01; LCH08; CHR04; Bim+03], and spatially [AB94; TGN02] (Fig. 3.1). Lacking any of these consistencies degrades the user experience.

Spatial consistency, in particular, is one of the most fundamental requirements for AR applications with OST-HMDs [Azu97; Hol97]. It is the focus of this dissertation. In the following, we briefly describe each consistency issue.

Note that we include the temporal consistency as a subset of the spatial consistency. Because, in common AR systems where users and the scene are dynamically moving, the users perceive the temporal delay as registration errors.

3.1 Spatial Consistency

Spatial consistency relates to presenting information geometrically aligned with real objects. An OST-HMD worn by a user must render virtual 3D information on the 2D screen in perfect alignment with real objects onto the user's retina. To this end, the path of the 3D-2D projection rays from the 3D world through the HMD into the user's eye needs to be calculated. In other words, spatial consistency requires careful calibration of an eye-HMD system consisting of an eyeball and the virtual screen of an OST-HMD.

Azuma and Bishop investigate the registration errors in OST-HMD systems [AB94]. They propose a manual calibration method where a user manually aligns a virtual boresight to reference 3D points in the scene, and compute display parameters such as the image center and field of view. Tuceryan and Navab propose another manual method that requires a single reference point with simpler manual data collection by a user [TN00]. Their method is one of the most commonly used method and we elaborate the detail in Chap. 6. Hua et al. also proposes a similar manual method designed for projector-based OST-HMDs [HG07; HGA02; HGA07].

Although these manual methods can estimate display parameters, the manual calibration is troublesome for users. Theoretically, each user has to recalibrate an OST-HMD whenever they wear a display or touches the display accidentally. Furthermore, manual calibrations can be erogenous if users are not used to them.

Unlike using human operators, Gilson et al. propose a camera-based method [GFG08]. They place a user-perspective camera behind the display, and calibrate the system by tracking virtual and real points by the camera. We elaborate related works deeper later in Chap. 6. However, the result is not necessarily usable with a real user since the eye position can be very different from

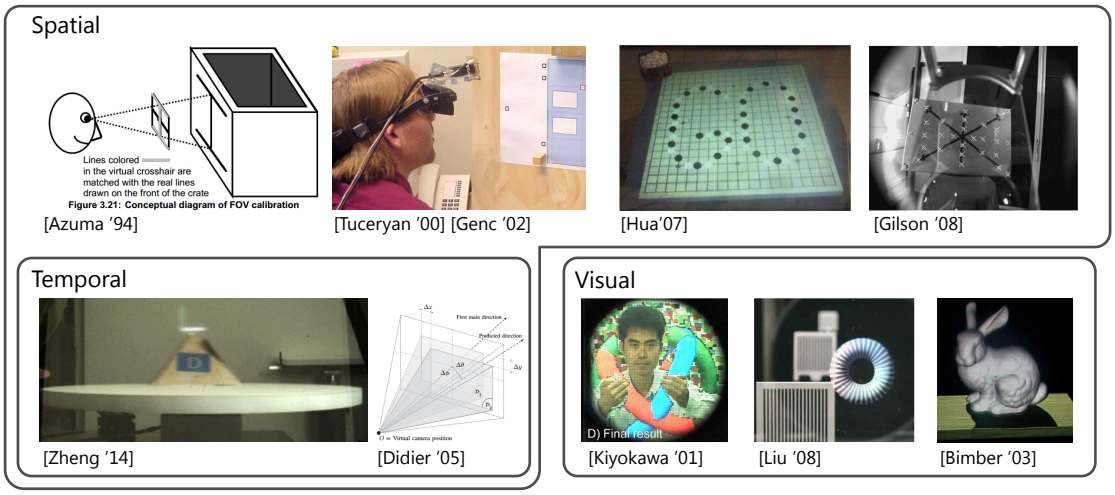


Figure 3.1: Existing works related to various consistency issues in AR applications with OST-HMDs [Azu95; TN00; HGA02; GFG08; Zhe+14; DRM05; KKO01; LCH08; Bim+03]. See related sections in this chapter for more detail.

the position where camera was placed.

This is why we propose an automated, and interaction free calibration method in Chap. 7.

Note that an inaccurate tracking system also degenerates the registration quality [AB94]. However, we do not focus on in this dynamic error issue in the dissertation – we analyze the eye-HMD system under the assumption that the 3D world information is perfectly estimated.

Part II is dedicated to explore the spatial calibration problem.

3.2 Temporal Consistency

There are two main sources of the delays: tracking delay and rendering delay.

3.2.1 Tracking Delay

The tracking delay stems from the fact that the AR world is always following after the physical world. In practical AR scenarios, users freely move in the world just like real world objects. When an AR system gets a 3D measurement of the world to render a 2D point on a display, the measurement is already a past one.

To mitigate this problem, one option is to introduce prediction in the tracking pipeline based on the past pose measurements. Azuma demonstrates predictive tracking on an OST-HMD system [Azu95]. The system predicts a future viewpoint pose based on past pose data from an outside-in tracker and an inertial measurement unit (IMU). While the prediction does reduce the registration errors, this approach does not work on non-stationary (i.e., unpredictable) signals, especially head motions.

One way to mitigate this limitation is a post-rendering technique [MMB97; DRM05; KYO01]. This technique applies 2D image warping on a rendered image in a GPU before the display is scanned out. This *last-minute* correction works as follows. An IMU on an OST-HMD keeps the measuring user's head motion while a GPU is rendering a frame. If the IMU has a higher measurement rate than the display frame rate, we can measure a small orientation change asynchronously to the GPU. Once the time-consuming 3D rendering is done, the GPU fetches the orientation change, and warps in 2D the rendered image to that from the closest viewpoint.

3.2.2 Rendering Delay

The rendering delay comes from the graphics pipeline of a system. Firstly, rendering of a scene on a GPU takes additional time even if the tracking delay is negligibly small. Secondly, the constraints of the modern graphic rendering pipeline imposes another delay. Since display is updated at a certain fixed frame rate, we cannot update the scene immediately with the latest tracking data.

There are several paths in the rendering pipeline, and each of them have to have low latency. Zheng et al. focus on minimizing the latency from when an image was written on a memory to when the data is converted to the photon [Zhe15; Zhe+14]. They demonstrate an AR system with a custom Digital Light Processing (DLP) projector which can update displayed images over 22k Hz with ideal from-memory-to-photon latency 0.4 ms.

3.3 Visual Consistency

The visual consistency relates to the photorealistic appearance of AR contents in various aspects. Even if we could register AR content perfectly in the 3D world in space and time, users would still easily notice that the content is fake if it lacks visual realism. Although non-photorealistic rendering can easily attract user's attention thanks to their distinctive appearances, this can be a problem in applications that focuses on the visual realism [Hal04; RD03]. Arief et al. have developed a real-time system to estimate illumination direction on a mobile system [AMH12].

3.3.1 Environmental Lighting

Environment lighting has a great roll in photorealistic rendering. Environmental light changes the appearance of physical objects depending on their surface properties such as reflectance and specularity. We can simulate those effects for AR contents if we know the lighting condition of the scene.

Bimber et al. demonstrate an AR system that can make the illumination between real and AR world consistent by using a video projector as a known light source[Bim+03]. Instead of controlling the scene illumination, Wang and Samaras propose a method that can estimate directional light sources from a single image [WS03].

3.3.2 Color Reproduction

Even if we consider illumination of the scene in AR renderings, it would still appear unrealistic if the display can not reproduce the exact colors as digitally specified. For example, as an extreme case, if our display is a monochrome display in green, we have no way to render a red apple. Even if we have an RGB display, the analog part of the display makes the problem difficult. When it converts a digital value to an analog light, the spectrum of the light source does not necessarily cover the illumination condition of the scene.

Menk and Koch present a spatial AR system which reproduces a desired input color on a physical, miniature 3D car model by a projector-camera system [MK13]. Amano and Kato propose a dynamic projector-camera system which controls the appearance of scene objects via dynamic camera projector feedback [AK10]. There also several works on color correction for OST-HMDs [Sri+13; Dav+14; Ito+15a].

3.3.3 Focal Distance

If we solved the above issues, then our OST-HMD should be able to emit photons of expected wavelength and intensity. A user however still complains if the image appears with wrong depth-of-focus, i.e., an image has to optically appear at the specified depth so that our eye accommodation agrees with. A common OST-HMD design can not handle this since it produces a flat image in mid air with certain fixed depth.

There are some possible solutions such as multiple display screens [LCH08], light-field displays [MF13], and retinal scanning displays with foveated rendering.

3.3.4 Occlusion

In reality, an object closer in depth occludes another object in the user's view. Since OST-HMDs render virtual images via half-mirror optics, light from both the world and the image source are always mixed. This ruins the realism of AR contents as it appears as if a semi-transparent object. There are software and hardware solutions.

The software solution is to subtract the background color from the rendered image. This approach would be complex when especially we render light-field instead of 2D images.

The hardware solution is to add an OST-HMD an occlusion layer, which can dynamically change the opacity of a part of the image screen [KKO00; KKO01; CHR04; MF13]. The obvious problem of this approach is that it requires extra hardware, and it has to handle blurry occlusion masks when the layer is placed in front of the display.

3.4 Social Consistency

Although this is not a technical challenge, it is important to mention that social acceptance is necessary for HMDs to be used in daily life. These displays are not yet used as common as smartphones or tablets in society. We would easily spot people who wear such displays at a public

place. Perhaps, this is just a matter of time like we, nowadays, are not perplexed to see people taking photos with smart devices with their hands outstretched to sky.

3.5 Summary

In the above sections, we briefly explained various consistency issues that need to be solved to make AR experience as immersive as possible. Among them, maintaining spatial consistency is of great importance because misaligned augmentation immediately looses user's AR experience. Therefore, our work primarily focuses on improving the current spatial calibration techniques for OST-HMDs. In the next part (Part II), we elaborate our study on this spatial calibration problem, which is the main contents of this dissertation.

4 Contributions of the Dissertation

Here, we summarize the contributions of this work.

Contribution 1: Automated calibration method for OST-HMDs (Chap. 7) We propose INteraction-*free* DIsplay CAlibation (~~IN~~DICA), an automated calibration method for the spatial calibration of OST-HMDs. Unlike the existing manual methods, our method tracks the 3D position of the user's eyeball with respect to the OST-HMD, and estimates calibration parameters automatically without the need of complex human interaction. The results show that our method gives more consistent calibration results than manual methods that rely on unstable user inputs.

Contribution 2: Error sensitivity analysis of the automated calibration method (Chap. 8) We investigate the influence of the calibration error on the overall spatial alignment quality. Through numerical simulation we found out that the orientation of the display's image screen has the highest impact on the alignment quality.

Contribution 3: Correction of viewpoint-dependent optical distortions (Chap. 9) We further break down the eye-HMD system to tackle the optical aberration of the display caused by its complex optics. The aberration is a nuisance since it is non-linear and, more importantly, is dependent on the user's viewpoint. We propose a light field model to naturally handle this 3D aberration. Using this model in our calibration method reduces the alignment error by 80 percent.

Contribution 4: Correction of image distortion (Chap. 10) The optics of OST-HMDs refract light from both the world and image light source. The above method (Contrib. 3) corrects only the optical distortion of the world view. We extend our method to also correct image distortions in OST-HMDs. The results indicate that correcting both world and image distortions simultaneously can only improve the calibration accuracy.

Contribution 5: Vision enhancement for defocus correction by OST-HMDs (Chap. 11) Finally, we present a vision enhancement concept for OST-HMDs. We aim to overcome eye aberration by augmenting human vision with OST-HMDs. The concept is to insert a filter image in the user's field of vision, and to cancel eye aberration. Our system demonstrates that we can increase the image quality more than 5db in peak signal-to-noise ratio compared to degraded images.

Before closing this part, the next chapter provides technical preliminaries required in the rest of this dissertation.

5 Technical Preliminaries

This section introduces mathematical tools used in this dissertation.

5.1 Mathematical Notations

Plain lower-case letters denote scalar values. Bold lower-case letters denote vectors such as a translation vector,

$$\mathbf{t} \in \mathbb{R}^3. \tag{5.1}$$

Upper-case typewriter letters denote matrices such as a rotation matrix,

$$\mathtt{R} \in \mathbb{R}^{3\times 3}. \tag{5.2}$$

A rotation matrix $\mathtt{R}$ satisfies

$$\mathtt{R}^{-1}\mathtt{R} = \mathtt{R}^{\mathrm{T}}\mathtt{R} = \mathtt{I},\ \det(\mathtt{R}) = 1, \tag{5.3}$$

where $(\cdot)^{-1}$ and $(\cdot)^{\mathrm{T}}$are the transpose/inverse of vectors and matrices, $\mathtt{I}$ is an identity matrix, and $\det(\cdot)$ is a determinant of a matrix. If a matrix is explicitly written with its elements, zero elements are left blank for clarity. Lower-case letters represent scalars. $\|\cdot\|$denotes the norm of a vector, e.g. $\|\mathbf{x}\| := \sqrt{\mathbf{x}^{\mathrm{T}}\mathbf{x}}$.

To avoid unnecessary complexity in notations, we sometimes ignore the above format to describe other concepts, but with explanations.

5.2 Coordinate System Convention

Upper-case letters denote coordinate systems such as the world coordinate system W. Given a coordinate system A, a 3D point in A is denoted by using vectors with the coordinate symbol as the lower index: $\mathbf{x}_A$.

Given coordinate systems A and B, the relative transformation from A to B is described by $(\mathtt{R}_{AB}, \mathbf{t}_{AB})$ where $\mathtt{R}_{AB}$ and $\mathbf{t}_{AB}$ stand for rotation and translation respectively, i.e., explicit transformation of a 3D point $\mathbf{x}_A$ in A to $\mathbf{x}_B$ in B can be written as (Fig. 5.1):

$$\mathbf{x}_B = \mathtt{R}_{AB}\mathbf{x}_A + \mathbf{t}_{AB}. \tag{5.4}$$

A homogeneous vector (Fig. 5.2) is created by adding $\tilde{\cdot}$ to a vector, e.g., a 3D point $\mathbf{x} \in \mathbb{R}^3$

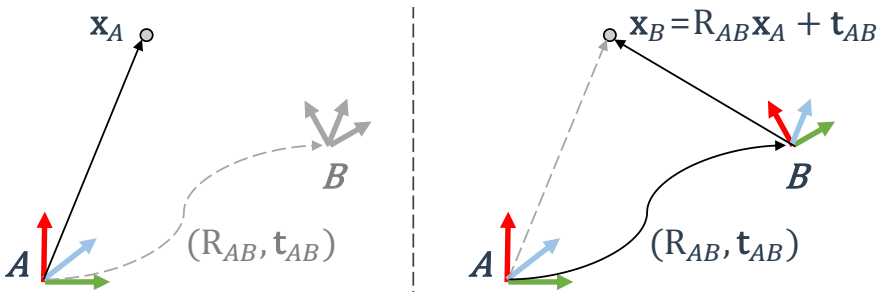


Figure 5.1: Our convention of the transformation between two coordinate systems.

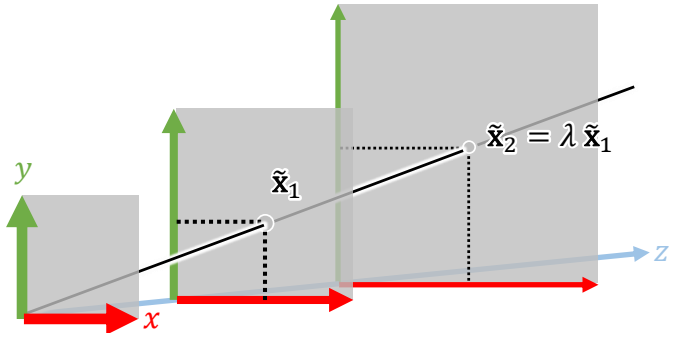


Figure 5.2: Homogeneous coordinate system.

becomes

$$\widetilde{\mathbf{x}} := \begin{bmatrix} \mathbf{x} \\ 1 \end{bmatrix}. \tag{5.5}$$

When two homogeneous vectors $\widetilde{\mathbf{x}_1}$ and $\widetilde{\mathbf{x}_2}$ are equal up to scale, we use $\sim$ to represent the homogeneous equality

$$\widetilde{\mathbf{x}_1} \sim \widetilde{\mathbf{x}_2} \Longleftrightarrow \exists \lambda \neq 0, \text{s.t.}\, \widetilde{\mathbf{x}_2} = \lambda \widetilde{\mathbf{x}_1}. \tag{5.6}$$

5.3 Pinhole Camera Model

In the pinhole camera model (Fig. 5.3), a 3D point $\mathbf{x} \in \mathbb{R}^3$ in the world is projected to a 2D image point $\mathbf{u}$ in the image plane of a camera as follows,

$$\widetilde{\mathbf{u}} \sim \mathtt{K}\mathbf{x}, \tag{5.7}$$

where $\mathtt{K}$:

$$\mathtt{K} := \begin{bmatrix} f_x & & c_x \\ & f_y & c_y \\ & & 1 \end{bmatrix} \in \mathbb{R}^{3\times 3} \tag{5.8}$$

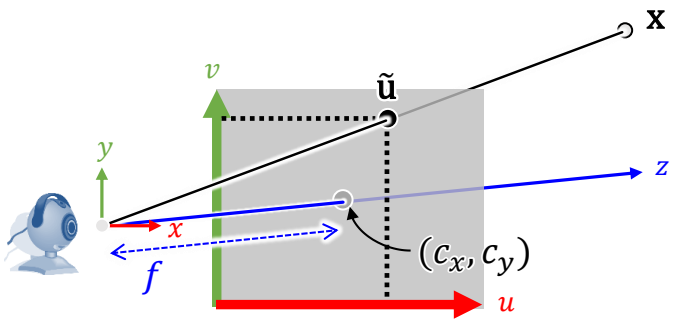


Figure 5.3: Pinhole camera model with a single focal length ($f := f_x = f_y$).

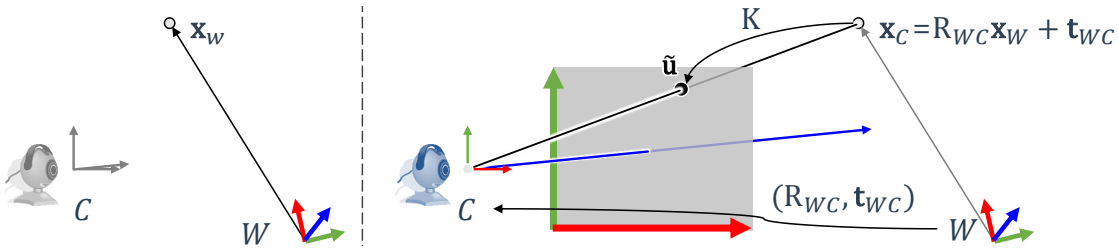


Figure 5.4: Extrinsic and intrinsic parameters under the pinhole camera model.

is an intrinsic matrix which maps the homogeneous 3D point to the homogeneous image point. $\{f_x, f_y\}$ are the focal lengths of the camera and (c_x, c_y) are the optical center of the camera. In general, the optical center and the center of the image plane are designed to coincide. If not, the model is called off-axis.

In general, a 3D point is in the camera coordinate system W (Fig. 5.4). We thus need to transform the point into the camera coordinate system C. Let $(\mathrm{R}_{WC}, \mathbf{t}_{WC})$ be a 6DoF transformation from the world to the camera coordinate system. Then the image point is computed as

$$\tilde{\mathbf{u}} \sim \mathrm{K} \begin{bmatrix} \mathrm{R}_{WC} & \mathbf{t}_{WC} \end{bmatrix} \tilde{\mathbf{x}}. \tag{5.9}$$

We call $\mathrm{P} := \mathrm{K} \begin{bmatrix} \mathrm{R}_{WC} & \mathbf{t}_{WC} \end{bmatrix} \in \mathbb{R}^{3\times 4}$ a perspective projection matrix.

Camera calibration refers to the process of estimating each component of the projection matrix P. The part $\begin{bmatrix} \mathrm{R}_{WC} & \mathbf{t}_{WC} \end{bmatrix}$ describes the extrinsic parameters, K describes the intrinsic parameters.

It is practically important to be aware of the fact that we have an option to choose the right-/left-handed coordinate systems (Fig. 5.6) and how to define the relationship between the chosen coordinate system and the image plane (Fig. 5.6).

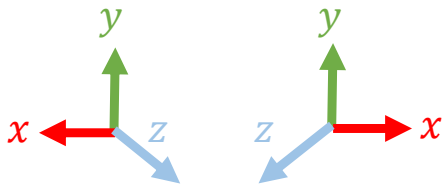


Figure 5.5: Left/right-handed coordinate systems.

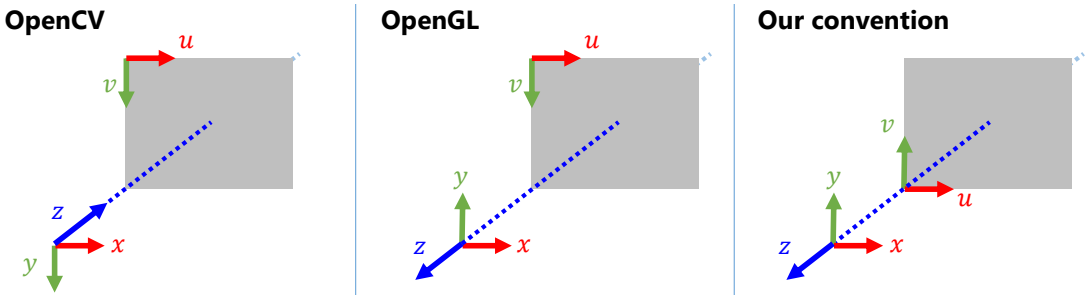


Figure 5.6: Coordinate system convention of different software frameworks. From left to right: OpenCV, OpenGL†, and Ubitrack. († Under a typical OpenGL programming convention before converting to Normalized Device Coordinates, which is *left-handed*.)

5.4 Parameter Estimation

Identifying an unknown system from observed inputs and outputs is a common problem in engineering. Such a problem is often abstracted as an estimation of a function f:

$$y = f(x) \tag{5.10}$$

given pairs of observation $\{(x_k y_k)\}_k^N$ that their output contains errors $\{e_k\}$, i.e.,

$$y_k = f(x_k) + e_k. \tag{5.11}$$

Under the assumption that the error is i.i.d. (independent and identically distributed) from zero-mean Gaussian with known variances, the maximum-likelihood estimate of the system is given by a least-squares estimate:

$$\hat{f} = \underset{f}{\operatorname{argmin}} \sum_{k=1}^{N} \|y_k - f(x_k)\|^2. \tag{5.12}$$

The actual solution of the above formula depends on the assumptions on the system f.

5.4.1 Linear Regression

Given that the system is linear, i.e., we can model the system as:

$$y = \mathbf{a}^{\mathrm{T}}\mathbf{x} \tag{5.13}$$

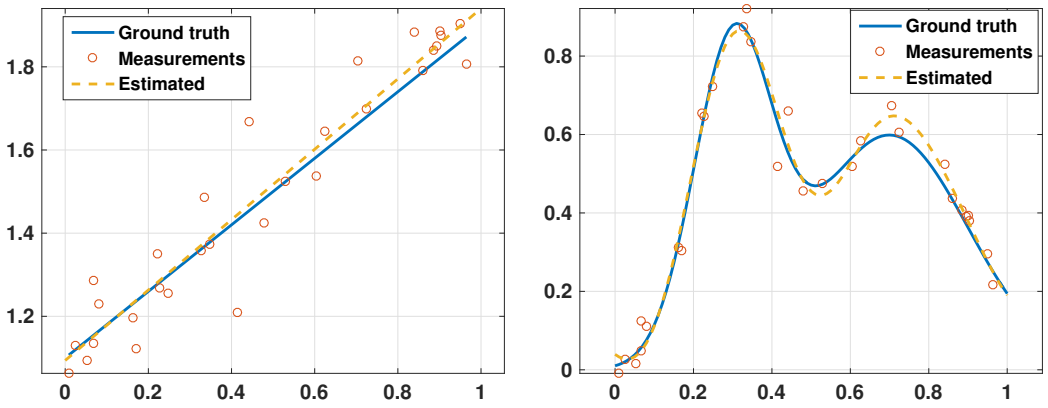


Figure 5.7: Toy examples of regression problems. (left) Linear regression applied to 1st-order function. (right) Kernel regression applied to a non-linear function.

where the output $y \in \mathbb{R}$ is scalar and the input $\mathbf{x} \in \mathbb{R}^d$ is a d-dimensional vector. Thus our least-squares estimate becomes

$$\hat{\mathbf{a}} = \underset{\mathbf{a}}{\operatorname{argmin}} \sum_{k=1}^{N} \left\| y_k - \mathbf{a}^{\mathrm{T}} \mathbf{x}_k \right\|^2. \tag{5.14}$$

$$= \underset{\mathbf{a}}{\operatorname{argmin}} \left\| \mathbf{y} - \mathtt{X}^{\mathrm{T}} \mathbf{a} \right\|^2 \tag{5.15}$$

where $\mathbf{y} := [y_1, \ldots, y_N]^{\mathrm{T}}$ and $\mathtt{X} := [\mathbf{x}_1, \ldots, \mathbf{x}_N]$ are collections of input and output data. Since the above function is convex in $\mathbf{a}$, by taking the derivative of the above in $\mathbf{a}$, and setting it to zero leads to

$$\mathtt{X}\mathbf{y} = \mathtt{X}\mathtt{X}^{\mathrm{T}} \hat{\mathbf{a}} \tag{5.16}$$

$$\hat{\mathbf{a}} = (\mathtt{X}\mathtt{X}^{\mathrm{T}})^{-1} \mathtt{X}\mathbf{y}. \tag{5.17}$$

Figure 5.7 left shows an example of applying the linear regression to a noisy dataset.

5.4.2 Non-parametric Regression

If the system f is nonlinear, we can apply kernel regression. The kernel regression yields a regression function $f : \mathbb{R}^N \to \mathbb{R}$ given a set of input and output pairs $\{(\mathbf{x}_k, y_k)\}_k$ [SS01]. We use the Gaussian kernel model, which approximates the true system $f(\mathbf{x})$ by the linear sum of Gaussian radial basis functions ϕ as:

$$f(\mathbf{x}) \approx \hat{f}(\mathbf{x} \mid \alpha) := \sum_{k=1}^{n_b} \alpha_k \phi(\mathbf{x}, \mathbf{x}_k), \tag{5.18}$$

$$\phi(\mathbf{x}, \mathbf{x}_k) := \exp\left(\frac{-(\mathbf{x} - \mathbf{x}_k)^{\mathrm{T}} (\mathbf{x} - \mathbf{x}_k)}{2\sigma^2} \right), \tag{5.19}$$

where σ is the kernel width, n_b is the number of basis functions, and $\alpha = [\alpha_1, \ldots, \alpha_{n_b}]^{\mathrm{T}}$ is the coefficient vector. The regularized least-squares estimator of the model is given by

$$\hat{f}(\mathbf{x} \mid \hat{\alpha}), \ \hat{\alpha} := (\Phi + \lambda \mathtt{I}_{n_b})^{-1} \mathbf{y}, \tag{5.20}$$

where Φ is the kernel matrix defined as $[\Phi]_{ij} = \phi(\mathbf{x}_i, \mathbf{x}_j)$, λ is the model regularization parameter, $\mathtt{I}_{n_b}$ is an n_b-by-n_b identity matrix, and $[\mathbf{y}]_k = y_k$.

Figure 5.7 right shows an example of applying the kernel regression to a noisy dataset.

Part II

Spatial Calibration of OST-HMDs

This part elaborate the spatial calibration problem and introduces solutions including our method.

We first introduce the problem and a conventional manual calibration method (Chapter 6). We then describe our automated calibration, which is the main contribution of this work (Chapter 7).

6 Manual Calibration

6.1 Introduction

A crucial issue in AR applications using OST-HMDs is to display 3D information from the current viewpoint of the user – and, more particularly, according to the user's eye position, relative to a not quite stable HMD pose on the user's head. Figure 6.1 top left shows a schematic diagram of a user wearing an OST-HMD system with a tracking camera.

An OST-HMD calibration aims to estimate the virtual camera system(s) representing the viewpoint(s) of the user's eye(s) behind the glass(s) of an HMD. The model consists of a screen of an OST-HMD and a user's eye by estimating a projection between them. Since the eye is a part of the system, the projection contains geometric information of the eye relative to the screen. To obtain such projections, many of the existing methods require user interactions. A very good, detailed discussion is given in Zhou's dissertation [Zho07; Owe+04]. We here provide a brief overview of the most relevant approaches.

First OST-HMD calibration algorithms required users to submit themselves to very complex, pre-arranged physical processes and settings. They were, for instance, requested to physically align their own view with a given world-based reference frame using a boresight approach [Azu95] or to position their heads on a head rest [CM92; KSR99].

More conveniently, Tuceryan et al. allowed (or even requested) users to move their heads freely within the physical environment in their Single Point Active Alignment Method (SPAAM) [TN00]. In their approach, users have to repeatedly align given a physical point of interest (at a known world position) with a cross hair drawn at random locations on their HMD screen. Each such 2D-3D correspondence describes a projection equation, with the head motion being compensated by world-based head tracking. In principle, 6 correspondences suffice to fully determine the 11 parameters of the projection matrix. Yet, to enhance robustness against user errors (imprecise alignments), Tuceryan et al. recommend using at least 12 correspondences. This has become a widely used method for display calibration in AR applications. Genc et al. [Gen+00] extended SPAAM for stereo OST-HMDs. To calibrate the left and right eye screens simultaneously, users need to align virtual 3D points to real 3D points by relying on their stereo perceptions.

6.2 Related Work

Yet, even though SPAAM is more convenient than the early methods, it suffers severely from the required large number of user interactions. They not only add physical burden on users, they also

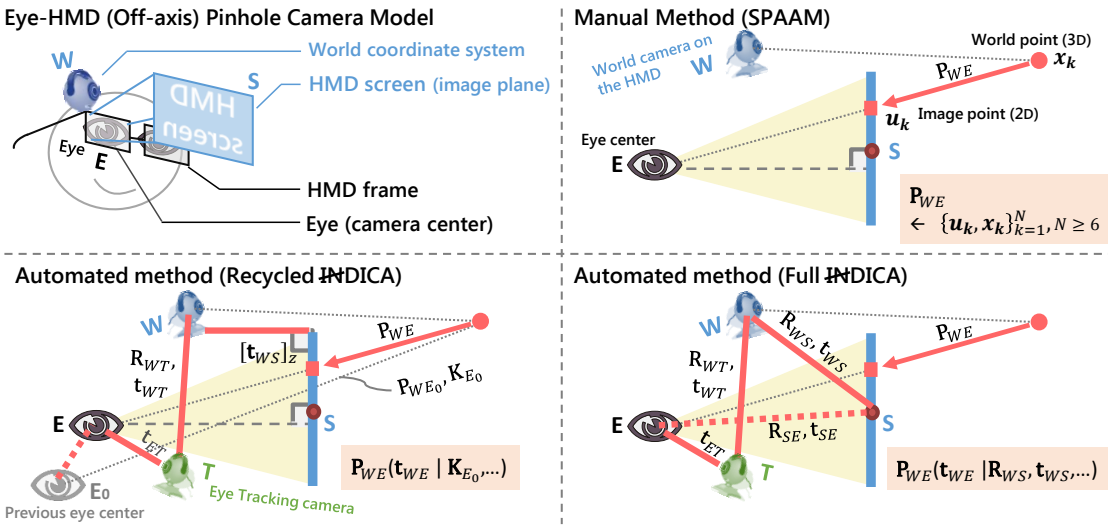


Figure 6.1: Schematic overview of OST-HMD calibration methods. (Top left) The image screen is fixed to the HMD coordinate system defined on the world camera. (Top right) Manual methods, e.g., SPAAM, construct an 11-DoF projection matrix P_{WE} from at least six 2D-3D correspondences collected by a user. (Bottom left and right) Automated methods (Recycled INDICA and Full INDICA) [IK14a; IK14b] reconstruct the projection as a function of the current eyeball position $\mathbf{t}_{WE}$ and various parameters of the current eye-HMD system. Note that none of the methods considers the distortion caused by optical elements of OST-HMDs. See Sec. 8.3.1 for a detailed definition of the parameters in the figure.

impact the calibration quality. Axholt et al. [Axh+11; Axh+10] analyzed estimation variance of SPAAM methods and reported that there is a significant effect on parameter estimation variance depending on how the 3D points are distributed in space – primarily in depth. In addition, the authors found that when users are occupied by the alignment task they tend to forget to change their postures and thus collect points within a limited depth range. The paper concluded that users need to be carefully instructed when employing SPAAM. A further source of error stems from the confirmation process for users to indicate when they have achieved a good alignment. Originally, users had to click a button. Maier et al. recently showed that more robust alignments can be achieved, if users are asked to merely hold still for a short amount of time – thereby signaling that they are done [Mai+12].

Some research effort has gone into reducing user interaction during the calibration procedure. Easy SPAAM [GTN02; Nav+04] reuses an old projection matrix from a previous SPAAM calibration and adjusts it only for a new user eye position. In this method, the change of eye position is approximated by 2D warping of the screen image including scaling – requiring fewer parameters than the full eye pose estimation. Thus, users need to establish only (at least) 2 2D-3D correspondences for online calibration. After Easy SPAAM, Owen et al. [Owe+04] proposed DRC (Display-Relative Calibration). The projection can be decomposed into display parameters and an eye position. The former depend only on an OST-HMD and can be determined

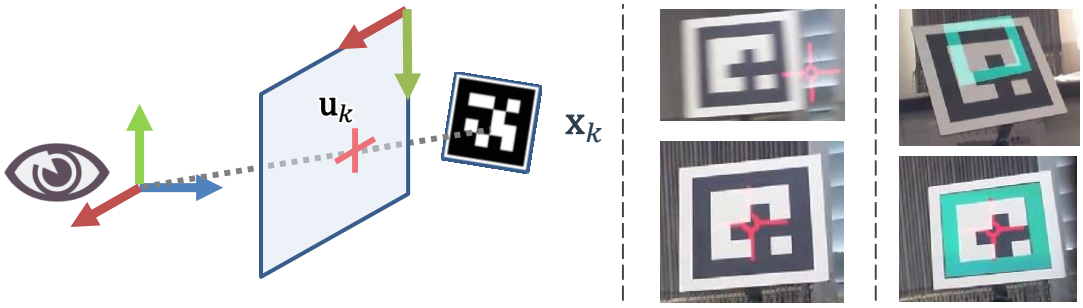


Figure 6.2: Manual data collection in SPAAM. (left) a schematic diagram of the relationship between a 3D point in the world and a 2D image point on the HMD screen, (middle) a user-perspective view matching a virtual crosshair to the center of a physical square marker board, and (right) a green virtual square frame overlaid on the marker before/after SPAAM calibration.

offline while the latter needs to be determined online. In DRC, the authors formalized this as a two-step calibration process. They described an offline calibration for the display parameters using mechanical jigs and proposed 5 different options for the second, online step, involving varying degrees of simplifying assumptions (ranging from not performing any online calibration over performing a simple warping such as Easy SPAAM to a full 6 DoF eye pose estimation). Hua et al. [HG07] developed an approach similar to Easy SPAAM for projection-based HMPDs using also this two-step calibration concept. Their difference is that they recalibrate the full set of display parameters in the online step.

6.3 Single Point Active Alignment Method (SPAAM)

We elaborate on SPAAM [TN00], the basic manual method for OST-HMD calibration. As shown in Fig. 6.1 top left, we can model our eye and the image screen of an OST-HMD as an off-axis pinhole camera (Fig. 5.3 in Sec. 5.3):

$$\widetilde{\mathbf{u}} \sim \underbrace{\mathtt{K}_E \begin{bmatrix} \mathtt{R}_{WE} & \mathbf{t}_{WE} \end{bmatrix}}_{=:\mathtt{P}_{WE} \in \mathbb{R}^{3\times 4}} \begin{bmatrix} \mathbf{x} \\ 1 \end{bmatrix}, \tag{6.1}$$

where $\mathtt{K}_E$ is an intrinsic matrix of the eye-HMD system and $(\mathtt{R}_{WE}, \mathbf{t}_{WE})$ is transformation from the world coordinate system W to the eye coordinate system E.

In the data collection phase, a user aligns a displayed 2D point to a 3D reference point in the world (Fig. 6.2), repeating this process N times with different head positions, we get a set of 2D-3D correspondences as $\{(\mathbf{u}_k, \mathbf{x}_k)\}_{k=1}^{N}$. Our goal is to estimate the 3x4projection matrix $\mathtt{P}_{WE}$

from this dataset. The matrix can be rewritten in row 4D vectors and elements as

$$\mathtt{P}_{WE} := \begin{bmatrix} \mathbf{p}_1^{\mathrm{T}} \\ \mathbf{p}_2^{\mathrm{T}} \\ \mathbf{p}_3^{\mathrm{T}} \end{bmatrix} = \begin{bmatrix} p_1 & p_2 & p_3 & p_4 \\ p_5 & p_6 & p_7 & p_8 \\ p_9 & p_{10} & p_{11} & 1 \end{bmatrix}. \tag{6.2}$$

Note that the last element of the matrix is set to 1 without loss of generality. From (6.1), a pair of the dataset, $(\mathbf{u}_{k,}\mathbf{x}_k)$, meets the following equation:

$$\begin{cases} u_k = \mathbf{p}_1^{\mathrm{T}}\widetilde{\mathbf{x}}_k / \mathbf{p}_3^{\mathrm{T}}\widetilde{\mathbf{x}}_k \\ v_k = \mathbf{p}_2^{\mathrm{T}}\widetilde{\mathbf{x}}_k / \mathbf{p}_3^{\mathrm{T}}\widetilde{\mathbf{x}}_k \end{cases}, \tag{6.3}$$

where $\mathbf{u}_{k,} = [u_k\, v_k]^{\mathrm{T}}$. This leads to

$$\begin{cases} u_k = \mathbf{p}_1^{\mathrm{T}}\widetilde{\mathbf{x}}_k - u_k\mathbf{p}_{3'}^{\mathrm{T}}\widetilde{\mathbf{x}}_k \\ v_k = \mathbf{p}_2^{\mathrm{T}}\widetilde{\mathbf{x}}_k - v_k\mathbf{p}_{3'}^{\mathrm{T}}\widetilde{\mathbf{x}}_k \end{cases}, \tag{6.4}$$

$$\Longleftrightarrow \mathbf{u}_k = \underbrace{\begin{bmatrix} \widetilde{\mathbf{x}}_k^{\mathrm{T}} & & -u_k\widetilde{\mathbf{x}}_k^{\mathrm{T}} \\ & \widetilde{\mathbf{x}}_k^{\mathrm{T}} & -v_k\widetilde{\mathbf{x}}_k^{\mathrm{T}} \end{bmatrix}}_{=:\mathtt{A}_k} \underbrace{\begin{bmatrix} \mathbf{p}_1 \\ \mathbf{p}_2 \\ \mathbf{p}_3' \end{bmatrix}}_{=:\mathbf{p}}, \tag{6.5}$$

where $\mathbf{p}_{3'} := [p_9\, p_{10}\, p_{11}]^{\mathrm{T}} \in \mathbb{R}^3$. By concatenating all measurements, we get a linear equation system:

$$\underbrace{\begin{bmatrix} \mathbf{u}_1 \\ \vdots \\ \mathbf{u}_N \end{bmatrix}}_{=:\mathbf{u}} = \underbrace{\begin{bmatrix} \mathtt{A}_1 \\ \vdots \\ \mathtt{A}_N \end{bmatrix}}_{=:\mathtt{A}} \mathbf{p} \Longleftrightarrow \mathbf{u} = \mathtt{A}\mathbf{p}, \tag{6.6}$$

where $\mathbf{u} \in \mathbb{R}^{2N}$ and $\mathtt{A} \in \mathbb{R}^{2N\times 11}$. To solve (6.6) uniquely, we need $N \geqq 6$ pairs of measurements. Then we can get

$$\begin{aligned} \mathbf{u} &= \mathtt{A}\mathbf{p} \\ \mathtt{A}^{\mathrm{T}}\mathbf{u} &= \mathtt{A}^{\mathrm{T}}\mathtt{A}\mathbf{p} \\ \mathbf{p} &= (\mathtt{A}^{\mathrm{T}}\mathtt{A})^{-1}\mathtt{A}^{\mathrm{T}}\mathbf{u}. \end{aligned} \tag{6.7}$$

Note that this solution is equivalent to the least-square estimate of the algebraic cost function:

$$\|\mathbf{u} - \mathtt{A}\mathbf{p}\|^2, \tag{6.8}$$

The estimate can thus be erroneous when the data contains outliers, which is likely to occur especially when a novice user conducts SPAAM.

6.3.1 Geometric optimization

The projection matrix $\mathbf{p}$ is so far estimated by minimizing the algebraic cost (6.8). We can refine the solution by minimizing the geometric cost function,

$$\sum_{k=1}^{N} \left\{ (u_k - \mathbf{p}_1^{\mathrm{T}} \widetilde{\mathbf{x}}_k / \mathbf{p}_3^{\mathrm{T}} \widetilde{\mathbf{x}}_k)^2 + (v_k - \mathbf{p}_2^{\mathrm{T}} \widetilde{\mathbf{x}}_k / \mathbf{p}_3^{\mathrm{T}} \widetilde{\mathbf{x}}_k)^2 \right\} \ , \tag{6.9}$$

over $\mathbf{p}$ via nonlinear optimization. The initial estimate of $\mathbf{p}$ is given by the previous linear solution.

7 Automated Calibration

This section is based on the work that the author published in IEEE 3DUI 2014 conference [IK14a].

7.1 Introduction

Although the manual calibrations works, they are cumbersome to use and disrupt the user's AR-experience. In practice, the calibration process is frequently skipped (e.g. when quickly showing an AR-demonstration to a visitor), or performed sub-optimally in order to enhance user convenience.

The issue becomes even more critical during extended use (i.e., in the context of an application lasting more than a few minutes [Ito+13]). In principle, the calibration has to be (re)done whenever the position of the HMD changes on the user's head. Such changes are likely to occur frequently. Reasons can be abrupt user motions, as well as situations when the user temporarily removes the glasses, e.g. to rub the eyes, to move through spatially complex terrain or to perform some tasks without the glasses. Quite often, subsequent (re)calibration is skipped since it is too much trouble.

Recently, gaze tracking cameras have become commercially viable. They are small enough to be combined with an OST-HMD. In combination with recent commercial activities to productize AR-glasses for a sizeable market, as well as the emerging trend to build mobile "intelligent" devices that include an increasing number of built-in sensors, we expect that future HMDs will include such cameras. Since such a camera has direct view of (one of) the user's eyes, it generates additional information that can be used to simplify and improve the display calibration process. Yet, the question arises, whether the eye position can be determined precisely and robustly enough to be usable for stable HMD calibrations. After all, small estimation errors of the eye position can have a significant impact on user-perceived offsets between real and virtual objects.

This section reports on an approach towards combining camera-based eye tracking with HMD calibration (see Fig. 7.1 and 6.1). An eye tracking camera (T) is rigidly attached to the bottom rim of an HMD, oriented towards one of the user's eyes. A second camera (W) determines the HMD pose within the surrounding world environment. The rigid setup of the two cameras and the HMD is pre-determined in an offline calibration process. Combining this static HMD calibration with dynamic eye tracking, we are able to generate world-related augmentations in the HMD even when the HMD is moved on the user's head. We have first, encouraging results that the setup generates a registration quality that is comparable to the state of the art – with potential for further improvements by employing more rigorous offline calibration procedures.

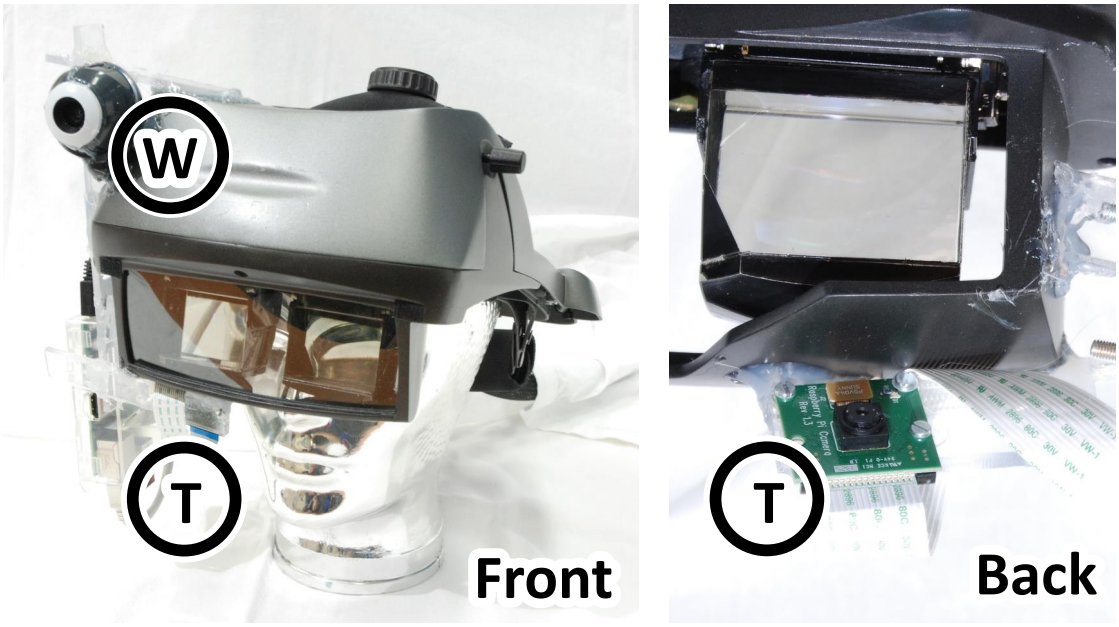


Figure 7.1: Our technical setup: a world camera, W, and an eye tracking camera, T, are connected to an OST-HMD.

7.2 Related Work

7.2.1 Head-mounted Eye Tracking

As discussed in the above, there are already a number of efforts on wearable head-mounted eye tracking systems including [TK12; Tsu+11; Ish+10; Sch+09; NNT13]. Those systems are developed mostly to collect and analyze a user's viewing direction for the purpose of gaze analysis. Among them, Tsukada et al. [TK12; Tsu+11] present first-person vision systems that formulate 3D model-based eye tracking using weak perspective projection. Nitschke et al. [NNT11; NNT13] have built an eye tracking system which reconstructs a user's view from a reflected image on his/her eye. They also employ a 3D eye model and use perspective projection for the eye pose estimation from the image.

7.2.2 Combinations of HMDs and eye trackers

Some research efforts combine eye trackers with HMDs. Nilsson et al. [NGC07] have developed a video see-through HMD with an eye tracker, and present hands-free interaction based on users' gaze. Also, Lee et al. [Lee+11] employ a monocular OST-HMD with a scene camera and an eye tracker for interaction purposes. Due to an HMD positioning issue, their eye tracker tracks the left eye while the graphics are shown on the right screen of the HMD.

Unlike such HMD systems where trackers are additionally attached to the display frames, Hua and Gao [HG12] have designed a compact eye-tracked HMD (ET-HMD) to which an eye tracker

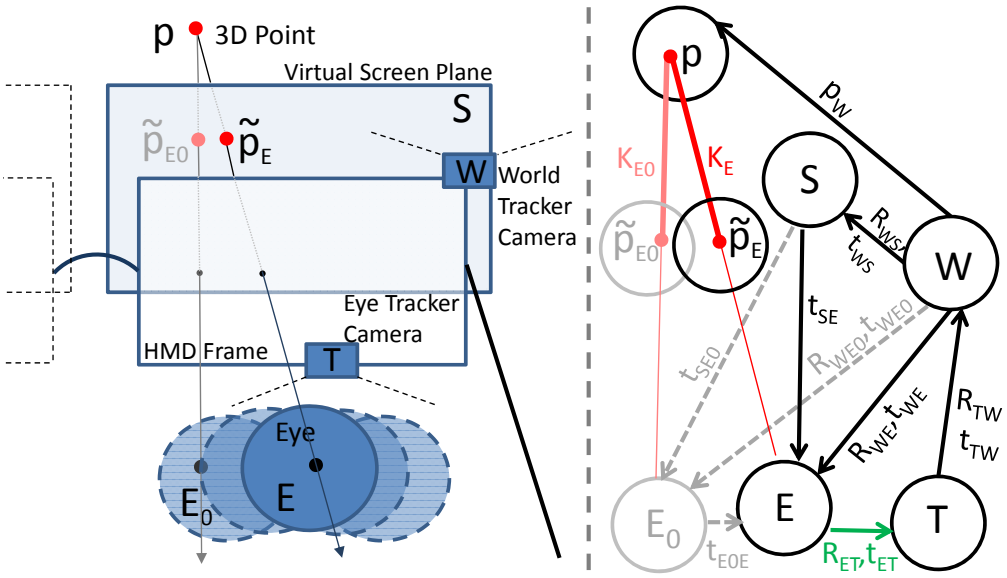


Figure 7.2: (left) Schematic drawing, illustrating the relevant internal coordinate systems of the right screen S with an eye tracking camera T, a world camera W, and the user's eye E (or E_0). (right) Spatial relationships necessary to determine the projection matrix from 3D object points p onto the screen point $\widetilde{\mathbf{p}_E}$ or $\widetilde{\mathbf{p}_{E_0}}$, respectively.

is integrated as a part of the optics system using a free-form prism.

Recently, Makibuchi et al. [MKY13] have developed a calibration method for OST-HMDs. They have attached a camera to an OST-HMD and estimated users' eye positions with a fiducial marker. However, they needed user interaction to calibrate the system.

7.3 Method

This section describes our calibration algorithm in two sub-sections: Sec. 7.3.1 formulates the overall calibration procedure using a 3D eye position, and Sec. 7.3.2 provides more detail on the 3D eye position estimation.

7.3.1 Calibration formulation

This section describes our method in four steps. Fig. 7.2 illustrates the spatial relationships of our calibration formulation.

In the first step, consider a virtual camera defined by an eye and the virtual screen of an OST-HMD, assuming that the screen with its coordinate system S is planar and located at $\mathbf{t}_{SE_0} := [s_x, s_y, s_z]^{\mathrm{T}}$ in the camera coordinate system E_0. The camera can be considered as an off-axis pinhole camera. Now, without loss of generality, assume E_0's z-axis is perpendicular to

the virtual screen. The camera is then expressed by the intrinsic camera matrix $\mathtt{K}_{E_0}$ as:

$$\mathtt{K}_{E_0} := \underbrace{\begin{bmatrix} \alpha_x & & \\ & \alpha_y & \\ & & 1 \end{bmatrix}}_{=:\mathtt{A}} \underbrace{\begin{bmatrix} s_z & & -s_x \\ & s_z & -s_y \\ & & 1 \end{bmatrix}}_{=:\mathtt{S}(\mathbf{t}_{SE_0})=\mathtt{S}(s_x,\, s_y,\, s_z)}. \tag{7.1}$$

The function $\mathtt{S}(\mathbf{t}_{SE_0})$ transforms 3D points in E_0 to the virtual image screen in real scale, and $\mathtt{A}$ is a diagonal matrix which transforms projected screen points into image pixel points by scaling factors $\{\alpha_x, \alpha_y\}$. Note that $\mathtt{S}(\mathbf{t}_{SE_0})$ is chosen so that $\mathbf{t}_{SE_0}$ is projected to the origin of the image pixel plane. Furthermore, the scaling factors $\{\alpha_x, \alpha_y\}$ are independent of the eye position, whereas $\mathbf{t}_{SE_0}$ is dependent.

Secondly, consider another eye position with its coordinate system E. We can define a new virtual camera consisting of the virtual screen and the new eye position. Following the same concept as for E_0, set the pose of E so that its z-axis is perpendicular to the virtual screen. Then, the transformation from E_0 to E is defined only by the translation $\mathbf{t}_{E_0E} := [t_x, t_y, t_z]^{\mathrm{T}}$. Thus the screen position in E can be written as $\mathbf{t}_{SE} = \mathbf{t}_{E_0E} + \mathbf{t}_{SE_0}$.

From Eq. 7.1, the intrinsic matrix $\mathtt{K}_E$ of the new virtual camera is obtained as

$$\mathtt{K}_E = \mathtt{A}\mathtt{S}(\mathbf{t}_{SE}) = \mathtt{K}_{E_0}\mathtt{S}(t_x/s_z,\, t_y/s_z,\, 1 + t_z/s_z). \tag{7.2}$$

The above shows that we can convert a virtual camera to another given by a new eye position and some display-specific parameters.

Thirdly, consider relocating the above coordinate systems together into a world coordinate system W defined somewhere on the HMD. By recalling the perspective projection of a pinhole camera, we obtain

$$\widetilde{\mathbf{p}_E} = \underbrace{\mathtt{K}_E \begin{bmatrix} \mathtt{R}_{WE} & \mathbf{t}_{WE} \end{bmatrix}}_{=:\mathtt{P}_{WE}(\mathbf{t}_{WE})} \begin{bmatrix} \mathbf{p}_W \\ 1 \end{bmatrix} \tag{7.3}$$

where $\mathtt{P}_{WE}$ is the projection matrix that projects world points onto the new virtual camera. Note that the rotations from the world to any eye coordinate systems including $\mathtt{R}_{WE}$ are actually identical since they are defined by the rotation of the screen $\mathtt{R}_{WS}$. Thus, it follows that $\mathtt{R}_{WS} = \mathtt{R}_{WE}(= \mathtt{R}_{WE_0})$. Then from Eq. 7.2, Eq. 7.3, and $\mathbf{t}_{SE} = \mathbf{t}_{WE} - \mathbf{t}_{WS}$, we obtain the following,

$$\mathtt{P}_{WE}(\mathbf{t}_{WE}) = \mathtt{A}\mathtt{S}(\mathbf{t}_{WE} - \mathbf{t}_{WS}) \begin{bmatrix} \mathtt{R}_{WS} & \mathbf{t}_{WE} \end{bmatrix} \tag{7.4}$$

$$= \mathtt{K}_{E_0}\mathtt{S}(t_x/s_z,\, t_y/s_z,\, 1 + t_z/s_z) \begin{bmatrix} \mathtt{R}_{WS} & \mathbf{t}_{WE} \end{bmatrix}. \tag{7.5}$$

Eq. 7.4 does not rely on the old eye position $\mathbf{t}_{WE_0}$. Instead it requires a complete set of display parameters: $\mathbf{t}_{WS}$ and the pixel scaling factors $\{\alpha_x, \alpha_y\}$. On the other hand, Eq. 7.5 does not rely on these parameters, except for $[\mathbf{t}_{WS}]_z$ (since $s_z + t_z = [\mathbf{t}_{WE} - \mathbf{t}_{WS}]_z$), – and it reuses the intrinsic matrix $\mathtt{K}_{E_0}$ from the old eye position. Both cases also require $(\mathtt{R}_{WE}, \mathbf{t}_{WE})$, the pose between the world and the eye.

Acq.	Condition	Param.	Relationship(From/To)
Online	Required	$\mathbf{t}_{ET}$	Eye(current) / Tracker
Offline	Required	$\mathtt{R}_{WS}$	World / Screen
		$\mathtt{R}_{WT}$	World / Tracker
		$\mathbf{t}_{WT}$	World / Tracker
	Option 1	$\mathbf{t}_{WS}$	World / Screen
		$\alpha_{\{x,y\}}$	Real scale / Img. pixel
	Option 2	$\mathtt{K}_{E_0}$	Eye(previous)
		$\mathbf{t}_{WE_0}$	World / Eye(prev.)
		$[\mathbf{t}_{WS}]_z$	World / Screen

Table 7.1: A summary of calibration parameters. Option 1 and 2 can be selected depending on available calibration environments.

Finally, consider an eye tracker rigidly mounted on the OST-HMD. Let T be the eye tracker's coordinate system. The tracker then provides the position of the eye E in T as $\mathbf{t}_{ET}$. Then the relationship between the eye and the world can then be written as

$$\mathbf{t}_{WE} = \mathtt{R}_{TE}\left(\mathbf{t}_{WT} - \mathbf{t}_{ET}\right) = \mathtt{R}_{WE}\mathtt{R}_{WT}^{\mathrm{T}}\left(\mathbf{t}_{WT} - \mathbf{t}_{ET}\right) = \mathtt{R}_{WS}\mathtt{R}_{WT}^{\mathrm{T}}\left(\mathbf{t}_{WT} - \mathbf{t}_{ET}\right). \tag{7.6}$$

Since the eye tracker and the OST-HMD are rigidly connected, $(\mathtt{R}_{WT}, \mathbf{t}_{WT})$ is constant and needs to be calibrated only once. All parameters except for the eye position $\mathbf{t}_{ET}$ relative to $\mathtt{P}_{WE}$ can be determined offline. Therefore, if such offline calibration is conducted beforehand, the system can reconstruct a projection matrix online for a given eye position $\mathbf{t}_{ET}$. Since the position is estimated by the tracker automatically, the system does not require user interaction at run time.

Table 7.1 summarizes the calibration parameters necessary to compute the projection matrix $\mathtt{P}_{WE}$. Two calibration *Options* exist by choosing either Eq. 7.4 or 7.5 for the derivation of $\mathtt{P}_{WE}$. Sec. 7.4 will present methods to obtain the calibration parameters for real settings.

7.3.2 Eye position acquisition

This section describes an eye tracking algorithm which estimates $\mathbf{t}_{ET}$, the 3D eye position relative to a tracker. In principle, any eye tracking method that determines the optical center of an eyeball can be used for our calibration system. In the current implementation, we employ a 3D eye position estimation method by Nitschke et al. [NNT11]. For the 2D ellipse extraction, we developed an ellipse fitting method based on work by Swirski et al. [SBD12] together with their open-source iris detector [HF98; FPF99].

7.3.2.1 3D Eye Position Estimation for Perspective Projection

We briefly describe the 3D eye position estimation method of Nitschke et al. [NNT11]. A more elaborated derivation is found in Sec. 2.2.1 of their paper.

Nitschke et al. model the eyeball as two overlapping spheres with different radii and separate

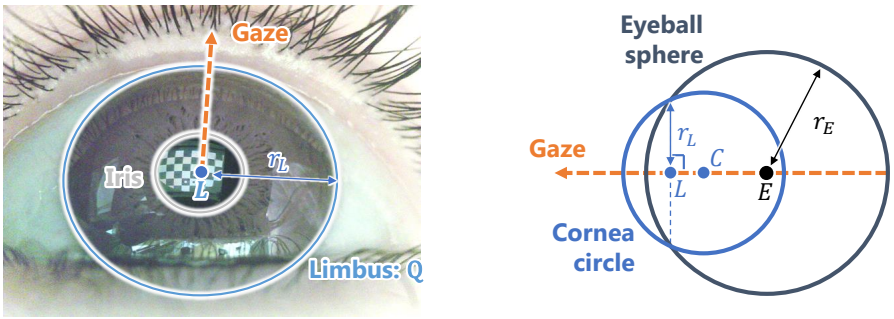


Figure 7.3: Explanation of our eye model parameters. (left) A 2D limbus circle is superimposed on an eye image. (right) Schematic illustration of the eyeball model.

centers of curvature (Fig. 7.3 right). Their method reconstructs the 3D position of the eye and its gaze direction through inverse projection of a 2D ellipse (typically used to describe the eye's limbus of the cornea) from an eye image. They assume that a 3D limbus circle with known constant radius r_L is observed as a 2D ellipse $\mathtt{Q}$ in an image captured by a camera with a known intrinsic matrix $\mathtt{K}$ (Fig. 7.3 left). $\mathtt{Q}$ is a matrix representation of the 2D ellipse with $\tilde{\mathbf{p}}^{\mathrm{T}}\mathtt{Q}\tilde{\mathbf{p}} = 0$ for all homogeneous 2D points $\tilde{\mathbf{p}}$ on the ellipse. Convert the ellipse $\mathtt{Q}$ to the physical scale as $\mathtt{Q}_e := \mathtt{K}^{\mathrm{T}}\mathtt{Q}\mathtt{K}$. Then factorize $\mathtt{Q}_e$ by the eigenvalue decomposition as $\mathtt{Q}_e = \mathtt{U}\mathtt{V}\mathtt{U}^{\mathrm{T}}$ to obtain the eigenvector matrix $\mathtt{U}$ and the eigenvalue matrix $\mathtt{V}$. Assume that the three eigenvalues, α, β and γ are ordered such that $\alpha\beta > 0$, $\alpha\gamma < 0$, and $|\alpha| > |\beta|$.

With $g := \sqrt{\frac{\beta-\gamma}{\alpha-\gamma}}$, $h := \sqrt{\frac{\alpha-\beta}{\alpha-\gamma}}$, and undetermined signs $\{s_k\}_{k=1}^{3}$, the 3D circle can be represented by the 3D limbus center position $\mathbf{t}_{LT}$ and the gaze normal vector $\mathbf{n}_T$ in T as:

$$\mathbf{t}_{LT} := \frac{s_3 r_L}{\sqrt{-\alpha\gamma}}\mathtt{U}\begin{bmatrix} s_2 h\gamma \\ 0 \\ -s_1 g\alpha \end{bmatrix}, \mathbf{n}_T := \mathtt{U}\begin{bmatrix} s_2 h \\ 0 \\ -s_1 g \end{bmatrix}. \tag{7.7}$$

Under the two-sphere eye model, the eye position $\mathbf{t}_{ET}$ can then be expressed for a given constant eye radius r_E as:

$$\mathbf{t}_{ET} = \mathbf{t}_{LT} - \sqrt{r_E^2 - r_L^2}\mathbf{n}_T. \tag{7.8}$$

Due to $\{s_k\}_{k=1}^{3}$, there are up to $2^3 = 8$ mathematically valid solutions for $\mathbf{t}_{ET}$. Applying the assumptions that the eyeball is in front of the camera and that the gaze vector is oriented toward the camera, the ambiguity can be reduced down to 2. In general, resolving this remaining ambiguity requires additional prior knowledge such as anthropometric properties of the eyeball or constraints from relationships between both eyes. The next section explains our disambiguation approach based on the temporal consistency of a single eyeball position. Fig. 7.4 top right and bottom right visualize $\mathtt{Q}$ and $\mathbf{t}_{ET}$ respectively. In our implementation, r_L is set to 5.5 [mm] and r_E to 12.6 [mm] according to Nitschke et al. [NNT11].

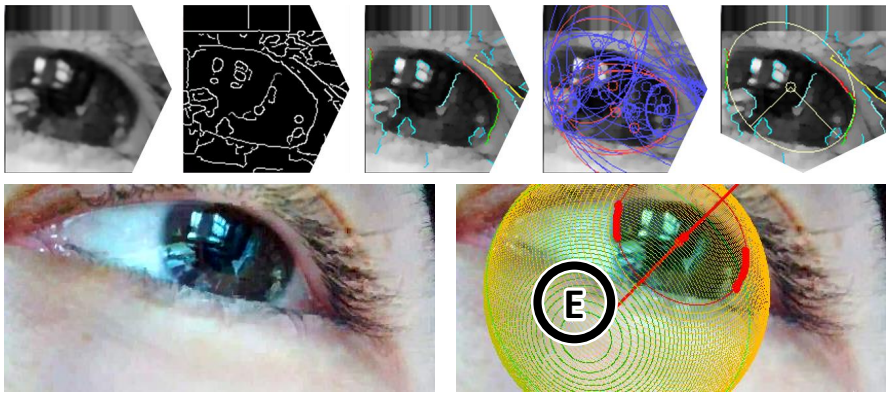


Figure 7.4: Eye position estimation overview. The images in the first row are output of Algorithm 1. From left to right: smoothing (line 4), Canny edge detection (line 5), edge segmentation (line 7), RANSAC ellipses $\{\mathtt{Q}_{\text{local\_best}}\}$, and the final $\mathtt{Q}$. The second row shows an original image and the visualization of an estimated 3D eyeball with an annotation of the eye coordinate system E.

7.3.2.2 Eye Position Disambiguation

The 3D eye position estimation method gives true and false estimates as described in the above section. Our basic idea is to disambiguate between them by sampling multiple eye images taken in a row, and finding a consistent combination of the estimates across time. The underlying assumption is that eye positions during a calibration stay the same or at least similar to one another.

Thus the correct estimates from several images should yield eye center positions that are very close to one another. Once a combination is obtained, the final estimate $\mathbf{t}_{ET}$ can be generated by taking their mean or median, or by estimating a time series of them. In the current system, k-means clustering [Mac+67] with cluster size $k = 2$ is employed to find the combination. After the clustering, the median of the cluster with the smallest within-class variance is used as the final eye position estimate.

Although the above method is applied in the experiment section, a simple median across all true and false estimates (namely, $k = 1$) also gave a result with a similar tendency.

Fig. 7.5 shows a set of eye position estimates and final estimates $\mathbf{t}_{ET}$. The sets of estimates were computed from each of 4 sets of eye images taken during 4 SPAAM calibrations (data sequence 1, as explained in Sec. 7.5). Each image set consisted of between 10 and 28 images. The k-means-based method and the simple median had a similar estimation variance of $\sigma^2 \approx 1.6e^{-6}$[m].

7.3.2.3 2D Limbus Ellipse Extraction

Our method to extract 2D limbus ellipses from eye images consists of 2 steps: detection of the limbus image region and extraction of limbus edges. Algorithm 1 describes our procedure.

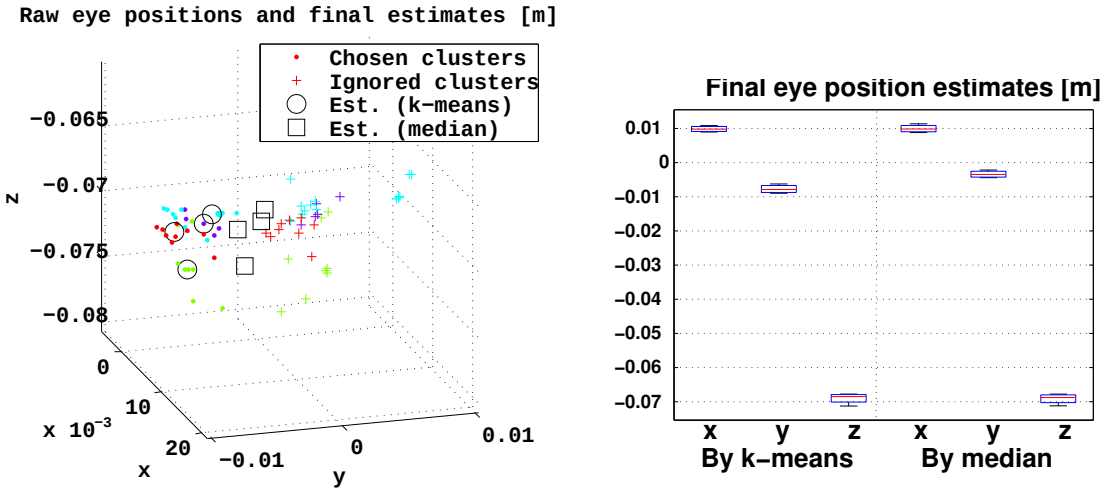


Figure 7.5: Disambiguation of raw 3D eye positions for 4 eye-image sets collected in a row: 3D visualization of the raw eye positions and final estimates $\mathbf{t}_{ET}$ by two different approaches, and a boxplot of the final estimates.

The first phase (line 2) employs an open-source Haar-like iris detector[1] made by Swirski et al. [SBD12]. Given an eye image, the detector returns an estimated 2D iris center and the surrounding region of interest (ROI) information. We expand the output ROI region from the central iris area to also cover the surrounding limbus area because the limbus radius is static while the iris's is not. The 3D eye position estimation algorithm needs a constant radius.

The second phase (lines 4-27) extracts the limbus ellipse from the ROI. We have adapted the algorithm by Swirski et al. [SBD12] to our purpose of detecting the limbus rather than the iris. The image is smoothed by a morphological opening operation and Gaussian blurring to obtain smoother edges around a limbus. Canny edge detection then computes a binary edge image (lines 4-5).

Next, isolated edge segments E are extracted from the edge image (line 7). In Swirski's code, the star burst algorithm was applied to obtain the iris edge pixels. However, this cannot be easily applied to our limbus case since eyelids often hide the top and bottom edges of the limbus. Thus, we erase edge pixels that are connected in horizontal chains and we subdivide the edge segments recursively (line 7). Due to this heuristic, the algorithm does not create long edges that contain false edge pixels stemming from eyelid-limbus borders. The obtained edges are then sorted by length. Only the top N edges are selected for further processing and the edges are re-fitted at subpixel precision (lines 9-12).

The algorithm subsequently (lines 18-24) uses a RANSAC approach fitting a tentative ellipse to each pair of edges (i.e. $N(N-1)/2$ pairs) and computing a score defined by the number of inliers among all pixels in the edge pair. Suitable heuristic criteria (ellipse properties such as its size, angle, center position etc.) are used to discard inappropriate ellipse candidates $\mathtt{Q}_{\text{local\_best}}$ (line

[1] http://www.cl.cam.ac.uk/research/rainbow/projects/pupiltracking/

Algorithm 1: Pseudo code for the limbus ellipse estimation.

input : Eye Image $\mathbf{I}$
output : 2D limbus ellipse $\mathtt{Q}$

```
// PHASE 1
I_L ← LIMBUS-DETECTION(I) // From [SBD12]
// PHASE 2
I_L ← IMAGE-SMOOTHING(I_L)
I_E ← CANNY-EDGE-DECETION(I_L)
// Collect non-horizontal edge segments
E ← EDGE-SEGMENTATION(I_E)
// Chose top N edges by their length
E ← TOP-LENGTH-EDGES(E)
// Refine points on each edge
foreach E_i ∈ E do
    E_i ← SUBPIXEL-FITTING(E_i)
// Find Ellipse for each edge pair
foreach {E_i, E_j} ⊂ E s.t. i ≠ j do
    edge ← E_i ∪ E_j
    // Ellipse fitting by RANSAC
    S_local_best ← 0, Q_local_best ← null
    repeat N_max times
        points ← RANDOM-SAMPLE(edge, 5)
        Q_k ← ELLIPSE-FITTING(points)
        // Count the # of inlier points
        S_k ← INLIER-COUNT(Q_k, edge)
        if S_k < S_local_best then
            S_local_best ← S_k, Q_local_best ← Q_k
    // Update the best ellipse
    if S_local_best < S_global_best then
        S_global_best ← S_local_best, Q ← Q_local_best
```

26). For instance, the eye center should stay inside the image. The top row of the pictures in Fig. 7.4 shows an example of the series of intermediate outputs of the second step.

7.4 Technical Setup

For the practical use of calibration methods, it is critical to establish a setup procedure. Following the mathematical formulation of our calibration algorithm in Sec. 7.3, this section describes procedures to obtain the required display parameters for the algorithm.

7.4.1 Hardware setup

We have built an OST-HMD system equipped with an eye tracker, as described below and in Fig. 7.1. We use nVisor ST60 from NVIS – an OST-HMD with 1280x1080 resolution. Although the display is stereo capable, only the right eye display is used for the current setup. An outward looking Logitech Webcam C200 serves as the world coordinate system W. It provides 640x480-pixel video and is attached to the OST-HMD. The display and the camera are both connected to a commodity laptop.

For the eye tracker T, a Raspberry Pi CSI Camera Module (or RaspiCam) is used. It is connected to a Raspberry Pi board and they are both attached to the OST-HMD (see Fig. 7.1). The position of the module was chosen to be at the bottom of the right display lens of the OST-HMD so that the module can capture the right eye of an operator easily. Although the module can provide 5MP (2592x1944 pixel) static images maximum, its hardware-encoded H.264 1080p (1920x1280) video stream is sent and resized to HVGA(480x320).

The video stream was transferred by the board to the laptop through a wired local network using gstreamer 1.0. It is received by the laptop using the VLC player. The default focal length of the module is too far for capturing eye images from near distance. Thus its lens component is carefully unsealed and reconfigured for suitable focal length.

7.4.2 System calibration

To apply our method to an OST-HMD system, such as the one described above, we have to precalibrate the system such that the calibration parameters listed in Table 7.1 become known.

In our system, both cameras are calibrated beforehand by using printed checkerboard patterns of different sizes. An open-source MATLAB toolbox[2] is used for the calibration.

Calibration of $\{\mathtt{R}_{WT}, \mathbf{t}_{WT}\}$: The parameters describe the relationship between the eye tracker and the world camera. Since both are optical sensors, visual tracking using fiducial markers can determine their 6 DoF poses relative to the markers. Therefore, by letting them observe several markers that are jointly registered to a common coordinate system, we can compute the required parameters. We employed a multi-marker setup as depicted in Figure 7.6. Our setup uses printed

[2] http://www.vision.caltech.edu/bouguetj/calib_doc/

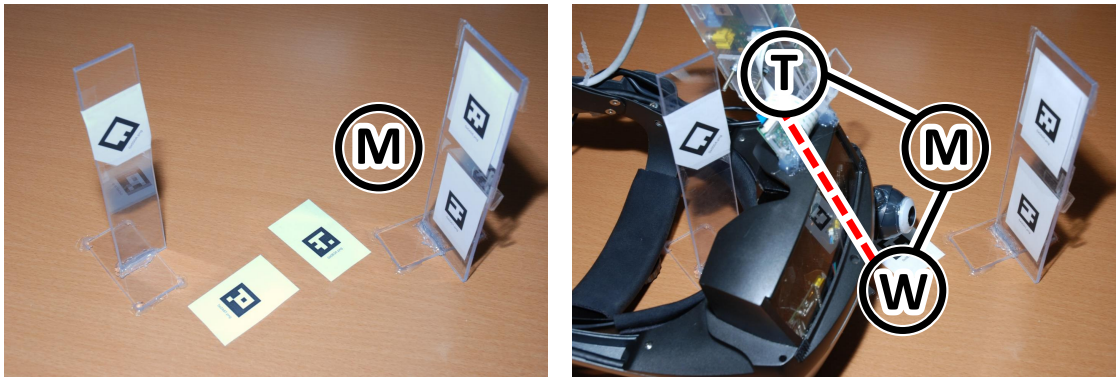


Figure 7.6: A multi-marker setup used for calibrating $(\mathtt{R}_{WT}, \mathbf{t}_{WT})$: multi markers only (left), with the OST-HMD (right). The distances between markers and the cameras are less than 10 cm.

square markers which are installed on planar world objects and rigidly fixed with respect to one another. The marker positions were measured beforehand to identify their spatial relationship by using the Ubitrack library [New+04; Hub+07]. Then our OST-HMD system was placed in the environment such that both cameras can see at least one of the markers of the scene. Finally, the relative pose between the tracker and the world camera is derived via the multi-marker coordinate system using the library.

Calibration of $\mathtt{R}_{WS}$: This parameter is the rotation from the world to the virtual screen coordinate system. The easier of two ways is to apply SPAAM for the OST-HMD once - independently of further use of SPAAM. In that case, $\mathtt{R}_{WS}$ can be obtained as one of the calibration parameters.

A second way towards recovering this rotation from the world to the screen coordinate system is to use the display calibration procedure of DRC [Owe+04]. The method uses a camera to capture the virtual screen from different view points. Then, it computes the parameters by reconstructing the 3D position of a pattern displayed on the virtual screen. The method is a bit more complicated, but it does not require any user interaction and thus might be more robust.

Calibration of $\{\mathbf{t}_{WS}, \alpha_x, \alpha_y\}$ (Option 1): These parameters are necessary when one uses the Eq. 7.4 for the method. These parameters define the position of OST-HMD's virtual screen relative to the world, and scaling factors that convert points on the virtual screen in real scale into the image as pixels. The DRC display calibration [Owe+04] can also provide this information.

Calibration of $\left\{\mathtt{K}_{E_0}, \mathbf{t}_{WE_0}, [\mathbf{t}_{WS}]_z\right\}$ (Option 2): These parameters are necessary when one uses the Eq. 7.5 for the method. This option was used instead of the other one for the simplicity of its calibration requirement. A SPAAM calibration can give the $\left\{\mathtt{K}_{E_0}, \mathbf{t}_{WE_0}\right\}$. The distance, $[\mathbf{t}_{WS}]_z$, from the world coordinate system to the virtual screen plane is ideally estimated by a method like DRC. Instead, we performed a calibration relying on a manual focus camera. We placed the camera so that it focuses on the virtual screen, then the obtained focal length was further

subtracted by manually-measured distance between this camera and the world camera on the display. In our case, the distance was about 78 cm.

7.5 Experiment

7.5.1 Design of the test process

As argued in the beginning of Chap. 7, OST-HMDs are not stable on users' heads during use in real AR applications. Currently, SPAAM-like methods are common practice, requiring users to align 3D targets to 2D points on the display screen. But, for the sake of time, users may compromise, staying on old calibration parameters rather than reperforming a tedious calibration routine (Degraded SPAAM). In sections 7.3 and 7.4, we have presented an eye tracking-based approach towards interaction-free display calibration.

We have evaluated the performance of our method (*proposed* condition) compared to SPAAM (*training-error* condition) and to Degraded SPAAM (*test-error* condition). Fig. 7.7 shows an overview of the process.

7.5.1.1 Data Acquisition

Prior to the evaluation, we acquired a series of data sets. Each set consists of 20 2D-3D point correspondences, with each 3D world point having been manually aligned to a 2D point on the screen (aka SPAAM). The 3D points were distributed across an area of about 90x50x60 cm^3 (width, height, depth) centered around position (-4, 3, -100) [cm] relative to the operator. During this process, we also recorded 15 eye images per 2D-3D point correspondence (i.e., 300 images in total). The top row of Fig. 7.7 illustrates the step in form of a pink and a green box. The 2D-3D correspondences formed the basis for a SPAAM-based estimation of the display projection matrix (blue box). The eye images were used to compute a series of 3D eye positions using our proposed algorithm (orange box). We call such a data collection session a *block*.

A total of 4 data collection sessions were performed while the HMD was kept as stably as possible on the user's head (top row of Fig. 7.7). After the first sequence, the OST-HMD was taken off from the head and put back on to simulate a degraded calibration situation. Then, the second set of blocks was collected in the same manner as the first one (indicated in Fig. 7.7 by variable $N = 1,2$). These two sequences form the ground-truth (GT) data which are the basis for subsequent evaluations of the three evaluation conditions.

Further details of the eye image acquisition process: The eye images were acquired manually and then processed automatically to obtain eye positions. Images for which the algorithm failed to extract eye positions were identified manually. This way, outlier images stemming, e.g., from blinking eyes, motion-blurred eyes, strong inward-light reflection etc. were eliminated manually. Yet, both true and false position estimates were passed to the calibration algorithm and disambiguated automatically, as described in Sec. 7.3.2.2. An extra visible light source was used due to the limited brightness caused by the OST-HMD.

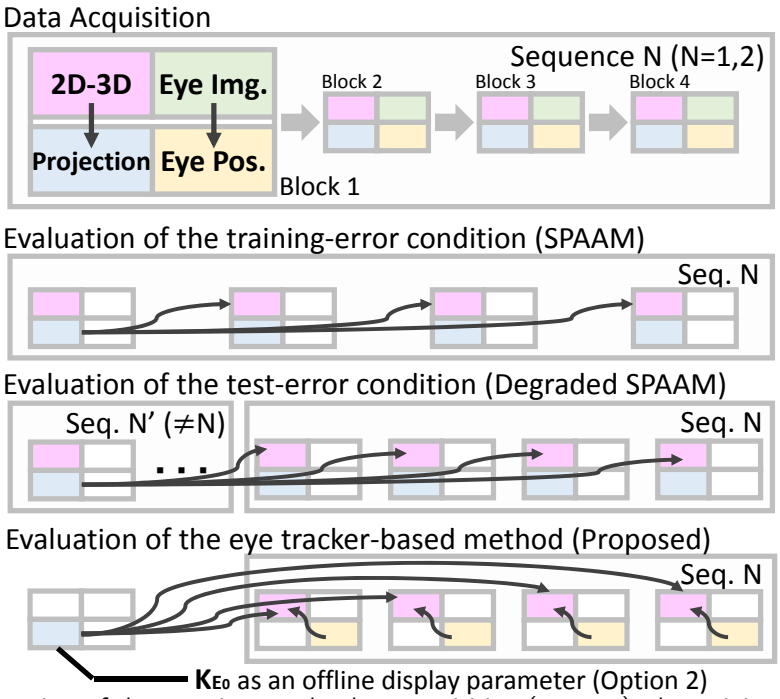


Figure 7.7: Overview of the experiment: the data acquisition (top row), the training-error condition (second row), the test-error condition (third row), and our proposed condition (bottom). Arrows in the evaluation diagrams indicate which data source from which block and sequence is to be projected and compared with respect to which GT data of which other block and sequence. For clarity, only one *Projection* cell is chosen to visualize the arrows.

7.5.1.2 Data Evaluation Process

***Training-error* evaluation:** We selected one of the four blocks of a sequence N to conduct a SPAAM calibration. The other three blocks of the same sequence N were subsequently used to evaluate the quality of the SPAAM calibration, using the evaluation procedure of Sec. 7.5.1.3. Switching the block for the calibration and redoing the same, a total of 24 ($4 \times 3 \times 2$) error measurement sets were obtained. The second row of Fig. 7.7 shows the procedure of this evaluation.

***Test-error* evaluation:** We used a block from one sequence, N, for the SPAAM calibration and tested the results against the four blocks from the other sequence N' – simulating the Degraded SPAAM condition in which a user continues using the same initial display calibration after the display was moved. This yields 32 ($4 \times 4 \times 2$) sets of error measurements. The third row of Fig. 7.7 shows this evaluation procedure.

Evaluation of our proposed method: For all eye image sets in each block, we applied the eye pose estimation method described in Sec. 7.3.2. The required precalibration parameters were estimated once, as described in Sec. 7.4. Option 2 in Table 7.1 was used to compute projection

matrices for our method. Remember that the dataset used for this precalibration was collected in the same manner as for the acquisition of the blocks, yet the dataset was not included in the 8 blocks for a fair comparison. The bottom row of Fig. 7.7 shows the evaluation procedure.

7.5.1.3 Evaluation Algorithm

Our evaluation aims to determine how well an estimated eye position approximates the true one that existed during the ground-truth data acquisition process. The following indirect and direct error measurements are employed.

2D projection error: This indirect error is considered as an image-based indicator of the estimation quality of the eye position. Firstly, 3D points of the GT data set are reprojected by the estimated projection matrices. Then the error is computed as the average distance between the reprojected points and the GT 2D points. This error is computed for each pair of estimated projections and the GT data set in the three evaluations of the previous section.

3D eye positions: 3D eye positions can be decomposed from the projection matrices by SPAAM. Thus, for each block, we compare the positions with the ones given by our 3D eye position estimation.

7.5.2 Results

Comparison of 2D Projection Error: Fig. 7.8 shows the analysis of the 2D projection error of each algorithm. In the following analysis, we have excluded a single outlier of the SPAAM calibration for fair comparison regarding mean values.

The mean 2D projection error of the proposed method is larger than that of SPAAM, and their difference is statistically significant in a two-sample t test $\left(p \approx 5.01e^{-5} < 0.05\right)$. However, the variance of the error for SPAAM $\left(\sigma^2 \approx 1.56\right)$ is larger than that for the eye tracker-based method $\left(\sigma^2 \approx 1.09\right)$ even though an outlier was excluded. Thus, in the display-head fixed situation, SPAAM achieves higher quality than our method, yet it might be unstable when users need to recalibrate the system often. This negative effect in SPAAM calibrations is further amplified in Degraded SPAAM. In turn, it can be expected that our proposed method is better than Degraded SPAAM since it is independent of the change of the display position relative to user's head.

The mean error of the proposed method is smaller than that of Degraded SPAAM at a statistically significant level $\left(p \approx 6.07e^{-8} < 0.05\right)$. Besides, the error variance of Degraded SPAAM $\left(\sigma^2 \approx 2.01\right)$ is worse than that of our proposed method. Thus, once users start compromising speed for precision by reusing old calibration parameters, our proposed method can provide more accurate and precise projection results.

A potential reason for the difference between our proposed method and the SPAAM methods can be illustrated in the following analysis of the 3D eye positions.

Comparison of 3D Eye Positions: Fig. 7.9 shows an analysis of the estimated eye positions in the world coordinate system, using eye positions from datasets $N = 1$ and 2 (left and right subfigures).

The first row illustrates the distribution of estimated eye positions as boxplots separately for x, y and z. It shows that the SPAAM method causes a large variance along the z-axis – the viewing direction of eye, with about ± 3 cm in each sequence. This error tendency coincides with the SPAAM analysis results by Axholt et al. [Axh+11; Axh+10]. On the other hand, eye positions estimated by the proposed method have smaller variance. This feature is preferable since it makes the calibration system more consistent.

The second row shows distances between eye positions and their mean position. The subfigures illustrate a similar trend. The unstable characteristic propagates to the projection matrix at the end, causing 2D projection errors with larger variance in SPAAM than in eye tracker-based calibration. The last row of Fig. 7.9 visualizes the 3D eye positions showing a much larger variance for SPAAM (green) than for our proposed method (red).

7.6 Discussion

Throughout the experiment, the eye tracker-based method achieved better calibration quality compared to Degraded SPAAM. Yet, it did not work better than SPAAM in terms of the 2D projection error. There are several types of possible reasons for this.

The first is the estimation quality of the offline calibration parameters. For example, in our implementation, the distance from the world camera to the virtual screen was crudely measured by hand with a manual focus camera (see Sec. 7.4.2). Furthermore, our formulation to compute projection matrices reuses a projection matrix obtained by another calibration method (here: SPAAM). Thus the maximal calibration accuracy that our method could achieve for the static head-display setup might be upper bounded by that of SPAAM. This hypothesis can be tested by conducting a complete virtual display calibration. A method such as the DRC [Owe+04] can be an option for such an investigation. Also, as mentioned in their work, a virtual display possibly has distortion due to its complicated optics, and thus certain "undistortion" might be required. For example, recent work by Lee and Hua [LH13] tackles this problem computer vision.

A second possible type of error is related to the eye tracking part. It requires several anatomical parameters related to the human eyeball, such as its radius and the limbus radius. These values should be different for each individual. Furthermore, the simplified 3D eye model used in the algorithm might be insufficient for the application. For example, the assumption that the visual axis of the eye is aligned with the optical axis does not hold in general. This might add a systematic error bias to the eye pose estimation. However, our informal examinations that analyze the noise tolerance of our method by perturbing parameters of the eyeball indicate that they had a lower impact on the projection error than adding noise to the virtual screen parameters.

Thorough investigations of potential error sources need to be conducted to further improve the performance of the interaction-free calibration method.

7.7 Summary

This section presented an interaction-free calibration method for OST-HMDs utilizing 3D eye localization. The method estimates the projection matrix by using static calibration parameters of an OST-HMD and online eye position measurements. The experiment shows that our calibration is more stable than the SPAAM calibration in terms of the 2D reprojection error and the estimated 3D eye position. Furthermore, our method performs better than a degraded SPAAM setup where users stay on an old set of calibration parameters – which is often the case in AR applications.

Future work directions involve the integration of a precalibration procedure to obtain complete virtual display parameters, the analysis of error sources, and sophistication of the eye tracking system with consideration to real-time capability.

Furthermore, many user-oriented issues arise – how can a system detect that the current calibration has collapsed and needs to be redone? Are there *good* gaze directions that produce better calibration accuracy? And if so, how can the system benefit from that at recalibration time without putting burden on the users? Are even *frame-wise* calibrations possible? These questions would be crucial for making OST-HMDs practically usable.

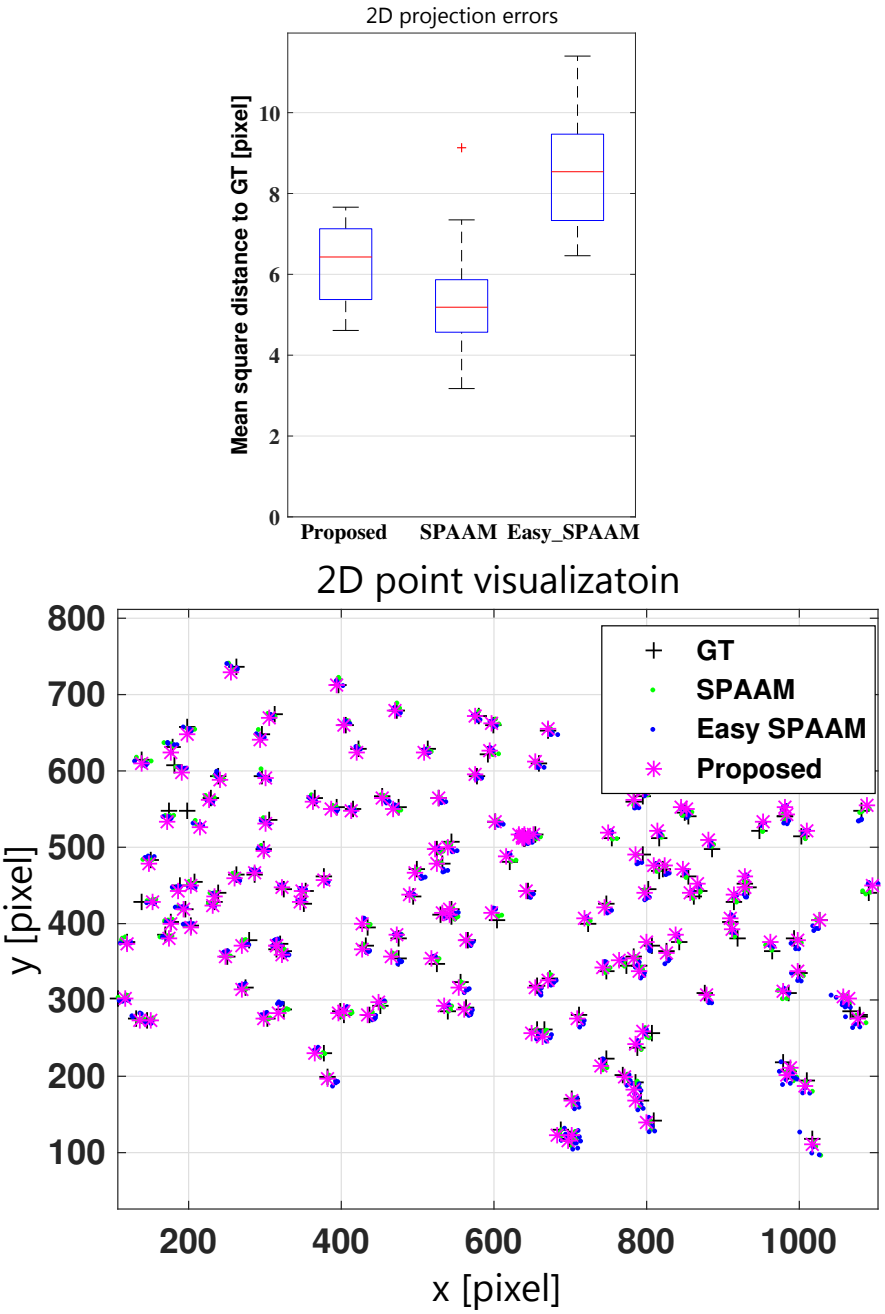
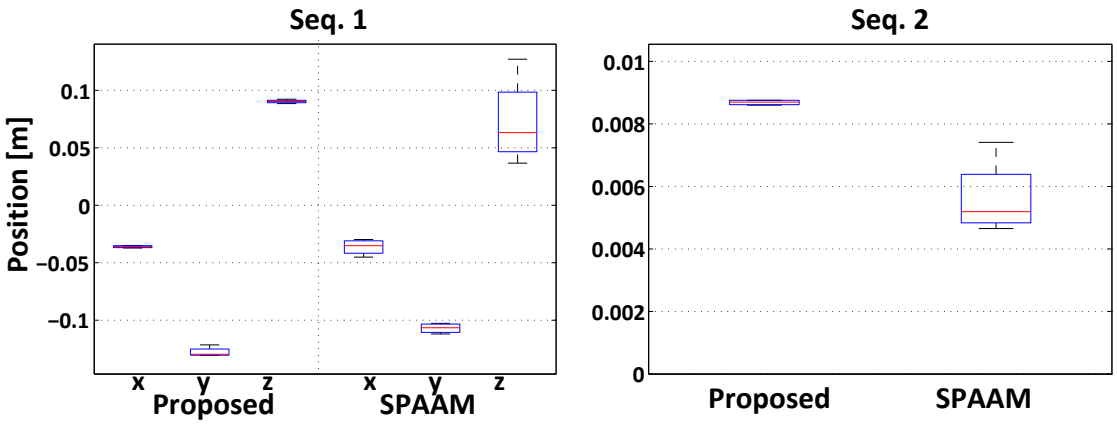
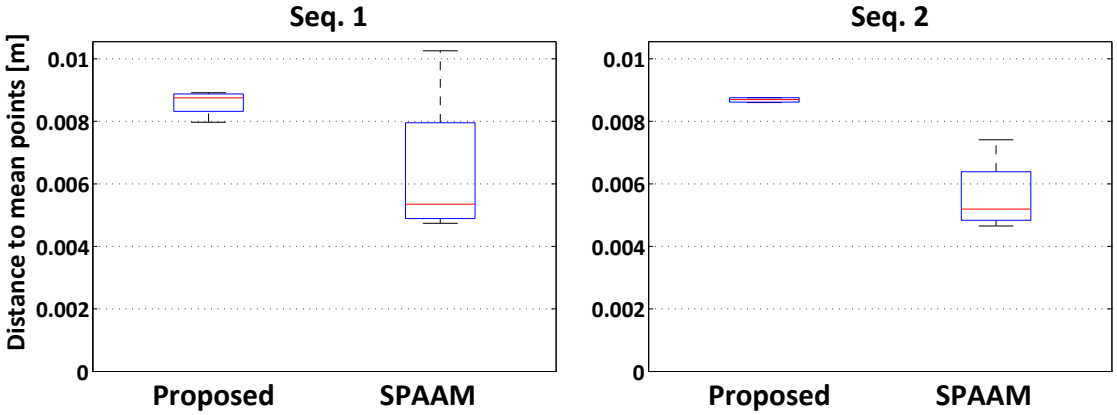


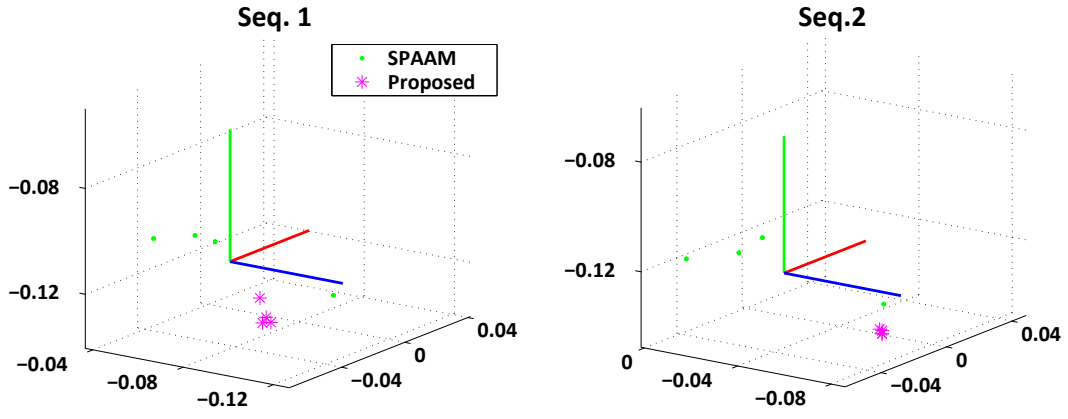
Figure 7.8: (top) A boxplot of the 2D projection analysis with the y axis showing the mean squared error distance. (bottom) Plot of both the projected points and the GT points .



(a) Distribution of the estimates



(b) Variance



(c) 3D visualization

Figure 7.9: Analysis of 3D eye positions $\mathbf{t}_{WE}$: (a) Boxplots of the positions, (b) Variance of their distance from their mean positions, and (c) a 3D visualization of the points.

8 Calibration Error Analysis for Automated Method

This section is based on the work that the author published in IEEE ISMAR 2014 conference [IK14b].

8.1 Introduction

As we mentioned in the previous sections, a crucial issue in AR applications using OST-HMDs is to render 3D information from the current viewpoint of the user, – and, more particularly, according to the user's eye position, relative to a not quite stable HMD pose on the user's head. Such rendering requires 2D-3D projection matrix from the world to a screen. Recall that manual calibration methods like SPAAM ([TN00], Chap. 6) find a projection which explains manually-collected 2D-3D alignments best. Thus the projection is in a black box, ignores spatial relationship of a display and an eye(Fig. 8.1a).

Our automated calibration method, ~~IN~~DICA (Chap. 7), on the other hand, generates the projection more explicitly with respect to the user's current eyeball position by combining the tracked eye position online. Depending on predetermined offline parameters, the method has two setups that require either: a partial set of display parameters in combination with a previous calibration result (Fig. 8.1b, Recycle Setup), or a full set of all display parameters (Fig. 8.1c, Full Setup).

The two setups represent the same, yet their interpretations are quite different: Recycle Setup (Fig. 8.1b) updates the black box from a manual method using a measured eye position. The box can be given by a previously performed SPAAM calibration or a camera-based HMD calibration such as in [GFG08]. On the other hand, Full Setup formulates the system as a combination of an explicit, display model and eye model (Fig. 8.1c). The setup requires an extra offline display calibration. Both setups have their pros and cons in practice, thus users would choose either setup over the other depending on the application (esp. means and convenience of performing the required calibrations). Therefore, evaluating and comparing both setups are valuable for the future use of ~~IN~~DICA.

In the previous section, we have compared our Recycle Setup with SPAAM calibrations in repeated experiments, and were able to demonstrate that the Recycle Setup performs more stably than SPAAM in estimating accurate 3D eye positions. However, we could not yet show whether the same holds for the Full Setup, which would model the system more accurately. More importantly, it is still unclear how estimation errors of the online/offline parameters affect the

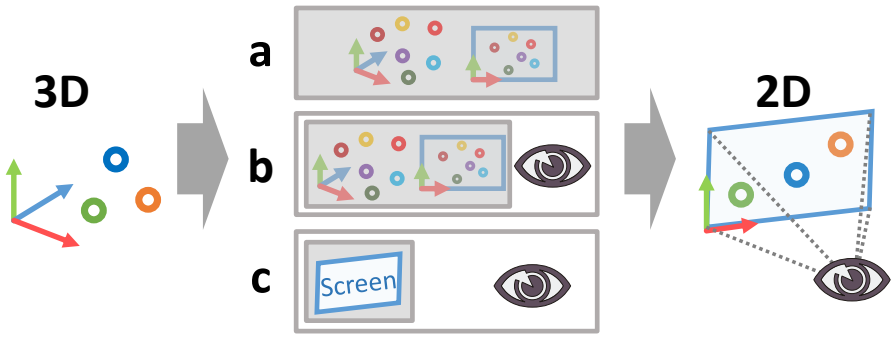


Figure 8.1: Interpretation of projection *black boxes* from different calibration methods: (a) SPAAM, (b) Recycle Setup, and (c) Full Setup.

calibration performance. Thus, it is still unclear how accurately each of the offline parameters must be determined in practice. Therefore, it is helpful to understand the sensitivity of the overall HMD calibration accuracy to imprecision of individual parameters such that the calibration process can be designed to place high priority on the most sensitive parameters.

This section evaluates the Full Setup by employing a marker-based offline display-parameter calibration. It confirms that the method performs as accurately as SPAAM and Recycle Setup. More importantly, this section reports on a theoretical analysis of the calibration sensitivity of both setups as well as SPAAM with respect to the various display calibration parameters, based on real observations. The analysis allows us to reason about the display calibration accuracy for each method and provides insight into designing a suitably optimized OST-HMD system.

Contributions As a summary, this section contributes to the research area on OST-HMD calibration by

- Reporting on a theoretical analysis of the calibration sensitivity of two setups in the automated OST-HMD calibration method as well as SPAAM
- Confirming that Full Setup of the automated method performs equally a SPAAM and Recycle Setup
- Providing insight into optimizing an OST-HMD system in terms of calibration stability

8.2 Related Work

A detailed discussion of OST-HMD calibrations is in the beginning this chapter.

Display Parameter Estimation Several research groups work on the display-parameter estimation. The offline step of Display-Relative Calibration (DRC) [Owe+04] estimates the display parameters through a standard vision-based calibration including first-order radial distortion.

Gilson et al. [GG12; GFG08] employ Tsai's method for estimating a camera frustum of an OST-HMD combined with an outside-in tracking system. Lee et al. [LH13] extended DRC to estimate higher-order radial distortion and showed that coefficients up to the 2nd order were actually effective.

Sensitivity Analysis to Calibration Errors Holloway [Hol97] provide a thorough end-to-end error analysis for AR applications with an OST-HMD system. The author's work includes a mathematical model of the system and an evaluation which confirms the model by taking measurements by a real system. Axholt et al. [Axh+10] modeled user-dependent noise for a SPAAM calibration and observed that the noise manifests itself as a poorly estimated eyepoint, primarily along the line of sight, both in simulation and real measurements.

8.3 Method

We elaborate our sensitivity analysis strategy in the following sections.

8.3.1 Two setups in interaction-free calibration

Fig. 6.1 bottom shows the various coordinates to be defined as part of the display calibration process. Calibrating an OST-HMD means to estimate a 3-by-4 projection matrix

$$\mathtt{P}_{WE}\left(\mathbf{t}_{WE}\right) := \mathtt{K}_E \begin{bmatrix} \mathtt{R}_{WE} & \mathbf{t}_{WE} \end{bmatrix} \tag{8.1}$$

of a virtual camera defined by the OST-HMD and an eye (See Sec. 7.3 for more detail). The intrinsic matrix $\mathtt{K}_E$ has two representations:

$$\mathtt{K}_E = \underbrace{\mathtt{K}_{E_0} \begin{bmatrix} 1+z_{EE_0}/z_{SE} & & -x_{EE_0}/z_{SE} \\ & 1+z_{EE_0}/z_{SE} & -y_{EE_0}/z_{SE} \\ & & 1 \end{bmatrix}}_{\text{Recycled ~~INDICA}}, \tag{8.2}$$

$$= \underbrace{\begin{bmatrix} \alpha_x & & c_x \\ & \alpha_y & c_y \\ & & 1 \end{bmatrix} \begin{bmatrix} z_{SE} & & -x_{SE} \\ & z_{SE} & -y_{SE} \\ & & 1 \end{bmatrix}}_{\text{Full ~~INDICA}}, \tag{8.3}$$

where $\mathbf{t}_{SE} = [x_{SE}, y_{SE}, z_{SE}]^{\mathrm{T}}$, $\mathbf{t}_{E_0E} = [x_{EE_0}, y_{EE_0}, z_{EE_0}]^{\mathrm{T}}$. $\mathbf{a} := [\alpha_x, \alpha_y]^{\mathrm{T}}$ is a scaling factor that converts 3D points on the screen to pixel points. $c_x := (w-1)/2$ and $c_y := (h-1)/2$ define the image center with the pixel width w and height h. $\mathtt{K}_{E_0}$ is the intrinsic matrix of another virtual camera defined by the old eye position E_0.

Equation 8.3 does not rely on the old eye position $\mathbf{t}_{WE_0}$. Instead, it requires the display pose $\mathbf{t}_{WS}$ and the scaling vector $\mathbf{a}$. On the other hand, Eq. 8.2 does not rely on these parameters, except for

$[\mathbf{t}_{WS}]_z$ since $\mathbf{t}_{SE} = \mathbf{t}_{WE} - \mathbf{t}_{WS}$, and it reuses the old intrinsic matrix $\mathtt{K}_{E_0}$. Both cases require $(\mathtt{R}_{WE}, \mathbf{t}_{WE})$, the pose between the world and the eye. We call calibration with Eq. 8.3 as Full Setup, and with Eq. 8.2 as Recycle Setup.

Let T be the coordinates of an eye tracker rigidly mounted on the OST-HMD, then $\mathbf{t}_{WE} = \mathtt{R}_{WS}\mathtt{R}_{WT}^{\mathrm{T}}(\mathbf{t}_{WT} - \mathbf{t}_{ET})$ (Eq. 6 in [IK14a]). In the previous work, we computed a 2D corneal limbus for estimating $\mathbf{t}_{ET}$ through the Canny edge detector. Instead, our current implementation uses the Line Segment Detector by Gioi et al. [Von+12].

8.3.2 Display parameter calibration

Our approach is similar to the work by Owen et al. [Owe+04] which reconstructs 3D shape of a virtual screen via the triangulation. They build a calibration jig for an HMD to obtain the pose of an calibration camera which captures the virtual screen of the HMD. Our method is modified in two-ways: we model the virtual screen as a 3D plane, and employ an inside-out marker tracking to obtain the calibration-camera poses. The following describes the calibration procedure:

Step 1: Place an OST-HMD so that a calibration camera observes a calibration pattern displayed on the virtual screen S. *Step 2*: Capture the pattern by the calibration camera, and capture a square marker M by a world camera W. *Step 3*: Remove the HMD carefully without touching the calibration camera, and capture the marker by the calibration camera directly. *Step 4*: Repeat the step 1 to 3 N_C times.

At the step 1, a real black sheet is placed in front of the HMD so that the calibration camera can see the pattern clearly. The position of the calibration camera is changed at every iteration of the step 1. After the above procedure, one obtains poses between the world camera and each calibration camera C_k as $\left(\mathtt{R}_{C_kW}, \mathbf{t}_{C_kW}\right)$. Ordinary camera calibration technique gives the virtual screen poses $\left\{\left(\mathtt{R}_{SC_k}, s\mathbf{t}_{SC_k}\right)\right\}_k$ up to a common scale factor s. This definition of s assumes that the scale factors $\{\alpha_x, \alpha_y\}$ are represented by a common factor α. We use this assumption through this section. Without loss of generality, the size of a checkerboard tile is set to its pixel size. Then α becomes equal to s^{-1}.

8.3.2.1 Linear Optimization Step

The rotation estimate $\widehat{\mathtt{R}_{WS}}$ can be obtained by taking the mean of $\{\mathtt{R}_{WS}^k := \left(\mathtt{R}_{C_kW}\mathtt{R}_{SC_k}\right)^{\mathrm{T}}\}_k$ in the quaternion space [Gra01]. The 3D position of the virtual screen in the world coordinates W can be written as, $\mathbf{t}_{SW}^k(s) = s\mathtt{R}_{C_kW}\mathbf{t}_{SC_k} + \mathbf{t}_{C_kW}$ for the k-th camera position. Define the averaged screen position

$$\overline{\mathbf{t}_{SW}}(s) := \frac{1}{N_C}\sum_{k=1}^{N_C}\mathbf{t}_{SW}^k(s), \tag{8.4}$$

and define,

$$\mathbf{a}^k := \frac{1}{N_C}\sum_{j=1}^{N_C}\mathtt{R}_{C_jW}\mathbf{t}_{SC_j} - \mathtt{R}_{C_kW}\mathbf{t}_{SC_k}, \quad \mathbf{b}^k := \frac{1}{N_C}\sum_{j=1}^{N_C}\mathbf{t}_{C_jW} - \mathbf{t}_{C_kW}. \tag{8.5}$$

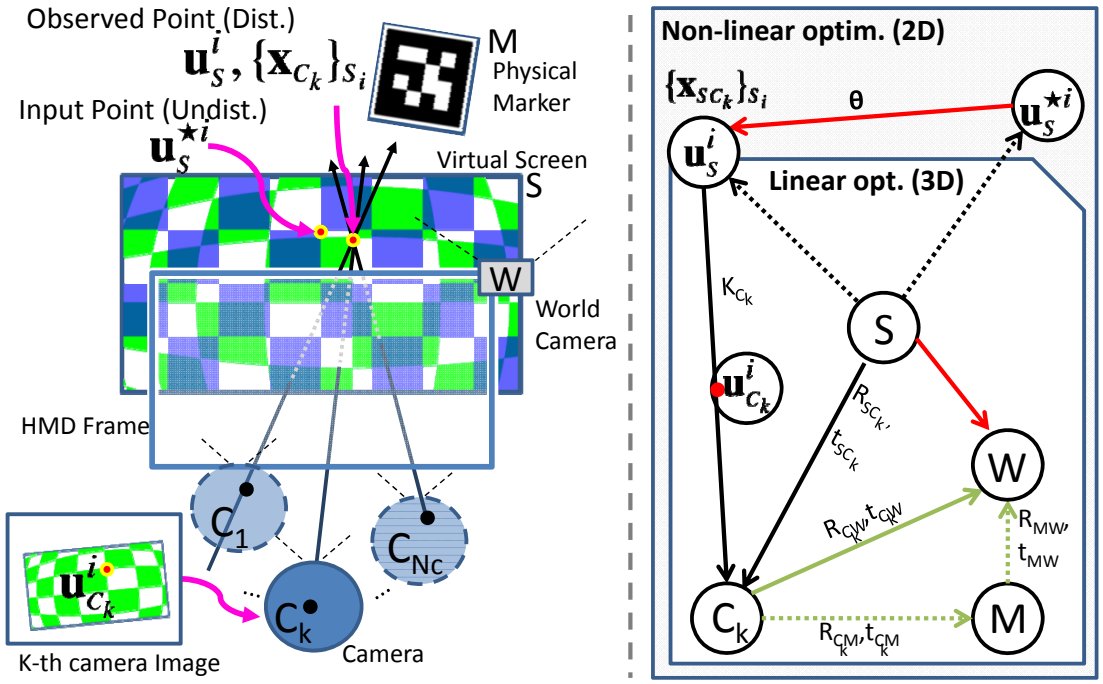


Figure 8.2: Display calibration setup including the optional non-liner optimization model. (left) Schematic drawing. (right) Spatial relationship. The blue regular checker pattern represents the original image sent to the display for rendering.

Since the positions $\left\{\mathbf{t}_{SW}^{k}\right\}_{k}$ represent the same, we obtain the following cost function over the scale factor s:

$$f(s) := \frac{1}{N_C}\sum_{k=1}^{N_C}\left\|\overline{\mathbf{t}_{SW}}(s) - \mathbf{t}_{SW}^{k}\right\|^2 = \frac{1}{N_C}\sum_{k}\left\|s\mathbf{a}^k + \mathbf{b}^k\right\|^2. \tag{8.6}$$

By solving this for $f'(s) = 0$, a linear estimate of the scale $\widehat{s}$ and the estimated display translation are obtained as follows,

$$\widehat{s} = -\sum_{k=1}^{N_C}\frac{\mathbf{a}^{k\,\mathrm{T}}\mathbf{b}^k}{\left\|\mathbf{a}^k\right\|^2},\ \ \widehat{\mathbf{t}_{WS}} = -\widehat{\mathtt{R}_{WS}}^{\mathrm{T}}\overline{\mathbf{t}_{SW}}(\widehat{s}). \tag{8.7}$$

8.3.2.2 Non-linear Optimization Step [Optional]

Having obtained a linear estimate, one can also apply non-linear optimization while taking the distortion parameters of the OST-HMD into account (Hua and Gao [HG12]). See Fig. 8.2 left for the relationship of the poses to be defined.

Let $\mathbf{u}_S^{\star i}$ be the i-th original pixel point on an OST-HMD –an integer pair which a software sends to render an image pixel on the display. Let $\{\mathtt{R}_{SW}, \mathbf{t}_{SW}, s\}$ be initial display parameters. In the ideal case, the k-th camera observes the original pixel on the 3D screen as:

$$\mathbf{x}_{C_k}(\mathbf{u}_S^{\star i}) := \mathtt{R}_{SC_k}\begin{bmatrix} s\,\mathbf{u}_S^{\star i} \\ 0 \end{bmatrix} + \mathbf{t}_{SC_k},\ \ \mathtt{R}_{SC_k} = \left(\mathtt{R}_{C_kW}\right)^{\mathrm{T}}\mathtt{R}_{SW},\ \mathbf{t}_{SC_k} = \left(\mathtt{R}_{C_kW}\right)^{\mathrm{T}}\left(\mathbf{t}_{SW} - \mathbf{t}_{C_kW}\right). \tag{8.8}$$

Let project$(\cdot)$ be a function $(\mathbb{R}^3 \to \mathbb{R}^2)$ which projects a 3D point in $\{C_k\}$ onto the image plane of the calibration camera, and $\mathbf{u}_{C_k}^i$ be the 2D point of the i-th display pixel captured by the k-th camera. Then, the reprojection error is defined as:

$$\mathrm{E}(\mathtt{R}_{SW}, \mathbf{t}_{SW}, s) := \sum_{i=1}^{N_C}\sum_{k\in S_i}\left\|\mathrm{project}(\mathbf{x}_{C_k}(\mathbf{u}_S^{\star i})) - \mathbf{u}_{C_k}^i\right\|^2, \tag{8.9}$$

where S_i is an index set of cameras that observed the i-th point. Consider distortion model $\mathbf{u}_S^i := \mathrm{distort}(\mathbf{u}_S^{\star i}, \theta)$ with the distortion parameter θ, then the calibration camera observes $\mathbf{u}_S^i$ (Fig. 8.2 left). Now, using Eq. 8.9, non-linear optimization problem is formed as:

$$\underset{\mathtt{R}_{SW}, \mathbf{t}_{SW}, s, \theta}{\mathrm{argmin}}\sum_{i}\sum_{k\in S_i}\left\|\mathrm{project}(\mathbf{x}_{C_k}(\mathbf{u}_S^i)) - \mathbf{u}_{C_k}^i\right\|^2. \tag{8.10}$$

Although our evaluation has not yet included the non-linear optimization step, Eq. 8.9 is still useful to evaluate estimated display parameters; if these error for an image is extremely high compared to the others, the image might be an outlier due to poor estimation of the initial pose, accidental motion of the calibration camera during the image capture, and so on.

8.3.3 Sensitivity measurement

Having introduced ~~I~~INDICA, an important question arises: how accurately should we estimate each parameter of the method to achieve enough registration quality for an AR application? This section proposes a formal way to answer this question by defining a sensitivity measurement to calibration errors.

OST-HMD projection matrix can be treated as a function of calibration parameters: $\mathtt{P}_{WE}(\lambda)$ where a vector λ encapsulates the parameters, e.g. $\{\mathbf{a}, \mathtt{R}_{WS}, \mathtt{R}_{WT}, \mathbf{t}_{WT}, \mathbf{t}_{ET}, \mathbf{t}_{WS}\}$ for Full Setup. In other words, λ represents a display configuration of one particular OST-HMD design.

Let $\mathbf{x}_W$ be a 3D point in the world coordinate system W, then $\mathbf{x}_W$ is projected to a 2D pixel $\mathbf{u}_{\mathbf{x}_W}$ by the projection matrix $\mathtt{P}_{WE}(\lambda)$ as $\mathbf{u}_{\mathbf{x}_W}(\lambda) := \begin{bmatrix} p/r & q/r \end{bmatrix}^{\mathrm{T}}$, where $\begin{bmatrix} p & q & r \end{bmatrix}^{\mathrm{T}} := \mathtt{P}_{WE}(\lambda) \begin{bmatrix} \mathbf{x}_W & 1 \end{bmatrix}^{\mathrm{T}}$. Let $\lambda^\star$ be true calibration parameters and $\Delta\lambda$ represents small perturbations added during a calibration procedure, then $\mathbf{u}_{\mathbf{x}_W}(\lambda^\star + \Delta\lambda)$ represents a perturbed 2D pixel.

Define the reprojection error vector

$$\mathbf{e}(\mathbf{x}_E) := \mathbf{u}_{\mathbf{x}_W}(\lambda^\star + \Delta\lambda) - \mathbf{u}_{\mathbf{x}_W}(\lambda^\star). \tag{8.11}$$

The first-order Taylor expansion gives an approximation of the vector as

$$\mathbf{e}(\mathbf{x}_E) \simeq \mathbf{J}_{\lambda^\star}(\mathbf{x}_E)\Delta\lambda + O(\Delta\lambda^2), \tag{8.12}$$

where

$$\mathbf{J}_{\lambda^\star}(\mathbf{x}_W) := \left(\frac{d\mathbf{u}_{\mathbf{x}_W}}{d\lambda}\right)_{\lambda^\star} \tag{8.13}$$

is a Jacobian matrix of $\mathbf{e}(\mathbf{x}_E)$ evaluated at $\lambda^\star$. This Jacobian determines the primary behavior of the error caused by $\Delta\lambda$, and requires the first derivative of $(p(\lambda), q(\lambda), r(\lambda))$ only:

$$\frac{d\mathbf{u}_{\mathbf{x}_W}}{d\lambda} = \frac{d}{d\lambda}\begin{bmatrix} p/r \\ q/r \end{bmatrix} = \begin{bmatrix} (p'r - pr')/r^2 \\ (q'r - qr')/r^2 \end{bmatrix}. \tag{8.14}$$

Although the calculation of Eq. 8.14 is straightforward for most of the parameters in λ, rotation $\{\mathtt{R}_{WS}, \mathtt{R}_{WT}\}$ still need care due to their implicit parametrization. In a similar manner described in [KS11], we treat the little change of a rotation as an infinitesimal rotation in Lie algebra, which is expressed by a vector $\omega := [\omega_x, \omega_y, \omega_z]^{\mathrm{T}}$ as

$$\Delta\mathtt{R} := [\omega]_\times \mathtt{R} = \begin{bmatrix} & -\omega_z & \omega_y \\ \omega_z & & -\omega_x \\ -\omega_y & \omega_x & \end{bmatrix} \mathtt{R}, \tag{8.15}$$

where $[\cdot]_\times$ is the skew-symetric matrix operator. For example, $\frac{dp(\mathtt{R})}{d\mathtt{R}}$ can be computed by$= \frac{dp(\Delta\mathtt{R})}{d\omega}$. Higher-order terms of the rotation parameters such as $\omega_k * \omega_{k'}$ are treated as zero.

Now, averaging $\mathbf{J}_{\lambda^\star}(\mathbf{x}_W)$ over the 3D point set $\{\mathbf{x}_W\}$ seems to behave as a sensitivity measurement. However, remember that $\mathbf{x}_W$ itself is dependent on some display parameters. Thus the

defined Jacobian does not take the same input set given different display configurations. Instead of $\mathbf{x}_W$, consider a 3D point $\mathbf{x}_E$ in E with a polar coordinate representation:

$$\mathbf{x}_E(\delta,\theta,\varphi) := \delta \begin{bmatrix} \sin\theta\cos\varphi \\ \sin\theta\sin\varphi \\ \cos\theta \end{bmatrix}, \ \theta \in \Theta, \varphi \in \Phi, \delta \in L, \tag{8.16}$$

then $\mathbf{x}_W = \mathrm{R}_{WS}^{\mathrm{T}}(\mathbf{x}_E - \mathbf{t}_{WE})$. Taking the mean of $\mathbf{J}_{\lambda^\star}(\mathbf{x}_W)$ over the polar coordinate domain $\{L,\Theta,\Phi\}$ gives an expected error sensitivity measurement:

$$\mathrm{E}[\mathbf{J}_{\lambda^\star}(\mathbf{x}_W)] = \frac{1}{V}\int_{L,\Theta,\Phi} \mathbf{J}_{\lambda^\star}(\mathbf{x}_W)\, d\delta\theta\varphi, \tag{8.17}$$

where V is the volume of the 3D space defined by $\mathbf{x}_E(\theta \in \Theta, \varphi \in \Phi, \delta \in L)$. Finally, by taking the sample mean of Eq. 8.17, we obtain our sensitivity measurement:

$$\overline{\mathbf{J}_{\lambda^\star}} := \frac{1}{N}\sum_{\delta,\theta,\varphi} \mathbf{J}_{\lambda^\star}\big(\mathrm{R}_{WS}^{\mathrm{T}}(\mathbf{x}_E(\delta,\theta,\varphi) - \mathbf{t}_{WE})\big), \tag{8.18}$$

where N is the number of 3D points sampled.

In summary, given a true OST-HMD configuration $\lambda^\star$ and a 3D space V in which an AR application needs to visualize AR contents, $\overline{\mathbf{J}_{\lambda^\star}}$ gives a prediction of the sensitivity of each calibration parameter to calibration errors. Each column of $\overline{\mathbf{J}_{\lambda^\star}}$ represents the sensitivity of a parameter with a different unit (e.g. scale, rotation and translation).

Note that, for the sake of intuitive understanding, we convert $\overline{\mathbf{J}_{\lambda^\star}}$ so that each calibration parameter has a scalar sensitivity measurement. For instance, let $\begin{bmatrix} \mathbf{e}_x & \mathbf{e}_y & \mathbf{e}_z \end{bmatrix}$ be a 2-by-3 submatrix of $\overline{\mathbf{J}_{\lambda^\star}}$ correspond to $\mathbf{t}_{WT}$, i.e. $\frac{d\mathbf{u}_{\mathbf{x}_W}}{d\mathbf{t}_{WT}}$. We define the scalar representation of the submatrix as $e_{\mathbf{t}_{WT}} := (\|\mathbf{e}_x\| + \|\mathbf{e}_y\| + \|\mathbf{e}_z\|)/3$. Other scalar measurements are defined in the same manner such as $e_{\mathbf{a}}$, $e_{\mathrm{R}_{WS}}$, and so on. To compensate the difference of the units in the measurements, each measurement should be scaled properly during comparisons as explained later in the experiment section.

8.4 Technical Setup

Following the mathematical formulation of the calibration algorithm in Sec. 8.3, this section describes procedures to obtain the required display parameters for the algorithm.

8.4.1 Hardware setup

We have built an OST-HMD system equipped with an eye tracker as described below and in Fig. 8.3. We use nVisor ST60 from NVIS –an OST-HMD with 1280x1080 resolution. The left-eye display is used for the current setup. An outward looking camera, Logitech C200, serves as the world camera W. For the eye tracker T, a PlayStation Eye camera is used. These cameras provide

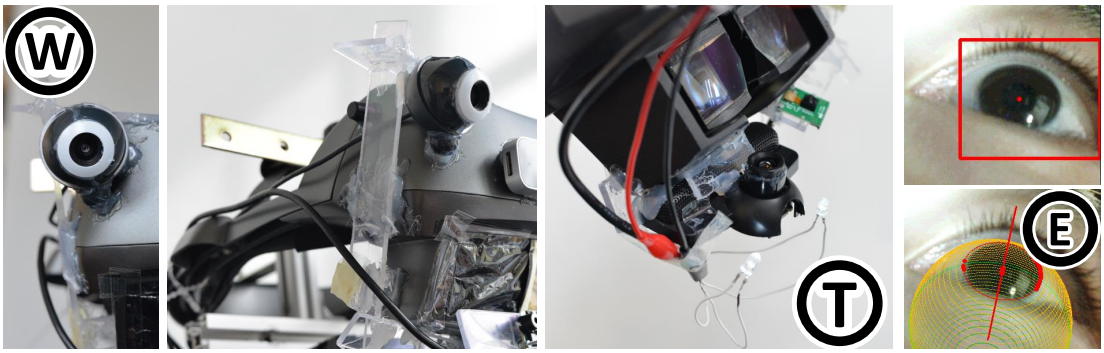


Figure 8.3: The OST-HMD setup used through the evaluations. The images contain annotations of the coordinate systems.

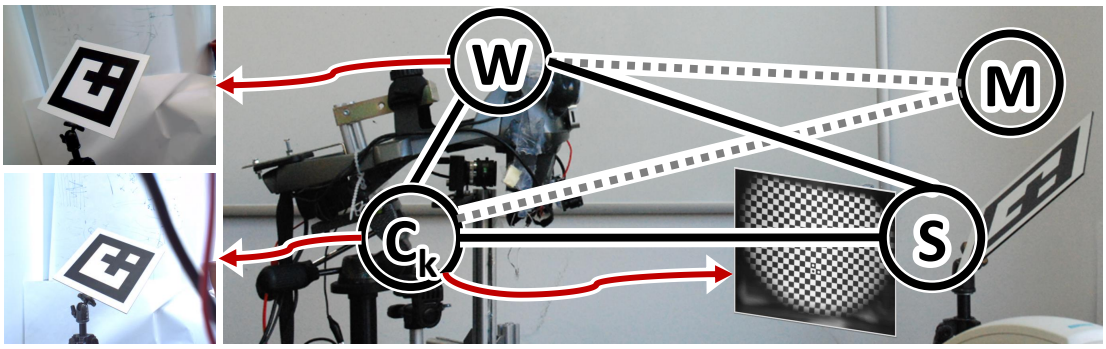


Figure 8.4: Display calibration setup for calibrating $\{\mathbf{a}, \mathrm{R}_{WT}, \mathbf{t}_{WS}\}$. (right) Spatial relationship with virtual screen. The screen is intentionally drawn in the image right for the schematic drawing. (left cuolumn) Sample images captured by the cameras.

640x480-pixel video and are attached to the HMD.

The position of the tracker is chosen to be at the bottom of the left display lens of the HMD. CL Eye Platform SDK[1] is used to capture images from the eye camera. The default focal length of its varifocal lens is manually adjusted and fixed to a suitable length.

8.4.2 System calibration

To apply Full/Recycle setup to an OST-HMD system, such as the one described above, we have to precalibrate the system such that the display parameters become known. We conduct the marker-based display-parameter calibration as explained in Sec. 8.3.2

Fig. 8.4 shows our calibration setup. For the calibration camera, we used iDS's UI-1240ML-C-HQ, an industrial camera which provides 1280x1024 color image, together with an 8-mm C-mount lens. World, tracker, and calibration cameras are calibrated beforehand by an open-

[1] http://codelaboratories.com/products/eye/sdk/

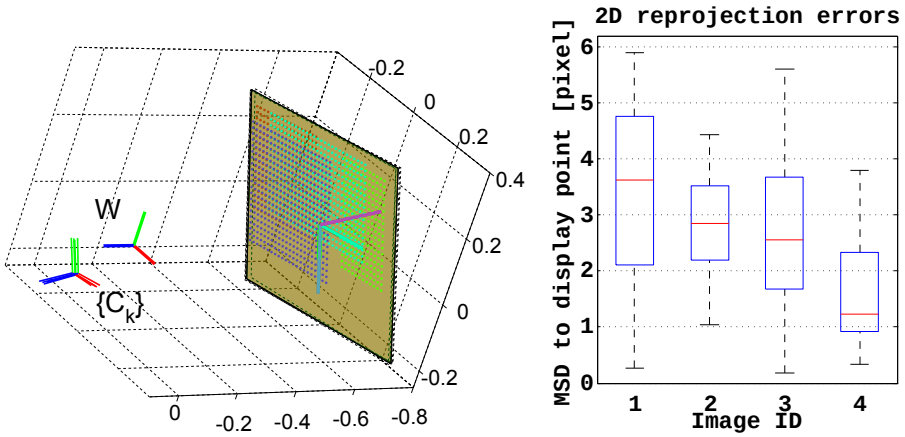


Figure 8.5: Display calibration result. (left) The estimated virtual screen plane and 3D grid points. (right) Reprojection error of each grid points per image. The error is relatively high compared to ordinary camera calibrations that yield subpixel errors in general.

source MATLAB toolbox[2] with printed checkerboard patterns. The poses $\left\{\left(\mathrm{R}_{C_kW}, \mathbf{t}_{C_kW}\right)\right\}$ were estimated via the marker coordinates M. The toolbox computes $\left(\mathrm{R}_{SC_k}, \mathbf{t}_{SC_k}\right)$ up to scale. Fig. 8.5 left shows the calibration result. The reprojection error plot in Fig. 8.5 right shows relatively high error variances compared to standard camera calibrations which yield a sub-pixel accuracy in practice. Nevertheless, we will show later in the experiment that this calibration quality was sufficient to achieve required accuracy.

Calibration of $\left\{\mathrm{K}_{E_0}, \mathbf{t}_{WE_0}\right\}$ for Recycle Setup are described in Chap. 7.

8.5 Experiment

8.5.1 Design of the test process

The test process mostly follows the one in [IK14a]. The main difference is that its data acquisition part is refined so that each data block can be collected individually. We have evaluated the performance of the interaction-free method (*Full*/*Recycle* Setup) compared to SPAAM (*training-error* condition) and to Degraded SPAAM (*test-error* condition). Fig. 8.7 shows an overview of the process.

8.5.1.1 Data Acquisition

Prior to the evaluation, we acquired a series of data sets. Each set consists of 20 2D-3D point correspondences, with each 3D world point having been manually aligned to a 2D point on the screen (aka SPAAM, Fig. 8.6 left). The 3D points were distributed across an area of about 105x66x121 cm^3 (width, height, depth) centered around position (-1, 16, -149) [cm] relative

[2] http://www.vision.caltech.edu/bouguetj/calib_doc/

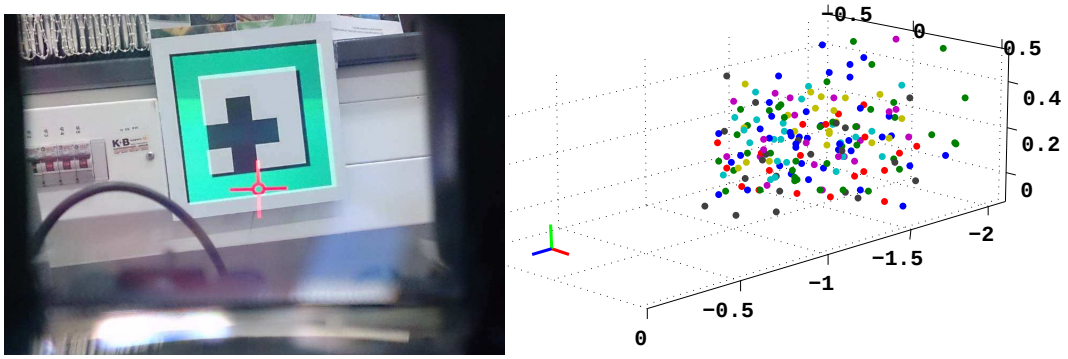


Figure 8.6: Data acquisition: (left) User's view during SPAAM calibration. A virtual 2D red crosshair (2D point) will be matched to the black square marker (3D point). The green frame is a virtual image overlayed for checking the SPAAM quality. (right) Measured 3D points in meter in different colors for different blocks.

to the operator (Fig. 8.6 right). Each 3D point set was also ensured to distribute well in depth for stable SPAAM calibration [Axh+10]. During this process, we also recorded at least 30 eye images per 2D-3D point correspondence. Fig. 8.7 (a) illustrates the step in form of a pink and a green box. The 2D-3D correspondences formed the basis for a SPAAM-based estimation of the projection matrix (blue box). The eye images were used to compute a series of 3D eye positions (orange box). We call such a data collection session a *block*.

A total of N(=9) data collection sessions were performed (Fig. 8.7 (a)). During each session, the HMD was kept as stably as possible on the user's head . After each session, the HMD was taken off from the head and put back on to simulate a degraded calibration situation. These blocks form the ground-truth (GT) data which are the basis for subsequent evaluations of the three evaluation conditions.

8.5.1.2 Data Evaluation Process

***Training-error* evaluation:** For each block among N blocks, a SPAAM calibration is conducted and its quality is evaluated on the same block by using the procedure described in section 8.5.2. At the end, a total of N(=9) error measurement sets were obtained. Fig. 8.7 (b) shows the procedure of this evaluation.

***Test-error* evaluation:** One block is chosen for the SPAAM calibration and the calibration is tested against the rest of blocks –simulating the Degraded SPAAM condition in which a user continues using the same initial display calibration after the display was moved. This yields $N(N-1)$ sets of error measurements. Fig. 8.7 (c) shows this evaluation procedure.

Data acquisition for Full/Recycle methods (Fig. 8.7 (d)) is same as in Sec. 7.5.1. Note that this experiment design is similar to the one in Sec. 7.5.1, yet is more concise and strict. In the previous design, two sequences of consecutive blocks were recorded and the head-display position was

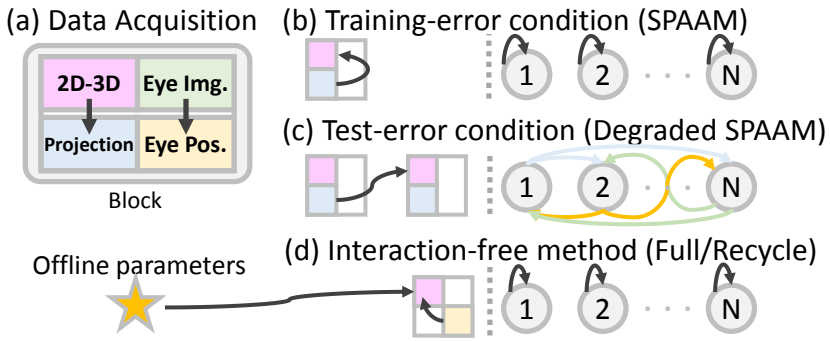


Figure 8.7: Overview of the experiment: (a) data acquisition , (b) training-error condition, (c) test-error condition, and (d) Full-/Recycle-setup conditions. Arrows between block nodes represent each evaluation: the source node is used for computing a projection by each method, and the destination node for evaluating the projection.

assumed to be the same among the blocks in the same sequence. Then, SPAAM setup is evaluated *between* those blocks. Since the assumption is not exactly valid due to the head movement during a SPAAM, evaluation between any two blocks should be treated as Degraded SPAAM setup rather than SPAAM. As same as Sec. 7.5.2, our evaluation has two error measurements: 2D reprojection error and 3D eye position.

Our evaluation aims to determine how well an estimated eye position approximates the true one that existed during the ground-truth data acquisition process. The following indirect and direct error measurements are employed.

8.5.1.3 2D Projection Error:

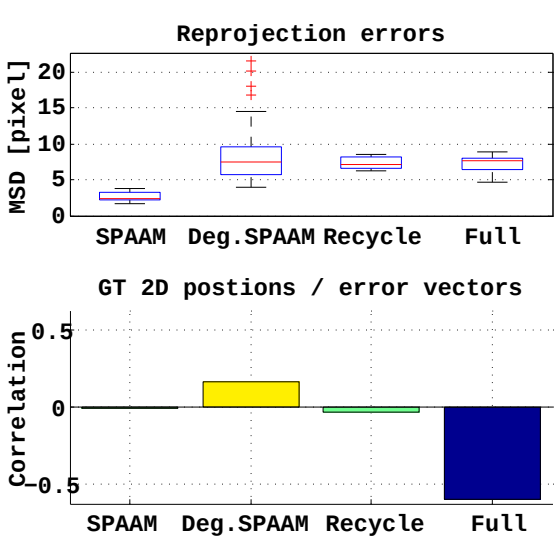
This indirect error is considered as an image-based indicator of the estimation quality of the eye position. Firstly, 3D points of the GT data set are reprojected by the estimated projection matrices. Then the error is computed as the average distance between the reprojected points and the GT 2D points. This error is computed for each pair of estimated projections and the GT data set in the three evaluations of the previous section.

8.5.1.4 3D Eye Positions:

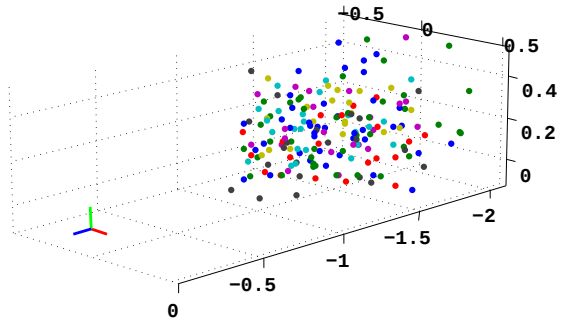
3D eye positions can be decomposed from the projection matrices by SPAAM. Thus, for each block, we compare the positions with the ones given by the 3D eye position estimation.

8.5.2 Performance analysis

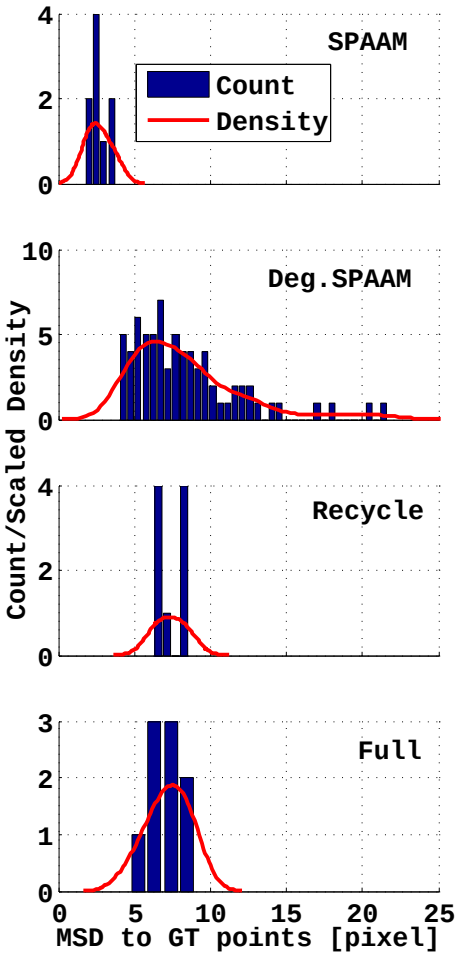
This section analyzes the effect of calibration errors against the final calibration accuracy by both theoretical and actual .



(a) Boxplot/correlation plot of the error



(b) Reprojected/GT 2D points



(c) Error histograms and distributions.

Figure 8.8: Comparison of 2D projection errors. (a) The high correlation seen at Full Setup suggests the existence of bias errors in the calibration procedure (this correlation is indeed observable in (b)). (c) Each distribution is normalized with the area of corresponding histograms for the visualization. The distribution of Degraded SPAAM has a gentle error tail toward error-increasing direction while the other three methods' distributions do not. The density estimations were done by applying the kernel density estimation.

8.5.2.1 Comparison of 2D projection error:

Fig. 8.8 summarizes the result. The boxplot of the reprojection error (Fig. 8.8a top) shows that SPAAM setup achieves the best accuracy. This is expected since the setup learns a projection from a dataset and tested on the same. On the other hand, the other three methods show the almost same average error. For comparison, one might consider applying a statistical testing immediately. Before doing that, we analyze the histograms of the error (Fig. 8.8c).

The error histogram of Degraded SPAAM gives inhomogeneous distribution somewhat similar to the Chi-square distribution. The reason can be explained by considering the re-wearing process during the calibrations. Every time an operator takes the OST-HMD off and on again, most of the time the display was set to almost the same position and few times to the position which is very different from the others. On the other hand, the histograms of ~~IN~~DICA give more homogeneous distributions with lower variances. Thus, when an OST-HMD is in a long-term use, ~~IN~~DICA is more reliable since the homogeneous property can upper bound the error range by the variance of the distributions. Overall, ~~IN~~DICA can be considered to be more stable than the Degraded SPAAM.

Furthermore, the correlation graph in Fig. 8.8a bottom gives another insight. The graph shows correlations between the GT points and the 2D reprojection error vectors –vectors from GT points to their corresponding reprojected 2D points. SPAAM has almost no correlation as DLT method computes an estimate which minimizes the error variance, which means that SPAAM tends to produce a projection over-fit to a given observation.

Degraded SPAAM holds some correlation, this is also understandable since different display positions on different head positions create constant bias errors. The correlation in Recycle Setup is even smaller, this implies that the method has achieved as good accuracy as it can under the combined use of SPAAM. Since Recycle Setup is relying on another projection matrix from SPAAM, the error might also reflects the *test error* of SPAAM that the method would actually achieve with other datasets taken in the same setup.

Notably, Full Setup has huge correlation while maintaining comparable calibration accuracy. This indicates that Full Setup still contains a bias error somewhere in the calibration procedure, thus has a room to further increase the calibration accuracy.

8.5.2.2 Comparison of 3D eye positions:

As reported in [Axh+11; Axh+10; IK14a], eye position estimates by SPAAM tend to have a large variance in z-axis, typically the viewing direction of an eye. Fig. 8.9 shows the estimated eye positions ($\mathbf{t}_{EW}$) in the world coordinates. It shows the similar tendency for SPAAM results while ~~IN~~DICA gives quite stable estimates as similar to Sec. 7.5.2. In the next section, We will provide a reasoning why SPAAM has this error tendency. In short, this is because the z-axis does not impact on the reprojection error as strong as x and y axes do.

There is a shift between the mean y position of SPAAM and that of ~~IN~~DICA (Fig. 8.9 right). This implies that the eye position estimates have a bias error in either or both methods.

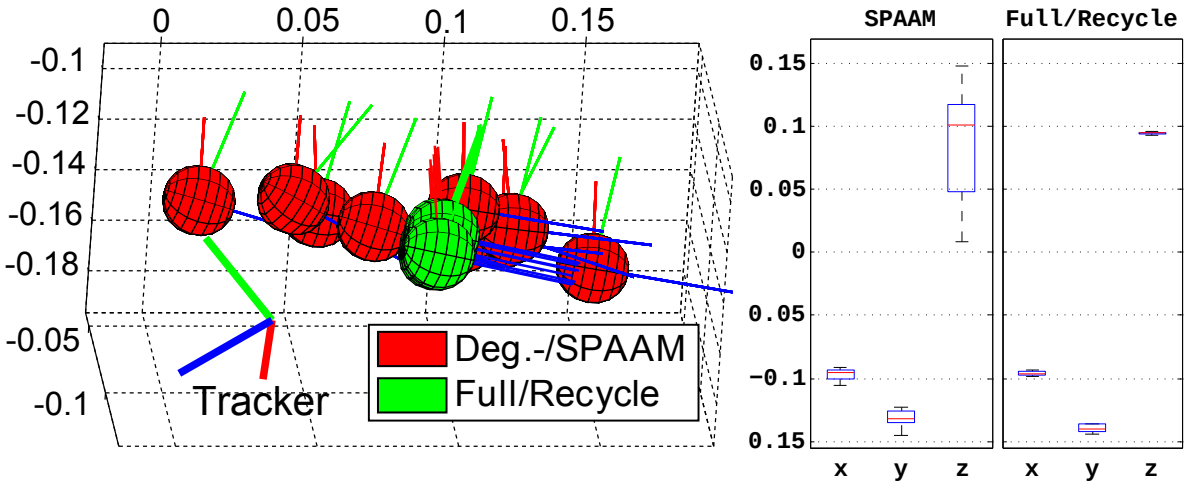


Figure 8.9: Comparison of 3D eye positions. (left) Visualization of estimated eye positions $\mathbf{t}_{EW}$[m]. Eyeballs are drawn with the radius used in the experiment. Orientations of the eyeballs show that of the screen ($\mathtt{R}_{SW}$). (right) $\mathbf{t}_{EW}$ w.r.t xyz axes in [m]. z-axis estimates from SPAAM are instable.

8.5.3 Sensitivity analysis

For sensitivity analysis of the obtained calibration parameters, we follow a general approach described at the section 3.4 in [Hol97]: We deliberately add perturbations to each calibration parameter, and recompute the calibration error for each perturbation to observe how the errors propagate to the reprojection error –the errors of most concern for users. We treat the calibration parameters and the eye position estimated during the experiment as $\lambda^{\star}$

As defined in Eq. 8.18, the sensitivity measurements for Full Setup: $\{e_X\}_X$, $X \in \{\mathbf{a}, \mathtt{R}_{WS}, \mathtt{R}_{WT}, \mathbf{t}_{WT}, \mathbf{t}_{ET}, \mathbf{t}_{WS}\}$ are computed for the given calibration parameters for the error prediction. In Full Setup, λ excludes display pixel center $\{c_x, c_y\}$ since the center is determined by a known display image resolution (and S is defined at this center). Since the unit of the measurement is in the form of [pixel/Y], the measurements are scaled depending on the units of related display parameters as the following: rotation ($\mathtt{R}_{WS}, \mathtt{R}_{WT}$) by 1 with Y=[deg], translation ($\mathbf{t}_{WT}, \mathbf{t}_{ET}, \mathbf{t}_{WS}$) by 0.01 with Y=[m], and pixel scaling $\mathbf{a}$ by 10 with Y=[pixel/m].

The sensitivity measurements for Recycle Setup, $X \in \left\{\mathbf{c}, \mathbf{f}_{E_0}, [\mathbf{t}_{WS}]_z, \mathbf{t}_{WT}, \mathbf{t}_{ET}, \mathbf{t}_{WE_0}, \mathtt{R}_{WE_0}, \mathtt{R}_{WT}\right\}$, and the SPAAM setup, $X \in \{\mathbf{c}, \mathbf{f}_E, \mathbf{t}_{WE}, \mathtt{R}_{WE}\}$, can be obtained in the same manner by setting $\mathtt{P}_{WE}(\lambda)$ with Eq. 8.2 or to $\mathtt{K}_E [\mathtt{R}_{WS} \;\; \mathbf{t}_{WE}]$ respectively. $\mathbf{f}_{E_0}$and $\mathbf{f}_E$ are focal-length vectors of an old and a new projection matrix respectively, and are scaled by 10 [pixel/m]. $\mathbf{c}$ is the image center vector and scaled by 10 [pixel/pixel], namely $e_{\mathbf{c}}$ becomes constant for both SPAAM, and Full/Recyle Setup. For computing the sensitivity measurements of each method, one of the block in Sec. 8.5.1.1 is used. For Full/Recycle Setup, an eye position estimate from the block is also used to compute the sensitivity of $\mathbf{t}_{ET}$. The distribution of 3D points are chosen in the range of the 3D GT dataset.

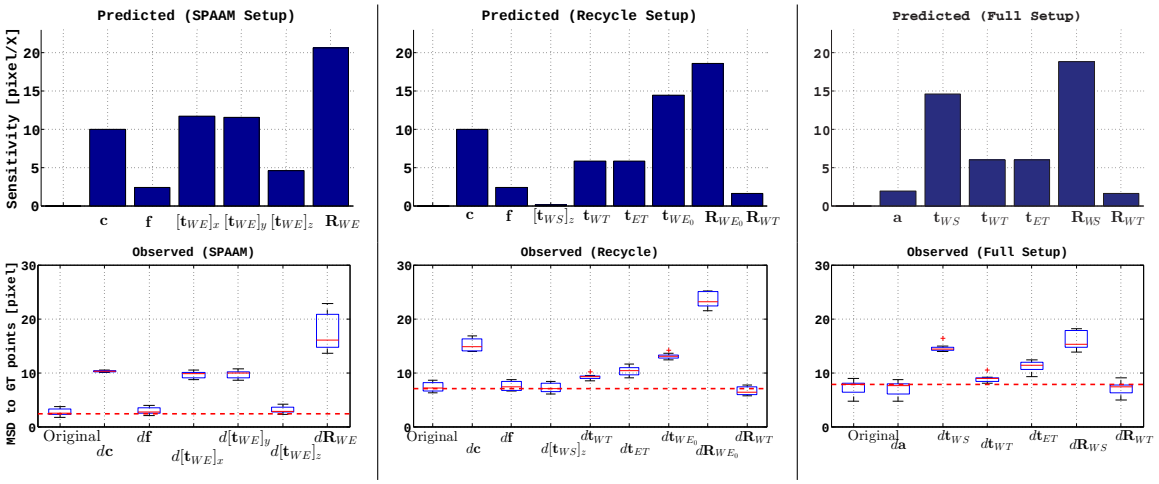


Figure 8.10: Sensitivity analysis against calibration errors. (top row) Predicted errors (bottom row) Observed errors. 'Original' are the original errors without perturbations, and red dotted-lines are baselines drawn at the mean values of the original errors. One can see that the predicted errors coincide with observations from the real datasets, and vice versa.

Fig. 8.10 shows the analysis result. The upper row shows predicted reprojection errors for each calibration parameter, and the lower shows errors actually observed. For SPAAM, we explicitly visualized each axis of $\mathbf{t}_{WE}$ for more detailed analysis. Overall, one can see that the predictions coincide with the observed errors. The figure provides several insights about the three methods including:

(1) SPAAM is relatively insensitive to the estimation error of z-axis of the eye position $\mathbf{t}_{WE}$ compared to that of the other axes. In other words, SPAAM tends to estimate a projection which has bigger variance in the z-axis direction. The rotation $\mathtt{R}_{WE}$ is a dominant parameter in SPAAM, thus if SPAAM gives an accurate projection, decomposed $\mathtt{R}_{WE}$ would be quite reliable. In turn, if Degraded SPAAM is used, HMD should be designed so that users can put it on the same orientation. (2) Both for Recycle/Full Setup, the eye position estimate $\mathbf{t}_{ET}$ is in the second dominant parameter group. (3) Recycle setup is sensitive to old eye poses while not to the display parameter ($[\mathbf{t}_{WS}]_z$) . Thus once an accurate old eye pose (and projection) is given by other methods such as SPAAM, $[\mathbf{t}_{WS}]_z$ does not require strict accuracy. This can be the reason why the evaluation of this setup by [IK14a] worked well with a rough $[\mathbf{t}_{WS}]_z$ which was measured by hand. (4) Full Setup is especially sensitive to the calibration quality of the virtual screen (S) relative to the world. This supports that the quality of our marker-based display calibration was well enough to be compared to SPAAM methods.

8.6 Discussion

Throughout the experiment, Full (and Recycle) Setup achieved more stable and comparably accurate calibration quality against Degraded SPAAM. The analysis of the projection errors tells

that Degraded SPAAM, a setup where a user compromises on a default or old calibration setting, is not a preferable solution for the long-term use of OST-HMDs; it is hard to guarantee maximum error bound and is not easy to predict how worse it can be. From the correlation analysis, the Recycle Setup seems to have achieved the ideal accuracy given that the partial calibration parameters are given by SPAAM. Thus, replacing SPAAM with camera-based methods, e.g. [GFG08], is a possible direction further improving the performance of the setup. On the one hand, Full Setup still shows potential to achieve better performance once the following error source is identified and eliminated.

It is still unclear why the Full Setup has: (a) huge correlation to the GT points and yet has (b) small reprojection error. We suspect the eye tracking as the cause because of two reasons observed: the existence of the offset in the 3D eye position estimates for (a) (Fig. 8.9), and the low error sensitivity of eye position found in the sensitivity analysis for (b) (Fig. 8.10). As mentioned in the discussion in Sec. 7.6, the source of the offset can be due to the discrepancy between the eye model and the real eyeball. One can explore these issues by, e.g., installing the eye tracker in a different configuration to see the change of correlation, improving the tracking method, and so on.

The two setups have a clear difference in the number of parameters to be estimated. By recalling Eq. 8.3 and 8.2, one can derive that Recycle/Full Setup yield 16/19 DoF respectively despite the fact that they represent the same projection matrix (Fig. 8.1). Recycle Setup aims to model the system more concisely and Full Setup does exactly. Each setup requires different precalibration procedures with different intricacy, thus one should consider the overall complexity of the calibration flow when applying ~~IN~~DICA.

The sensitivity analysis (Fig. 8.10) gave various insight about the use of the three calibration methods. However, strictly speaking, the result is valid only for the particular OST-HMD configuration we tested –an indoor setup with a configuration where a world camera is set on the top of the HMD and an eye tracker on the bottom. Different OST-HMDs do yield radically different configurations, and different AR applications (and FoV of HMDs) do yield different 3D point space of interest. Thus the result itself might not directly be applied to quite different scenario such as outdoor setups or HMDs with large FoV, yet one can conduct their own analysis based on our formulation once they identified their current configuration or have HMD design at hand.

The proposed sensitivity-analysis framework has potential to impact on designing an optimal OST-HMD configuration. Searching the display parameter domain with the sensitivity measurement might give the optimal configuration for a certain application. For instance, our informal investigation shows that Full Setup becomes less sensitive to the eye-position error when the eye tracker and the world camera are *straightly aligned on the eye axis*, which requires half mirror optics[MaN12]. Contrarily, this setup becomes more sensitive to the virtual screen pose error. This setup would be preferable when an OSD-HMD can be finely calibrated in a factory, then used by variety of people with less accurate eye tracking.

8.7 Summary

We conduct intensive analysis of the interaction-free OST-HMD calibration method. The evaluation demonstrates the Full Setup performs as accurately as the Recycle Setup under the use of a marker-based display calibration. Furthermore, we formulates an error sensitivity analysis for both SPAAM and the interaction-free method by deriving the Jacobian of reprojection error over eye positions and display parameters. The analysis formulation is then investigated on an HMD with justification of the theory by the real measurements, which brings various insight including: high sensitivity of the virtual screen parameters, middle sensitivity of the eye position, the reasoning of SPAAM's error tendency etc.

Part III

Distortion Correction of OST-HMDs

Through the previous part, we have broken the OST-HMD system down into an eye and HMD part, and modeled them separately for the spatial calibration. The eye part involves modeling an eye optics and locating the eyeball with respect to the display coordinate system [IK14a; IK14b; Plo+15]. The HMD part involves modeling the image screen. The previous chapters estimated the screen parameters by measuring an eye and/or an OST-HMD screen by a camera. This part focuses on the HMD model even deeper. We particularly focuses on optical distortions in the display system in a camera-based approach.

We first introduce a non-parametric distortion model for display optics (Chapter 9.4.2) and extend the model to image distortions (Chapter 10).

9 Light-field Correction

This section is based on the work that the author presented at IEEE VR 2015 and published in IEEE TVCG journal in 2015 [IK15a].

9.1 Introduction

Although the automated method we proposed in Chap. 7 frees users from manual calibration, our calibration results so far still contain systematic errors due to simplistic eye-HMD modeling ([IK14b], Fig. 6.1). Chapter 8 presented a sensitivity analysis of a number of calibration and registration parameters, indicating which of them are most critical for reducing systematic errors. Yet, they neglected to model an important fact of the system: optical elements of OST-HMDs distort incoming world-light rays before they reach the eye (Fig. 9.1), just as corrective glasses do.

In common OST-HMD designs, the light rays of an image that arrive at the user's eye are collimated, showing images at virtual infinity [HB11] or are perceived as a 3D planar screen floating mid air in practice. While this property is desirable to align the image and the user's eye easily, it requires curved optical elements which inevitably distort light rays incoming from the world [Hol97]. Since users see a distorted world through these elements, ignoring the distortion degenerates the registration quality.

This section proposes a method to compensate the world light-ray distortion caused by OST-HMDs' optical elements. The method estimates a 4D-to-4D mapping between the original light field and the light field distorted by the optics in offline. After computing the mapping, the method compensates distortions on the virtual screen with respect to the eyeball center. We first validate the compensation method in the camera-based OST-HMD setup and show that the method significantly reduces the calibration error. We then further evaluate our method in an actual interaction-free OST-HMD calibration setup with a real user involved. The result shows that the compensation reduces the systematic error, and again significantly improves the overall calibration quality.

Contributions As a summary, our contribution of this section includes the following:

- We provide a formulation of the lens distortion caused by the optical elements of HMDs that distort the transmission of light rays.
- We provide an offline calibration procedure to learn a mapping which corrects the light-field distortion. The procedure is required only once per HMD.

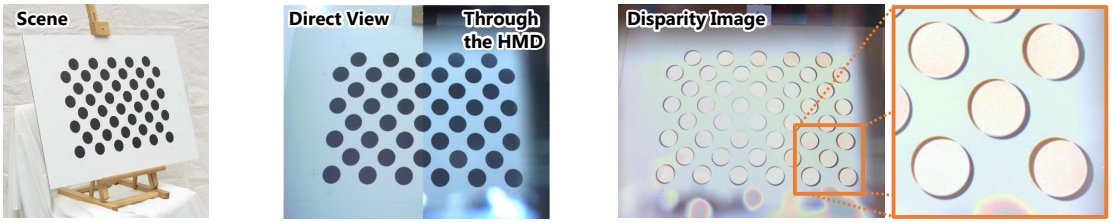


Figure 9.1: An illustration of our problem: an optical distortion caused by an optical element of an OST-HMD. (a) A target in the scene. (b) A direct view by a camera and a view through the OST-HMD from the same viewpoint. (c) The absolute difference of the two images (intensity is inverted). (d) A zoomed part of (c). The distortion is radial in appearance. Note that distortion shapes can vary depending on the position of the view point and the HMD. Fig. 9.2 shows a corresponding HMD setup.

- We demonstrate that applying the correction method reduces the systematic error which has existed in conventional camera-based and user-based calibrations and significantly increases the calibration accuracy.

9.2 Related Work

The following subsections present an overview of topics related to the correction of optical distortion effects in imaging systems.

9.2.1 Spatial calibration of OST-HMDs revisited

Existing calibration methods model an eye-HMD vision system, such as ours in Fig. 9.2, as an off-axis pinhole camera where the virtual screen S of the display is the image plane, and the eyeball center E is the camera center (Fig. 6.1). The model is represented by a 3-by-4 projection matrix P_{WE} which projects a 3D point from the world W to a user view on the screen S.

Manual methods, such as SPAAM, require at least six 3D-2D correspondences to estimate the matrix (Fig. 6.1 top right). On the other hand, an automated method such as ~~I~~NDICA (Fig. 6.1 bottom left and right) does not require such user-based alignment. Instead, it tracks the eye position and computes the projection matrix together with some precalibrated parameters.

The automated method actually has two formulations. The formulation of **Recycled ~~I~~NDICA** (Fig. 6.1 bottom left) reuses an old projection matrix from a prior eye position E_0 and updates this old projection matrix by taking the new eye position E into account. The formulation of **Full ~~I~~NDICA** (Fig. 6.1 bottom right) calculates the current projection according to precalibrated HMD parameters such as the relative pose between an eye-tracker T and a world camera W, the pose of the HMD's virtual screen S with respect to the world camera, and the apparent size (α_x, α_y) of the virtual screen pixels [IK14b]. This formulation has 17 degree of freedom (DoF) including the eye position.

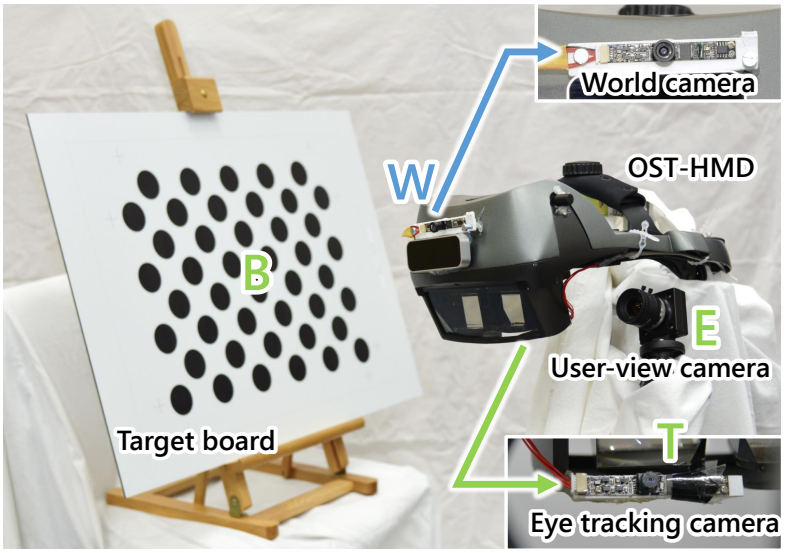


Figure 9.2: Hardware setup. It shows our OST-HMD (nVisor ST60), the world camera W attached on the HMD, the eye tracking camera T fixed beneath the left optical element of the HMD, the user-view camera E, and a target board for calibration experiments.

Although ~~IN~~DICA is suitable for practical use, it is prone to contain systematic errors possibly stemming from its simplified eye-HMD system modeling ([IK14b], Fig. 6.1). The model mainly consists of two independent parts: eye-dependent and HMD-dependent parameters. The former relate to anatomical eye parameters such as the eye-ball radius. The latter relate to optical characteristics of the optical components of an HMD such as the virtual screen pose and optical distortion. This distortion effect is the prime issue that we investigate in this section.

9.2.2 Undistortion for cameras

As mentioned in the previous section, existing OST-HMD calibration methods assume the eye-HMD system to be an (off-axis) pinhole camera [Axh11]. The model is commonly used in computer vision, where lens distortion is one of the most essential problems [Stu+11]. Parametric distortions in the 2D image space, e.g. radial and tangential distortions, affect ordinary lenses the most, and thus are commonly sufficient to perform image undistortions [Zha00; DF01]. For heavily-distorted lenses, such as fish-eye lenses or catadioptric optics, some approaches employ non-parametric distortion models [HK07; QM95].

An important difference between conventional cameras and eye-HMD systems is that camera models may assume that the camera center with respect to camera's image plane is static, while HMD models must expect that the center, i.e. the user's eyeball center, is dynamic with respect to the image screen of the OST-HMD. Therefore, to undistort an eye-HMD system, it is necessary to estimate distortions relative to the possibly moving eyeball center.

9.2.3 Undistortion for HMDs

Vincent and Tjahjadi [VT05] propose a non-parametric approach for Head-Up Display (HUD) calibration. Their method undistorts images by first estimating a homography between ideal and distorted grid images and then computing further offsets per grid by fitting a B-spline surface to the data to compute a non-parametric undistortion model. While their method can handle complex distortions, such as those caused by HUD optics, it needs to re-learn the distortion parameters whenever the eyeball center moves.

A key observation of these undistortion methods is that they only estimate a 2D mapping between a distorted and the original image. Given a camera, a 2D mapping is only valid for one camera center known beforehand. Unlike in cameras, the eyeball center changes dynamically in an eye-HMD system. A naive way to apply these methods to the eye-HMD system is to estimate a 2D mapping once for a predefined eyeball center, and then reuse the mapping for different users [VT05]. Obviously, this does not assure that the learned mapping undistorts images properly for arbitrary eyeball centers. A second possible option would be to learn several mappings at different eyeball centers, then select a mapping of a predefined eyeball center nearest to the current position at runtime. This approach might work more accurately than the first one, yet again it does not produce a correct undistortion for every new eyeball center. The third approach would be to further learn a regression function of those 2D mappings, namely learn a super function which returns a 2D mapping given an eyeball center. This approach assumes that two 2D mappings of two eyeball centers close to each other are similar in some sense. This assumption requires a careful definition of the distance between the 2D mappings used, e.g. a distance of radial distortion coefficients, factorial distortion parameters, etc.

In the following, we extend the last idea to the 4D domain – the light field space. Remember that we are concerned with the distortion caused by an optical element of an OST-HMD. Physically speaking, the distortion is due to the fact that the optical element distorts all incoming light rays from the scene passing through the element. Under the assumption that the optical element smoothly distorts the light rays, i.e. similar incoming light rays are distorted similarly, it is our problem to find a 4D-to-4D mapping between the original light field and the distorted light field. Once the mapping is given, we can readily create a 2D mapping for a given eyeball center.

9.2.4 Light-field representation

A light field or Lumigraph is a 4D function representing the light rays passing through a 3D space (Fig. 9.3 bottom) [Gor+96; LH96]. The representation has been used for rendering photorealistic visual effects such as reflection and refraction in computer graphics [Hei+99], and applied to model light-field displays [Jon+07; Hua+14] and light-field cameras [Ng+05] in computational photography.

9.2.5 Non-parametric regression

We use non-parametric regression to estimate the mapping between light fields. In machine learning, regression is one of the most fundamental methods. Given training data $\{(x_k, y_k)\}_k$ with

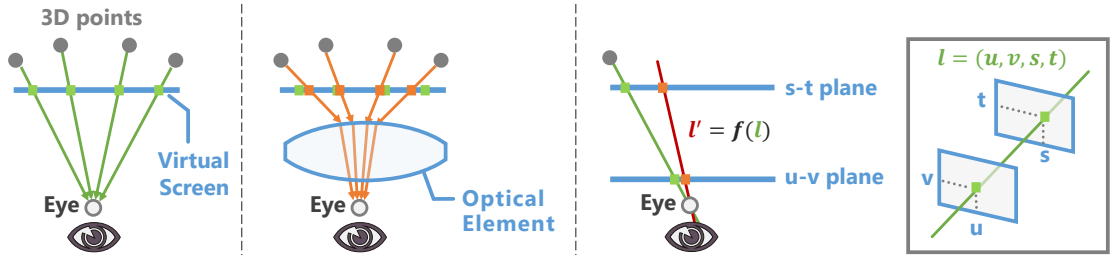


Figure 9.3: Schematic drawing of the real-world distortion effect caused by the optical element of an OST-HMD. (Top left) Light rays from a user's eye to the world. The rays intersect with the image plane of the virtual screen. (Top right) The optical element of the display distorts the rays. It thus also shifts the pixel positions of the intersections. (Bottom) Modeling of the distortion as a 4D-to-4D mapping between light fields. We use the 4D Lumigraph parameterization: (u,v,s,t). Note that a distorted light ray l' is modeled to pass through the eye center and a shifted intersection pixel position.

k samples, a regression method finds a function $y = f(x)$ which explains the dataset best in a statistical sense. If candidates of f is limited within a function class $g(x \mid \theta)$ with parameters θ, then the problem is called parametric regression. Image undistortion based on a radial distortion model is an example of this problem. On the other hand, if f is estimated locally based on the data itself, it is called non-parametric regression. For example, the B-spline regresses a function f by tuning the amplitude of each basis function which is uniformly distributed in the data domain. The so-called kernel regression method is similar to B-splines. Yet, it is more concise in the sense that the method regresses f by radial basis functions located at each data point [SS01].

9.3 Method

This section explains the spatial calibration of the eye-HMD system and the distortion estimation of the optical elements of the display. See Sec. 7.3 and 8.3.1 for the detail of the automated calibration method ~~I~~NDICA.

9.3.1 Distortion estimation for OST-HMD optics

An optical element of an OST-HMD distorts light rays incoming from the world to an eye, i.e. each light ray is mapped to a distorted light ray (Fig. 9.3). Our goal is to obtain this mapping $f : \mathbb{R}^4 \rightarrow \mathbb{R}^4$ between an original light field and a distorted light field after distortions by the optical element. We use the 4D Lumigraph parameterization by assigning point pairs on two planes denoted as *u-v plane* and *s-t plane* (Fig. 9.3 bottom).

9.3.1.1 Light Field Computation in OST-HMDs

In this section, we first formulate a light ray passing through a plane in a coordinate system (Fig. 9.4 top). We then apply the formulation to our OST-HMD calibration setup, and define original

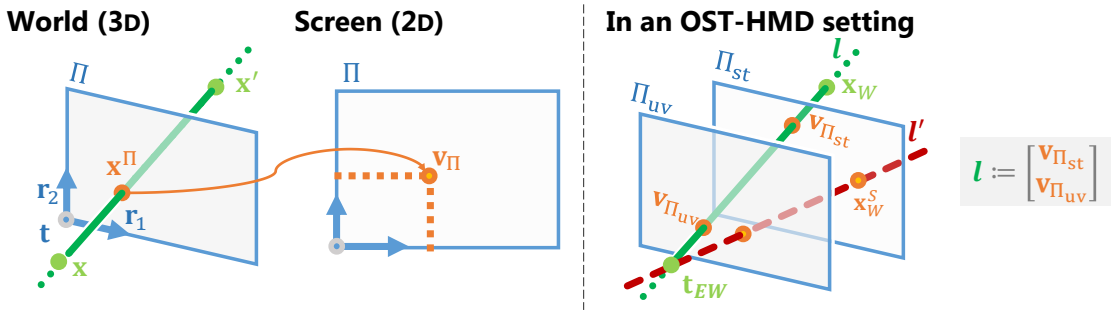


Figure 9.4: Schematic diagram of the definitions of the light field with respect to the HMD coordinate system. (Top) A 3D point $\mathbf{x}^{\Pi}$ is the intersection between a 3D plane Π and a 3D line passing through 3D points $\mathbf{x}$ and $\mathbf{x}'$. In the plane coordinate system, $\mathbf{x}^{\Pi}$ can be defined by a 2D point v_{Π}. (Bottom) Light rays in our OST-HMD setup in more detail, using the notation of Sec. 9.3.1.1.

and distorted light rays (Fig. 9.4 bottom).

Given a position $\mathbf{t}$ and an orientation $\mathtt{R} := \begin{bmatrix} \mathbf{r}_1 & \mathbf{r}_2 & \mathbf{r}_3 \end{bmatrix}^{\mathrm{T}}$, $\mathbf{t}$ and the first two row vectors $\mathbf{r}_1$ and $\mathbf{r}_2$ span a 3D plane as $\Pi(\mathtt{R},\mathbf{t}) := \{a\mathbf{r}_1 + b\mathbf{r}_2 + \mathbf{t} \mid a,b \in \mathbb{R}\}$. A light ray passing through two 3D points $\mathbf{x}$ and $\mathbf{x}'$ intersects with the 3D plane Π at $\mathbf{x}^{\Pi}$ as follows (Fig. 9.4 Left):

$$\mathbf{x}^{\Pi}(\mathbf{x},\mathbf{x}') := \mathbf{x}' + \frac{(\mathbf{t}-\mathbf{x}')^{\mathrm{T}}\mathbf{r}_3}{(\mathbf{x}-\mathbf{x}')^{\mathrm{T}}\mathbf{r}_3}(\mathbf{x}-\mathbf{x}') \in \mathbb{R}^3. \tag{9.1}$$

Note that $\mathbf{x}^{\Pi}$ is commutative, i.e. $\mathbf{x}^{\Pi}(\mathbf{x},\mathbf{x}') = \mathbf{x}^{\Pi}(\mathbf{x}',\mathbf{x})$. The 3D intersection point is represented by a real-scale 2D vector in the plane's coordinate system as:

$$v_{\Pi}(\mathbf{x},\mathbf{x}') := \begin{bmatrix} \mathbf{r}_1^{\mathrm{T}} \\ \mathbf{r}_2^{\mathrm{T}} \end{bmatrix} (\mathbf{x}^{\Pi}(\mathbf{x},\mathbf{x}') - \mathbf{t}) \in \mathbb{R}^2. \tag{9.2}$$

Now consider our interaction-free OST-HMD calibration setup. We treat the HMD coordinate system as the world W. Let the virtual screen orientation and position be $\mathtt{R}_{SW} = \mathtt{R}_{WS}^{\mathrm{T}}$ and $\mathbf{t}_{SW} = -\mathtt{R}_{WS}^{\mathrm{T}}\mathbf{t}_{WS}$ respectively, and let $\mathbf{t}_{SW}^{0} := [[\mathbf{t}_{SW}]_x\ [\mathbf{t}_{SW}]_y\ 0]^{\mathrm{T}}$. Then, we define the s-t and u-v plane as $\Pi_{st} := \Pi(\mathtt{R}_{SW},\mathbf{t}_{SW})$ and $\Pi_{uv} := \Pi(\mathtt{R}_{SW},\mathbf{t}_{SW}^{0})$ respectively. Given a point $\mathbf{x}_W$ in W, we define a light ray l passing through $\mathbf{x}_W$ and the eyeball center $\mathbf{t}_{EW} = -\mathtt{R}_{WS}^{\mathrm{T}}\mathbf{t}_{WE}$ as

$$l_k := l(\mathbf{t}_{EW},\mathbf{x}_W,\mathtt{R}_{SW},\mathbf{t}_{SW}) := \begin{bmatrix} v_{\Pi_{st}}(\mathbf{t}_{EW},\mathbf{x}_W) \\ v_{\Pi_{uv}}(\mathbf{t}_{EW},\mathbf{x}_W) \end{bmatrix} \in \mathbb{R}^4. \tag{9.3}$$

Equation 9.3 represents light rays from the eyeball center when there is no distortion induced by the optical element. If we have such a distortion, then $\mathbf{x}_W$ matches, from a view point, to a 3D screen point $\mathbf{x}_W^S$ which is slightly shifted from $\mathbf{x}^{\Pi_{st}}(\mathbf{x},\mathbf{x}') \neq \mathbf{x}_W^S$. We define the distorted light ray as:

$$l'_k := l\left(\mathbf{t}_{EW},\mathbf{x}_W^S,\mathtt{R}_{SW},\mathbf{t}_{SW}\right) \in \mathbb{R}^4. \tag{9.4}$$

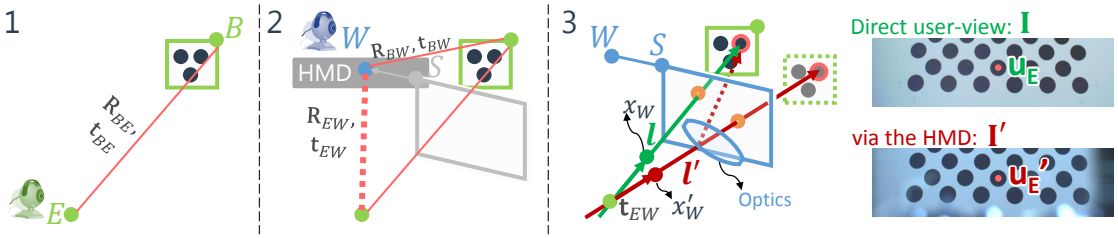


Figure 9.5: Light-field collection overview. It describes each step of the collection procedure in Sec. 9.4.2.

Finally, given a set of the light-ray pairs $\{(l_k, l'_k)\}_k$, our objective is now to learn the regression f which returns a distorted light ray given an original light ray so that the output explains the data set well.

9.3.1.2 Non-parametric Regression for the Distorted Light Field

Since our dataset $L := \{(l_k, l'_k)\}_k$ has multivariate (4D) output, we learn four kernel regression functions (See. Sec 5.4.2) for each output dimension of the distorted light ray l'. the ease of notation, we use $\mathbf{f}$ for representing the bundle of the four functions so that we can write $l' = \mathbf{f}(l)$. Note that, by switching the input and output, we can also learn an *undistortion* mapping $l = \mathbf{f}^{-1}(l')$.

In general, the performance of the kernel regression depends on the parameters of the kernel function and of the regularizer, i.e. σ and λ [SS01; TFM07]. We use a standard cross-validation technique [Sto74] to automatically choose those parameters. Another pragmatic step for stable estimation is to normalize the training data so that they have zero mean and identity covariance matrices. If we apply this normalization technique to the training data, we also need to un-/normalize the out-/input by the mean and variance of the training data used for the regression.

9.3.1.3 Rendering with a Distorted Light Field

Now, we are ready to correct the optical distortion in AR visualization. In the original Full ~~IN~~DICA setup [IK14a; IK14b], we would project a 3D point $\mathbf{x}_w$ on the known display image plane by a projection matrix $\mathtt{P}_{we}(\mathbf{t}_{we})$. Instead, we now first convert $\mathbf{x}_w$ to a light ray $l(\mathbf{t}_{ew}, \mathbf{x}_w, \mathtt{R}_{sw}, \mathbf{t}_{sw})$, and find a distorted ray $l' = \mathbf{f}(l)$. Then, we compute a distorted 2D pixel point $\mathbf{u}'$ as:

$$\mathbf{u}' = \begin{bmatrix} \alpha_x [l']_s + c_x \\ \alpha_y [l']_t + c_y \end{bmatrix}, \tag{9.5}$$

where $[\cdot]_s$ and $[\cdot]_t$ denote functions that return s and t elements of an input light ray respectively.

Note that we can define another 2D pixel point $\mathbf{u}$ from l, which represents the same pixel point that the conventional projection matrix gives. Thus, if we collect all pairs of $(\mathbf{u}, \mathbf{u}')$ corresponding to light rays that pass through each image pixel and the eye center, it generates a look-up table

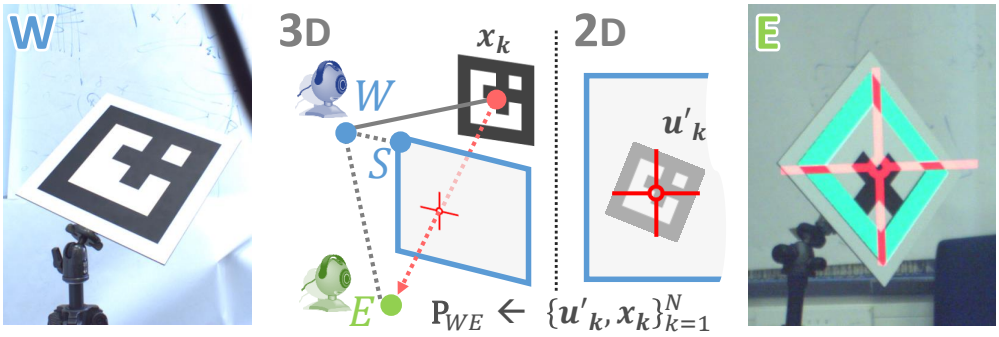


Figure 9.6: Camera-based SPAAM setup. From left to right: 3D reference marker seen by the world camera, a schematic illustration of SPAAM; a 2D crosshair is matched to the marker in the user-view camera E, and a calibration result where a 2D virtual green frame is overlaid on the board using the estimated calibration result.

representing a *2D* distortion map – a common representation of lens distortions in computer vision.

9.4 Technical Setup

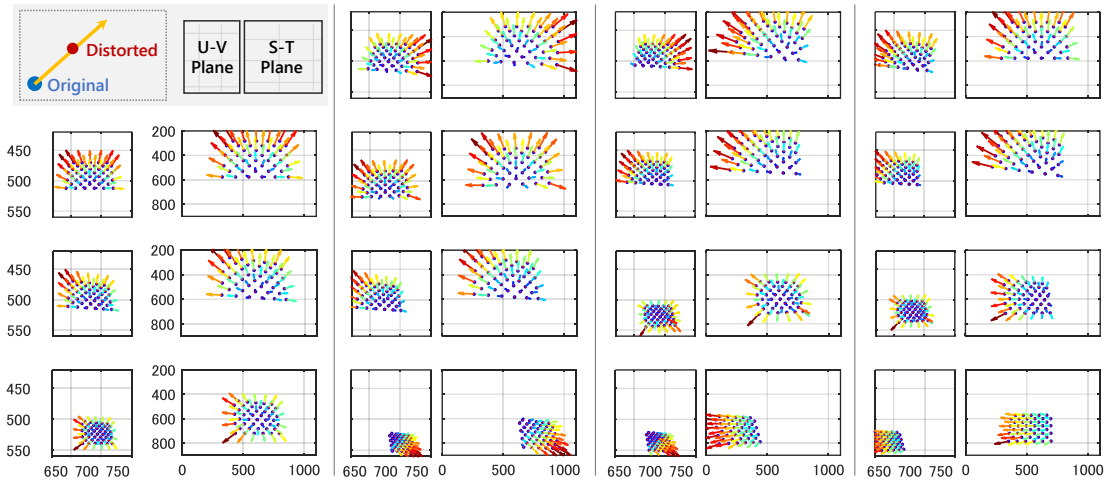
We explain our hardware setup, as well as an offline procedure for collecting original and distorted light fields for an HMD.

9.4.1 Hardware setup

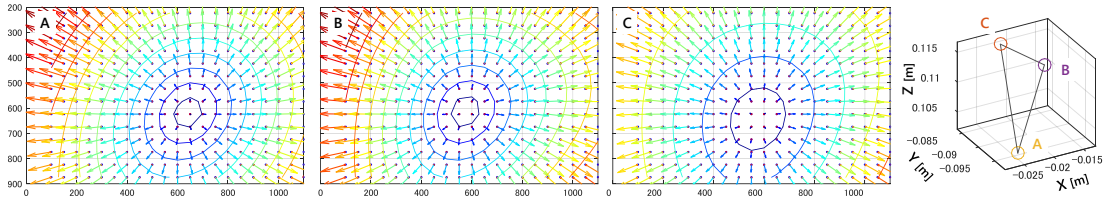
We have built an OST-HMD system equipped with an eye tracker as described below and in Fig. 9.2. We use an nVisor ST60 from NVIS – an OST-HMD with 1280×1024 resolution. The left-eye display is used for the current setup. An outward looking camera, a Delock USB 2.0 Camera with a 64-degree lens, serves as the world camera W. For the eye tracker T, another Delock Camera with a 55-degree lens is used. These cameras provide 1600×1200-pixel video and are attached to the HMD. The eye tracker is positioned at the bottom of the left display lens of the HMD. The default focal length of its fixed-focus lens is manually adjusted and fixed to a suitable length.

We calibrated the eye-HMD system as described in [IK14b] to obtain offline parameters (Sec. 8.3.1): pose between the HMD and the eye-tracking camera $(\mathtt{R}_{WT}, \mathbf{t}_{WT})$, pose between the HMD and its virtual screen $(\mathtt{R}_{WS}, \mathbf{t}_{WS})$, and the scaling vector $\mathbf{a}$ [pixel/meter].

For a camera-based SPAAM experiment and for the light field estimation, we replace the human eye E by a camera. We use the UI-1240ML-C-HQ camera of iDS's together with an 8mm C-mount lens. The camera provides 1280×1024 images.



(a) Measured light fields. We visualize u-v and s-t planes of each light field $\{L_i\}$ measured from different viewpoints. Colored arrows originate from original image points and pass through corresponding distorted points. Their color-coded length is proportional to the distance between the point pairs. Distortions are mainly radial but their shape changes for each viewpoint.



(b) Testing the learned regression with three artificial eyeball positions different from those in the training dataset. Three color plots show the original and distorted light rays on the s-t plane. The 3D plot on the right visualizes the three eye positions $\mathbf{t}_{EW}$ used in this example. The positions are within a 1.5 cm^3 space. Different eyeball positions result in different distortions.

Figure 9.7: Light-field mapping computation. (a) Measured light fields. (b) Estimated distortion maps.

9.4.2 Light field collection

This section describes our offline calibration procedure for collecting training data of original and distorted light fields. For learning the regression function $l' = \mathbf{f}(l)$, we collect a set of original and distorted light ray pairs: $L_i = \left\{(l_{ik}, l'_{ik})\right\}_k$ for a number of viewpoints i. Measurements from different viewpoints are necessary so that the regression can cover various eye positions in applications. Our collection procedure requires the following (Fig. 9.2): a user-view camera E, an OST-HMD with a world camera W, and a fiducial target board B fixed in a scene. We assume that the cameras and the OST-HMD's virtual screen are already calibrated. The procedure is as follows (Fig. 9.5):

1. Place the user-view camera E and the 3D target B in the scene, and let the camera capture a direct-view image $\mathbf{I}$. Then from $\mathbf{I}$ and the camera's intrinsic matrix $\mathtt{K}_E$, estimate the pose of the target as $(\mathtt{R}_{BE}, \mathbf{t}_{BE})$.

2. Place the OST-HMD in front of the user-view camera, and let the camera capture a distorted-view image $\mathbf{I}'$. Let the world camera W capture the 3D target and estimate the pose $(\mathtt{R}_{BW}, \mathbf{t}_{BW})$. Using this pose and $(\mathtt{R}_{BE}, \mathbf{t}_{BE})$, compute $(\mathtt{R}_{EW}, \mathbf{t}_{EW})$.

3. From $\mathbf{I}$ and $\mathbf{I}'$, extract corresponding 2D points $\mathbf{u}_E$ and $\mathbf{u}'_E$. Then compute their 3D position in W as
$$\mathbf{x}_W := \mathtt{R}_{EW}\mathtt{K}_E^{-1}\widetilde{\mathbf{u}_E} + \mathbf{t}_{EW},\ \mathbf{x}'_W := \mathtt{R}_{EW}\mathtt{K}_E^{-1}\widetilde{\mathbf{u}_E}' + \mathbf{t}_{EW}, \tag{9.6}$$
where $\widetilde{\cdot}$ represents homogeneous vectors. Finally, compute an original light ray $l := l\left(\mathbf{t}_{EW}, \mathbf{x}_W, \mathtt{R}_{SW}, \mathbf{t}_{SW}\right)$ and its distorted $l' = l\left(\mathbf{t}_{EW}, \mathbf{x}'_W, \mathtt{R}_{SW}, \mathbf{t}_{SW}\right)$.

As the result, we get a set of the light-ray pairs $L_i = \left\{(l_{ik}, l'_{ik})\right\}_k$.

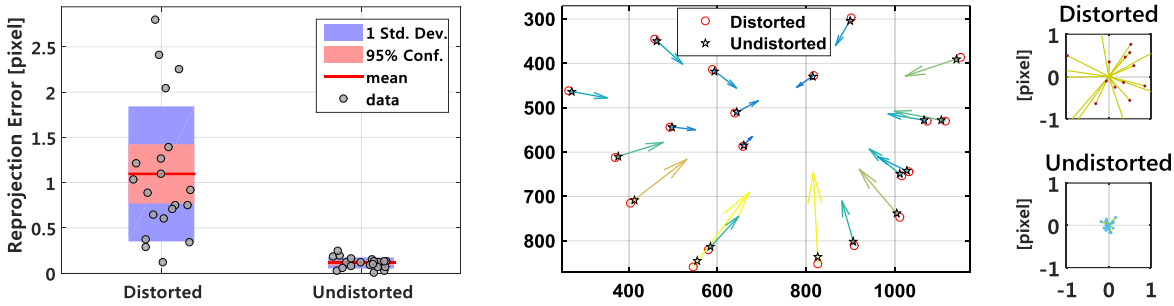
In our experiment, we used a calibration board with a 4-by-11 asymmetrical circle grid, and measured the distortion from 19 different view points, $\mathbf{t}_{EW}$. This yielded total 836 ($= 4 \times 11 \times 19$) light ray pairs. We have not analyzed how many viewpoints are sufficient to estimate the mapping correctly.

9.5 Experiment

We conducted two calibration experiments: a camera-based OST-HMD calibration experiment and a user-based calibration. The camera-based calibration purely assesses the validity of our distortion correction method, and the user-based calibration further demonstrates its performance in a realistic OST-HMD calibration with a real user. Before going into the calibration experiments, we first elaborate the result of the light-field distortion learning.

9.5.1 Distortion model learning

After we collected a training data set $\{L_i\}_i$ as explained in Sec. 9.4.2, we learned the light-field mapping function $\mathbf{f}: \mathbb{R}^4 \to \mathbb{R}^4$ through the kernel regression method (Sec. 9.3.1.2). We used a



(a) Distorted 2D points ($\{\mathbf{u}_k\}$ in Fig. 9.6) observed during the camera-based SPAAM and their undistorted 2D points. Colored arrows originate from the distorted points and go through the undistorted points. The arrows' lengths are proportional to the distance of each point pair.

(b) Reprojection errors of original (Distorted) and corrected (Undistorted) datasets. The corrected dataset results in a smaller reprojection error than the original. The difference of the errors was statistically significant ($p \approx 8.28\text{e-}07 \ll 0.05$, two-sample t-test).

(c) Distributions of reprojection error vectors aligned to the origins.

Figure 9.8: Camera-based calibration experiment. (a) Observed and corrected 2D points. (b) Reprojection errors. (c) Error vectors.

Matlab implementation of the regression[1], which includes the cross validation step. We chose n_b (=100) random light rays from the training data for the basis functions in each regression. Note that we can also compute the inverse map $\mathbf{f}^{-1}$ by switching the input and output.

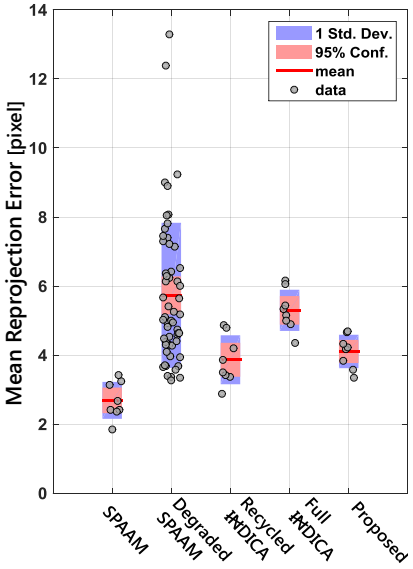
Figure 9.7 summarizes the result. Fig. 9.7a visualizes the u-v and s-t planes of several L_i among the 19 sets. The figure illustrates the difference between each corresponding light-ray pair (l_{ik}, l'_{ik}) by drawing direction vectors from original to distorted 2D points on the planes. The lengths of the vectors are proportional to their point pairs' distances, for intuitive understanding. Since the virtual screen defines the s-t plane, the s-t plane figures show actual distortions observed by each view point. The visualizations show that different distortions occur at each viewpoint. Overall, the distortions are concentric similar to radial distortions.

Figure 9.7b tests the obtained regression function for three different view points – eyeball positions that moved within a 1.5 cm^3 space (the rightmost column). The left three columns demonstrate that the regressed function outputs different distortions for each new eye position.

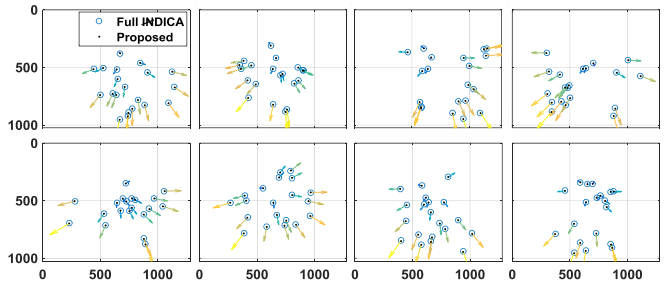
9.5.2 Distortion correction for camera-based calibration

Recall that eye-HMD system relies on the eye model and the OST-HMD model. We focus on improving the HMD model by taking the distortion from optical elements into account. Therefore, we first separate the eye part, another source of systematic error, for the primary validation of the distortion compensation method. As in the work by Gilson et. al. [GFG08], our procedure uses a user-view camera E (Fig. 9.6) instead of human operators. We next evaluate our distortion compensation method in a camera-based manual calibration (SPAAM). The user-view camera

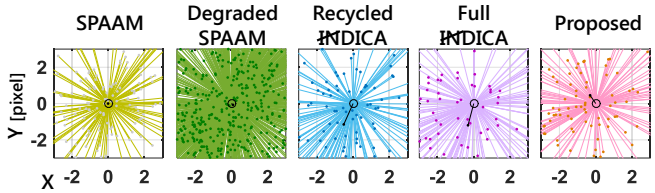
[1] http://www.ms.k.u-tokyo.ac.jp/software.html



(a) Boxplot of the mean reprojection errors for each calibration session of each method. Our method improves Full INDICA accuracy significantly ($p \approx 6.37\text{e-}04 \ll 0.05$, two-sample t-test), while it did not defeat the Recycled INDICA ($p \approx 0.46$).



(b) Reprojected 2D points by the two methods from each acquisition step. The amount of distortion correction is visualized by arrows with scaling. The estimated distortions are similar to what we learned in the regression dataset.



(c) Distribution of the reprojection error vectors. Black vectors drawn from the center of each plots are the mean value of each distribution. As expected, SPAAM has a mean vector which is almost at the center. The proposed method made the mean vectors of Full INDICA closer to the center, thus our method reduced the systematic errors existed in our previous automated calibration method.

Figure 9.9: User-based calibration experiment. (a) Box plot of the mean errors. (b) Distribution of reprojection error vectors. (c) Visualization of reprojected 2D points on the virtual screen.

was the same as the one used in the training dataset acquisition.

In the camera-based SPAAM (Fig. 9.6), we set up the user-view camera and the OST-HMD as described in Sec. 9.4.2. We rendered a 2D crosshair on the display. We placed a square marker in the world such that the camera saw both the center of the marker and the crosshair at the same pixel $\mathbf{u}'_k$ on S. Then we recorded $\mathbf{u}'_k$. At the same time, the world camera computed the 3D position of the marker center $\mathbf{x}_k$ in W. We repeated this procedure $N(=20)$ times, resulting in N pairs of 2D-3D correspondences $\left\{(\mathbf{u}'_k, \mathbf{x}_k)\right\}_k$. After the data collection, we compared two conditions. The first condition was an ordinary SPAAM, where we computed a projection matrix from the raw data and computed its reprojection error. In Fig. 9.8a, circles (Distorted) denote the original distorted 2D points measured during this step.

The second condition incorporated our distortion compensation method. First of all, before starting the above data collection, we let the user-view camera see the marker without placing the HMD. We thereby obtained the camera pose with respect to the marker. Then, we placed the HMD in front of the camera without moving it. Since the world camera on the HMD saw the

same marker (as in Sec. 9.3.1.1), we could compute $\mathbf{t}_{EW}$ and a 3D ray $\mathbf{x}_W^S$ by back projecting the distorted 2D point $\mathbf{u}'_k$. We thus obtained the distorted light ray l'_k.

Using the inverse mapping $\mathbf{f}^{-1}$, we estimated the original light ray as $l_k = \mathbf{f}^{-1}(l'_k)$: We computed *undistorted* 2D positions that the camera would have observed if there had been no distortion by the optical element (*Undistorted* points in Fig. 9.8a). Based on these corrected 2D points and the original 3D points, we estimated a projection matrix and compute its reprojection error.

Figure 9.8b is the comparison of the reprojection errors from the two conditions. It shows that our corrected condition (the right bar) leads to a significantly lower reprojection error compared to the original (the left bar). In SPAAM, we used the Direct Linear Transform (DLT) and Levenberg-marquadt (LM) method for estimating initial and optimized projection matrices. The DLT method does not model distortion in 2D space. The LM method we used does not include any distortion terms. And, Fig. 9.8c visualizes the distributions of the error vectors.

The fact that our correction method significantly reduced the reprojection error indicates that the method removed a systematic error caused by the optical element of an HMD which has not been considered in the standard HMD calibration framework.

9.5.3 Distortion correction for user-based calibration

We further evaluate our method in a user-based calibration experiment where a real user conducts 2D-3D correspondence acquisitions manually.

We follow the experiment design in [IK14b]. An expert user of SPAAM has to collect sets of 2D-3D correspondences while letting the eye tracking camera record eye images to compute eye positions offline. The user has eight data collection sessions. The user is asked to take the HMD off and put it back on after each session to simulate a realistically degrading condition (Degraded SPAAM) with users staying on the initial calibration parameters even when their HMDs have moved on their head. For each session, the user collects 20 correspondences. We use the collected correspondence sets to analyze SPAAM, Degraded SPAAM, Recycled INDICA, Full INDICA and our correction method. Since our method requires the spatial parameters of the virtual screen of an OST-HMD, the method can be seen as an extension of Full INDICA which uses the same parameters. Figure 9.9 summarizes the result of the experiment.

Figure 9.9a shows the box plot of the average reprojection errors for each calibration session. Our proposed correction method improved the reprojection error compared to Full INDICA to a statistically significant level. On the other hand, the improvement was not significant compared to Recycled INDICA. The discussion section analyzes this observation. All INDICA-based methods demonstrate more stable results than the Degraded SPAAM, corroborating the findings of other work.

Figure 9.9b visualizes the effect of the distortion correction. It draws reprojected 2D points of Full INDICA and of the proposed method for each data acquisition session. From the error vectors between the points, estimated distortions look concentric and radial.

Figure 9.9c presents the error vectors of each method in separate boxes. The error vectors are defined as vectors from 2D points corrected by the user to 2D points reprojected from

corresponding 3D points. In the visualization, the error vectors are shifted such that they all start at the origin. Each box also visualizes the mean of the end points of the error vectors. SPAAM shows an almost centered mean value. This is expected since the LM method estimates a projection matrix such that the mean error is minimized even if there are outliers. Since the 2D-3D *ground truth* data would contain noisy samples due to the manual alignment, the SPAAM result is likely to be overfitted.

On the other hand, the mean errors of our previous INDICA methods exhibit large offsets from the center. In other words, the reprojected 2D points of the methods are shifted in a particular direction – the methods contain systematic errors in their projection matrices. However, our correction method *shifts back* the mean error of Full INDICA closer to the center. Therefore, our method reduces the systematic errors that the previous automated method (Full INDICA) had.

9.6 Discussion

Throughout the two experiments, our correction method increased the calibration accuracy significantly and reduced the systematic errors which have existed in our previous interaction-free calibrations.

In the camera-based experiment, our method demonstrated that it improved the SPAAM calibration to subpixel level by precorrecting the distortion caused by the OST-HMD's optical elements. A natural question following this result was how much our method is contributing to the real, user-based OST-HMD calibration.

In the user-based calibration, our method also improved the calibration accuracy against Full INDICA. However, the accuracy had no significant difference against Recycled INDICA. A reason might lie in the recycled projection matrix in Recycled INDICA. In the experiment, the recycled projection matrix was from a standard user-based SPAAM, which means that the user aligned *distorted* 2D points to 3D points. And the DLT and LM methods estimated a projection matrix which best fit the distorted correspondences to the extent allowed by the perspective camera model. Thus, the recycled projection matrix partially contributed to an implicit compensation of the optical distortion in Recycled INDICA.

We conject that this is why the Recycled INDICA is yielding as low a mean error as our correction method while showing higher error variance – a systematic error possibly induced by the forcibly fit projection matrix.

Even though the original SPAAM is prone to overfit the given 2D-3D correspondences, the automated methods have not performed as accurately as the SPAAM calibration, yet. Why? Why do such gaps still exist? We have several hypotheses stemming from the fact that INDICA models the eye-HMD vision system as a naive pinhole camera, which is not true when we scrutinize the optical models of OST-HMD optics and the anatomical model of the human eye.

9.7 Summary

We have proposed an OST-HMD calibration method. It corrects an optical distortion which conventional eye-HMD models have not considered – the distortion of the light rays passing through the optical elements of OST-HMDs. Our method consists of both offline and online steps. In the offline step, we learn a 4D-to-4D light field mapping which converts each original light ray to its distorted ray. The learning is done by first collecting the light rays measured with/-out an optical element, then computing the mapping via a non-parametric regression. Then, at the online step, the method compensates the distortion by using the mapping given an eyeball position from the interaction-free OST-HMD calibration method. Our experiments show that the correction method reduces the systematic error which has existed in both conventional camera-/user-based calibrations, and also significantly improves calibration accuracy.

Future work directions involve: considering the distortion of the virtual screen [KHS14; Owe+04; LH13] which we assume to be planer, deepening the understanding of the eye-dependent parameters [Plo+15], investigating the possibility of automated frame-wise OST-HMD calibrations, establishing and refining ways to compare different calibration methods with both subjective [Mos+15] and objective error measurements, overcoming the latency issue which is also another dominant aspects directly affects to the spatial registration quality [Zhe+14], and so on.

Firstly, OST-HMDs have distortions in their virtual screen, whereas projection has been assumed to be planar in our current model. Our correction method considers an optical phenomenon that the optical elements distort incoming world light rays. In the same manner, the elements also distort virtual screens perceived by users into a non-planar surface [KHS14; Owe+04; LH13]. Even the assumption that we treat the virtual screen as a collection of 3D points floating in mid air is violated when the light sources are collimated as in retinal displays. Camera-based experiments would suffice to justify and evaluate those HMD-related hypotheses.

Secondly, the visual axis of the human eye differs from the optical axis in the two-sphere eye model we employed [Plo+15] This issue requires actual users. A camera integrated in a prosthetic eye might be an alternative, yet we have no clue how accurately such a system can mimic the real eye.

Yet another issue is the reliability of our 2D-3D correspondence dataset, which is collected manually. Although the dataset was produced by a human expert, the 2D-3D dataset may still contain a large amount of noise: if the noise is more than a few pixels in the true projected 2D points, it would be meaningless to argue about calibration errors in subpixel range – or impossible to obtain major significance despite the potential of a new method.

What would help to justify this last hypothesis is to conduct the same experiment with many different subjects in a very controlled environment such as in [Mos+15]. Perhaps such a study can create a benchmark dataset as a by product. Similarly, our method would also require a proper user-study as a follow-up.

As a final remark, let us reconsider the distortions by the optical elements. In the experiments with the nVisor ST60, the estimated distortion was rather concentric. Other OST-HMDs may have different characteristics. For example, EPSON BT-100 has a linear distortion due to its thick

planar optical element. As another example, Lumus DK32, a high-end OST-HMD with ultra-thin light-guide elements, seems to have little distortion. Thus it might not benefit from our distortion corrections as much as the ST60 does. In this way, as a follow up study, it would be interesting to apply our non-parametric distortion correction to various HMDs.

10 Unified Light-field Correction

This section is based on the work that the author presented at IEEE ISMAR 2015 [Ito+15b].

10.1 Introduction

In the previous section, we have focused on correcting optical distortion of incoming light rays from the real world through an OST-HMD to a user's eye, which we call from now on Direct-View Distortion (DVD). For calibrating the HMD part, an issue often over-looked is actually this DVD and another optical aberrations caused by the optical media of OST-HMDs – Augmented-View Distortion (AVD) (See Fig. 10.1 too). AVD is the image distortion of a perceived image. A common OST-HMD design employs an optical combiner to guide the light from a light source of an OST-HMD to a user's eye [RH05]. As the result, the user perceives the light ray as if it appears as a virtual image floating mid air or at infinity in the user's view (Fig. 10.1 left). Since the combiner is an optical medium, it inevitably refracts light rays passing through itself [RH05] including those from a physical object in the world, i.e. DVD, and image light rays from the OST-HMD, i.e. AVD (Fig. 10.1 right).

The intricacy of these distortions is that the amount of each distortion depends on where and at what angle a light ray hits the combiner and passes through its medium, i.e., each user viewpoint suffers from different amounts of both distortions. Furthermore, due to the imperfection of the display optics, the image light rays may suffer from additional aberration out of the design during its passage through various optical media from the light source to the combiner.

Importantly, DVD and AVD share the same distortion characteristic in part (Fig. 10.1). Consider a light ray from an eye towards the optical combiner, the ray first reaches to the half-mirror of the combiner while refracted by the medium. The ray is then split into two paths: one towards the world and the other the light source while receiving additional aberrations separately. D/AVD thus consist of a shared distortion part and a individual part. This section and existing works, however, do not explicitly separate these mixture model.

While methods exist for correcting either of the distortions independently, there is, to our knowledge, no method handling both distortions simultaneously for an arbitrary eye position with respect to an HMD screen.

Adapting existing AVD correction methods to DVD is not straightforward. They model the image screen of an OST-HMD as a 3D plane/surface for modeling AVD. Such modeling does not transfer to DVD directly. The previous section proposes a camera-based calibration which corrects DVD for arbitrary eye positions based on a 4D light-field model [IK15a]. This method treats DVD as a mapping between original and distorted light-fields; learns the mapping via

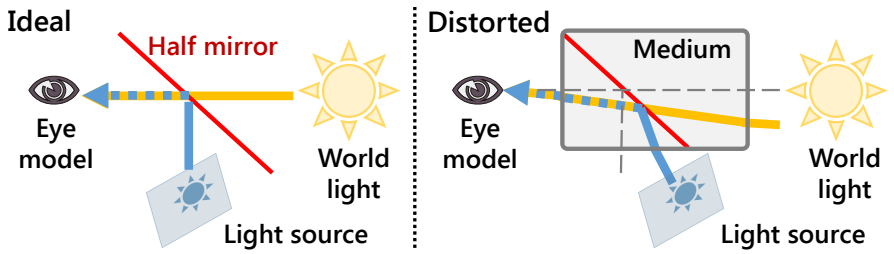


Figure 10.1: Schematic visualization of optical aberrations in an OST-HMD system. (Left) Ideal case. (Right) A practical case where the A/DVD present in the system. If we back-trace a ray from an eye, one notices that both distortions partially share a distortion path.

non-parametric regression from a training data set offline; and computes DVD for a given new eye-position online. Under the light-field model, the AVD can be treated in a similar manner.

We propose a camera-based calibration method that corrects both distortions simultaneously for arbitrary eye positions given an OST-HMD system. Our method adapts the light-field approach to AVD and has an offline/online step. The offline step learns a cascaded mapping which consists of two light-field mappings corresponding to each distortion. The online step applies the cascaded mapping to given 3D world points and returns 2D image points. The 2D points will appear on a distorted image plane and will match the 3D points, which are also distorted, from the user's current viewpoint.

The evaluation with two OST-HMDs (a professional and a consumer OST-HMD) show that our model significantly improves the calibration quality compared to a conventional HMD model and the previous DVD-only model. The results also indicate that only correcting both distortions simultaneously can improve the quality.

We discuss limitations of the current approach mostly due to the limited capabilities of current OST-HMDs, and conclude by noting some open questions toward practical OST-HMD calibrations.

Contributions

- Providing a calibration method which corrects the D/AVD of OST-HMDs simultaneously for arbitrary eye positions.
- Demonstrating that the method improves the calibration quality of two OST-HMDs. The qualities are comparable up to a human eye of 20/50 visual acuity.
- Showing, with a reasoning, that only correcting both distortions can improve the final quality.

10.2 Related Work

A key of successful calibration is how to model the eye-HMD system. Although eye model is equivalently important, this section focuses on the HMD model which holds both DVD and AVD.

10.2.1 Direct-view distortion of OST-HMD

As we elaborated in Chap. 9 [IK14b], we tackled the DVD by employing the light-field correction approach. Our approach aims at capturing the optical characteristics of OST-HMDs as the shift of optical rays.

10.2.2 Augmented-view distortion of OST-HMDs

A user perceives an image of an OST-HMD as a virtual screen floating in mid air. For correct registration, we need to know how this image appears in a user perspective view. We categorize existing approaches in three types: parametric and semi-/non-parametric.

Parametric Approach Parametric approaches model the screen images as a certain class of functions. A common approach is to treat the image produced by an OST-HMD as a 3D plane floating in mid air [JMC93; TGN02; GFG08; Gen+00; AB94]. Under this model, an OST-HMD system is treated as an off-axis pinhole camera.

This model, however, is incapable of describing real OST-HMD optics. Owen et al. [Owe+04] demonstrate that the plane model does not coincide with a 3D geometry of the display measured via triangulation, and they propose a curved 3D surface model. Their surface model respects the spherical distortion caused by a curved mirror in their OST-HMD, which falls into the first-order radial distortion model. Robinett and Rolland [RR92] use the same distortion model. Hua et al. [HGA02] apply a similar model to their head-mounted projective display. Lee and Hua [LH13] extend the surface model to 6th order radial distortions and tangential distortion.

These approaches have a common drawback in the change of the user's eye position w.r.t the HMD screen. Since the optics of an OST-HMD may distort the light of an image pixel differently at different viewpoints, the above models, learned at a single viewpoint, can cause registration errors when the eye position changes.

Semi-Parametric Approach Wientapper et al. [Wie+13] propose a semi-parametric model for Head-Up Displays (HUDs). HUDs are essentially the same as OST-HMDs except that their images are reflected on the wind shields of vehicles. Their model combines the 3D-plane and a view-dependent polynomial model. The latter employs a higher-order polynomial function of 5 parameters: a 2D image point and a 3D eye position. We call their model semi-parametric since their polynomial model is essentially non-parametric, which is based on local data points and is represented by a linear sum of polynomial kernels.

Non-Parametric Approach Klemm et al. [KHS14] upgrade the 3D plane model by triangulating every pixel of an OST-HMD screen via the photogrammetry with structured image patterns. Their non-parametric, point-cloud approach requires an additional user adaptation since a few millimeters of error in the viewpoint position causes non-negligible registration errors.

Recall that the refraction of an optical medium causes the AVD. The amount of refraction depends on which path the ray takes through in the medium – a light field (LF) of an image

changes the shape based on the user's eye position.

We adapt our original light-field model used for the DVD – we model the image screen as a function of light rays defined by 4D rays and learn the function via non-parametric kernel regression.

10.3 Method

We first explain the DVD and the AVD correction separately. We then introduce a unified approach. The notations are same as [IK15a].

10.3.1 Direct-view distortion correction in the nutshell

As same as Sec. 9.3, let ${}^{D}l_{W}$ a light ray in the world coordinate system as a 4D Lumigraph: ${}^{D}l_{W} := l(\mathbf{t}_{EW}, \mathbf{x}^{S}_{W}, \mathtt{R}_{SW}, \mathbf{t}_{SW}) := l(\mathbf{x}^{S}_{W}) \in \mathbb{R}^4$. See Sec. 9.3 for the exact definition.

The DVD causes a distorted ray ${}^{D}l'_{W}$. Given a set of $({}^{D}l_{W}, {}^{D}l'_{W})$ measured from various viewpoints within the eyebox of the HMD, our LF correction approach in Sec. 9.3 gives a function ${}^{D}f(\cdot)$ so that ${}^{D}f({}^{D}l_{W})$ is close to ${}^{D}l'_{W}$ via a non-parametric kernel regression.

10.3.2 Augmented-view distortion correction

We measure the LF of the OST-HMD screen in a similar way as in the previous section. Instead of letting a camera see 3D world points through the medium of the HMD, we let the camera capture the image of the image screen such that the camera can identify which image pixel is corresponding to that of the camera sensor.

Let $\mathbf{u}^{S}_{E} \in \mathbb{R}^2$ be an image pixel of a camera which corresponds to $\mathbf{u}_{S} \in \mathbb{R}^2$, an image pixel of the virtual screen of the HMD. Let $\mathtt{K}_{E} \in \mathbb{R}^{3\times 3}$ be the intrinsic matrix of the user-perspective camera located at $\mathbf{t}_{EW}$. We compute a point $\mathtt{K}_{E}^{-1}\widetilde{\mathbf{u}}^{S}_{E}$, where $\widetilde{\cdot}$ denotes the homogeneous vector. Given a 6DoF pose between the eye and the world coordinate systems as $(\mathtt{R}_{EW}, \mathbf{t}_{EW})$, an eye sees a ray ${}^{A}l'_{W}$ as

$$ {}^{A}l'_{W} := l(\mathbf{t}_{EW}, \mathtt{R}_{EW}\mathtt{K}_{E}^{-1}\widetilde{\mathbf{u}}^{S}_{E} + \mathbf{t}_{EW}, \mathtt{R}_{SW}, \mathbf{t}_{SW}) = l(\mathtt{R}_{EW}\mathtt{K}_{E}^{-1}\widetilde{\mathbf{u}}^{S}_{E} + \mathbf{t}_{EW}). \tag{10.1} $$

${}^{A}l'_{W} \in \mathbb{R}^4$ is a *distorted* ray since $\mathbf{u}^{S}_{E}$ contains the AVD already. We define an original ray ${}^{A}l_{W}$ *virtually*: a ray that would appear as $\mathbf{u}_{S}$ if there were not for AVD and if the 3D plane model were correct. Let α a scale parameter with a unit of [meter/pixel], then ${}^{A}l_{W}$ becomes,

$$ {}^{A}l_{W} := l(\mathbf{t}_{EW}, \alpha\mathtt{R}_{SW}\widetilde{\mathbf{u}}_{S} + \mathbf{t}_{SW}, \mathtt{R}_{SW}, \mathbf{t}_{SW}) = l(\alpha\mathtt{R}_{SW}\widetilde{\mathbf{u}}_{S} + \mathbf{t}_{SW}) \in \mathbb{R}^4. \tag{10.2} $$

Finally, we learn a function ${}^{A}f^{-1}(\cdot)$ so that ${}^{A}f^{-1}({}^{A}l'_{W})$ is close to ${}^{A}l_{W}$. We now introduce a way to correct D/AVD simultaneously.

Figure 10.2: Hardware setup. (Left) nVisor ST60. (Right) Moverio BT-100. The camera on the left image is displaced for the photography; it was set closer to the screen during actual data collections.

10.3.3 Unified distortion correction

For aligned visualization, a world point and a corresponding image point must eventually travel along the same light path from the combiner to the eye. However, ${}^{D}l_{W}$ reaches the user's eye as ${}^{D}f({}^{D}l_{W})$ after having undergone the DVD; the corresponding virtual ray ${}^{A}l_{W}$ as ${}^{A}f({}^{A}l_{W})$ after AVD. The virtue of these LFs is that they are defined in a common coordinate system: they share the same u-v and the s-t plane of the 4D lumigraph. We can thus directly bypass both distortion effects.

Our goal is to achieve ${}^{A}f({}^{A}l_{W}) = {}^{D}f({}^{D}l_{W})$. Since ${}^{D}l_{W}$ is defined in the same space as ${}^{A}l_{W}$, we may apply the inverse of the AVD: ${}^{A}l_{W} = {}^{A}f^{-1}({}^{D}f({}^{D}l_{W}))$. Thus, if $\mathbf{u}_s$ corresponds to ${}^{A}l_{W}$, then $\mathbf{u}_s$ and the corresponding 3D point in the world align in the user's view. By definition, $\mathbf{u}_s$ is the s-t elements of ${}^{A}l_{W}$: $\mathbf{u}_s = [[{}^{A}l_{W}]_s[{}^{A}l_{W}]_t]^{\mathrm{T}}$.

10.4 Technical Setup

We arranged two experimental setups with two different OST-HMDs. The first one is a professional HMD which has a higher resolution and a larger field of view than the second one – a consumer OST-HMD. Both setups use a 4×11-asymmetrical circle-grid board as the 3D world reference. We place it at about 1.5m away from the displays so that user-perspective cameras can see the board and the image of each display sharply at the same time.

10.4.1 Professional OST-HMD setup

The first setup (Fig. 10.2 left) uses the exact same one in [IK14b].

10.4.2 Consumer OST-HMD setup

The second setup is a consumer OST-HMD (Fig. 10.2 right) – a Moverio BT-100 from EPSON with 960×540 resolution and 23° diagonal FoV. The left-eye display is used for the setup. The

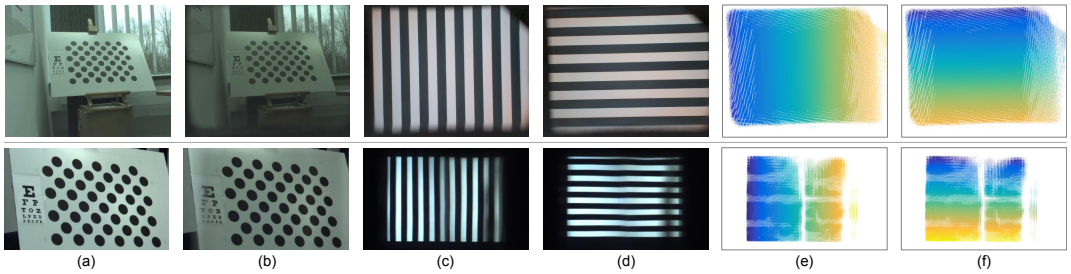


Figure 10.3: Examples of training data and processed images. Top row is for the ST60 and the bottom for BT-100. Each column is: (a) a scene captured by the user-perspective camera directly, (b) through the optical combiner, (c,d) a gray-code pattern displayed on the screen and captured by the camera, and (e,f) learned camera-to-hmd pixel mappings of horizontal and vertical directions. The mappings are color coded.

HMD employs an HTPS TFT LCD panel with a color filter for each display. Light from the panel is guided to semi-transparent, half mirrors in front of a user's eye, and the mirrors reflect the light to the eye.

This HMD has a composite video input. We use a VGA-composite converter (Ligawo PC-TV). It generates a composite video signal from an input digital image with 656×496 resolution. As a result, the HMD renders a stretched image compared to the original resolution.

The outward looking camera is the same Delock USB 2.0 model as the professional setup. Since this HMD has much narrower FoV than the ST60, we select a different user-perspective camera with narrower FoV. The camera is a UI-2280SE-C-HQ Rev.3 from iDS. It has a 2/3" sensor and provides 2448×2048 images, together with a 25mm C-mount lens (FL-CC2514-5M from RICOH).

10.4.3 Image light field acquisition via structured patterns

We first describe how to measure the LF of a display screen, i.e. $\{{}^{A}l'_{W}\}$. Given a user-perspective camera which is seeing the image screen of an OST-HMD, we need to collect 2D-to-2D correspondences $\{(\mathbf{u}^{S}_{E}, \mathbf{u}_{S})\}$. We display structured patterns (gray-code/sinusoidal for pixel-/subpixel-level matching) on the screen, and match screen image points $\{\mathbf{u}^{S}_{E}\}$ to camera image pixels $\{\mathbf{u}_{S}\}$. We use a software[1] by Yamazaki et al. [YMK11]. Figure 10.3 (c,d) are some patterns shown on the display of HMDs captured by a camera. Figure 10.3 (e,f) show learned mappings.

We follow the calibration procedure in Sec. 9.4.2 to determine an initial 3D plane pose $(\mathtt{R}_{SW}, \mathbf{t}_{SW})$ and a pixel-to-meter scale α from $\{(\mathbf{u}^{S}_{E}, \mathbf{u}_{S})\}$, and the intrinsic matrix $\mathtt{K}_{E}$ of the user-perspective camera. Finally, Eq. 10.1 and 10.2 give ${}^{A}l'_{W}$ and ${}^{A}l_{W}$ respectively.

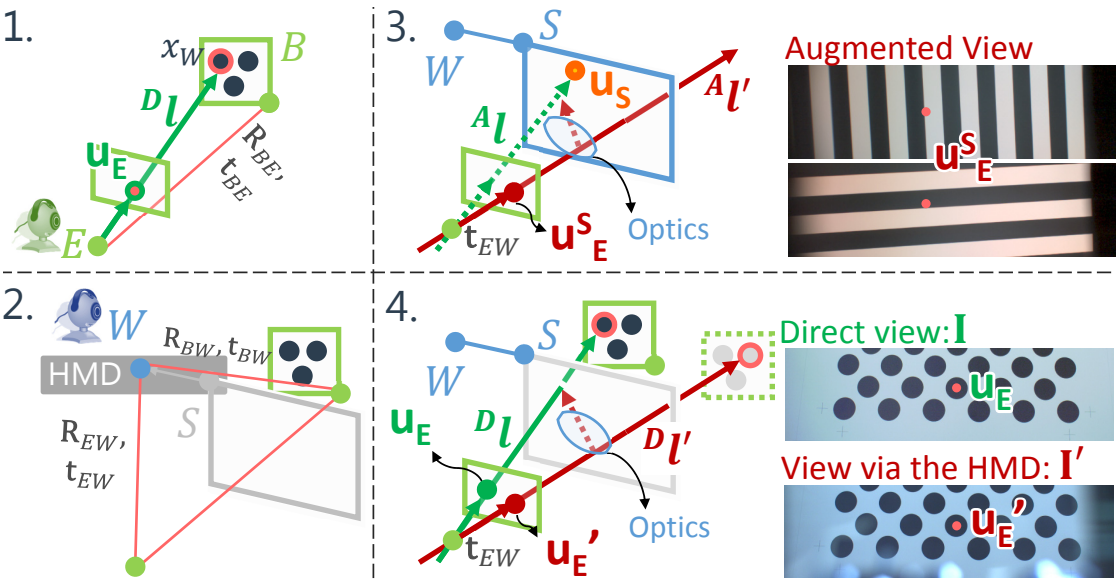


Figure 10.4: An overview of the data collection steps described in Sec. 10.4.4. Images are from the professional HMD setup.

10.4.4 Training data sampling

We describe how to collect training data of the original and the distorted LF offline. We need a set of original and distorted light ray pairs $\left\{\left({}^{D}l_W, {}^{D}l'_W, {}^{A}l'_W, {}^{A}l_W\right)\right\}$ for various viewpoints within the eye box such that the learned regressions can cover various eye positions in applications. Our collection procedure requires the following: a user-view camera E, an OST-HMD with a world camera W, and a fiducial target board B fixed in a scene. We assume that the cameras are calibrated. The procedure is as follows (see Fig. 10.4):

1. Place the user-perspective camera E and the 3D target B in the scene, and let E capture a direct-view image $\mathbf{I}$. Then from $\mathbf{I}$ and the intrinsic matrix $\mathtt{K}_E$ of E, estimate the pose of the target as $\left(\mathtt{R}_{BE}, \mathbf{t}_{BE}\right)$.

2. Place the OST-HMD in front of the user-view camera, and let the user-view camera capture a distorted-view image $\mathbf{I}'$. Let the world camera W on the HMD capture the 3D target and estimate the pose $\left(\mathtt{R}_{BW}, \mathbf{t}_{BW}\right)$. Using this pose and $\left(\mathtt{R}_{BE}, \mathbf{t}_{BE}\right)$, compute the pose of the user-view camera relative to the world, $\left(\mathtt{R}_{EW}, \mathbf{t}_{EW}\right)$.

3. Without touching the hardware, block the world light coming through the display, e.g. by putting a black sheet in front of the HMD, then capture the structured patterns as described in Sec. 10.4.3.

4. From $\mathbf{I}$ and $\mathbf{I}'$, extract corresponding 2D points $\mathbf{u}_E$ and $\mathbf{u}'_E$. Compute their 3D position in W as $\mathbf{x}_W := \mathtt{R}_{EW}\mathtt{K}_E^{-1}\widetilde{\mathbf{u}}_E + \mathbf{t}_{EW}$ and $\mathbf{x}'_W := \mathtt{R}_{EW}\mathtt{K}_E^{-1}\widetilde{\mathbf{u}}'_E + \mathbf{t}_{EW}$. Finally, compute an original light ray $l := l\left(\mathbf{x}_W\right)$ and its distorted $l' = l\left(\mathbf{x}'_W\right)$. Figure 10.3 shows collected samples.

[1] http://www.dh.aist.go.jp/~shun/research/calibration/

As a result, we get a set of desired light-ray pairs with a maximum of 44 ($= 4 \times 11$) pairs. We collect such sets for a number of viewpoints. Due to the limited FoV of the image screen and/or failures in the structured-light matching, some 3D points on the board do not have corresponding 2D points on the screen (Fig. 10.3 (e,f)).

For the professional HMD setup, we collected training data from 17 viewpoints. We used all 17 sets for learning the DVD, and 10 for the AVD. For the consumer HMD setup, we collected training data from 8 viewpoints. We used 8 for the DVD and 7 for the AVD. Subsequently, for both setups, we took test data from new viewpoints – different from those of the training data sets.

The differences of the number of data sets used for each distortion estimation are from various reasons: a) Some data sets were old one that captured only for DVDs. b) image screens were not visible from a viewpoint, i.e., the user-perspective camera was outside of the ideal eyebox. c) A missing pose between the cameras since the user-perspective camera could not see the calibration board due to the occlusion caused by the frame of an HMD. The discussion section covers remaining issues towards establishing simpler and stabler calibration procedure.

10.5 Experiment

10.5.1 Error measurements based on viewing angles

Existing calibration methods evaluate their accuracy by 2D projection errors: the distance between a measured and an estimated 2D point in a planar coordinate system (e.g. the image plane of a user-perspective camera [GFG08; LH13] or a physical board in the scene [McG+01]).

Instead, we employ the Viewing Angle (VA) error between the forward-projected rays of the two 2D points from a user's viewpoint (Fig. 10.5). Let us call a ray from the original 2D point as the base ray $\mathbf{n} \in \mathbb{R}^3$, $\|\mathbf{n}\| = 1$, and let the other from the perturbed 2D point $\mathbf{n}' \in \mathbb{R}^3$, $\|\mathbf{n}'\| = 1$. We first consider a plane which is tangential to the unit sphere at $\mathbf{n}$. The plane has its x/y axis along a latitude/longitude line on $\mathbf{n}$ heading to west/north (Fig. 10.5 left). We then define the VA error $\theta := \arccos(\mathbf{n}^{\mathrm{T}}\mathbf{n}')$ and the direction angle ρ (Fig. 10.5 right).

The VA error constitutes a common measurement over OST-HMDs of different resolutions and is compatible to the human visual acuity; thus should be used instead of the reprojection error.

10.5.2 Experiment procedure

For each HMD setup, our procedure was the following: collect a training and a test data set as described in Sec. 10.4.4, compute mappings ${}^{D}f(\cdot)$ and ${}^{A}f^{-1}(\cdot)$ via a kernel regression following Sec. 9.3.1.2, and test them with the test data. The number of basis functions was 50.

Experiments were done mostly with MATLAB R2014b. For the pose estimation of the calibration board, we used our open-source C++ tracking framework, Ubitrack, with the OpenCV library.

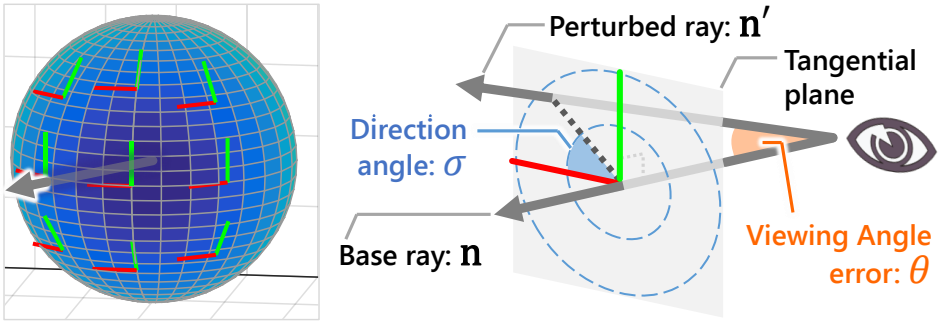


Figure 10.5: Illustration of the VA error. (Left) A 3D unit sphere with tangential coordinate systems. (Right) The VA error and the direction angle. The x/y axis of the tangential plane is orthogonal to the base ray and is aligned with the longi-/latitude lines of the sphere.

10.5.3 Results with the professional OST-HMD (Fig. 10.6)

We first explain the results of the independent distortion estimations, i.e. ${}^{D}f({}^{D}l_w)$ vs. ${}^{D}l'_w$ and ${}^{A}f^{-1}({}^{A}l'_w)$ vs. ${}^{A}l_w$, with Fig. 10.6a. The top row shows the boxplots of the estimation errors of the AVD and the DVD, respectively. The bottom row shows their histograms. Note that the blue lines in the plots are at one arc minute, i.e. 1/60 (0.016...) degree, which is equivalent to the critical gap size for emmetropic (standard) human visual acuity. If a calibration result crosses these lines, such an AR experience would be, as it were, *retinally* aligned, thus indistinguishable to the eyes. Both corrections improved the calibration accuracy. Especially, the DVD estimation half crossed the threshold.

We now examine the results of the actual calibrations, i.e.${}^{A}l_w$ vs. ${}^{A}f^{-1}({}^{D}f({}^{D}l_w))$, with Fig. 10.6b. The boxplot at the top summarizes the calibration errors. Correcting either of the two distortions separately did improve the overall calibration quality compared to without any corrections. Correcting both of them further decreased the errors. However, the mean error was higher than the visual acuity. We conjecture that this stems from the relatively high errors in the AVD estimation compared to the DVD.

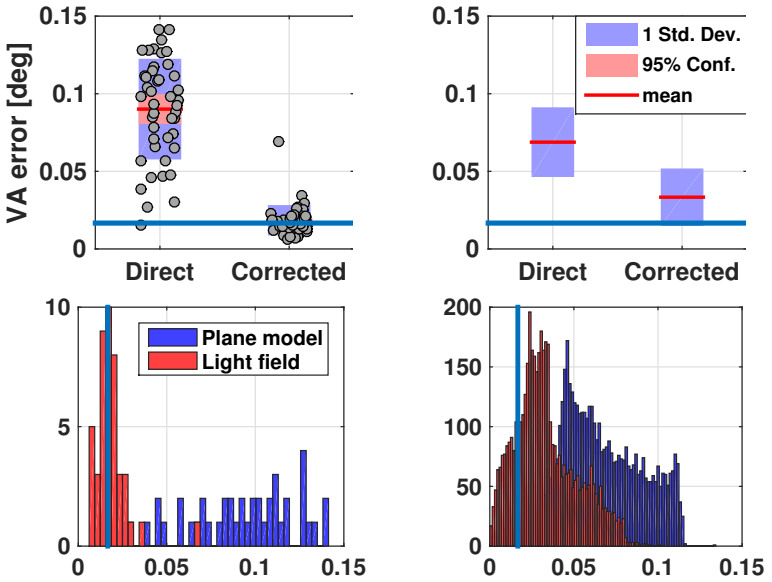
Nevertheless, the mean error reaches to the level of around 20/50 visual acuity. If those people see AR contents with this calibration without eyeglasses, it would appear in indistinguishable registration quality – if the eye model and position are perfectly estimated.

Figure 10.6b bottom shows each error with corresponding direction angle. The bias decreases by applying the distortion corrections.

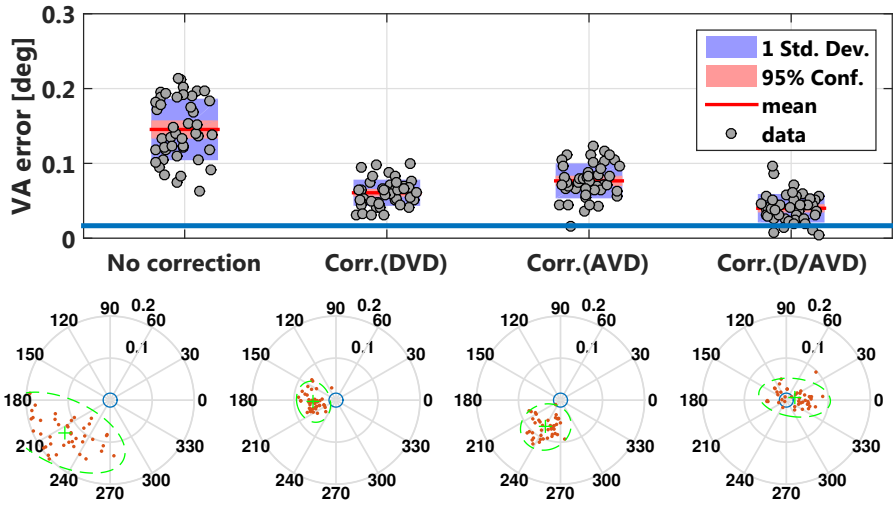
10.5.4 Results with the consumer OST-HMD (Fig. 10.7)

Figure 10.7a left shows that the DVD estimation achieves the mean error to fall below the visual acuity line – as for the professional setup. In the AVD estimation (Fig. 10.7a right), the errors decreased overall, yet, the majority of the samples stay above the line.

Compared to the results of ST60, that of the BT100 (Fig. 10.7b) reveal an intriguing fact: correcting either of the D/AVD independently makes the results worse than applying no corrections –

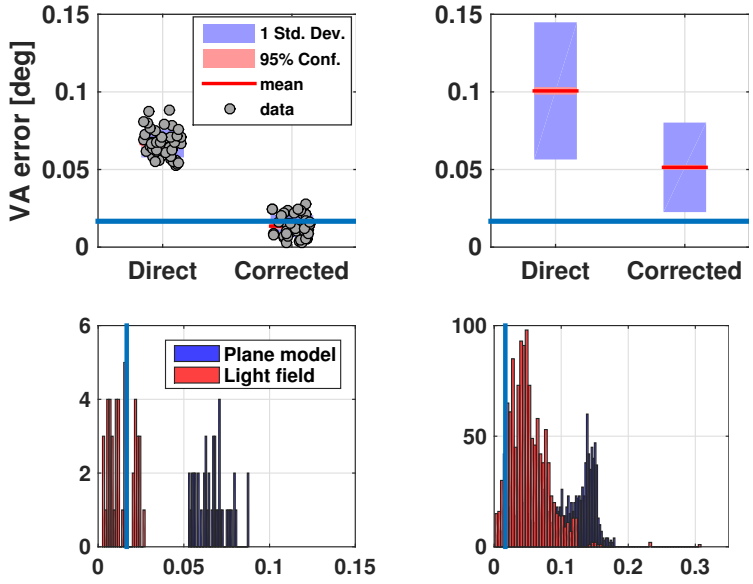


(a) Results of D/AVD estimations, i.e. ${}^{D}l'_{w}$ vs. ${}^{D}f({}^{D}l_{w})$ and ${}^{A}l_{w}$ vs. ${}^{A}f^{-1}({}^{A}l'_{w})$. (Top) Boxplots of VA errors of each estimation. (Bottom) Their histograms. In both cases two-sample t test returned p values almost zero.

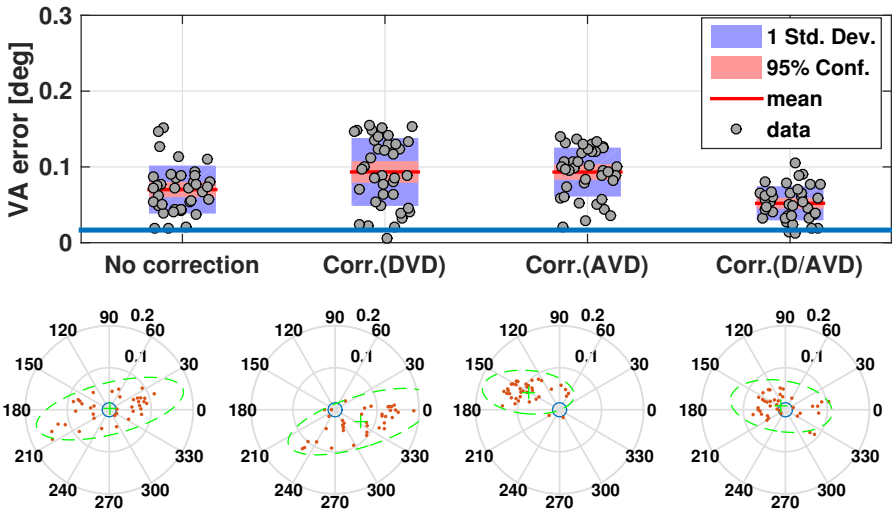


(b) Results of calibrations. (Top) The boxplot of each calibration methods in angle errors. (Bottom) Error vectors in polar coordinates. Dashed green lines visualize covariances with a confidence value 95 %. A multiple comparison test confirmed any of the two conditions are significantly different mean values.

Figure 10.6: Calibration results for the professional OST-HMD (NVIS nVisor ST60). Blue lines represent the standard visual acuity value: 1 arcsec.



(a) Results of D/AVD estimations. The two-sample t test returned p values almost zero for both cases.



(b) Results of calibrations. Unlike the previous setup, a multiple comparison test found significant difference except: between original and the proposed (p=0.090); and between the single correction methods (p=1.0).

Figure 10.7: Calibration results for the consumer OST-HMD (Moverio BT100). All notations are same as the Fig. 10.6.

even though the DVD estimation achieves the error almost to the level of human visual acuity (Fig. 10.7a). Only when the two corrections are combined, the error back down below the level of the original error.

We suppose this is because the combined part of the rays from the world and from the screen partially share an optical path inside the optical combiner (Fig. 10.1). The BT-100 employs a flat plastic housing for its combiner. Both light rays are possibly almost parallel when they join at the half-mirror, then they pass through the same medium while receiving the same aberration. Thus, correcting one of the two rays for A/DVD degrades the overall quality. Only the combined correction of both rays improves the accuracy.

10.6 Discussion

We examine the results of the two setups in combination. We start with the estimation results of the DVD (Fig. 10.6a/10.7a left) and the AVD (Fig. 10.6a/10.7a right) separately, then look at the actual calibration results with those estimated distortions (Fig. 10.6b and 10.7b). We further discuss the limitations of our method and open issues.

DVD Estimation | Fig. 10.6a left and 10.7a left Without corrections, the professional HMD setup has a higher mean error (0.09 deg.) than the consumer setup (0.067 deg.). This result is understandable since the HMD uses a thick cubic prism combiner which yields larger DVD than the flat-plate combiner of the consumer HMD. After the corrections, the DVD reduces the mean error as low as the standard human visual acuity in both setups.

AVD Estimation | Fig. 10.6a right and 10.7a right Without corrections, both setups show VA errors that are larger than in the DVD case. This can be that the AVDs were not as smooth as the DVDs: viewpoint changes could not be accommodated as well since the regression can not account for radical local changes.

Unlike the DVD case, the professional HMD has a lower mean AVD error (0.07 deg.) than the consumer (0.1 deg.). The consumer HMD has a huge non-linear distortion due to two separate half mirrors used in the optical combiner (Fig. 10.3 (c,d)). The center columns of the screen are especially blurry, and the pattern matching in the region failed (Fig. 10.3 (e,f)). On the other hand, the professional HMD achieved almost perfect matching thanks to its clear image.

Overall Calibration Results| Fig. 10.6b and 10.7b As mentioned in the experiment section, correcting the A/DVD simultaneously improves the total calibration accuracy in the professional setup (Fig. 10.6b) significantly. In the consumer setup (Fig. 10.7b), the result was not significant, yet it shows that the unified correction reverts errors that happen when only one distortion is corrected.

In both setups, the final accuracy does not reach at the human visual acuity level. Since the DVD estimation reached at the visual acuity level in both setups, it is likely that improving the AVD estimation does so the overall calibration.

Limitations and Issues of the Proposed Method The current method has a number of issues that need to be solved in order to establish a practical HMD calibration routine as easy as current camera calibration software.

Estimation quality for the AVDs: Our methods can not yet estimate AVD to the level of the human visual acuity while the DVD is corrected at the desirable level (Fig. 10.6a and 10.7a). Possible causes are: the number of data sets, tuning parameters in the regressions, the image matching accuracy, and the pose tracking accuracy of cameras. We discuss some issues in the following paragraphs.

Number of data sets: We do not know how much training data is sufficient to achieve desirable accuracy and which viewpoints are best for the training phase. At least, if we knew a valid eye box in which any user's eye would stay, we could limit the positions of the view point to this box for collecting the data.

Hyper parameters of nonlinear regression: As Sec. (10.5.2) says, the nonlinear regressions need several parameters. We do not know what the best ones. We assumed that the number of basis functions might have the strongest effect. However, our informal examination did not generate strong matching differences somewhere between 50 to 200. Too few basis functions, such as 10, failed as expected.

Image matching and: As in the BT100 case, a partially blurry image of the screen makes it difficult to detect image pixel correspondences for computing the LF. This questions the use of the LF model which is based on the geometrical optics: light from a point source is treated as an ideal ray at each viewpoint. Expanding the model such that it can handle *blurred* light rays might improve the matching. This also requires a different matching approach than our current matching based on the structured patterns.

Camera tracking: We used a calibration board for the pose tracking between the user-perspective camera and the world camera on the HMD. We do not know if the pose accuracy was high enough or not. In this sense, a sensitivity analysis via simulations, as we did for the interaction-free calibration [IK14b], might reveal the key factors in the entire calibration procedure.

DVD as a LF model: There is yet another issue in the current LF model for the DVD. The model treats both original and distorted light rays as those that pass through the center of the eyeball. The assumption is true for the original ray. However, a distorted ray (${}^{D}l'_{w}$) does not necessarily do the center – e.g. a convex lens shift the focal point of convergent light rays. Nevertheless, the experiments of the DVD estimation achieved desirable accuracy even though we did not consider this possible misalignment.

Computation time: We have not yet implemented a real time system to render the corrected virtual view. Our projection function is not a simple perspective projection any more. To compute the complex function, we might need a sampling approach somewhat similar to a ray tracing approach. Since we can not directly compute which image pixel corresponds to which 3D point in the scene, a naive way would be to sample a bundle of light rays passing through a given eye position, and to check where they hit the image screen.

User-based evaluation: Although this section focuses on the HMD-dependent factors only, it

is necessary to see how much our correction method contributes to the user-based interaction-free calibration [IK14a]. It would be also interesting to consider how to measure the VA error, if we can not get the ground truth eye position. Given an accurate eye-tracking algorithm, one might make a compromise and use the position from the tracker as the ground-truth.

Hardware Approach for DVD There are OST-HMDs that employ retinal-scanning (e.g., Brother AirScouter) or pupil-division optics (e.g., Olympus MEG4.0). Theoretically, such displays do not suffer from the DVD problem: either they do not use the optical combiner or the combiner is small enough such that the world light reaches to user's eye directly.

10.7 Summary

This section presents a calibration method which corrects optical aberrations that degrade the quality of OST-HMD calibration. Unlike existing methods, our calibration method corrects both DVD and AVD simultaneously for arbitrary eye position. Our approach expands a light-field correction approach developed for DVD to the AVD, and cascades two distortion corrections to cancel both distortions at the same time. Our method is camera-based and has an offline learning and an online correction step. The evaluation shows that the method significantly improves the calibration quality compared to conventional methods. The overall registration accuracy was comparable to 20/50 visual acuity. Furthermore, the results indicate that only by correcting both distortions simultaneously can improve the accuracy. We also analyzes limitations of the method and possible research directions.

Part IV

Vision Enhancement with Calibrated OST-HMDs

This part presents a fundamental study of Vision Enhancement (VE) in defocus correction via OST-HMDs to improve human vision. Our idea is to add a visual stimulus to a user's natural vision such that the user regains visual acuity. The stimulus is given as a compensation image displayed on an OST-HMD.

11 Defocus Correction via OST-HMDs

This section is based on the work that the author presented at AH 2015 conference [IK15b].

11.1 Introduction

Vision is our primary means to perceive the physical world. We can achieve various tasks through vision, combined with higher-order brain functions. We have developed numerous Vision Enhancement (VE) devices for supporting and/or boosting the capability of our vision in various aspects, e.g., dynamic range (sunglasses, night vision goggles), focal length (corrective glasses), field of view (telescopes, microscopes, low-vision glasses [Pel01; Pel02]), spectrum range (thermal goggles), and exposure (stroboscopic/shutter glasses [Koi+12]).

VE devices can be categorized into two different types based on their principles: *direct* and *indirect*. Direct devices like corrective glasses and sunglasses consist of optical elements that directly make use of phenomena of optical physics such as refraction and transmittance. The capabilities of direct devices are limited by those of their optical elements and the human eyes. On the other hand, indirect devices such as night-/thermal-vision goggles use external vision sensors to obtain *super-vision* which is hard or impossible to obtain by direct devices, and users see post-processed images on a display.

Indirect devices can substitute direct devices given an appropriate sensor which reproduces the same vision as the optical components of the direct devices provide. For example, if we capture an image by a camera with a huge zoom lens, and display it on a VST-HMD, we get a virtual telescope [Osk+13]. Furthermore, indirect devices can benefit from the power of computational photography by post-processing raw sensor data as professional astrometric telescope systems do.

However, as a wearable vision system, indirect devices have an essential limitation: they dispose and intercept the user's *direct* view by occluding the real world from the user's eye. Direct devices do not have this limitation, yet they are inflexible in applying different VE effects since each optical effect requires different optical components.

As we mentioned earlier in Sec. 2.2, OST-HMDs integrate digital images into the user's view while keeping the real scene visible through (semi-)transparent optical combiners. The OST-HMDs are potentially capable of incorporating the benefits of both types of VE devices. Along with the recent developments of mobile sensing devices, we believe that the future OST-HMD system will substitute many direct VE devices.

This ultimate goal is, unfortunately, far from what we currently have. The limitations of the current display hardware and computer vision/graphics technologies impede the realization of practical VE devices. Yet, our community has developed essential technologies such that, if

combined all-together, they could establish a more practical system. However, few works tackle VE problems along the context of OST-HMD systems.

Contributions As a summary, our main contributions of this section include the following:

- We provide a theoretical formulation of VE for the defocus correction via OST-HMDs.
- We demonstrate that our VE formulation can improve the defocus effect through conceptual experiments.
- We provide a thorough analysis of the current VE setup including limitations and possible research directions toward the realization of a practical VE system.

11.2 Related Work

The VE concept has close relations to the following areas.

11.2.1 Projector-Camera Systems

An OST-HMD is a projector which displays an image mid air; an eye is a camera which captures the image. Thus, they form a Projector-Camera System (PCS). PCSs have two topics related to VE: improving the appearance of the projected image, and modifying that of the surface on which the projected image appears. The former includes: a defocus-blur correction [OS08], a temporal super-resolution system [SGM12], and a color-correction system [Bim+08]. The latter includes: contrast and resolution enhancement of ePapers [BST11], a dynamic-range enhancement technique [BI08], and a color-enhancement system for visually impaired people [AK10].

The OST-HMD system differs from conventional PCSs in several ways: the image plane floats in mid air rather than being projected on physical surfaces; the spatial relationship between the eye and the display plane is dynamic; and the user-perspective image, i.e., what a user exactly sees, is not easy to obtain. The floating image plane makes a calibration of a system nuisance, and the difficulty of the user-view acquisition even devastates the image rendering process due to missing feedback from the eye.

11.2.2 Computational Photography for Aberration Corrections

Several computational displays account for visual aberrations of human eyes [Pam+12; Hua+13]. Such systems consist of multilayer/light-field displays and render pre-filtered images designed to cancel estimated aberrations. These displays require the user's eye position and the eye aberration profile, in practice, as a point spread function (PSF).

11.2.3 Low-Vision Devices for Visual Impairments

Apart from the pure computer vision/graphics fields, researchers in clinical fields have developed various low-vision devices to help visually impaired people [Pel01; Pel02]. These *direct* devices are composed of several optical lenses to create tailored vision to compensating the patients' degraded vision.

Recently, Huang and Peli [HP14] developed a VE system which provides a patient with an edge-enhanced image for contrast enhancement. They create a user-perspective image by using a planar OST-HMD screen model and a display pose measured manually.

11.2.4 Vision Enhancement in Augmented Reality

In Augmented Reality (AR), there are works on overlaying images on users' views to modify/enhance their vision. Such work includes: an AR-Xray system which integrates occluded scenes into a user's view via a VST-HMD system [AST09], and an AR microscope that provides depth-dependent image augmentations so that viewers can grasp the focal depth of microscope imagery effortlessly [Giu+11].

11.3 Method

VE systems have three key computation steps: To transform sensor images so that the displayed image will coincide with the user's perspective (Step A), to estimate a user's vision, e.g., a PSF and the color sensitivity of a user's eye (Step B), and to preprocess the image to be displayed (or to modulate light from the display) so that the combined stimulus of the image and user's vision creates the desired optical effect (Step C). We refer these steps in the discussion section to associate topics to the steps. We first formulate our method based on ideal assumptions. We then relax it with practical assumptions. Through out this section, we assume that cameras are the pinhole cameras and images are undistorted. $\mathbf{I}$ represents a 2D image. We attach $\star$ to variables for representing that their are true values.

11.3.1 Formulation

Step A is analogous to the rendering process in graphics engines: we need to bake a 2D image out of a 3D scene from the view point of a user-perspective. The intricacy of our setups is that neither we can align a camera to a user's eye position perfectly nor the camera sensor can replicate the eye sensor exactly.

We start with a User-Perspective (UP) camera U, which represents the human eye, placed in the world (Fig. 11.1). The camera sees a scene structure $\mathrm{X_W}$ in the world coordinate system. Typically, $\mathrm{X_W}$ is modeled as a set of 3D surfaces with material information. As the result, the camera records a user-perspective image $\mathbf{I}^{\star}_{\mathrm{U}}$. $\mathbf{I}^{\star}_{\mathrm{U}}$ is a ground-truth image that an emmetropic eye would see, whereas an ametropic eye suffers from an optical aberration $f_{\mathrm{abr}}(\cdot)$ and sees a degraded view $\mathbf{I}^{\star\prime}_{\mathrm{U}} := f_{\mathrm{abr}}(\mathbf{I}^{\star}_{\mathrm{U}})$. Typical aberrations include myopia, hyperopia, and astigma. If we

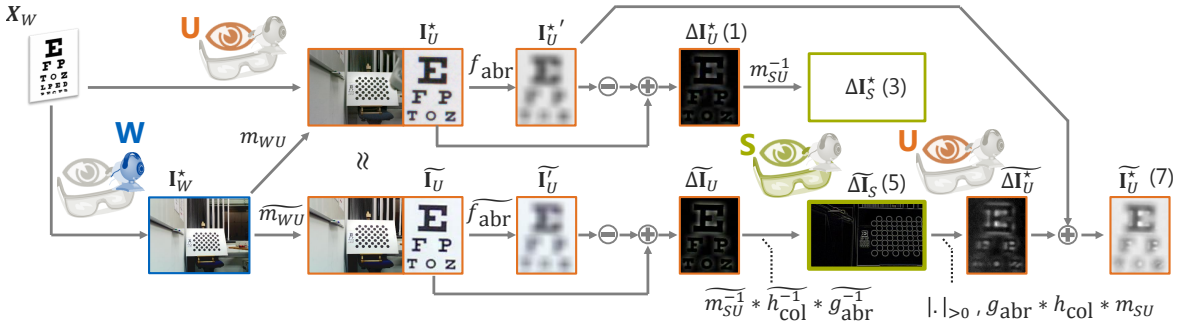


Figure 11.1: Schematic diagram of a general formulation of the vision enhancement for defocus correction. See the method section for more details.

can present a visual stimulus:

$$\Delta\mathbf{I}^\star_{\text{U}} := \mathbf{I}^\star_{\text{U}} - \mathbf{I}^{\star\prime}_{\text{U}} \tag{11.1}$$

to the user's view, the user would regain the regular view as $\mathbf{I}^\star_{\text{U}} = \Delta\mathbf{I}^\star_{\text{U}} + \mathbf{I}^{\star\prime}_{\text{U}}$. We call $\Delta\mathbf{I}^\star_{\text{U}}$ as a compensation image.

We estimate $\Delta\mathbf{I}^\star_{\text{U}}$ from an image taken by a world camera W attached to an OST-HMD. Without loss of generality, we treat W as the origin of the world coordinate system. Similar to the UP camera, W also sees X_W and captures a world-view image $\mathbf{I}^\star_{\text{W}}$. Note that $\mathbf{I}^\star_{\text{W}}$ is different from $\mathbf{I}^\star_{\text{U}}$ since the two cameras have different extrinsic and intrinsic parameters. Therefore, we need to warp $\mathbf{I}^\star_{\text{W}}$ to $\mathbf{I}^\star_{\text{U}}$ by using these parameters. Let $\mathbf{P}_{\text{WU}}$ and $\mathbf{P}_{\text{WW}}$ be world-to-image projection matrices of U and W respectively ($\mathbf{P}_{\text{WW}}$ only has the intrinsic part since W is at the origin). Given $\mathbf{P}_{\text{WU}}$, $\mathbf{P}_{\text{WW}}$ and the 3D structure of X_W, we define an image warping function $m(\cdot \mid \cdot,\cdot,\cdot)$ as

$$\mathbf{I}^\star_{\text{U}} = m(\mathbf{I}^\star_{\text{W}} \mid \text{X}_\text{W}, \mathbf{P}_{\text{WW}}, \mathbf{P}_{\text{WU}}) =: m_{\text{WU}}(\mathbf{I}^\star_{\text{W}}) \tag{11.2}$$

where $m_{AB}(\cdot) := m(\cdot \mid \text{X}_\text{W}, \mathbf{P}_{\text{W}A}, \mathbf{P}_{\text{W}B})$ for given coordinate systems A and B. If the 3D structure is complex, $m_{\text{WU}}(\cdot)$ can be computed by the epipolar geometry with known depth. If the structure is a plane, like our experiment setups, $m_{\text{WU}}(\cdot)$ becomes an image transformation via a homography matrix.

Although we can now compute $\Delta\mathbf{I}^\star_{\text{U}}$ from $\mathbf{I}^\star_{\text{W}}$, we can not directly provide this stimulus to the user's eye: we have to do so via the OST-HMD screen. This introduces the third camera: a screen camera S. Its position is the same as the UP camera, and its image plane (thus orientation) is defined by the image screen S of the OST-HMD. Note that S can be defined as an off-axis pinhole camera under the assumption that the screen, which is floating in mid air, is planar. If S is a real camera, it has projection matrix $\mathbf{P}_{\text{WS}}$ and records an image $\mathbf{I}^\star_{\text{S}}$. When properly transformed, $\mathbf{I}^\star_{\text{S}}$ exactly matches to a part of $\mathbf{I}^\star_{\text{U}}$ since their cameras share the same view point. We now obtain the ideal screen image to be displayed on the OST-HMD:

$$\Delta\mathbf{I}^\star_{\text{S}} := m_{\text{SU}-1}(\Delta\mathbf{I}^\star_{\text{U}}) = m_{\text{SU}-1}(m_{\text{WU}}(\mathbf{I}^\star_{\text{W}}) - f_{\text{abr}}(m_{\text{WU}}(\mathbf{I}^\star_{\text{W}}))) \tag{11.3}$$

where $m_{\mathrm{SU}-1}(\cdot)$, the inverse mapping of another warping function $m_{\mathrm{SU}}(\cdot)$, warps $\mathbf{I}^{\star}_{\mathrm{S}}$ to $\mathbf{I}^{\star}_{\mathrm{U}}$. Finally, displaying $\Delta\mathbf{I}^{\star}_{\mathrm{S}}$ on the screen gives the stimulus:

$$\mathbf{I}^{\star\prime}_{\mathrm{U}} + m_{\mathrm{SU}}(\Delta\mathbf{I}^{\star}_{\mathrm{S}}) = \mathbf{I}^{\star\prime}_{\mathrm{U}} + \Delta\mathbf{I}^{\star}_{\mathrm{U}} = \mathbf{I}^{\star}_{\mathrm{U}}. \tag{11.4}$$

In practice, we only have the estimates of the functions in the above formulation: $\widetilde{m_{\mathrm{WS}}}(\cdot)$, $\widetilde{m_{\mathrm{SU}-1}}(\cdot)$, $\widetilde{f_{\mathrm{abr}}}(\cdot)$, and $\widetilde{m_{\mathrm{WS}}}(\cdot)$. Thus the screen image that we actually display becomes

$$\widetilde{\Delta\mathbf{I}_{\mathrm{S}}} := \widetilde{m_{\mathrm{SU}-1}}(\widetilde{\Delta\mathbf{I}_{\mathrm{U}}}),\ \widetilde{\Delta\mathbf{I}_{\mathrm{U}}} := \widetilde{\mathbf{I}_{\mathrm{U}}} - \widetilde{\mathbf{I}'_{\mathrm{U}}}, \tag{11.5}$$

$$\widetilde{\mathbf{I}_{\mathrm{U}}} := \widetilde{m_{\mathrm{WU}}}(\mathbf{I}^{\star}_{\mathrm{W}}),\ \widetilde{\mathbf{I}'_{\mathrm{U}}} := \widetilde{f_{\mathrm{abr}}}(\widetilde{\mathbf{I}_{\mathrm{U}}}). \tag{11.6}$$

By displaying $\widetilde{\Delta\mathbf{I}_{\mathrm{S}}}$ on the HMD screen, the user perceives

$$\widetilde{\mathbf{I}^{\star}_{\mathrm{U}}} := \mathbf{I}^{\star\prime}_{\mathrm{U}} + \widetilde{\Delta\mathbf{I}^{\star}_{\mathrm{U}}} \approx \mathbf{I}^{\star}_{\mathrm{U}}, \tag{11.7}$$

where $\widetilde{\Delta\mathbf{I}^{\star}_{\mathrm{U}}} := m_{\mathrm{SU}}(\widetilde{\Delta\mathbf{I}_{\mathrm{S}}})$. Here we applied the true warping function $m_{\mathrm{SU}}(\cdot)$ since this is a physical process.

Note that ordinary displays cannot handle *negative* values. We can only display $|\widetilde{\Delta\mathbf{I}_{\mathrm{S}}}|_{>0}$ instead of $\widetilde{\Delta\mathbf{I}_{\mathrm{S}}}$ where $|\cdot|_{>0}$ is a function which sets negative values of the image pixels to zero. We refer the VE with the former as Enhanced + (plus) and the latter Enhanced +- (plus-minus).

We have, so far, ignored some effects related to the OST-HMD optics: a color distortion by the virtual screen and an optical aberration effect against the displayed image. We further integrate these effects into the above formulation.

Color Distortion of Virtual Screen Image Eye-HMD system has a color distortion stems from several conversions between analog and digital image signals. The world camera receives light from the world and sends converted digital values to the display. The display emits new light based on the values. Finally, the Up camera receives the display light, and outputs new digital values. In general, this final color differs from the color from the world camera.

Let $h_{\mathrm{col}}(\cdot)$ be such a digital color distortion applied to the final image perceived by the UP camera. Then, the UP camera sees an image as $h_{\mathrm{col}}(\widetilde{\mathbf{I}_{\mathrm{U}}})$ at Eq. 11.6. We estimate the inverse of the distortion as $\widetilde{h^{-1}_{\mathrm{col}}}(\cdot)$, and redefine the displayed image $\widetilde{\Delta\mathbf{I}_{\mathrm{S}}}$ (Eq. 11.5) as

$$\widetilde{\Delta\mathbf{I}_{\mathrm{S}}} = \widetilde{m_{\mathrm{SU}-1}}(\widetilde{h^{-1}_{\mathrm{col}}}(\widetilde{\Delta\mathbf{I}_{\mathrm{U}}})) = \widetilde{m_{\mathrm{SU}-1}}(\widetilde{h^{-1}_{\mathrm{col}}}(\widetilde{\mathbf{I}_{\mathrm{U}}} - \widetilde{\mathbf{I}'_{\mathrm{U}}})). \tag{11.8}$$

The UP camera finally perceives an enhanced view as $\widetilde{\mathbf{I}^{\star}_{\mathrm{U}}} = \mathbf{I}^{\star\prime}_{\mathrm{U}} + h_{\mathrm{col}}(m_{\mathrm{SU}}(\widetilde{\Delta\mathbf{I}_{\mathrm{S}}}))$ instead of Eq. 11.7.

Optical Aberration of Virtual Screen Image An HMD screen causes another aberration effect. In general, the image screen created by an OST-HMD does not necessarily appear at the same distance as an object on which a user is currently focusing. This misfocus causes another aberration on the screen image. Let $g_{\mathrm{abr}}(\cdot)$ such an aberration which is different from $f_{\mathrm{abr}}(\cdot)$,

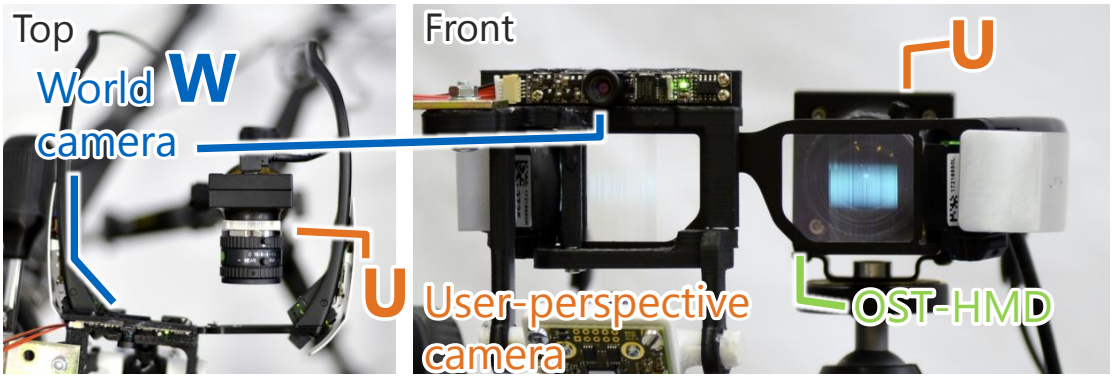


Figure 11.2: An OST-HMD setup used in our experiments. (Left) A top view of the setup. (Right) Front view. A blue color is displayed on the OST-HMD: Lumus DK32. The UP camera is placed behind the HMD, and the world camera is mounted on the display frame. Both cameras see a printed Snellen Chart (image by Jeff Dahl) as a reference object .

following the same derivation in the previous section, we redefine $\widetilde{\Delta \mathbf{I}_{\mathrm{S}}}$ as

$$\widetilde{\Delta \mathbf{I}_{\mathrm{S}}} := \widetilde{m_{\mathrm{SU}^{-1}}}(\widetilde{g_{\mathrm{abr}}^{-1}}(\widetilde{\Delta \mathbf{I}_{\mathrm{U}}})) = \widetilde{m_{\mathrm{SU}^{-1}}}(\widetilde{g_{\mathrm{abr}}^{-1}}(\widetilde{\mathbf{I}_{\mathrm{U}}} - \widetilde{\mathbf{I}'_{\mathrm{U}}})), \tag{11.9}$$

where $\widetilde{g_{\mathrm{abr}}^{-1}}(\cdot)$ is an estimate of the inverse of $g_{\mathrm{abr}}(\cdot)$. The UP camera finally perceives $\widetilde{\mathbf{I}^{\star}_{\mathrm{U}}} = \mathbf{I}^{\star\prime}_{\mathrm{U}} + g_{\mathrm{abr}}(m_{\mathrm{SU}}(\widetilde{\Delta \mathbf{I}_{\mathrm{S}}}))$.

Generalized Formulation We merge the color and aberration correction functions. Assuming that the color distortion $h_{\mathrm{col}}(\cdot)$ is pixel-wise, we place the $h_{\mathrm{col}}(\cdot)$ inside the aberration function $g_{\mathrm{abr}}(\cdot)$ as $g_{\mathrm{abr}}(h_{\mathrm{col}}(\cdot))$. Then we obtain the image to be displayed on the screen as follows:

$$\begin{aligned} \widetilde{\Delta \mathbf{I}_{\mathrm{S}}} &= \widetilde{m_{\mathrm{SU}^{-1}}}(\widetilde{h_{\mathrm{col}}^{-1}}(\widetilde{g_{\mathrm{abr}}^{-1}}(\widetilde{\mathbf{I}_{\mathrm{U}}} - \widetilde{\mathbf{I}'_{\mathrm{U}}}))) && (11.10) \\ &= \widetilde{m_{\mathrm{SU}^{-1}}}(\widetilde{h_{\mathrm{col}}^{-1}}(\widetilde{g_{\mathrm{abr}}^{-1}}(\widetilde{\mathbf{I}_{\mathrm{U}}} - \widetilde{f_{\mathrm{abr}}}(\widetilde{\mathbf{I}_{\mathrm{U}}})))). && (11.11) \end{aligned}$$

The perceived view finally becomes:

$$\widetilde{\mathbf{I}^{\star}_{\mathrm{U}}} = \mathbf{I}^{\star\prime}_{\mathrm{U}} + g_{\mathrm{abr}}(h_{\mathrm{col}}(m_{\mathrm{SU}}(\widetilde{\Delta \mathbf{I}_{\mathrm{S}}}))). \tag{11.12}$$

The above formulation does not define how to compute each function. Because their definitions are depending on various assumptions on the scene, the display, and the eye. In the experiment section, we introduce our assumptions and show a simplified, concrete formulation based on the above general formulation.

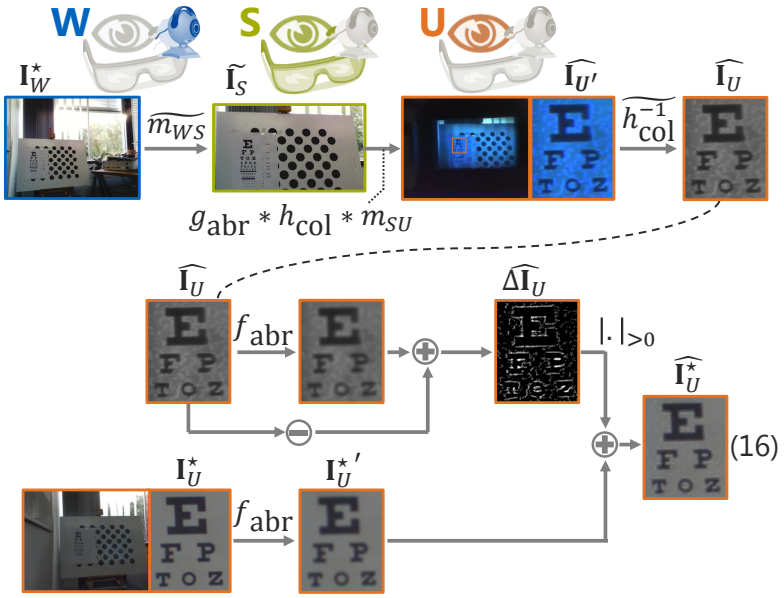


Figure 11.3: Schematic diagram of vision enhancement flow in the conceptual setup (Experiment 1).

11.4 Experiments

We conduct two proof-of-concept experiments. Experiment 1 demonstrates the formulation under a controlled environment with various assumptions to investigate the potential of the VE. Experiment 2 relaxes the assumptions to show the limited performance of the VE with the current technology. In both the experiments, we use a UP camera instead of a real user for obtaining objective measurements.

A thorough discussion including the limitations of the setups and possible solutions for realizing a practical system is presented in the discussion section.

11.4.1 Hardware Setup

We have built an OST-HMD system equipped with an outward looking camera as described below and in Fig. 11.2. We use Lumus DK-32, an OST-HMD with 1280×720 resolution. The left eye display is used for the current setup. For the world camera, we use a USB 2.0 camera from Delock. The camera has 1600×1200 image resolution with 64° field of view. For the UP camera placed behind the OST-HMD, we use UI-1240ML-C-HQ from iDS. The camera holds 1280×1024 image resolution together with an 8mm C-mount lens. As a scene object $\mathrm{X_w}$, we use an acuity chart set on a planar calibration board. The board is placed about 1.5 m away from the display.

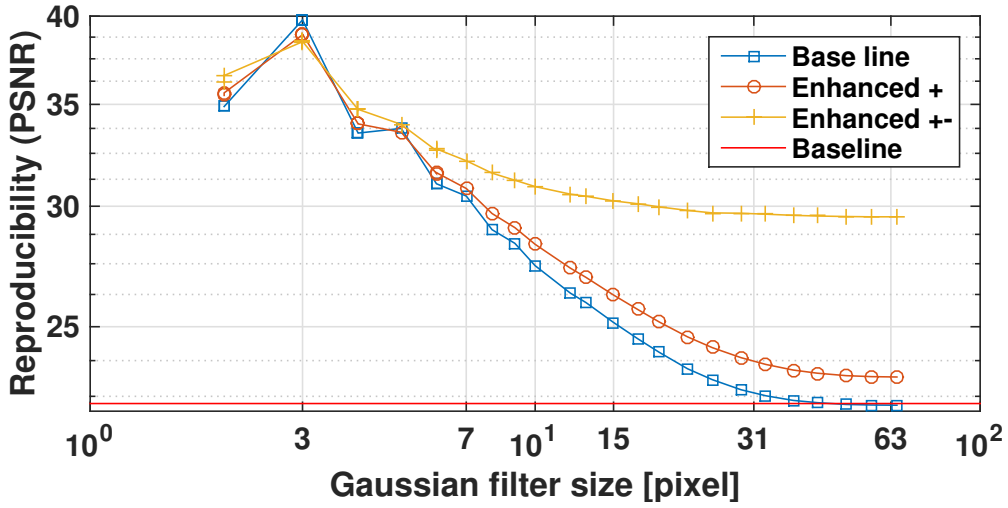


Figure 11.4: The result of the experiment 1. The log-scale X axis is the size of Gaussian blur filter used to compute degraded user-perspective images $\mathbf{I}^{\star\prime}_{\mathrm{U}}$. The log-scale Y axis is the PSNR between the ground truth image $\mathbf{I}^{\star}_{\mathrm{U}}$ and: $\mathbf{I}^{\star\prime}_{\mathrm{U}}$ (Degraded), the enhanced images with the positive image $\widehat{\mathbf{I}}^{\star}_{\mathrm{U}>0}$ ($\mathrm{E}^{\star}$, Enhanced $+$); with the complete filter $\widehat{\mathbf{I}}^{\star}_{\mathrm{U}}$ ($\mathrm{E}^{\star}_{>0}$, Enhanced $+$-), and the UP image $\widehat{\mathbf{I}}_{\mathrm{U}}$ created from the world view $\mathbf{I}^{\star}_{\mathrm{W}}$ (Baseline). The VE technique improves the quality of the degraded images.

11.4.2 Experiment 1

This experiment is to assesses a potential performance of the VE by simulating a setup where a perfect OST-HMD is available. We first explain the simulation formulation, then the actual data collection procedure, and the simulation results.

Simulation Procedure and Assumptions Instead of directly obtaining $\widetilde{\Delta\mathbf{I}^{\star}_{\mathrm{U}}}$ by displaying $\widetilde{\Delta\mathbf{I}_{\mathrm{S}}}$, we partially synthesize it by introducing several assumptions (Fig. 11.3). Some of the assumptions naturally stem from the fact that we use an UP camera instead of an user. Some others are for simplifications due to our lack of methods to estimate some of the true functions. We emphasize it here that the upcoming sections investigate potential solutions for incorporating real users and removing the simplifications. We now explain our simplified VE procedure.

First of all, the UP camera and the world camera capture $\mathbf{I}^{\star}_{\mathrm{U}}$ and $\mathbf{I}^{\star}_{\mathrm{W}}$ respectively as same as the original formulation. We then warp $\mathbf{I}^{\star}_{\mathrm{W}}$ directly by $\widetilde{m_{\mathrm{WS}}}(\cdot)$, and display $\widetilde{\mathbf{I}}_{\mathrm{S}} := \widetilde{m_{\mathrm{WS}}}(\mathbf{I}^{\star}_{\mathrm{W}})$ on the HMD. While blocking the world light, we let the UP camera capture $\widetilde{\mathbf{I}}_{\mathrm{S}}$ as $\widehat{\mathbf{I}_{\mathrm{U}'}} := g_{\mathrm{abr}}(h_{\mathrm{col}}(m_{\mathrm{SU}}(\widetilde{\mathbf{I}}_{\mathrm{S}})))$. Finally, based on the assumption that light is additive, we process $\mathbf{I}^{\star}_{\mathrm{U}}$ and $\widehat{\mathbf{I}_{\mathrm{U}'}}$ in a software to synthesize a deblurred image.

We introduce another assumption to simplify the warping function: we limit the scene structure X_{W} to a planar 3D surface with a 2D marker. The UP camera and the world camera track the 2D marker, and we compute the relative poses among the cameras and the plane. As the result, we obtain projection matrices $\mathbf{P}_{\mathrm{WU}}$ and $\mathbf{P}_{\mathrm{WW}}$. Also, given an OST-HMD calibration method explained

in the next section, we obtain the projection matrix $\mathbf{P}_{\mathrm{WS}}$ as well. As the result, $\widetilde{m_{\mathrm{WS}}}(\cdot)$ becomes an image translation by a homography matrix.

We compute a compensation image $\widehat{\Delta \mathbf{I}_{\mathrm{U}}}$ by applying the estimates of the inverse of $h_{\mathrm{col}}(\cdot)$ and $g_{\mathrm{abr}}(\cdot)$:

$$\widehat{\mathbf{I}}_{\mathrm{U}} := \widetilde{h_{\mathrm{col}}^{-1}}(\widetilde{g_{\mathrm{abr}}^{-1}}(\widehat{\mathbf{I}}_{\mathrm{U}'})),\ \widehat{\Delta \mathbf{I}}_{\mathrm{U}} := \widehat{\mathbf{I}}_{\mathrm{U}} - f_{\mathrm{abr}}(\widehat{\mathbf{I}}_{\mathrm{U}}), \tag{11.13}$$

where we assume $f_{\mathrm{abr}}(\cdot)$ as a Gaussian blur with a known diameter σ [pixel]. We synthesize a degraded UP view as $\mathbf{I}_{\mathrm{U}}^{\star\prime} := f_{\mathrm{abr}}(\widehat{\mathbf{I}}_{\mathrm{U}})$, and evaluate the VE performance by changing σ. To ignore $g_{\mathrm{abr}}^{-1}(\cdot)$, We also place X_{W} almost at the same distance as the image screen when seen by the UP camera. To simplify the color correction, we only consider a gray-scale color, and we approximate the inverse of the color correction $h_{\mathrm{col}}^{-1}(\cdot)$ as a scaling factor c defined as follows:

$$\widetilde{h_{\mathrm{col}}^{-1}}(\cdot) = c^{-1} * (\cdot), \tag{11.14}$$

where $c := \mathrm{mean}(\widehat{\mathbf{I}_{\mathrm{U}'}}) / \mathrm{mean}(\mathbf{I}_{\mathrm{U}}^{\star})$ .mean$(\cdot)$ is the mean color of a given image.

Now, our approximated compensation image and our enhanced image become

$$\widehat{\Delta \mathbf{I}_{\mathrm{U}}} = c^{-1} * \widehat{\mathbf{I}_{\mathrm{U}'}} - f_{\mathrm{abr}}(c^{-1} * \widehat{\mathbf{I}_{\mathrm{U}'}}). \tag{11.15}$$

$$\widehat{\mathbf{I}}_{\mathrm{U}}^{\star} := \mathbf{I}_{\mathrm{U}}^{\star\prime} + \widehat{\Delta \mathbf{I}_{\mathrm{U}}},\ \widehat{\mathbf{I}}_{\mathrm{U}_{>0}}^{\star} := \mathbf{I}_{\mathrm{U}}^{\star\prime} + |\widehat{\Delta \mathbf{I}_{\mathrm{U}}}|_{>0}. \tag{11.16}$$

We compute the enhancement error as

$$\mathrm{E}^{\star} := \mathrm{PSNR}(\mathbf{I}_{\mathrm{U}}^{\star}, \widehat{\mathbf{I}}_{\mathrm{U}}^{\star}),\ \mathrm{E}_{>0}^{\star} := \mathrm{PSNR}(\mathbf{I}_{\mathrm{U}}^{\star}, \widehat{\mathbf{I}}_{\mathrm{U}_{>0}}^{\star}), \tag{11.17}$$

where $\mathrm{PSNR}(\cdot, \cdot)$ represents Peak Signal-to-Noise Ratio (PSNR) of two given images. Similar images yield higher PSNR, and go to infinity if the two images are identical. Note that $\mathrm{E}_{>0}^{\star} \geqq \mathrm{E}^{\star}$ holds. Note that the above derivation of the compensation image is valid only if our generalized formulation is true (Eq. 11.11 and 11.12).

Data Acquisition for $\mathbf{I}_{\mathrm{U}}^{\star}$ and $\widehat{\mathbf{I}_{\mathrm{U}'}}$ Our data collection procedure is as follows:

1. Calibrate the world/UP cameras and the HMD screen.
2. Place the UP camera toward the board X_{W}, and capture $\mathbf{I}_{\mathrm{U}}^{\star}$.
3. Place the HMD in front of the UP camera and capture $\mathbf{I}_{\mathrm{W}}^{\star}$.
4. Compute $\mathbf{P}_{\mathrm{WW}}$ and $\mathbf{P}_{\mathrm{WS}}$ (also $\mathbf{P}_{\mathrm{WU}}$ for the experiment 2) from the Step 1 and Step 2.
5. From $\mathbf{I}_{\mathrm{U}}^{\star}$ and $\mathbf{I}_{\mathrm{W}}^{\star}$, compute the 6-DoF pose between the cameras and the parameters of the board as a 4D vector.
6. From the vector, $\mathbf{P}_{\mathrm{WW}}$, and $\mathbf{P}_{\mathrm{WS}}$, estimate $\widetilde{m_{\mathrm{WS}}}(\cdot)$ as a 2D homography mapping.

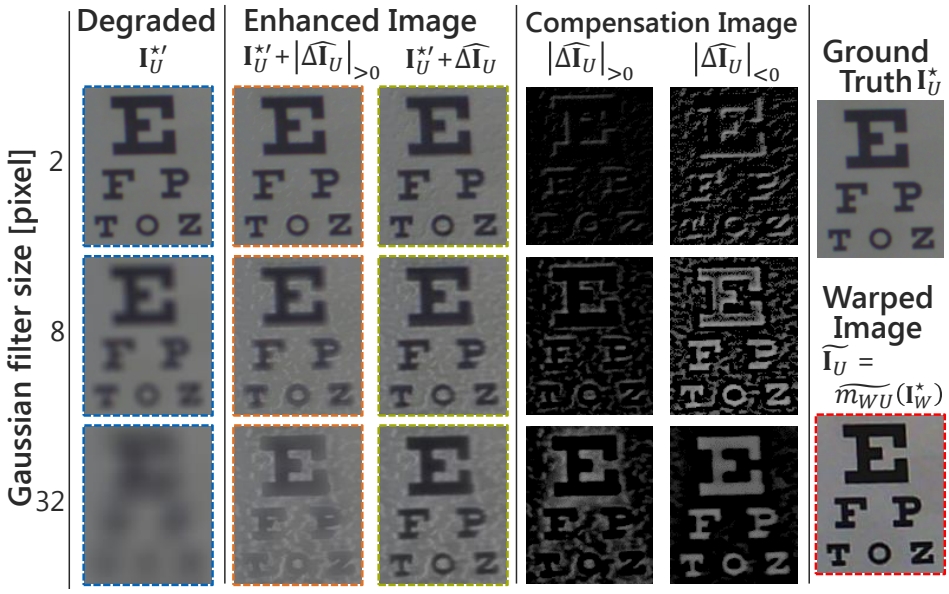


Figure 11.5: Sample image from the experiment 1. The enhanced images are clearer compared to the degraded images. Note that the brightness and contrast of the compensation images here are modified to +40/-40% for a presentation purpose. See also Fig. 11.4.

7. Display the warped image $\widetilde{\mathbf{I}_S}$ on the display. Let the UP camera capture the displayed image as $\widehat{\mathbf{I}_{U'}}$ while blocking the world light by, e.g., making the room completely dark.

At Step 1, we treat the display screen as a virtual 3D plane with its pose defined relative to the UP camera. For the OST-HMD calibration, we used method described in [IK14a].

After this process, we obtain $\mathbf{I}_U^{\star}$ and $\widehat{\mathbf{I}_{U'}}$ for the evaluation. Note that we only compare a region of the images which includes the acuity chart (see Fig. 11.5).

Enhancement Result of Experiment 1 Figures 11.4 and 11.5 show the results. The red line (Baseline) in Fig. 11.4 shows $\mathrm{PSNR}(\mathbf{I}_U^{\star}, \widetilde{\mathbf{I}_U})$ as a baseline. This is equivalent to a user perspective VST-HMD setup where the display replaces the actual user view by the world camera. Note that we also adjusted the color balance of $\widetilde{\mathbf{I}_U}$ to that of $\mathbf{I}_U^{\star}$ by Eq. 11.14. The blue line with square markers (Degraded) is a plot of $\mathrm{PSNR}(\mathbf{I}_U^{\star}, \mathbf{I}_U^{\star\prime})$, which is what a degraded eye would see the world without VE. The orange line with circle markers (Enhanced +) is $\mathrm{E}^{\star}$ and yellow with crosses (Enhanced +-) is $\mathrm{E}_{>0}^{\star}$ from Eq. 11.17.

As expected, all methods suffer from the increasing amount of the Gaussian blur. However, both Enhanced + and Enhanced +- in general improve the degraded images while keeping the qualities higher than just displaying the warped world-camera view, i.e., Baseline. Enhanced +- shows a significant improvement over Enhanced +. This suggests developing an adjustable opaque layer in the display is desirable for practical VE systems. Such systems can visualize *negative* values. The negative color contributes dominantly when extreme blur is present (see the third row of Fig. 11.5).

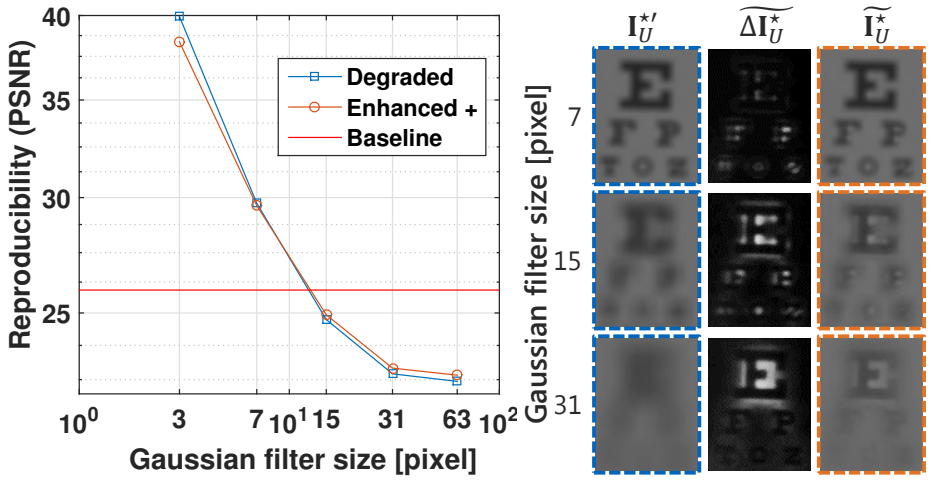


Figure 11.6: Result of experiment 2. (Left) Error plot. (Right) Result images, where we modified the brightness of the compensation images +70% for the presentation purpose.

11.4.3 Experiment 2

This experiment follows the original formulation in the method section (Fig. 11.1) to assess a more realistic setup that a practical VE system should follow. Data collection is done in a similar manner as the previous section.

We first prepared $\widetilde{\Delta\mathbf{I}_S}$. For computing $\widetilde{\mathbf{I}'_U}$, we again treated $\widetilde{f_{abr}}(\cdot)$ as a Gaussian blur. The same blur is added to $\mathbf{I}_U^{\star}$ for synthesizing $\mathbf{I}_U^{\star\prime}$. The UP camera captured the displayed image $\widetilde{\Delta\mathbf{I}_U^{\star}}$. When capturing, we blocked the world light from the display in the same way as the experiment 1. Then we computed $\widetilde{\mathbf{I}_U^{\star}}$ (Eq. 11.12). Note that we had to tune the image gamma of $\widetilde{\Delta\mathbf{I}_U^{\star}}$ before fusing it with $\mathbf{I}_U^{\star\prime}$ to reduce the backlight ambient color of the display panel that made pixels bluish even if the digital color of the pixels are set to zero.

Figure 11.6 summarizes the result. Compared to the experiment 1, the improvement was negligibly small. One of the main cause might be the color distortion $h_{col}(\cdot)$ of the display, which distorts the brightness of image and thus the compensation image.

11.5 Discussion

11.5.1 Limitations of the Current Experiments

Our current setups, especially in experiment 1, use various assumptions and simplifications. We list them with the corresponding step characters from the method section:

L1 (A) X_W is limited to a 3D plane and 6-DoF pose between the cameras are known.

L2 (A) We refined manually the 2D position between $\mathbf{I}_U^{\star}$ and $\widehat{\mathbf{I}_{U'}}$ before computing the compensation images.

L3 (A) An ideal user-perspective image $\mathbf{I}_{\mathrm{U}}^{\star}$ is available, and the eye position is known with respect to the HMD camera.

L4 (B) The aberration model $f_{\mathrm{abr}}(\cdot)$ is a Gaussian blur.

L5 (B,C) Images are in gray. Only the distortion of color intensity is corrected by a simple color correction (Eq. 11.14).

L6 (B,C) Optical elements do not distort world illumination.

L7 (C) $g_{\mathrm{abr}}^{-1}(\cdot)$ is the identity function, i.e. we ignored it.

L8 (C) The relative image resolution of the display screen is higher than that of the user-perspective image.

These issues have to be solved for a practical VE. In the next section, we discuss practical solutions for each relevant topic.

11.5.2 Issues toward Practical Vision Enhancement

We analyze issues reaming in the current VE system and discuss possible solutions. We clustered the issues along the three steps of VE, and referred corresponding limitations in the headlines.

11.5.2.1 (A) Transformation of Sensor Images

UP Rendering with Arbitrary Scene Structure (L1,L2,L3): Our VE method requires to map the world view to the user perspective view. The mapping changes by both the eye position of a user and the structure of a scene. There are hardware and software solutions to estimate a correct mapping.

A hardware solution is to align the optical paths of the world camera and the eye (the UP camera) by a half mirror [SF11]. This way, both cameras optically share the same viewpoint, and we may opt out the X_{w} from the warping function. However, the mirror has to be dynamically configurable to keep the virtual center of the camera according to the dynamic eye position with respect to the world camera.

A software solution is to compute the depth of the scene X_{w} and warp the current world view to the UP view while taking the depth of each pixel into account [TIS13; Bar+12]. This solution requires a 3D sensing. Since two viewpoints are different, we face with the occlusion problem that a part of the scene visible by an UP camera is not visible by the world camera. Such occluded part of the scene thus can not be displayed.

Both the solutions require a high-speed 3D eye tracking. With an eye-tracking camera, another possible technique to provide $\mathbf{I}_{\mathrm{U}}^{\star}$ directly is the corneal imaging [NNT13]. It analyzes reflected image on eye cornea to see what a user actually see.

OST-HMD Calibration (L2,L3): Related to the mapping issue above, another issue is pertaining to the calibration of the OST-HMD screen. The pose of the screen image is necessary to compute the correct projection for the virtual camera S. Although common HMD calibration methods calibrate the screens as 3D planes, this is often invalid due to the complex optics of the HMD. Furthermore, the HMD optics distort the user perspective view itself as corrective glasses do. An ideal HMD calibration method must take these issues into its HMD model for the calibration.

11.5.2.2 (B) Estimation of User's Vision

Appearance Correction (L4,L5,L6): Even if we have a correct UP rendering, we still need to estimate the color distortion, $h_{\mathrm{col}}(\cdot)$, so that a displayed color perceived by a user is consistent with the visual stimulus that the user receives from the scene directly. A solution is to calibrate the color of the display (and the world camera) beforehand by a UP camera. Some work for such OST-HMD color corrections already exists [Sri+13; Dav+14].

Another unsolved, challenging issue is that photoreceptor cells of eyes have different sensitivity than image sensors.

Eye Aberration Estimation (L4): In our experiment, we assumed that the eye aberration, $f_{\mathrm{abr}}(\cdot)$, is a simple Gaussian blur. Namely, we only considered a defocus basis of Zernike polynomials, a common aberration model employed in optometry [BQ11]. Optometry researchers have worked on estimating the profile with wave-front sensors [G+94; Pla+01; LL10] or video-based techniques [Sur+07]. Recently a mobile system has been proposed [Pam+10].

11.5.2.3 (C) Preprocessing and Rendering of Filter Images

Image Rendering (L5,L6,L7): In addition to $\widetilde{f_{\mathrm{abr}}}(\cdot)$, we need the second aberration $\widetilde{g^{-1}_{\mathrm{abr}}}(\cdot)$ for the virtual screen. If the image screen is at the same distance as an object at which a user is focusing, we may thus ignore this aberration or treat is as $\widetilde{f_{\mathrm{abr}}}(\cdot)$. Practically, this assumption is unlikely. If we have the aberration estimate $\widetilde{f_{\mathrm{abr}}}(\cdot)$, the display screen pose, and the scene information $\mathrm{X_w}$, then it would be possible to estimate $\widetilde{g_{\mathrm{abr}}}(\cdot)$. We then invert $\widetilde{g_{\mathrm{abr}}}(\cdot)$. A software solution is to employ an image deconvolution method such as [OS08] to compute $\widetilde{g^{-1}_{\mathrm{abr}}}(\cdot)$.

Another solution by hardware is a retinal HMD. Its rendering is unaffected by crystalline lenses. The display employs the Maxwellian-view optical system [Wes66], and realizes focus-free images [Asa+03], thus we can physically ignore $\widetilde{g_{\mathrm{abr}}}(\cdot)$.

Resolution between eyes and OST-HMDs (L8): This would be most challenging issue in terms of hardware. The maximum angular resolution of the human eye around the fovea is approximately half an arcsecond, about 8.3e-3 degrees. The display we used has 40-degrees field of view with the 1280×720 pixels, which yields about 31e-3 degrees at the finest. Thus the display resolution is still far lower than that of our eyes. As far as we know, there have been no

OST-HMDs thus far with resolutions comparable to that of human eye. Again retinal displays would be a possible solution with a foveated rendering.

11.5.2.4 Other Issues

Vision Enhancement against Corrective Glasses Corrective glasses are well-established direct, analog VE devices for aberration corrections. Potential benefits of our VE system based on OST-HMDs over such devices are: personalization of enhancement depending on users' aberration types that change over age, correction of higher-order aberration that the glasses can not correct, and simultaneous enhancement of other vision problems such as color blindness.

Overcoming Physical Limit of our Visual Acuity In other word, can we make $\widetilde{\mathbf{I}}^{\star}_{\mathrm{U}} = \mathbf{I}^{\star\prime}_{\mathrm{U}} + \widetilde{\Delta\mathbf{I}}^{\star}_{\mathrm{U}}$ better than $\mathbf{I}^{\star}_{\mathrm{U}}$?

An interesting question of the VE for the defocus correction is whether emmetropic people can benefit from the system. For example, if our display has a higher resolution than that of eye retina, then the display can create a *super-resolution* image $m_{\mathrm{SU}}(\widetilde{\Delta\mathbf{I}}_{\mathrm{S}})$ which is finer than the eye retina can perceive. There is no benefit of doing this in terms of the spatial resolution. Perhaps, a vibro-imaging system [FA12] can overcome this limitation by temporally modulating the display image to achieve a perceptual *super-resolution*.

11.6 Summary

This section proposes a VE concept for defocus correction of human eyes via OST-HMDs. Our main contributions are: (1) We provide a theoretical formulation of VE while incorporating with constraints of the optical relationship between the HMDs and human eyes. (2) We conduct proof-of-concept experiments, with cameras and an OST-HMD, to demonstrate that the method improves a degraded image quality. (3) More importantly, we provide a thorough analysis of the current VE setup including limitations, issues and possible solutions toward the realization of a practical VE system.

Future work directions involve: OST-HMD color correction, eye aberration estimation, deconvoluted image rendering, and extension to the full-color imagery. Besides on that, a study with actual users is also desirable. We hope this work will serve as a foundation for improving the VE techniques.

Part V

Conclusion and Future Work

12 Conclusion and Future Work

We present the conclusion of this dissertation, and suggest future research directions.

12.1 Conclusion

Over the dissertation, we discussed how we can improve the realism of AR experiences with OST-HMDs. We focused on the spatial calibration problem in the displays, and proposed an automated method which eliminates troublesome manual interaction. We further tried to understand the eye-HMD system, and proposed a distortion correction method to remove optical aberration effects in OST-HMDs.

In my vision, all the above works are actually a few of the many milestones to fully utilize near-eye displays to enhance our vision capability. If we have a perfect OST-HMD that can create imageries indistinguishable to the human vision, I believe such a system would support our visual judgments in daily life by assisting our vision skills. We even might not notice that we are actually using the system – it is already an extension of your body like shoes and clothes.

Having this vision, we also explored a potential future application that can only be possible with such perfectly calibrated OST-HMDs. We built a proof-of-concept OST-HMD setup and demonstrate, in simulation, that it is possible to correct eye aberration by overlaying a compensation image into a user's field of view.

Our future works naturally follow this path to improve the realism of AR experiences further and to seek for vision augmentation applications.

12.2 Future Works

We describe our short-term and our long-term research directions. The short term directions involve in to further improve the current eye-HMD model for more accurate calibration. The long-term directions is related to enhancing our vision capability via OST-HMD systems.

Improving the eye-HMD model for indistinguishable registration quality Although we have improved the calibration accuracy. We have not yet achieved calibration accuracy which is smaller than the standard human eyesight (0.016° in the viewing angle at our fovea). Up to now, we estimate the optical characteristic of OST-HMDs in the geometrical optics sense. A more rigorous model for the optical aberration (such as point spread function [Hei+13]) might improve the accuracy even further. Furthermore, our current model does not consider chromatic aberration.

Implementing a real time system incorporated with the automated method A practical OST-HMD has to keep tracking our eyes and, accordingly, update the mapping between the 3D world and the image screen at every single frame. Strictly speaking, we even need to update the mapping during a GPU is rendering the current frame due to the saccadiac eye movement. To realize such a system, we would need to integrate a high-speed eye tracking system in an OST-HMD.

Vision Adapting Image Rendering As we introduced in Part IV, OST-HMDs have potential to enhance and augment our visual perception. Realizing such technology requires an OST-HMD system to have an automated calibration system and, more importantly, an optometric system which can measure the current state of our eyes in real time.

This future requirement brings us a concept: Vision Adapting Image Rendering (VAIR) for OST-HMDs. With VAIR, an OST-HMD would dynamically adapt an AR content based on the state of our eyes. Such a state cab be: eye position, divergence, accommodation, and even the chromatic profile of eye's retinas. An example is our automated calibration method. The method updates a projection matrix from the world to the image screen based on the current 3D position of a user's eye. Another example is deconvolutional displays that optimize images based on user's eye sight [Hua+12; Hua+13]. The VAIR concept is also related to some works that investigate how to model and measure eye's aberration profile [BQ11; Pam+10].

12.3 Closing Remark

I wonder what did researchers in the 60's think when they saw Ivan Sutherland's HMD for the first time. Would they be impressed by the demonstration of the display? Or, would they rather think his remark on the ultimate display room – a display which can control the existence of matter, is far-fetched?

Working on OST-HMDs has been a hard fun. Up to now, the current AR experiences with the displays do not even show us how far we are from his ultimate vision. If the well-known dictum from Arthur C. Clarke, "*Any sufficiently advanced technology is indistinguishable from magic*"[1], is true, the current HMDs are not sufficiently advanced yet. There is thus still some room for us to improve them. The one thing I am sure is that at least this thesis has made an additional step towards the distant horizon. I think that is fine for now; I will keep walking.

[1] Interestingly, it was mentioned just two years before Sutherland's work [Cla62]

List of Figures

List of Tables

Bibliography

[AB94] R. Azuma and G. Bishop. “Improving static and dynamic registration in an optical see-through HMD.” In: *Proceedings of ACM SIGGRAPH 1994*. 1994, pp. 197–204. DOI: 10.1145/192161.192199.

[AB95] R. Azuma and G. Bishop. “A frequency-domain analysis of head-motion prediction.” In: *Proceedings of ACM SIGGRAPH 1995*. 1995, pp. 401–408.

[AK10] T. Amano and H. Kato. “Appearance control by projector camera feedback for visually impaired.” In: *CVPRW*. IEEE. 2010, pp. 57–63.

[AMH12] I. Arief, S. McCallum, and J. Y. Hardeberg. “Realtime estimation of illumination direction for augmented reality on mobile devices.” In: *Color and Imaging Conference*. Vol. 2012. 1. Society for Imaging Science and Technology. 2012, pp. 111–116.

[Asa+03] N. Asai, R. Matsuda, M. Watanabe, H. Takayama, S. Yamada, A. Mase, M. Shikida, K. Sato, M. Lebedev, and J. Akedo. “Novel high resolution optical scanner actuated by aerosol deposited PZT films.” In: *Micro Electro Mechanical Systems (MEMS)*. IEEE. 2003, pp. 247–250.

[AST09] B. Avery, C. Sandor, and B. H. Thomas. “Improving spatial perception for augmented reality x-ray vision.” In: *VR*. 2009, pp. 79–82.

[Axh+10] M. Axholt, M. Skoglund, S. D. Peterson, M. D. Cooper, T. B. Schön, F. Gustafsson, A. Ynnerman, and S. R. Ellis. “Optical See-Through Head Mounted Display Direct Linear Transformation Calibration Robustness in the Presence of User Alignment Noise.” In: *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*. Vol. 54. 28. SAGE Publications. 2010, pp. 2427–2431.

[Axh+11] M. Axholt, M. A. Skoglund, S. D. O’Connell, M. D. Cooper, S. R. Ellis, and A. Ynnerman. “Parameter Estimation Variance of the Single Point Active Alignment Method in Optical See-Through Head Mounted Display Calibration.” In: *Proceedings of VR*. IEEE. 2011, pp. 27–34.

[Axh11] M. Axholt. “Pinhole Camera Calibration in the Presence of Human Noise.” In: *Linköping University Electronic Press* (2011).

[Azu95] R. Azuma. “"Predictive Tracking for Augmented Reality".” PhD thesis. Computer Science, University of North Carolina, Chapel Hill, NC, 1995.

[Azu97] R. T. Azuma. “A survey of augmented reality.” In: *Presence: Teleoperators and Virtual Environments* 6.4 (Aug. 1997), pp. 355–385.

[Bar+12] D. Baričević, C. Lee, M. Turk, T. Höllerer, and D. A. Bowman. "A hand-held AR magic lens with user-perspective rendering." In: *ISMAR*. 2012, pp. 197–206.

[Bed95] B. B. Bederson. "Audio augmented reality: a prototype automated tour guide." In: *Conference companion on Human factors in computing systems*. ACM. 1995, pp. 210–211.

[BI08] O. Bimber and D. Iwai. "Superimposing dynamic range." In: *TOG*. Vol. 27. 5. 2008, p. 150.

[Bic+07] C. Bichlmeier, F. Wimmer, S. M. Heining, and N. Navab. "Contextual anatomic mimesis hybrid in-situ visualization method for improving multi-sensory depth perception in medical augmented reality." In: *Mixed and Augmented Reality, 2007. ISMAR 2007. 6th IEEE and ACM International Symposium on*. IEEE. 2007, pp. 129–138.

[Bim+03] O. Bimber, A. Grundh?fer, G. Wetzstein, and S. Kn?del. "Consistent Illumination within Optical See-Through Augmented Environments." In: *Proceedings of IEEE International Symposium on Mixed and Augmented Reality (ISMAR) 2003*. 2003, pp. 198–207.

[Bim+08] O. Bimber, D. Iwai, G. Wetzstein, and A. Grundhöfer. "The Visual Computing of Projector-Camera Systems." In: *Computer Graphics Forum*. Vol. 27. 8. Wiley Online Library. 2008, pp. 2219–2245.

[BQ11] M. Bennett and A. Quigley. "Creating personalized digital human models of perception for visual analytics." In: *User Modeling, Adaption and Personalization*. Springer, 2011, pp. 25–37.

[BR05] O. Bimber and R. Raskar. *Spatial augmented reality: merging real and virtual worlds*. CRC Press, 2005.

[BR06] O. Bimber and R. Raskar. "Modern approaches to augmented reality." In: *ACM SIGGRAPH 2006 Courses*. ACM. 2006, p. 1.

[Bre+96] D. E. Breen, R. T. Whitaker, E. Rose, and M. Tuceryan. "Interactive occlusion and automatic object placement for augmented reality." In: *Computer Graphics Forum*. Vol. 15. 3. Wiley Online Library. 1996, pp. 11–22.

[BST11] M. Broecker, R. T. Smith, and B. H. Thomas. "Adaptive substrate for enhanced spatial augmented reality contrast and resolution." In: *ISMAR*. 2011, pp. 251–252.

[Car+11] J. Carmigniani, B. Furht, M. Anisetti, P. Ceravolo, E. Damiani, and M. Ivkovic. "Augmented reality technologies, systems and applications." In: *Multimedia Tools and Applications* 51.1 (2011), pp. 341–377.

[CHR04] O. Cakmakci, Y. Ha, and J. P. Rolland. "A Compact Optical See-Through Head-Worn Display with Occlusion Support." In: *Proceedings of IEEE International Symposium on Mixed and Augmented Reality (ISMAR) 2004*. 2004, pp. 16–25. DOI: 10.1109/ISMAR.2004.2.

[Cla62] A. C. Clarke. “Hazards of prophecy: the failure of imagination.” In: *Profiles of the Future, Gollancz, London* (1962).

[CM92] T. Caudell and D. Mizell. “Augmented Reality: An Application of Heads-Up Display Technology to Manual Manufacturing Processes.” In: *25th Hawaii International Conference on Systems Sciences (HICCS'92)*. 1992, pp. 659–669.

[Coo+01] J. R. Cooperstock et al. “The classroom of the future: enhancing education through augmented reality.” In: *Proc. HCI Inter. 2001 Conf. on Human-Computer Interaction*. Vol. 1. 2001, pp. 688–692.

[CR06] O. Cakmakci and J. Rolland. “Head-worn displays: a review.” In: *Journal of Display Technology* 2.3 (2006), pp. 199–216.

[Dav+14] J. David Hincapie-Ramos, L. Ivanchuk, S. K. Sridharan, and P. Irani. “SmartColor: Real-time color correction and contrast for optical see-through head-mounted displays.” In: *ISMAR*. 2014, pp. 187–194.

[DF01] F. Devernay and O. D. Faugeras. “Straight lines have to be straight.” In: *Journal of Machine Vision and Applications* 13.1 (2001), pp. 14–24. DOI: 10.1007/PL00013269.

[DRM05] J. Didier, D. Roussel, and M. Mallem. “A Time Delay Compensation Method Improving Registration for Augmented Reality.” In: *Proceedings of IEEE International Conference on Robotics and Automation (ICRA) 2005, April 18-22, 2005, Barcelona, Spain*. 2005, pp. 3384–3389. DOI: 10.1109/ROBOT.2005.1570633.

[FA12] N. Fujimori and S. Ando. “Super-resolution reconstruction algorithm using vibro-imaging and correlation image sensor.” In: *Society of Instrument and Control Engineers (SICE), Annual Conference on*. IEEE. 2012, pp. 2028–2033.

[FMS93] S. Feiner, B. MacIntyre, and D. D. Seligmann. “Knowledge-Based Augmented Reality.” In: *Commun. ACM* 36.7 (1993), pp. 53–62.

[FPF99] A. Fitzgibbon, M. Pilu, and R. B. Fisher. “Direct Least Square Fitting of Ellipses.” In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 21.5 (1999), pp. 476–480.

[Fuc+98] H. Fuchs, M. A. Livingston, R. Raskar, K. Keller, J. R. Crawford, P. Rademacher, S. H. Drake, A. A. Meyer, et al. *Augmented reality visualization for laparoscopic surgery*. Springer, 1998.

[G+94] B. Grimm, S. Goelz, J. F. Bille, et al. “Objective measurement of wave aberrations of the human eye with the use of a Hartmann-Shack wave-front sensor.” In: *The Journal of the Optical Society of America A (JOSA A)* 11.7 (1994), pp. 1949–1957.

[Gen+00] Y. Genc, F. Sauer, F. Wenzel, M. Tuceryan, and N. Navab. “Optical See-Through HMD Calibration: A Stereo Method Validated with a Wideo See-Through System.” In: *Augmented Reality, 2000. (ISAR 2000). Proceedings. IEEE and ACM International Symposium on*. IEEE. 2000, pp. 165–174.

[GFG08] S. J. Gilson, A. W. Fitzgibbon, and A. Glennerster. "Spatial calibration of an optical see-through head-mounted display." In: *Journal of neuroscience methods* 173.1 (2008), pp. 140–146.

[GG12] S. Gilson and A. Glennerster. "High fidelity immersive virtual reality." In: *Virtual reality-human computer interaction* (2012).

[Giu+11] A. Giusti, P. Taddei, G. Corani, L. Gambardella, C. Magli, and L. Gianaroli. "Artificial defocus for displaying markers in microscopy z-stacks." In: *TVCG*. Vol. 17. 12. 2011, pp. 1757–1764.

[Gor+96] S. J. Gortler, R. Grzeszczuk, R. Szeliski, and M. F. Cohen. "The Lumigraph." In: *Proceedings of ACM SIGGRAPH 1996*. 1996, pp. 43–54. DOI: 10.1145/237170.237200.

[Gra01] C. Gramkow. "On averaging rotations." In: *Journal of Mathematical Imaging and Vision* 15.1-2 (2001), pp. 7–16.

[GTN02] Y. Genc, M. Tuceryan, and N. Navab. "Practical Solutions for Calibration of Optical See-Through Devices." In: *IEEE and ACM International Symposium on Mixed and Augmented Reality, ISMAR*. 2002, pp. 169–175.

[Hal04] M. Haller. "Photorealism or/and non-photorealism in augmented reality." In: *Proceedings of the 2004 ACM SIGGRAPH international conference on Virtual Reality continuum and its applications in industry*. ACM. 2004, pp. 189–196.

[Här+04] A. Härmä, J. Jakka, M. Tikander, M. Karjalainen, T. Lokki, J. Hiipakka, and G. Lorho. "Augmented reality audio for mobile and wearable appliances." In: *Journal of the Audio Engineering Society* 52.6 (2004), pp. 618–639.

[Has+06] Y. Hashimoto, N. Nagaya, M. Kojima, S. Miyajima, J. Ohtaki, A. Yamamoto, T. Mitani, and M. Inami. "Straw-like user interface: virtual experience of the sensation of drinking using a straw." In: *Proceedings of the 2006 ACM SIGCHI international conference on Advances in computer entertainment technology*. ACM. 2006, p. 50.

[HB11] R. R. Hainich and O. Bimber. *Displays: fundamentals and applications*. CRC press, 2011.

[Hei+13] F. Heide, M. Rouf, M. B. Hullin, B. Labitzke, W. Heidrich, and A. Kolb. "High-quality computational imaging through simple lenses." In: *ACM Transactions on Graphics (TOG)* 32.5 (2013), p. 149.

[Hei+99] W. Heidrich, H. P. A. Lensch, M. F. Cohen, and H.-P. Seidel. "Light Field Techniques for Reflections and Refractions." In: *Rendering Techniques*. 1999, pp. 187–196.

[HF09] S. J. Henderson and S. Feiner. "Evaluating the benefits of augmented reality for task localization in maintenance of an armored personnel carrier turret." In: *Mixed and Augmented Reality, 2009. ISMAR 2009. 8th IEEE International Symposium on*. IEEE. 2009, pp. 135–144.

[HF11] S. Henderson and S. Feiner. “Exploring the benefits of augmented reality documentation for maintenance and repair.” In: *Visualization and Computer Graphics, IEEE Transactions on* 17.10 (2011), pp. 1355–1368.

[HF98] R. Halır and J. Flusser. “Numerically Stable Direct Least Squares Fitting of Ellipses.” In: *Proc. 6th International Conference in Central Europe on Computer Graphics and Visualization. WSCG*. Vol. 98. Citeseer. 1998, pp. 125–132.

[HG07] H. Hua and C. Gao. “A Systematic Framework for On-line Calibration of a Head-Mounted Projection Display for Augmented-Reality Systems.” In: *Journal of the Society for Information Display* 15.11 (2007), pp. 905–913.

[HG12] H. Hua and C. Gao. “A compact, eye-tracked optical see-through head-mounted display.” In: *Proc. SPIE*. Vol. 8288. 2012, 82881F.

[HGA02] H. Hua, C. Gao, and N. Ahuja. “Calibration of a head-mounted projective display for augmented reality systems.” In: *Proc. ISMAR*. IEEE. 2002, pp. 176–185.

[HGA07] H. Hua, C. Gao, and N. Ahuja. “Calibration of an HMPD-based augmented reality system.” In: *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on* 37.3 (2007), pp. 416–430.

[HK07] R. I. Hartley and S. B. Kang. “Parameter-Free Radial Distortion Correction with Center of Distortion Estimation.” In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)* 29.8 (2007), pp. 1309–1321.

[Hol97] R. L. Holloway. “Registration error analysis for augmented reality.” In: *Presence: Teleoperators and Virtual Environments* 6.4 (1997), pp. 413–432.

[HP14] A. D. Hwang and E. Peli. “An augmented-reality edge enhancement application for Google Glass.” In: *Optometry & Vision Science* 91.8 (2014), pp. 1021–1030.

[Hua+12] F.-C. Huang, D. Lanman, B. A. Barsky, and R. Raskar. “Correcting for optical aberrations using multilayer displays.” In: *TOG*. Vol. 31. 6. 2012, p. 185.

[Hua+13] F.-C. Huang, G. Wetzstein, B. A. Barsky, and R. Raskar. “Computational light field display for correcting visual aberrations.” In: *SIGGRAPH Posters*. 2013, p. 40.

[Hua+14] F.-C. Huang, G. Wetzstein, B. A. Barsky, and R. Raskar. “Eyeglasses-free display: towards correcting visual aberrations with computational light field displays.” In: *ACM Transactions on Graphics (TOG)* 33.4 (2014), p. 59.

[Hub+07] M. Huber, D. Pustka, P. Keitler, F. Echtler, and G. Klinker. “A System Architecture for Ubiquitous Tracking Environments.” In: *Proceedings of the 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality*. IEEE Computer Society. 2007, pp. 211–214.

[IK14a] Y. Itoh and G. Klinker. “Interaction-free calibration for optical see-through head-mounted displays based on 3D eye localization.” In: *Proceedings of IEEE Symposium on 3D User Interfaces (3DUI), Minneapolis, MN, USA, March 29-30, 2014*. 2014, pp. 75–82. DOI: 10.1109/3DUI.2014.6798846.

[IK14b] Y. Itoh and G. Klinker. "Performance and Sensitivity Analysis of INDICA: INteraction-Free DIsplay CAlibration for Optical See-Through Head-Mounted Displays." In: *Proceedings of IEEE International Symposium on Mixed and Augmented Reality (ISMAR) 2014*. 2014, pp. 171–176.

[IK15a] Y. Itoh and G. Klinker. "Light-Field Correction for Spatial Calibration of Optical See-Through Head-Mounted Displays." In: *IEEE Transactions on Visualization and Computer Graphics (Proceedings Virtual Reality 2015)* 21.4 (Apr. 2015), pp. 471–480. DOI: http://doi.ieeecomputersociety.org/10.1109/TVCG.2015.2391859.

[IK15b] Y. Itoh and G. Klinker. "Vision Enhancement: Defocus Correction via Optical See-Through Head-Mounted Displays." In: *6th Augmented Human International Conference, AH '15, Singapore, March 9-11, 2015*. 2015, pp. 1–8.

[Ina+00] M. Inami, N. Kawakami, D. Sekiguchi, Y. Yanagida, T. Maeda, and S. Tachi. "Visuo-haptic display using head-mounted projector." In: *Virtual Reality, 2000. Proceedings. IEEE*. IEEE. 2000, pp. 233–240.

[Ish+10] Y. Ishiguro, A. Mujibiya, T. Miyaki, and J. Rekimoto. "Aided eyes: eye activity sensing for daily life." In: *Proceedings of the 1st Augmented Human International Conference*. ACM. 2010, p. 25.

[Ito+13] Y. Itoh, F. Pankratz, C. Waechter, and G. Klinker. *Calibration of Head-Mounted Finger Tracking to Optical See-Through Head Mounted Display*. Demonstraton at ISMAR. 2013.

[Ito+15a] Y. Itoh, M. Dzitsiuk, T. Amano, and G. Klinker. "Semi-Parametric Color Reproduction Method for Optical See-Through Head-Mounted Displays." In: *IEEE Transactions on Visusalization and Computer Graphics (TVCG)* 21.11 (Nov. 2015). Proceedings of ISMAR 2015: long papers, pp. 1269–1278. DOI: 10.1109/TVCG.2015.2459892.

[Ito+15b] Y. Itoh, M. Dzitsiuk, T. Amano, and G. Klinker. "Simultaneous Direct and Augmented View Distortion Calibration of Optical See-Through Head-Mounted Displays." In: *Proceedings of 14th IEEE International Symposium on Mixed and Augmented Reality, ISMAR 2015, FUkuoka, Japan, Sep. 29 - Oct. 3, 2015*. 2015, pp. 43–48.

[Iwa+04] H. Iwata, H. Yano, T. Uemura, and T. Moriya. "Food simulator: A haptic interface for biting." In: *Virtual Reality, 2004. Proceedings. IEEE*. IEEE. 2004, pp. 51–57.

[JMC93] A. L. Janin, D. W. Mizell, and T. P. Caudell. "Calibration of head-mounted displays for augmented reality applications." In: *Virtual Reality Annual International Symposium*. IEEE. 1993, pp. 246–255.

[Jon+07] A. Jones, I. McDowall, H. Yamada, M. Bolas, and P. Debevec. "Rendering for an interactive 360 light field display." In: 26.3 (2007), p. 40.

[Kat+00] H. Kato, M. Billinghurst, I. Poupyrev, K. Imamoto, and K. Tachibana. “Virtual object manipulation on a table-top AR environment.” In: *Augmented Reality, 2000.(ISAR 2000). Proceedings. IEEE and ACM International Symposium on*. Ieee. 2000, pp. 111–119.

[KB99] H. Kato and M. Billinghurst. “Marker tracking and hmd calibration for a video-based augmented reality conferencing system.” In: *Augmented Reality, 1999.(IWAR’99) Proceedings. 2nd IEEE and ACM International Workshop on*. IEEE. 1999, pp. 85–94.

[KHS14] M. Klemm, H. Hoppe, and F. Seebacher. “Non-Parametric Camera-Based Calibration of Optical See-Through Glasses for Augmented Reality Applications.” In: *Proceedings of IEEE International Symposium on Mixed and Augmented Reality (ISMAR) 2014*. IEEE. 2014, pp. 273–274.

[KKO00] K. Kiyokawa, Y. Kurata, and H. Ohno. “An optical see-through display for mutual occlusion of real and virtual environments.” In: *Augmented Reality, 2000.(ISAR 2000). Proceedings. IEEE and ACM International Symposium on*. IEEE. 2000, pp. 60–67.

[KKO01] K. Kiyokawa, Y. Kurata, and H. Ohno. “An optical see-through display for mutual occlusion with a real-time stereovision system.” In: *Computers & Graphics* 25.5 (2001), pp. 765–779. DOI: 10.1016/S0097-8493(01)00119-4.

[Koi+12] N. Koizumi, M. Sugimoto, N. Nagaya, M. Inami, and M. Furukawa. “Stop motion goggle: augmented visual perception by subtraction method using high speed liquid crystal.” In: *AH*. 2012, p. 14.

[Kol93] J. Kollin. “A retinal display for virtual-environment applications.” In: *SID International Symposium Digest of Technical Papers*. Vol. 24. SOCIETY FOR INFORMATION DISPLAY. 1993, pp. 827–827.

[KP10] D. W. F. van Krevelen and R. Poelman. “A Survey of Augmented Reality Technologies, Applications and Limitations.” In: *The International Journal of Virtual Reality* 9.2 (June 2010), pp. 1–20.

[KS03] H. Kaufmann and D. Schmalstieg. “Mathematics and geometry education with collaborative augmented reality.” In: *Computers & Graphics* 27.3 (2003), pp. 339–345.

[KS11] K. Kanatani and Y. Sugaya. “Bundle Adjustment for 3-D Reconstruction: Implementation and Evaluation.” In: *Memoirs of the Faculty of Engineering, Okayama University* 45 (2011), pp. 1–9.

[KSR99] G. Klinker, D. Stricker, and D. Reiners. “"Augmented Reality: A Balancing Act Between High Quality and Real-Time Constraints".” In: *Proc. 1. International Symposium on Mixed Reality (ISMR’99)*. Yokohama, Japan, Mar. 1999, pp. 325–346.

[KYO01] R. Kijima, E. Yamada, and T. Ojika. “A development of reflex HMD-HMD with time delay compensation capability.” In: *Proc. 2nd Int’l Symp. Mixed Reality*. Citeseer. 2001, pp. 40–47.

[LCH08] S. Liu, D. Cheng, and H. Hua. “An optical see-through head mounted display with addressable focal planes.” In: *Proceedings of IEEE International Symposium on Mixed and Augmented Reality (ISMAR) 2008*. 2008, pp. 33–42. DOI: 10.1109/ISMAR.2008.4637321.

[Lee+11] J.-Y. Lee, H.-M. Park, S.-H. Lee, T.-E. Kim, and J.-S. Choi. “Design and Implementation of an Augmented Reality System Using Gaze Interaction.” In: *Information Science and Applications (ICISA), 2011 International Conference on*. IEEE. 2011, pp. 1–8.

[LH13] S. Lee and H. Hua. “A Robust Camera-based Method for Optical Distortion Calibration of Head-Mounted Displays.” In: *Virtual Reality Conference (VR), 2013 IEEE*. IEEE. 2013, pp. 27–30.

[LH96] M. Levoy and P. Hanrahan. “Light Field Rendering.” In: *Proceedings of ACM SIGGRAPH 1996*. 1996, pp. 31–42. DOI: 10.1145/237170.237199.

[LL10] M. Lombardo and G. Lombardo. “Wave aberration of human eyes and new descriptors of image optical quality and visual performance.” In: *Journal of Cataract & Refractive Surgery (JCRS)* 36.2 (2010), pp. 313–331.

[LN07] R. W. Lindeman and H. Noma. “A classification scheme for multi-sensory augmented reality.” In: *Proceedings of the 2007 ACM symposium on Virtual reality software and technology*. ACM. 2007, pp. 175–178.

[Luk+15] S. Lukosch, M. Billinghurst, L. Alem, and K. Kiyokawa. “Collaboration in Augmented Reality.” In: *Computer Supported Cooperative Work (CSCW)* (2015), pp. 1–11.

[LW04] Y.-C. Liu and M.-H. Wen. “Comparison of head-up display (HUD) vs. head-down display (HDD): driving performance of commercial vehicle operators in Taiwan.” In: *International Journal of Human-Computer Studies* 61.5 (2004), pp. 679–697.

[Mac+67] J. MacQueen et al. “Some Methods for Classification and Analysis of Multivariate Observations.” In: *5th Berkeley Symposium on Mathematical Statistics and Probability*. Vol. 1. 281-297. 1967, p. 14.

[Mai+12] P. Maier, A. Dey, C. Waechter, C. Sandor, M. Tönnis, and G. Klinker. “An Empiric Evaluation of Confirmation Methods for Optical See-Through Head-Mounted Display Calibration.” In: *Joint Virtual Reality Conference of ICAT - EGVE-EuroVR*. 2012, pp. 73–80. DOI: 10.2312/EGVE/JVRC12/073-080.

[MaN12] S. MaNN. “Through the glass, lightly [viewpoint].” In: *Technology and Society Magazine, IEEE* 31.3 (2012), pp. 10–14.

[McG+01] E. McGarrity, Y. Genc, M. Tuceryan, C. B. Owen, and N. Navab. "A New System for Online Quantitative Evaluation of Optical See-through Augmentation." In: *ISAR*. IEEE. 2001, pp. 157–166. DOI: 10.1109/ISAR.2001.970525.

[MF13] A. Maimone and H. Fuchs. "Computational augmented reality eyeglasses." In: *Mixed and Augmented Reality (ISMAR), 2013 IEEE International Symposium on*. IEEE. 2013, pp. 29–38.

[Mil+95] P. Milgram, H. Takemura, A. Utsumi, and F. Kishino. "Augmented reality: A class of displays on the reality-virtuality continuum." In: *Photonics for Industrial Applications*. International Society for Optics and Photonics. 1995, pp. 282–292.

[MK13] C. Menk and R. Koch. "Truthful color reproduction in spatial augmented reality applications." In: *Visualization and Computer Graphics, IEEE Transactions on* 19.2 (2013), pp. 236–248.

[MKY13] N. Makibuchi, H. Kato, and A. Yoneyama. "Vision-Based Robust Calibration for Optical See-Through Head-mounted Displays." In: *IEEE International Conference on Image Processing (ICIP)*. IEEE. 2013, pp. 2177–2181.

[MMB97] W. R. Mark, L. McMillan, and G. Bishop. "Post-rendering 3D warping." In: *Proceedings of the 1997 symposium on Interactive 3D graphics*. ACM. 1997, 7–ff.

[Mos+15] K. Moser, Y. Itoh, K. Oshima, E. Swan, G. Klinker, and C. Sandor. "Subjective Evaluation of a Semi-Automatic Optical See-Through Head-Mounted Display Calibration Technique." In: *IEEE Transactions on Visualization and Computer Graphics (Proceedings Virtual Reality 2015)* 21.4 (Apr. 2015), pp. 491–500.

[Myn+97] E. D. Mynatt, M. Back, R. Want, and R. Frederick. "Audio Aura: Light-weight audio augmented reality." In: *Proceedings of the 10th annual ACM symposium on User interface software and technology*. ACM. 1997, pp. 211–212.

[Nar+11] T. Narumi, S. Nishizaka, T. Kajinami, T. Tanikawa, and M. Hirose. "Augmented reality flavors: gustatory display based on edible marker and cross-modal interaction." In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM. 2011, pp. 93–102.

[Nav+04] N. Navab, S. Zokai, Y. Genc, and E. M. Coelho. "An On-line Evaluation System for Optical See-Through Augmented Reality." In: *Virtual Reality Conference (VR), 2004*. IEEE. 2004, pp. 245–246.

[Nav04] N. Navab. "Developing killer apps for industrial augmented reality." In: *Computer Graphics and Applications, IEEE* 24.3 (2004), pp. 16–20.

[New+04] J. Newman, M. Wagner, M. Bauer, A. MacWilliams, T. Pintaric, D. Beyer, D. Pustka, F. Strasser, D. Schmalstieg, and G. Klinker. "Ubiquitous Tracking for Augmented Reality." In: *Mixed and Augmented Reality, 2004. ISMAR 2004. Third IEEE and ACM International Symposium on*. IEEE. 2004, pp. 192–201.

[Ng+05] R. Ng, M. Levoy, M. Brédif, G. Duval, M. Horowitz, and P. Hanrahan. "Light field photography with a hand-held plenoptic camera." In: *Computer Science Technical Report CSTR*. Vol. 2. 11. 2005.

[NGC07] S. Nilsson, T. Gustafsson, and P. Carleberg. "Hands Free Interaction with Virtual Information in a Real Environment." In: (2007), pp. 53–57.

[Nic+11] S. Nicolau, L. Soler, D. Mutter, and J. Marescaux. "Augmented reality in laparoscopic surgical oncology." In: *Surgical oncology* 20.3 (2011), pp. 189–201.

[NNT11] C. Nitschke, A. Nakazawa, and H. Takemura. "Image-based Eye Pose and Reflection Analysis for Advanced Interaction Techniques and Scene Understanding." In: *IPSJ Computer Vision and Image Media (CVIM) (Doctoral Theses Session)* (2011), pp. 1–16.

[NNT13] C. Nitschke, A. Nakazawa, and H. Takemura. "Corneal Imaging Revisited: An Overview of Corneal Reflection Analysis and Applications." In: *IPSJ Transactions on Computer Vision and Applications* 5 (2013), pp. 1–18.

[Noj+02] T. Nojima, D. Sekiguchi, M. Inami, and S. Tachi. "The smarttool: a system for augmented reality of haptics." In: *Virtual Reality, 2002. Proceedings. IEEE*. IEEE. 2002, pp. 67–72.

[OS08] Y. Oyamada and H. Saito. "Defocus blur correcting projector-camera system." In: *Advanced Concepts for Intelligent Vision Systems*. Springer. 2008, pp. 453–464.

[Osk+13] T. Oskiper, M. Sizintsev, V. Branzoi, S. Samarasekera, and R. Kumar. "Augmented Reality binoculars." In: *ISMAR*. Oct. 2013, pp. 219–228. DOI: 10.1109/ISMAR.2013.6671782.

[Owe+04] C. B. Owen, J. Zhou, A. Tang, and F. Xiao. "Display-Relative Calibration for Optical See-Through Head-Mounted Displays." In: *Proceedings of IEEE International Symposium on Mixed and Augmented Reality (ISMAR) 2004*. 2004, pp. 70–78.

[Pam+10] V. F. Pamplona, A. Mohan, M. M. Oliveira, and R. Raskar. "NETRA: interactive display for estimating refractive errors and focal range." In: *TOG*. Vol. 29. 4. ACM, 2010, p. 77.

[Pam+12] V. F. Pamplona, M. M. Oliveira, D. G. Aliaga, and R. Raskar. "Tailored displays to compensate for visual aberrations." In: *TOG*. Vol. 31. 4. 2012, p. 81.

[Pel01] E. Peli. "Vision multiplexing: an engineering approach to vision rehabilitation device development." In: *Optometry & Vision Science* 78.5 (2001), pp. 304–315.

[Pel02] E. Peli. "Treating with spectacle lenses: a novel idea!?" In: *Optometry & Vision Science* 79.9 (2002), pp. 569–580.

[Pla+01] B. C. Platt et al. "History and principles of Shack-Hartmann wavefront sensing." In: *Journal of Refractive Surgery* 17.5 (2001), S573–S577.

[Plo+15] A. Plopski, Y. Itoh, C. Nitschke, K. Kiyokawa, G. Klinker, and H. Takemura. "Corneal-Imaging Calibration for Optical See-Through Head-Mounted Displays." In: *TVCG (Proc. VR)* 21.4 (Apr. 2015), pp. 481–490.

[PT02] W. Piekarski and B. Thomas. "ARQuake: the outdoor augmented reality gaming system." In: *Communications of the ACM* 45.1 (2002), pp. 36–38.

[QM95] M. Qiu and S. D. Ma. "The Nonparametric Approach for Camera Calibration." In: *Proceedings of IEEE International Conference on Computer Vision (ICCV) 1995*. 1995, pp. 224–229.

[Ras+01] R. Raskar, G. Welch, K.-L. Low, and D. Bandyopadhyay. *Shader lamps: Animating real objects with image-based illumination*. Springer, 2001.

[Ras+04] R. Raskar, P. Beardsley, J. Van Baar, Y. Wang, P. Dietz, J. Lee, D. Leigh, and T. Willwacher. "RFIG lamps: interacting with a self-describing world via photosensing wireless tags and projectors." In: *ACM Transactions on Graphics (TOG)*. Vol. 23. 3. ACM. 2004, pp. 406–415.

[Ras99] C. E. Rash. *Helmet mounted displays: Design issues for rotary-wing aircraft*. Vol. 93. SPIE Press, 1999.

[RD03] M. Roussou and G. Drettakis. "Photorealism and non-photorealism in virtual heritage representation." In: *First Eurographics Workshop on Graphics and Cultural Heritage (2003)*. Eurographics. 2003, p. 10.

[Rei+98] D. Reiners, D. Stricker, G. Klinker, and S. Müller. "Augmented Reality for Construction Tasks: Doorlock Assembly." In: *1st International Workshop on Augmented Reality (IWAR 1998), San Francisco* (1998).

[RF00] J. P. Rolland and H. Fuchs. "Optical versus video see-through head-mounted displays in medical visualization." In: *Presence: Teleoperators and Virtual Environments* 9.3 (2000), pp. 287–309.

[RH05] J. Rolland and H. Hua. "Head-mounted display systems." In: *Encyclopedia of optical engineering* (2005), pp. 1–13.

[RHF94] J. P. Rolland, R. L. Holloway, and H. Fuchs. "A Comparison of Optical and Video See-Through Head-Mounted Displays." In: *SPIE Telemanipulator and Telepresence Technologies* 2351 (1994), pp. 293–307.

[RN95] J. Rekimoto and K. Nagao. "The world through the computer: Computer augmented interaction with real world environments." In: *Proceedings of the 8th annual ACM symposium on User interface and software technology*. ACM. 1995, pp. 29–36.

[RR92] W. Robinett and J. P. Rolland. "A computational model for the stereoscopic optics of a head-mounted display." In: *Presence* 1.1 (1992), pp. 45–62.

[RWF98] R. Raskar, G. Welch, and H. Fuchs. "Spatially augmented reality." In: *First IEEE Workshop on Augmented Reality (IWAR'98)*. Citeseer. 1998, pp. 11–20.

[San+15] C. Sandor, M. Fuchs, A. Cassinelli, H. Li, R. Newcombe, G. Yamamoto, and S. Feiner. "Breaking the Barriers to True Augmented Reality." In: *arXiv preprint arXiv:1512.05471* (2015).

[SBD12] L. Swirski, A. Bulling, and N. A. Dodgson. "Robust Real-Time Pupil Tracking in Highly Off-axis Images." In: *Eye Tracking Research and Applications (ETRA)*. Santa Barbara, CA, USA, Mar. 2012, pp. 173–176.

[Sch+08] B. Schwerdtfeger, D. Pustka, A. Hofhauser, and G. Klinker. "Using laser projectors for augmented reality." In: *Proceedings of the 2008 ACM symposium on Virtual reality software and technology*. ACM. 2008, pp. 134–137.

[Sch+09] E. Schneider, T. Villgrattner, J. Vockeroth, K. Bartl, S. Kohlbecher, S. Bardins, H. Ulbrich, and T. Brandt. "EyeSeeCam: An Eye Movement-Driven Head Camera for the Examination of Natural Visual Exploration." In: *Annals of the New York Academy of Sciences* 1164.1 (2009), pp. 461–467.

[SF11] S. Shimizu and H. Fujiyoshi. "Acquisition of 3D gaze information from eyeball movements using inside-out camera." In: *AH*. 2011, p. 6.

[SGM12] B. Sajadi, M. Gopi, and A. Majumder. "Edge-guided resolution enhancement in projectors via optical pixel sharing." In: *TOG*. Vol. 31. 4. 2012, p. 79.

[Sig+13] R. Sigrist, G. Rauter, R. Riener, and P. Wolf. "Augmented visual, auditory, haptic, and multimodal feedback in motor learning: A review." In: *Psychonomic bulletin & review* 20.1 (2013), pp. 21–53.

[Sri+13] S. K. Sridharan, J. D. Hincapié-Ramos, D. R. Flatla, and P. Irani. "Color correction for optical see-through displays using display color profiles." In: *VRST*. 2013, pp. 231–240.

[SS01] B. Scholkopf and A. J. Smola. *Learning with kernels: support vector machines, regularization, optimization, and beyond*. MIT press, 2001.

[Sta+96] A. State, G. Hirota, D. T. Chen, W. F. Garrett, and M. A. Livingston. "Superior Augmented Reality Registration by Integrating Landmark Tracking and Magnetic Tracking." In: *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*. SIGGRAPH '96. New York, NY, USA: ACM, 1996, pp. 429–438. ISBN: 0-89791-746-4. DOI: 10.1145/237170.237282.

[Sto74] M. Stone. "Cross-validatory choice and assessment of statistical predictions." In: *Journal of the Royal Statistical Society. Series B (Methodological)* (1974), pp. 111–147.

[Stu+11] P. F. Sturm, S. Ramalingam, J. Tardif, S. Gasparini, and J. Barreto. "Camera Models and Fundamental Concepts Used in Geometric Computer Vision." In: *Foundations and Trends in Computer Graphics and Vision* 6.1-2 (2011), pp. 1–183. DOI: 10.1561/0600000023.

[Sur+07] R. Suryakumar, J. P. Meyers, E. L. Irving, and W. R. Bobier. "Application of video-based technology for the simultaneous measurement of accommodation and vergence." In: *Vision research* 47.2 (2007), pp. 260–268.

[Sut65] I. E. Sutherland. "The Ultimate Display." In: *Proceedings of the Congress of the Internation Federation of Information Processing (IFIP) 65*. Vol. 2. 1965, pp. 506–508.

[Tan+03] A. Tang, C. Owen, F. Biocca, and W. Mou. "Comparative effectiveness of augmented reality in object assembly." In: *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM. 2003, pp. 73–80.

[TFM07] H. Takeda, S. Farsiu, and P. Milanfar. "Kernel Regression for Image Processing and Reconstruction." In: *IEEE Transactions on Image Processing* 16.2 (2007), pp. 349–366. DOI: 10.1109/TIP.2006.888330.

[TGN02] M. Tuceryan, Y. Genc, and N. Navab. "Single-Point Active Alighment Method (SPAAM) for Optical See-Through HMD Calibration for Augmented Reality." In: *Presence: Teleoperators and Virtual Environments* 11.3 (2002), pp. 259–276.

[TIS13] M. Tomioka, S. Ikeda, and K. Sato. "Approximated user-perspective rendering in tablet-based augmented reality." In: *ISMAR*. 2013, pp. 21–28.

[TK12] A. Tsukada and T. Kanade. "Automatic Acquisition of a 3D Eye Model For a Wearable First-Person Vision Device." In: *Eye Tracking Research and Applications (ETRA)*. 2012, pp. 213–216.

[TN00] M. Tuceryan and N. Navab. "Single Point Active Alignment Method (SPAAM) for Optical See-Through HMD Calibration for AR." In: *Proceedings of ISAR*. IEEE. 2000, pp. 149–158.

[Tsu+11] A. Tsukada, M. Shino, M. Devyver, and T. Kanade. "Illumination-free gaze estimation method for first-person vision wearable device." In: *ICCV Workshops*. June 2011, pp. 2084–2091.

[VB99] J. Vallino and C. Brown. "Haptics in augmented reality." In: *Multimedia Computing and Systems, 1999. IEEE International Conference on*. Vol. 1. IEEE. 1999, pp. 195–200.

[Von+12] R. G. Von Gioi, J. Jakubowicz, J.-M. Morel, and G. Randall. "LSD: a line segment detector." In: *Image Processing On Line* (2012).

[VT05] C. Y. Vincent and T. Tjahjadi. "Multiview camera-calibration framework for non-parametric distortions removal." In: *IEEE Transactions on Robotics* 21.5 (2005), pp. 1004–1009. DOI: 10.1109/TRO.2005.851383.

[Wes66] G. Westheimer. "The maxwellian view." In: *Vision research* 6.11 (1966), pp. 669–682.

[Wie+13] F. Wientapper, H. Wuest, P. Rojtberg, and D. Fellner. “A camera-based calibration for automotive augmented reality head-up-displays.” In: *Proc. ISMAR*. IEEE. 2013, pp. 189–197.

[WS03] Y. Wang and D. Samaras. “Estimation of multiple directional light sources for synthesis of augmented reality images.” In: *Graphical Models* 65.4 (2003), pp. 185–205.

[Yen+11] C.-Y. Yen, K.-H. Lin, M.-H. Hu, R.-M. Wu, T.-W. Lu, and C.-H. Lin. “Effects of virtual reality–augmented balance training on sensory organization and attentional
controlled trial.” In: *Physical therapy* 91.6 (2011), pp. 862–874.

[YMK11] S. Yamazaki, M. Mochimaru, and T. Kanade. “Simultaneous self-calibration of a projector and a camera using structured light.” In: *Proc. CVPRW*. June 2011, pp. 60–67. DOI: 10.1109/CVPRW.2011.5981781.

[Zha00] Z. Zhang. “A Flexible New Technique for Camera Calibration.” In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)* 22.11 (2000), pp. 1330–1334. DOI: 10.1109/34.888718.

[Zhe+14] F. Zheng, T. Whitted, A. Lastra, P. Lincoln, A. State, A. Maimone, and H. Fuchs. “Minimizing latency for augmented reality displays: Frames considered harmful.” In: *IEEE International Symposium on Mixed and Augmented Reality, ISMAR 2014, Munich, Germany, September 10-12, 2014*. 2014, pp. 195–200. DOI: 10.1109/ISMAR.2014.6948427.

[Zhe15] F. Zheng. “Spatio-temporal registration in augmented reality.” PhD thesis. University of North Carolina at Chapel Hill, 2015.

[Zho07] J. Zhou. “Calibration of Optical See Through Head Mounted Displays for Augmented Reality.” PhD thesis. Michigan State University, 2007.