Eye Tracking Based Navigation for Proton Beam Therapy

Inaugural dissertation

to
be awarded the degree of Dr. sc. med.

presented at
the Faculty of Medicine
of the University of Basel

by
Stephan Wyder
from Bern, Switzerland

Basel, 2017

Approved by the Faculty of Medicine
on application of

Prof. Dr. Philippe C. Cattin, University of Basel, *First examiner*
Prof. Dr. Raphael Sznitman, University of Bern, *Second examiner*
Prof. Dr. med. Francis Munier, Jules-Gonin Eye Hospital Lausanne, *External expert*

Basel, 2nd May 2017

Prof. Dr. med. Thomas Gasser
Dean

Contents

Acknowledgments

I am very grateful to my supervisor, Prof. Dr. Philippe C. Cattin, for his excellent guidance and support throughout the time of my doctoral studies. His deep theoretical knowledge in all our projects, as well as his flair for practical approaches impresses me time and again. It is admirable how Philippe has a keen grasp for the tradeoff between advising his students and giving them the freedom to be creative.

I am thankful to all current and former team members of the *Medical Image Analysis Center, MIAC* of the University of Basel, Switzerland: Simon Andermatt, Dr. Patrik Arnold, Natalia Chicherova, Corinne Eymann-Baier, Dr. Beat Fasel, Dr. Ketut Fundana, Alina Giger, Antal Horváth, Lorenzo Iafolla, Dr. Christoph Jud, Dr. Silja Kiriyanthan, Dr. Tahir Majeed, Nadia Möri, Dr. Uri Nahum, Peter von Niederhäusern, Dr. Simon Pezold, Dr. Frank Preiswerk, Tiziano Ronchetti, Samaneh Manavi Roodsari, Robin Sandkühler, Dr. Adrian Schneider, Jörg Wagner, and Dr. Lilian Witthauer. The inspiring discussions about practical engineering problems, more theoretical mathematical stuff, mountaineering, sports in general, and more were very much appreciated. Especially I would like to thank Simon Pezold for being such a careful and passionate reviewer. He seems to know the *The Chicago Manual of Style* [1] inside and out.

Furthermore, I would like to thank Fabian Hennings and Dr. Jan Hrbacek, both from *Paul Scherrer Institute (PSI)* in Villigen, Switzerland for the fruitful collaboration.

Finally, I thank my family and friends for their support and encouragement, especially Silja Wenk for always being there for me.

Summary

Cancers of the eye, so-called ocular tumors, are a severe disease that may lead to blindness or even death if left untreated. A possibility to remove the tumor from the body of the patient is a so-called enucleation surgery, the removal of the eye. However, it is a drastic action and oncologists usually try to avoid it. Another treatment option is the therapy with protons. The actual proton therapy to treat ocular tumors is very successful and non-invasive. However, the navigation method that is applied for this kind of therapy requires a pre-treatment surgery, where radio-opaque clips are sutured onto the affected eyeball. These clips are used during the actual treatment to align the diseased eye with two orthogonal X-ray units. Hence, the overall treatment is invasive.

The work at hand presents an alternative, completely non-invasive navigation method based on eye tracking technology. We present a new treatment scheme with a first eye tracking prototype integrated into the treatment facility at *Paul Scherrer Institute (PSI)*. This system together with a patient specific eye model enables the medical physicist to align the patient's eye such that the tumor gets accurately treated by the proton beam.

Further, we present a second, improved eye tracking system. This time, we propose a stereo eye tracker, which only uses one physical camera to save physical space. We combine a stereo eye tracking algorithm with a clever arrangement of two planar mirrors and a single camera to get high accuracy, precision, and a compact design altogether.

Finally, we present a method to quantitatively evaluate the proposed navigation system. Verifying the accuracy of the location estimate of a volunteer's eye center is not easily possible. This is because the eye center is an intangible point, that does not correspond to an anatomical structure. Our evaluation method is based on an eye phantom on microstages and a corresponding kinematic model.

Our research and development may lead to an ocular tumor treatment which will be safer, more cost-effective, and more accessible to patients suffering from this serious disease.

1 Introduction

People with light colored eyes are somewhat more likely to develop melanoma of the eye than are people with brown eyes.

(cancer.org)

1.1 Motivation

The human eye can be affected by a variety of primary and secondary intraocular tumors. *Posterior uveal melanomas* are the most common primary intraocular malignant tumors of the eye and are a life-threatening disease.

Today, tumors of the eye such as uveal melanomas are often treated by charged particle beam irradiation. Within this general class of radiotherapies, proton beam therapy is particularly successful, because it allows for a very precise irradiation of the tumor. This is especially important due to the close proximity of many crucial structures in the eye, such as the optic nerve head and the lens.

By nature, proton beam therapy is a non-invasive treatment. However, the way it is performed nowadays requires a surgery prior to the actual treatment. During this surgery, the surgeon sutures four to five radio-opaque markers to the outer surface of the sclera at the edge of the tumor. This enables the medical physicists to localize the tumor by transillumination and to align the affected eye of the patient during the actual therapy with protons. Obviously, the required surgery bears additional risks for the patient and costs time and money.

On the one hand, proton therapy is a very successful, by nature non-invasive treatment. On the other hand, proton therapy depends on a navigation method with X-rays and clips, which requires a complicated surgery prior to the actual treatment. Hence the whole treatment is an invasive treatment, which leads to the following two research questions that formed the basis of our project:

> **Research question 1:** *Is it possible to replace today's invasive navigation system with a novel, non-invasive method?*

> **Research question 2:** *Is the potential replacement method able to localize the eye in space with sub-millimeter accuracy?*

1.2 Contribution

We propose a new treatment scheme, which may eliminate the need for a pre-treatment surgery for this particular proton therapy for eye tumors. The scheme essentially consists of a patient

specific statistical eye model [2, 3, 4], showing the tumor outline, and an eye tracking based navigation [5, 6], intended to localize the patient's eye in space. The work at hand describes the novel navigation method and is one part of the new treatment scheme. The proposed system will be safer, more cost-effective, and more accessible to patients suffering from this serious disease.

1.3 Outline

For a clear understanding of the enclosed publications, we deliver the required medical and technical background in Chapter 2. In Sections 2.1 and 2.2, we provide an introduction into the anatomy of the eye, cancer of the eye, and its treatment with protons. In Section 2.3, we give an introduction into eye tracking, the technology able to localize the eye in 3D space. Additionally, in Section 2.4, an introduction into coordinate transformations is given, which enables us to connect coordinate systems in 3D space.

Chapters 3, 4, and 5 contain the publications as they were submitted or accepted. The first enclosed publication introduces the new treatment scheme with a first eye tracking prototype. Within the second publication, an improved eye tracking system is proposed. In the third publication, we present a method to evaluate the accuracy of the eye tracker.

Finally, in Chapter 6 we present our final thoughts and ideas for future work.

2 Background

2.1 Medical Background

Cancers within the eye, so-called intraocular tumors, are a severe disease. Everywhere in the body, also in the eye, cells can suddenly start to grow in an uncontrolled way: a tumor arises. Cancer describes malignant tumors which grow in an abnormal way and spread into other regions of the body (*metastases*). According to the *World Health Organization (WHO)*[1], cancer is the leading cause of death worldwide. Compared to other organs, the eye is affected relatively seldom. However, eye cancer is almost always linked with the risk of visual loss.

The *American Cancer Society (ACS)* provides on their website[2] profound information about cancer in general, but also about eye cancer in particular. We used information from their website to write this chapter and we recommend it for further reading.

Anatomy of the Eye

The biggest part of the eye is the eyeball, sometimes referred to as *globe*. It is filled with a transparent, jelly-like liquid called *vitreous humor*. The eye contains three main layers:

1. The *sclera* is the white, outermost layer of the eyeball, which is visible around the *iris*. It directly passes into into the transparent *cornea*.

2. The *uvea* is the middle layer of the eye and itself consists of the iris, the colored part of the eye, the *choroid*, which provides blood to the retina, and finally the *ciliary body*. The latter includes the muscles responsible for the shape change of the lens. Further, the ciliary body is responsible for the production of the *aqueous humor*, the transparent liquid in the *anterior chamber*.

3. The *retina* is the innermost layer of the eyeball and consists of specialized, light-sensitive cells. The image which is focused by the lens onto the retina is transferred over the *optic nerve* to the *visual cortex*, which is part of the brain.

The *orbit* contains structures outside of the eyeball: eye muscles, nerves, and *adnexal structures*, consisting of eyelids and tear glands. Figure 2.1 illustrates a healthy human eye with labels for the most important structures.

---

[1] http://www.who.int/mediacentre/factsheets/fs297/en/ (accessed December 2016)
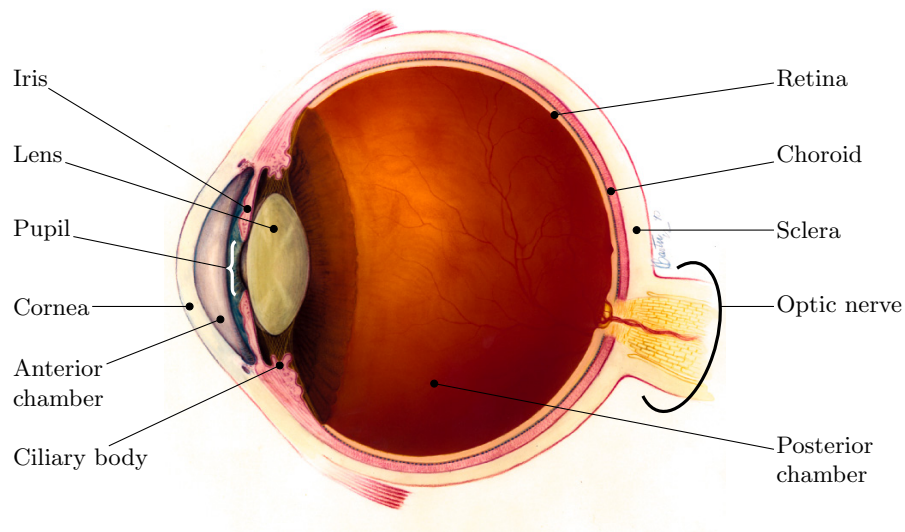[2] http://www.cancer.org/cancer/eyecancer/ (accessed December 2016)

Figure 2.1: Anatomy of the eye[3]

Intraocular Tumors

Eye cancer can be classified into primary and secondary intraocular tumors. Primary intraocular tumors have their origin in the eyeball. For adults, *ocular melanoma* is the most common type of eye cancer with an incidence rate of about 6 new cases per one million people per year [7]. Melanoma of the eye is related to melanoma of the skin. Both develop from *melanocytes*, cells making pigment (color). Secondary intraocular tumors have their origin in another organ and spread to the eye afterwards. Most common cancers spreading to the eye are lung and breast cancer. Even though secondary eye cancers are not truly eye cancers (in the sense of cancers that originate from the eye), nevertheless they are more common than primary eye cancers. Figure 2.2 illustrates a malignant choroidal tumor.

Not much is known about risk factors for eye cancer and what causes it. Consequently, it is not yet possible to prevent it. However, there is a link between melanoma of the skin and sunlight, accordingly protecting against sunlight with UVA and UVB absorbing sunglasses might reduce the risk for getting an intraocular tumor.

Since eye cancer is uncommon, regular screenings are not recommended. Intraocular tumors are often found during a routine eye examination. Symptoms, amongst others, may include: problems with vision (blurred vision or sudden flashes), visual field loss, changes of shape or color of pupil or iris.

If symptoms are spotted, an ophthalmologist will most likely use a *fundus camera* or a *slit lamp* to see through the lens into the posterior chamber of the eye. This is usually the first examination before further imaging tests are done. *Ultrasound imaging* or *magnetic resonance imaging* are both suitable imaging modalities for the further examination of eye tumors. Sometimes a biopsy is required for the identification of an intraocular tumor. However, there is a risk, that some critical structures within the eye are damaged or that the tumor spreads because of the needle injection.

---

[3]Original image from: *National Eye Institute, National Institutes of Health* (https://nei.nih.gov)

Figure 2.2: Malignant choroidal tumor[4]

Intraocular tumors can be treated with surgery, radiation therapy, and laser therapy, amongst others. Sometimes therapies are combined. Ophthalmologists and oncologists propose the treatment or a combination of treatments, which promise best success. An eye cancer treatment might affect the vision. Obviously, doctors try to preserve vision but sometimes even enucleation of the eye is unavoidable. Radiation therapy is most often used to treat intraocular tumors, because some vision can often be saved and the eye structure is preserved, which results in a better appearance after the treatment. *Brachytherapy* is the most common radiation therapy, where small radioactive pellets are sutured on or close to the cancer. The pellets remain there for a couple of days, before they are removed again. When using lasers, a highly focused beam (e.g. infrared) is used to destroy the tumor cells by heating them up. *External beam radiation therapy* is an approach, in which radiation is coming from a source outside of the body, where X-rays or protons can be used [8]. Details about the proton therapy can be found in the next section.

2.2 Proton Therapy

Proton therapy is a good option to treat intraocular tumors, the local tumor control rate is about 95 % [9, 10]. Thereby the eye can usually be conserved (eye retention rate of approx. 90 %), and frequently the eye function is preserved [9, 10, 11].

There are about 60 clinical proton therapy facilities worldwide, thereof 20 are in Europe. In Switzerland currently one clinically operating center for proton therapy (CPT) exists, which is located at the *Paul Scherrer Institute (PSI)* in Villigen. Intraocular tumors have been treated at PSI since 1984.

To write this chapter we used information provided by two centers for proton therapy, PSI[5] and *Proton therapy center, Prague, Czech Republic*[6]. Further details about the technology and

---

[4]Image from: *PIER digital library, Pathology Education Instructional Resource*
[5]https://www.psi.ch (accessed December 2016)
[6]http://www.proton-cancer-treatment.com (accessed December 2016)

the treatment going beyond what we provide here can be found on their websites and in the publication of Damato et al. [9].

Principles of Proton Therapy

The basis of this particular therapy are the protons, which are *positively charged elementary particles of hydrogen atom nuclei*. Protons can be accelerated in a *cyclotron* to a speed equal to approximately half the speed of light. This results in an energy of up to 230 MeV, which also determines the maximum depth of penetration of about 30 cm. OPTIS2, the facility to treat eye cancer at PSI, has a fixed horizontal beam line, which operates with an energy of 70 MeV.
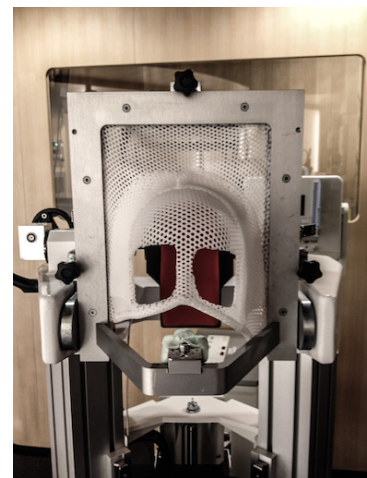
After the acceleration of the protons, they are guided through several components (*range shifter*, *scatter foil*, *modulator wheels*), such that a homogeneous circular field with a maximal diameter of 35 mm gets delivered at the *snout / nozzle*. This field is reduced for each patient, depending on the size of the tumor, using an individually milled copper collimator aperture. Figure 2.3a illustrates the *snout*, where the proton beam exits.

The proton beam releases its energy by deceleration and this results in an ionization of the tumor cells, which then ultimately die.

One of the key success factors of the therapy is the fact that the radiation dose can be delivered very precisely, due to the *Bragg peak*. The maximum of the Bragg curve, the Bragg peak, corresponds to the highest energy loss of charged particles at a certain depth in the tissue. There is a low entry dose, where the lens is, a maximal dose of energy at the required depth where the tumor cells are and a zero exit dose, close to the optic nerve, the retina and the brain. This is notably important for organs with a high density of critical structures like the human eye.
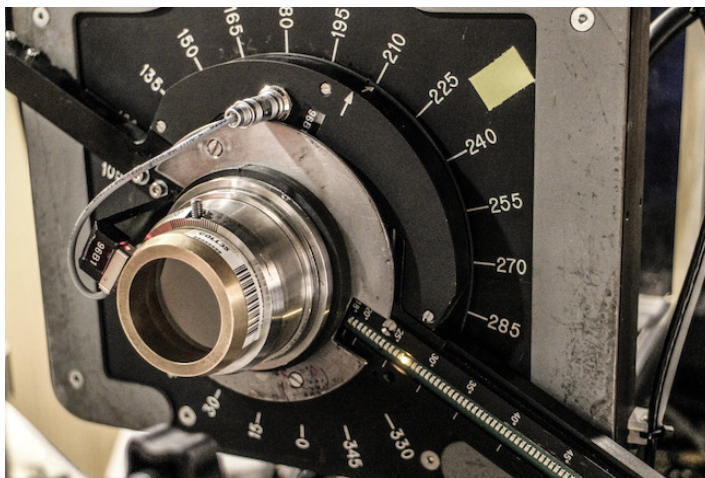
(a) Snout - the proton beam exit   (b) Headmask for the patient

Figure 2.3: OPTIS2 treatment facility at PSI, Villigen

Today's Navigation

For the treatment, the patient is seated in a motorized chair with six degrees of freedom (DOF) and the patient's head is fixated using an individually produced head mask and a bite block.

Figure 2.3b shows the upper part of the chair with the head mask. For an accurate patient positioning with respect to the treatment beam, OPTIS2 is equipped with two orthogonal X-ray units, which are aligned with the treatment isocenter. Since the tumor cannot be visualized with X-rays, a pre-treatment surgery is required. Therefore, a surgeon sutures radio-opaque clips to the outer scleral surface around the tumor periphery in order to define the tumor location within the eye and to be able to target the tumor during the proton therapy. X-rays are used to localize the clips allowing for an iterative alignment of the eye for the actual therapy.

The orientation of the eye (gazing angle) is defined through a fixation light, a small LED that can be placed at any point in the proximity of the nozzle and that the patient has to fixate during therapy. The LED bar can be seen in Figure 2.3a. Eye movement during the radiation is monitored using a camera.

The current method of choice for therapy planning by Goitein and Miller [12] has been in use with small alterations for more than three decades. Despite its successful outcome, the main limitation of the current method is that it requires an invasive patient preparation.

2.3 Eye Tracking

Hansen et al. [13], Narcizo et al. [14], as well as Duchowski [15, 16] and Morimoto et al. [17] provide excellent overviews of eye tracking technologies, algorithms, and applications. To write this chapter, we used a lot of valuable information from their publications, which we also recommend for further reading.

What is Eye Tracking?

Eye tracking, sometimes called gaze tracking, is a multidisciplinary research field, which can involve mechanical engineering, optical engineering, image processing, mathematical modeling, and pattern recognition, amongst others. As diverse as the involved disciplines are, as diverse are the applications. Eye trackers are typically used for:

- Usability testing of graphical user interfaces or websites

- Market research to discover what attracts the customers

- Human-computer interactions to replace for instance mouse and keyboard

- Monitoring of pilots and car drivers to improve safety while flying or driving

The common purpose of all the applications is the possibility to monitor and process eye movement of humans. The maybe most frequently calculated eye tracking result is the point of gaze, the point, where the line of sight intersects a plane in space (e.g. a computer screen). Almost everything is possible between simply detecting the presence of an eye in a scene and complex interactions, where the eyes are used to take over the control of a system. Using eye trackers for 3D navigation is, however, not common.

Properties of Eye Trackers

When deciding for a certain eye tracking architecture, a few system characteristics have to be considered, since they may vary considerably:

- What is the required eye tracker accuracy and reliability?

- How robust will the illumination conditions be during the eye tracking?

- What is an acceptable effort for a system- and user-calibration?

- Is the device size important?

- How much may the system cost?

Depending on the requirements, different eye tracker constructions, methods, and algorithms might be suitable. The following section provides a short summary of different types of eye trackers.

Classification of Eye Trackers

Eye trackers can be classified in several different ways. First of all, eye trackers can be subdivided into *intrusive* and *non-intrusive* devices [16]. The intrusive techniques, based on electrodes or contact lenses with search coils, are in physical contact with the eye or the skin. The non-intrusive techniques are mostly video signal based, which means they process images from a video camera. The video based eye trackers (video-oculography systems) are the most widely used eye trackers because of their simplicity and the good availability of the required hardware components.

The video based eye trackers can further be subdivided, dependent on their construction into *head-mounted* devices, *desktop* devices, and *embedded* devices. Head-mounted, wearable devices are mostly mounted onto a helmet or on eye glass frames. These types of eye trackers are ideal for mobile applications, however, they are usually not as accurate as stationary devices. Stationary eye trackers my be placed on a desk or they are embedded into another device, like an airplane or a medical device.

Considering video based eye trackers, further classifications depending on the algorithms can be done [13]. Basically, every video based eye tracker has to perform two tasks in order to determine the point of gaze, its main intended use. First, the video image has to be evaluated and at least one eye has to be detected. Second, the extracted information about the eye has to be processed for the gaze estimation.

To distinguish between different eye detection methods, the most important property is maybe whether the image gets interpreted in its entirety or if there are certain features, for which an algorithm searches. Appearance based methods detect eyes directly, for instance by comparing an input image with template images. In contrast to that, feature based methods look for some features within the input image, typically the pupil and some glints, reflections of lamps on the corneal surface.

To detect the line of sight and the point of gaze, again, lots of methods exist with their own advantages and disadvantages. For instance, *interpolation based* methods, also called *2D regression based* eye trackers, assign a certain feature constellation to a certain point of gaze on a screen based on two calibrated polynomials. The gaze points in between the calibrated points

get interpolated. This is maybe the easiest gaze estimation method in terms of implementation, but it has also a few limitations.

3D model based methods are much more flexible and powerful than the interpolation based methods, because they model the whole scene with the camera and the eye in 3D.

For our purpose only the 3D model based eye trackers are suitable, since they model also the 3D location of the eye.

Figure 2.4 visualizes the most important distinction for video based eye tracking algorithms. From here on, we focus on non-intrusive, feature based eye tracking, which is relevant for this project.
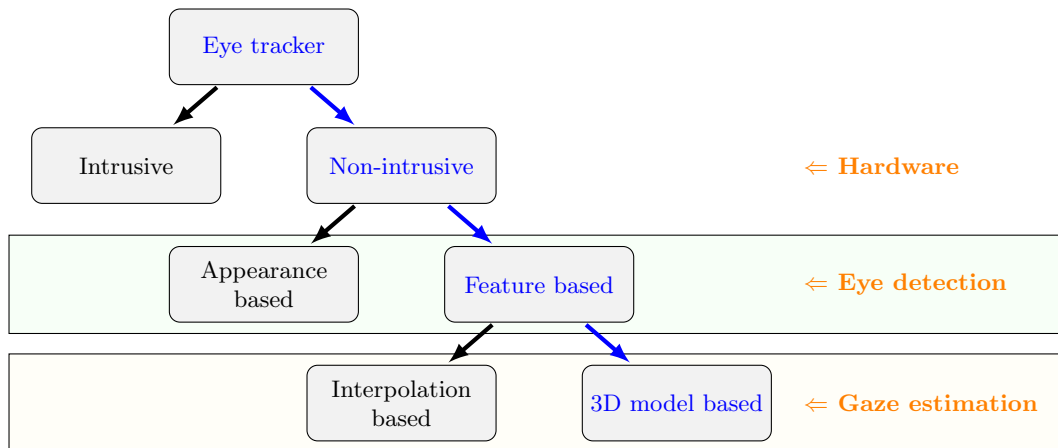

Figure 2.4: Eye tracker classification

Illumination and Pattern Projection

When working with videos, it is crucial to have a well illuminated scene. Therefore, an active light source is recommended. This enables an operator to use an eye tracker also in dark environments. Mostly, the active light source is not only used as ambient light, but also to produce reflections on the cornea, which are tracked and processed for the point of gaze estimate.

For this, usually near infrared light is used ($\lambda = 800\,\text{nm}$ to $900\,\text{nm}$). The advantage is, that the user does not see the corresponding wavelength and is therefore not disturbed by the active illumination. Furthermore, an eye tracker based on near infrared illumination can also be used in the dark. In combination with an infrared-pass filter for the camera, stable light conditions can be produced for indoor applications.

The drawback of infrared light is, that the human eye has no natural defense mechanism like blinking for visible light. If the safety norms are not considered, there is a danger of heating up sensitive structures within the eye too much.

Image Acquisition

For video based eye trackers, the image acquisition may be seen as the first step in the pipeline towards a point of gaze estimate. To construct a video based eye tracker, obviously a video camera is required. Depending on the requirements on accuracy, already a cheap webcam can

possibly be used. However, a more expensive industry camera may be worth the money in other scenarios, because it provides higher resolution, bigger sensor size and details about the sensor location within the case. A bigger sensor usually has bigger pixels, which are more light sensitive than smaller ones. Besides this advantage, however, the slightly smaller depth of focus of a bigger sensor compared to a smaller sensor is a disadvantage. An industry camera can further be used with different lenses, with different focal lengths, depending on the requirements. High focal length (tele lens) will lead to higher accuracy but does not allow for big head movements compared to a lens with low focal length (wide angle lens). Having a suitable camera and lens, it is crucial to find a good tradeoff for the aperture. The smaller the aperture number, the more light can pass through the lens during a certain timeframe. This results in shorter exposure times and with that in sharper images, in case of fast eye movement. In this case, however, the depth of focus is shallow. The bigger the aperture number, the higher the depth of focus but also the less light may pass thorugh the lens, which may result in motion artifacts, because of longer exposure time. Consequently the aperture of the lens has a direct and indirect influence on the brightness and the sharpness of the image.

Feature Detection

The second step, after an image of an eye has been acquired, is the detection of eye features. Some methods detect the pupil, some the limbus, some methods even have a model for the whole eye (including the eye corners and the eye lids) [18, 19, 20, 21]. Depending on the extent and the complexity, eye detection algorithms may even detect blinks. Figure 2.5 illustartes a typical image acquired for eye tracking purpose with some detected features. The coordinates of one or more glints and the pupil or the limbus are passed to the gaze estimation, which is responsible for estimating the point of gaze.
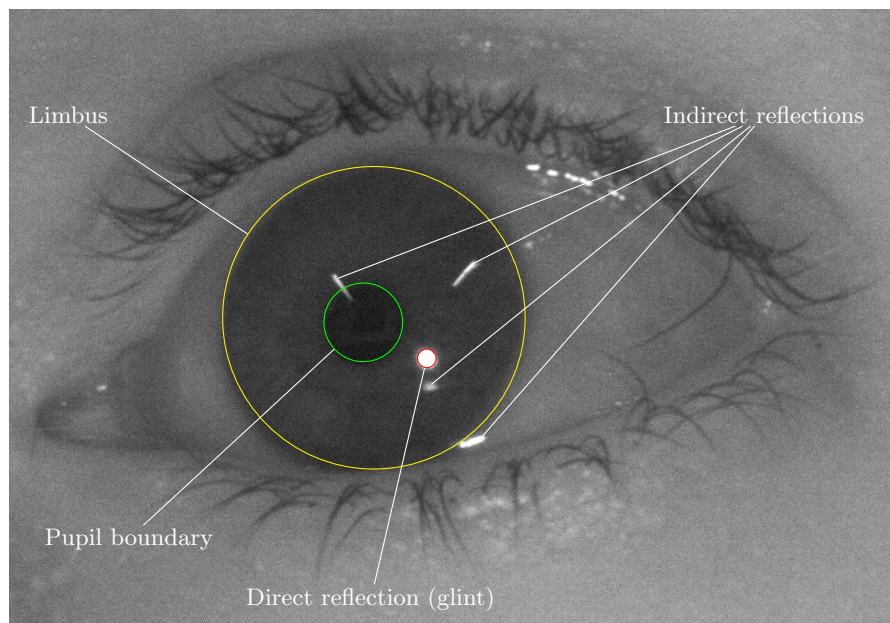

Figure 2.5: Typical camera image with detected features

Gaze Estimation

The ultimate goal of this step in the pipeline is to determine the point of gaze, having certain eye features available from the previous step. Lots of different methods exist to achieve this goal [22, 23, 24, 25], some are particularly optimized for little calibration effort, others for high accuracy.

The simplest video based gaze estimator uses 2D regression [17, 26, 27]. In the following, we show the working principle of this gaze tracker.

The cornea is considered to be a perfect spherical mirror and consequently a cornea reflection (glint) of a stationary light source should stay at the same point in a camera image, even if the eye is rotated around its center. However, it is assumed that the eye stays at a fixed position.

The relative constellation of the detected pupil center to the glint is used to determine the point of gaze. Two non-linear second order polynomials with altogether 12 degrees of freedom are used for the mapping of the pupil-glint vector to a point of gaze. Let v_x and v_y be

$$v_x = a_0 + a_1 \cdot u_x + a_2 \cdot u_y + a_3 \cdot u_x u_y + a_4 \cdot u_x^2 + a_5 \cdot u_y^2, \tag{2.1}$$

$$v_y = b_0 + b_1 \cdot u_x + b_2 \cdot u_y + b_3 \cdot u_x u_y + b_4 \cdot u_x^2 + b_5 \cdot u_y^2, \tag{2.2}$$

where $a_0, a_1, ..., a_5$ and $b_0, b_1, ..., b_5$ correspond to the parameters, which have to be trained by a user calibration. The vector $[u_x, u_y]$ corresponds to a 2D pupil-glint vector (from feature detection) and $[v_x, v_y]$ denotes the 2D point of gaze coordinates on a computer screen.

A set of calibration points is used to determine the 12 polynomial parameters. One calibration point consists of a known point of gaze $[v_x, v_y]$ in a computer screen coordinate system and a corresponding pupil-glint vector $[u_x, u_y]$ in the image coordinate system. With either *QR decomposition* or *normal equations*, the 12 parameters can be determined with at least one such calibration point. However, the capability of the gaze tracker would be very limited. It is recommended to use as many calibration points as possible, ideally distributed over the whole computer screen. As soon as the polynomial parameters are determined for a certain person, the gaze tracker is able to interpolate a newly given and a priori unknown pupil-glint vector and return a new point of gaze.

For the following example, we used a 50 mm to 50 mm plate with 16 evenly distributed calibration points. The eye tracking hardware is shown in Figure 3.1a. The polynomials were trained by a user calibration: The user fixated all 16 calibration points $[v_x, v_y]_i, i \in \{1, 2, 3, ..., 16\}$ one after the other, while at the same time, images were acquired from the camera, which is located in the center of the plate. For all images, the pupil and the glint are automatically detected and a vector $[u_x, u_y]_i, i \in \{1, 2, 3, ..., 16\}$ is built for each image.

To solve for the unknown user specific parameters $a_0, a_1, ..., a_5$ and $b_0, b_1, ..., b_5$ we rewrite the Equations 2.1 and 2.2 and convert them to the matrix form, such that we get $Ac_m = d_m$, where $m \in \{1, 2\}$ encodes either the first or the second polynomial. Matrix A consists of 16 row vectors and is the same for both equations

$$A = \begin{bmatrix} 1 & u_x & u_y & u_x u_y & u_x^2 & u_y^2 \end{bmatrix}_i, i \in \{1, 2, 3, ..., 16\}. \tag{2.3}$$

The column vectors c_m, representing the unknown parameters, are:

$$c_1 = \begin{bmatrix} a_0 & a_1 & a_2 & a_3 & a_4 & a_5 \end{bmatrix}^{\mathrm{T}}, \; c_2 = \begin{bmatrix} b_0 & b_1 & b_2 & b_3 & b_4 & b_5 \end{bmatrix}^{\mathrm{T}}. \tag{2.4}$$

The column vectors d_m correspond to either the x or y coordinates of all the 16 calibration points: $d_1 = [v_x]_i, i \in \{1, 2, 3, ..., 16\}$, $d_2 = [v_y]_i, i \in \{1, 2, 3, ..., 16\}$. The matrix A and the vectors d_1 and d_2 are filled with the observations from the user calibration:

$$\begin{bmatrix} \begin{bmatrix} 1 & u_x & u_y & u_x u_y & u_x^2 & u_y^2 \end{bmatrix}_1 \\ \begin{bmatrix} 1 & u_x & u_y & u_x u_y & u_x^2 & u_y^2 \end{bmatrix}_2 \\ \vdots \\ \begin{bmatrix} 1 & u_x & u_y & u_x u_y & u_x^2 & u_y^2 \end{bmatrix}_{16} \end{bmatrix} \cdot \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \end{bmatrix} = \begin{bmatrix} [v_x]_1 \\ [v_x]_2 \\ \vdots \\ [v_x]_{16} \end{bmatrix}, \tag{2.5}$$

$$\begin{bmatrix} \begin{bmatrix} 1 & u_x & u_y & u_x u_y & u_x^2 & u_y^2 \end{bmatrix}_1 \\ \begin{bmatrix} 1 & u_x & u_y & u_x u_y & u_x^2 & u_y^2 \end{bmatrix}_2 \\ \vdots \\ \begin{bmatrix} 1 & u_x & u_y & u_x u_y & u_x^2 & u_y^2 \end{bmatrix}_{16} \end{bmatrix} \cdot \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \end{bmatrix} = \begin{bmatrix} [v_y]_1 \\ [v_y]_2 \\ \vdots \\ [v_y]_{16} \end{bmatrix}. \tag{2.6}$$

Using the normal equations, the problem can be solved in a least-square sense:

$$c_1 = (A^{\mathrm{T}} \cdot A)^{-1} \cdot (A^{\mathrm{T}} \cdot d_1), \tag{2.7}$$

$$c_2 = (A^{\mathrm{T}} \cdot A)^{-1} \cdot (A^{\mathrm{T}} \cdot d_2). \tag{2.8}$$

Having the a priori unknown parameters c_1 and c_2 calculated, a new point of gaze can be determined using the Equations 2.1, 2.2 and a new pupil-glint vector $[u_x, u_y]$ coming from the previously done feature detection.

Figure 2.6 illustrates the result of the described interpolation based eye tracker, which we calibrated with 16 calibration points. The arrows illustrate the orientation and the magnitude of some pupil-glint vectors.

Head movement during the calibration and the actual eye tracking is not permitted and the effort required for the user calibration is rather high compared to other gaze estimation methods. This method is, however, not usable to localize the eye in 3D space, since no eye structures are modeled with this approach.

Eye Model

Compared to the 2D regression based gaze estimation, the 3D model based gaze estimation relies on a 3D model of the scene, including the eye tracker and the eye itself. To model the eye, engineers and researchers often make use of the theoretical eye model of Gullstrand-Le Grand [28]. Figure 2.7 illustrates this theoretical eye model. There, the eye is basically modeled with two spheres, one is approximating the cornea the other the eyeball. It is the same with the model
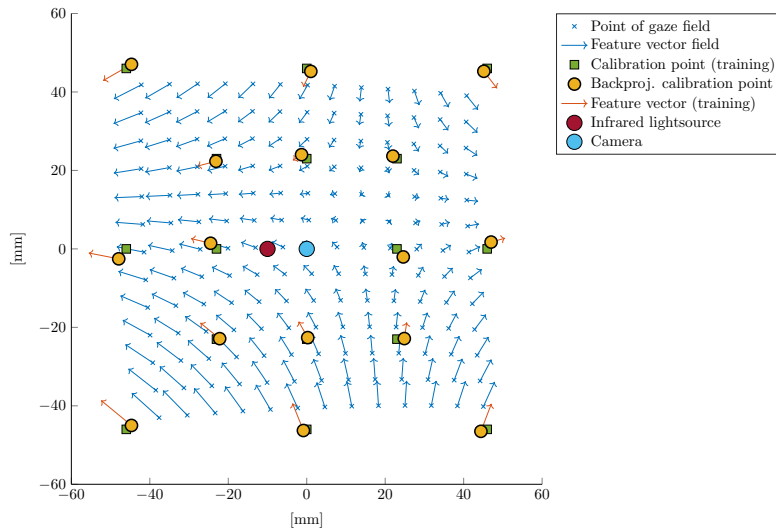
Figure 2.6: 2D regression based gaze estimation

of Guestrin and Eizenman [29], where the eye is also modeled with two spheres. The corneal radius and the refractive index, and so forth, are user specific. The eye tracking model first uses tabulated values as initial guess for the user specific parameters, afterwards the parameters get customized by a user calibration.
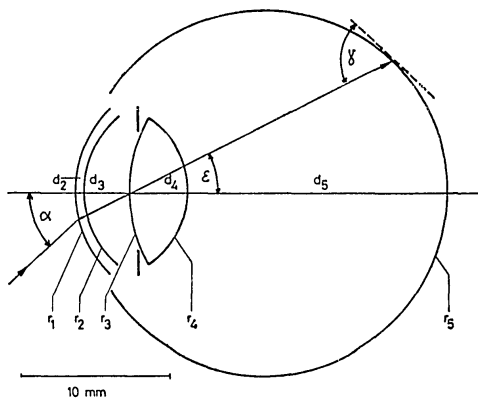
Radius (mm)	Spacing (mm)	Refractive index n_d	Medium
$r_1 = + 7.8$		1.0	air
$r_2 = + 6.5$	$d_2 = 0.55$	1.3771	cornea
$r_3 = +10.2$	$d_3 = 3.05$	1.3374	aqueous
$r_4 = - 6.0$	$d_4 = 4.0$	1.420	lens
$r_5 = -12.3$	$d_5 = 16.60$	1.336	vitreous
	$f = 22.29$ mm		

(a) Sketch: visual angle α; internal angle ϵ; angle of acceptance γ.

(b) Corresponding values

Figure 2.7: Theoretical eye model of Gullstrand-Le Grand[7].

User Calibration

For almost all feature based eye tracking systems a user calibration is required, before the eye can be tracked accurately. The calibration is required for both the interpolation based algorithms and

---

[7]Image and table from: [28]

the 3D model based algorithms. The gaze tracker usually has to be calibrated for each subject individually before the actual tracking. For the calibration, the user has to gaze at certain fixation points. However, there also exist calibration methods without an explicit procedure, so-called automatic calibration methods [30]. The fixation points are typically shown on a computer screen for a desktop eye tracker or sometimes LEDs are used for embedded eye tracking systems without display. Depending on the eye tracking algorithm more or fewer calibration points are required.

In the case of interpolation based methods, polynomial parameters have to be trained, such that they can relate certain input features to a point of gaze for instance on a screen. The calibration points have to be evenly distributed over the whole area, which is relevant for the eye tracking. For a good training it is further relevant to use lots of points, such that noise can be canceled out.

In the case of 3D model based methods, typically some internal model parameters have to be trained. Usually these parameters have an anatomical meaning. Typical parameters are α and β, representing the horizontal (pan) and the vertical (tilt) angle between the visual and the geometrical axis of the eye.

Device Calibration

The device calibration determines a priori required device parameters, including information from manufacturing, data sheets and homography. In contrast to the user calibration, the device calibration does not have to be performed for every user. It is sometimes even done only once after manufacturing. Sometimes, the operator has to repeat a device calibration after some hardware parameters were changed.

Such a calibration may include *camera calibration* to determine the camera matrix and undistortion parameters. It may further include a manual optimization of the field of view of the camera (including aperture and focus). Almost always an eye tracker has to be co-referenced with another device, since it is not sufficient to have any results, e.g. the point of gaze, just in a local camera coordinate system.

Eye Movement Analysis

With an eye tracker we are able to determine a point of gaze. As soon as we estimate a series of points of gaze over a certain period of time, we actually can talk about eye tracking. Looking at protocols of a series of points of gaze, the question arises how to interpret these points. Depending on the speed and resolution of our eye tracking system more or fewer details can be recognized. Most of the time we are looking at a certain object, this is called a *fixation*. To make sure, that the image does not fade-out, we have to refresh it with tiny, fast eye rotations, so-called *microsaccades*. The eye movement between fixations, when we redirect the fovea to a new location, is called *saccade*. Depending on the application it makes sense to only watch out for fixations. This would require a low pass filter of the signal. More advanced filters are in use, which for instance recognize unconscious eye movements occurring when certain processes in the brain happen. More information about the interpretation of eye movement protocols can be found in Duchowski [16] and Young et al. [31].

2.4 Coordinate Transformations

Medical navigation systems usually work in 3D space, as it is the case with the eye localization system presented in this thesis. Depending on the complexity of the system, more or fewer subsystems are involved, for instance cameras or displays. Subsystems communicate with each other and they may have to exchange information in the form of 3D coordinates. These subsystems, however, have their own 2D or 3D coordinate systems defined by an origin and a certain orientation of the perpendicular *unit* or *basis vectors*. Hence, 3D coordinates have to be *transformed* when they are passed from one subsystem to another.

Below we provide an introductory example to illustrate how to work with different coordinate systems and their transformations. To illustrate this, we introduce three coordinate systems (CS): CS_I, CS_W, and CS_C. The 2D coordinate system CS_I represents the camera image CS, it is a projection of the scene with the wooden cube. The camera contains a 3D camera coordinate system CS_C, which has its origin at the camera center. The camera center is at the point where all projection lines pass, the pinhole aperture of the camera. The checkerboard on the wooden cube defines a third CS, which represents the world coordinate system CS_W. Figure 2.8 shows the situation.
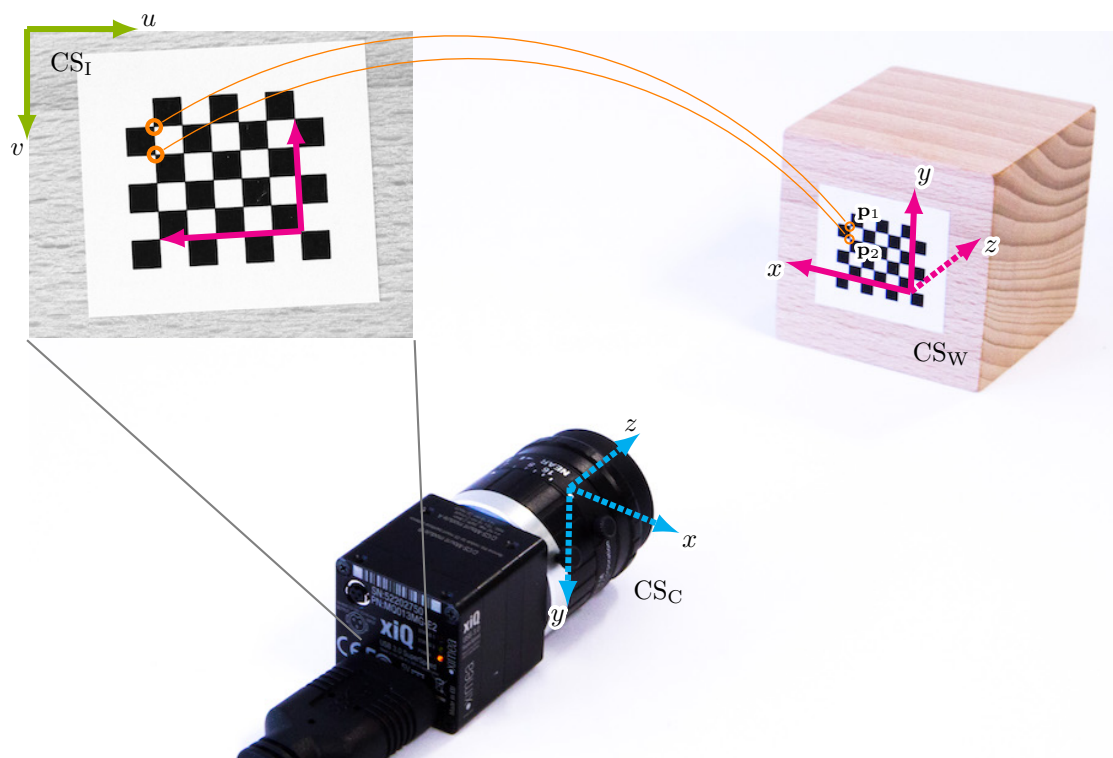

Figure 2.8: Visualized homography estimation

A typical situation, where coordinate system transformations get important is for instance when we would like to determine the camera position (camera center) relative to the world CS CS_W.

The works of Zhang [32] and Hartley and Zisserman [33] are important in the field of camera calibration, homography estimation and in general about projective geometry. Their publications can be recommended for further reading.

Homogeneous Coordinates

Homogeneous coordinates are a system of coordinates in projective geometry. Cartesian coordinates are lifted into a higher space, for instance a 3D coordinate $[x, y, z]$ gets $[x, y, z, w]$. Usually, for vectors $w = 0$ and for points $w = 1$. If after a projection, a point's component $w \neq 1$, then a normalization is required such that $[x, y, z, w]$ results in $[x/w, y/w, z/w, 1]$. This homogeneous representation of points, vectors, and transformations enables us to use a matrix for vector operations such as rotations and translations. To apply the transformation a vector can simply be multiplied with the corresponding transformation matrix.

Rigid 3D to 3D Transformations

A rigid transformation $^C T_W$ from $\mathbb{R}^3 \rightarrow \mathbb{R}^3$ is used to transform for instance the 3D coordinates of a checkerboard corner $\mathbf{p}_i, i \in \{1, 2, ..., 30\}$ from CS_W to CS_C. The orange circles in Figure 2.8 mark two of the overall 30 *inner* checkerboard corner points. The transformation $^C T_W$ is a representation of the coordinate system CS_C relative to CS_W. The *coordinate basis* of $\mathbf{p}_i$ is changed with:

$$^C\mathbf{p}_i = {}^C T_W \cdot {}^W\mathbf{p}_i, \tag{2.9}$$

where $^C\mathbf{p}_i$ and $^W\mathbf{p}_i$ represent the same homogeneous checkerboard corner with different bases:

$$^C\mathbf{p}_i = \begin{bmatrix} ^C px_i \\ ^C py_i \\ ^C pz_i \\ 1 \end{bmatrix}, {}^W\mathbf{p}_i = \begin{bmatrix} ^W px_i \\ ^W py_i \\ ^W pz_i \\ 1 \end{bmatrix}. \tag{2.10}$$

The opposite direction of the transformation can be realized with:

$$^W\mathbf{p}_i = ({}^C T_W)^{-1} \cdot {}^C\mathbf{p}_i. \tag{2.11}$$

The transformation consists of a rotation matrix R with nine values r_{ij} and a translation vector t consisting of three values:

$$^C T_W = \begin{pmatrix} R & t \\ 0^T & 1 \end{pmatrix} = \begin{pmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix}. \tag{2.12}$$

Rigid transformations, also called isometries, preserve *Euclidean distances* and consist of rotations, translations, reflections, or their combination.

Such rigid transformations, formed by 4×4-matrices are used to represent a CS and to transform homogeneous 3D coordinates from one CS to another. This is equivalent to a basis change of a certain fixed 3D point in space: the same checkerboard corner can be represented in CS_C and also in CS_W.

Projective 3D to 2D Transformations

A projective transformation ${}^I T_W$ from $\mathbb{R}^3 \rightarrow \mathbb{R}^2$ is used to transform for instance the 3D coordinates of a checkerboard corner $\mathbf{p}_i$ from CS_W to CS_I:

$$
\begin{bmatrix} {}^I p u_i \\ {}^I p v_i \\ 1 \end{bmatrix} \propto \underbrace{\begin{pmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{pmatrix}}_{\text{Intrinsic parameters}} \cdot \underbrace{\begin{pmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{pmatrix}}_{\text{Extrinsic parameters}} \cdot \begin{bmatrix} {}^W p x_i \\ {}^W p y_i \\ {}^W p z_i \\ 1 \end{bmatrix}.
\tag{2.13}
$$

It is only a proportional relationship ($\propto$), because the coordinates in CS_I have to be normalized such that the third component becomes one. The *intrinsic parameters* correspond to ${}^I T_C$ and are constant for a certain camera and are a priori not known. They can be determined with a *camera calibration*. The *extrinsic parameters* correspond to the rigid transformation ${}^C T_W$ without the fourth row. This transformation represents the camera position in the scene with a 3×3 rotation-matrix R and a 3×1 translation-vector t:

$$
\begin{pmatrix} R & t \end{pmatrix} = \begin{pmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{pmatrix}.
\tag{2.14}
$$

These parameters are a priori not known either. They can be determined by a *homography estimation*.

Camera Calibration and Homography Estimation

A *camera calibration* is a procedure to determine the intrinsic parameters, including lens distortions and the basic projective behavior. This enables us for instance to determine the relation between the camera's unit (pixels) and real world units (e.g. millimeters), and to determine the location of the camera in the scene.

During the camera calibration a set of parameters are estimated, which include intrinsics and distortion coefficients. Five parameters are used to model radial and tangental distortion. The intrinsic parameters can be modeled with four parameters f_x, f_y, c_x, and c_y:

$$
C = \begin{pmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{pmatrix},
\tag{2.15}
$$

where C is called the *camera matrix*, f_x and f_y correspond to the focal length, and c_x and c_y correspond to the *principal point* coordinate, where the principal axis intersects the sensor. For a camera with sensor dimensions of $1024\,\text{px} \times 768\,\text{px}$ in the ideal case $c_x = 512$ and $c_y = 384$.

The parameter estimation is based on algorithms and models from Zhang et al. [32] and Heikkilä et al. [34]. Implementations of these algorithms are freely available as part of the *Open Source Computer Vision, OpenCV* library[8] and from the *Camera calibration toolbox for Matlab*[9].

The camera calibration is performed by taking snapshots of a known pattern with different orientations and positions. The pattern usually consists of a grid of dots or a checkerboard with known geometry on a flat surface, like we use in our exemplary setup. For every image, all the checkerboard corner points are searched and the corresponding constant 3D coordinates (expressed in CS_W) are registered with the appropriate 2D coordinates extracted from the image (expressed in CS_I). The orange lines connecting the orange circles in Figure 2.8 visualizes this registration.

Because the camera image is only a projection of the checkerboard, it is a priori not possible to estimate the transformation between CS_I and CS_W. However, because we know the size of the checkerboard (e.g. $30\,\text{mm} \times 35\,\text{mm}$), we still can estimate the distance between CS_C and CS_W.

A homography H is nothing more than a projective transformation from 3D to 2D [35]. Basically, it is the combination of the extrinsic and intrinsic camera parameters:

$$H = {}^{I}T_{C} \cdot {}^{C}T_{W} = C \cdot {}^{C}T_{W}. \tag{2.16}$$

A homography estimation is about estimating the extrinsic parameters having C already calculated with a camera calibration. This estimation can be done with a single image of a checkerboard pattern.

The camera calibration and the homography estimation are thus important to estimate the camera position in space, relative to a world coordinate system.

---

[8]http://www.opencv.org
[9]https://www.vision.caltech.edu/bouguetj/calib_doc

3 Eye Tracking Supported Treatment

The publication in this chapter describes the new treatment scheme, which potentially enables to replace today's navigation method for the proton therapy of eye tumors.

Figure 3.1a shows our very first eye tracking prototype, with an industry camera, two infrared LEDs, and 16 colored LEDs as calibration points. The LEDs were accessible over an *Arduino* microcontroller and two shift registers. On the way to the eye tracker presented in this publication, we also made some experiments with MRI-compatibility. This is the reason for the golden color of the cable in Figure 3.1b. The golden mantle is a copper shield to reduce interference with the magnetic fields of the MRI device. We hoped to gain two things from the MRI-compatibility of the eye tracker. First, the idea was, that we could co-register the eye tracker eye model with the patient specific eye model, containing the tumor outline. Second, we thought, that the MR-data of a certain patient would reduce the user calibration cost of the eye tracker, by providing some a priori invisible anatomical lengths. The intention failed, because of the limitations of the MRI. It is difficult to acquire sharp high resolution images from the eye, when the patient is awake.

Finally, we found a solution for the co-registration of the two eye models, which does not require an extra eye tracking session in the MRI device. MRI-compatibility was therefore not an issue anymore. The new treatment scheme together with the proposed eye tracker and its integration into the treatment facility at PSI is presented in the following publication.
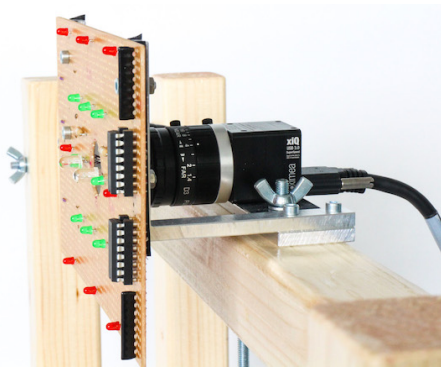
(a) First eye tracking prototype on a wooden frame   (b) Next generation eye tracker with mirror

Figure 3.1: First eye tracking prototypes

Publication

The following paper was published 2016 in the journal *IEEE Transactions on Biomedical Engineering, (TBME)*:

S. Wyder, F. Hennings, S. Pezold, J. Hrbacek, and P. C. Cattin, "With Gaze Tracking Toward Noninvasive Eye Cancer Treatment," *Biomedical Engineering, IEEE Transactions on*, vol. 63, no. 9, pp. 1914–1924, 2016

With Gaze Tracking Toward Noninvasive Eye Cancer Treatment

Stephan Wyder*, Fabian Hennings, Simon Pezold, Jan Hrbacek, and Philippe C. Cattin

Abstract—We present a new gaze tracking-based navigation scheme for proton beam radiation of intraocular tumors and we show the technical integration into the treatment facility. Currently, to treat a patient with such a tumor, a medical physicist positions the patient and the affected eye ball such that the radiation beam targets the tumor. This iterative eye positioning mechanism requires multiple X-rays, and radio-opaque clips previously sutured on the target eyeball. We investigate a possibility to replace this procedure with a noninvasive approach using a 3-D model-based gaze tracker. Previous work does not cover a comparably extensive integration of a gaze tracking device into a state-of-the-art proton beam facility without using additional hardware, such as a stereo optical tracking system. The integration is difficult because of limited available physical space, but only this enables to quantify the overall accuracy. We built a compact gaze tracker and integrated it into the proton beam radiation facility of the Paul Scherrer Institute in Villigen, Switzerland. Our results show that we can accurately estimate a healthy volunteer's point of gaze, which is the basis for the determination of the desired initial eye position. The proposed method is the first crucial step in order to make the proton therapy of the eye completely noninvasive.

Index Terms—Applied optics, camera calibration, gaze tracking, homography, infrared (IR) ray-tracing, intraocular tumors, navigation system, ocular oncology, proton beam therapy, tumor targeting.

I. INTRODUCTION

UVEAL Melanomas are the most frequent primary intraocular tumors. Untreated, they may lead to blindness and to death (caused by metastasis). Nowadays, such tumors can be treated successfully with proton radiation while conserving the eye (eye retention rate of approx. 90%), and frequently preserving the eye function [1]. One of the key success factors of the therapy is the fact that the radiation dose can be delivered very precisely, due to the Bragg peak. The maximum of the Bragg curve, the Bragg peak, corresponds to the highest energy loss of charged particles at a certain depth in the tissue. This is notably important for organs with a high density of critical structures like the human eye [2].

The current method of choice for therapy planning by Goitein and Miller [3] has been in use with small alternations for more than three decades. Despite its successful outcome, the main limitation of the current method is that it requires an invasive

patient preparation. The surgeon sutures radio-opaque clips to the outer scleral surface around the tumor periphery in order to define the tumor location within the eye and to be able to target the tumor during radiation therapy. Two orthogonal X-ray units and a motorized chair are integral parts of the treatment facility. X-rays are used to localize the clips allowing for an iterative alignment of the eye for the actual therapy.

Our objective is to bridge the gap between the fields of computer based navigation and clinical proton therapy with the goal to make the entire workflow noninvasive. A suitable gaze tracking integration is the key. The main purpose of a gaze tracker is to estimate where a human subject is looking. Additionally, some gaze trackers are able to estimate the exact position and orientation of a subject's eye in three-dimensional (3-D) space. We reuse this by-product of the gaze tracker to align the eye to the treatment beam. An integral part of those types of gaze trackers are infrared (IR) LEDs at a known position allowing to determine the eye location in a 3-D space.

By introducing a gaze tracking system into the treatment scheme, we aim to overcome the invasive preparation of patients for the treatment.

Gaze tracking systems have already been used in proton beam therapy to gate the proton beam in case of sudden eye motion [4]. To target tumors noninvasively, Rüegsegger *et al.* introduced an OCT-based tumor targeting system [5]. Fassi *et al.* [6] as well as Via *et al.* [7] also introduced gaze tracking into proton therapy to treat eye tumors. Their work covers the general idea of gaze tracking-supported tumor targeting, and they propose a specific gaze tracking system, presented independently of the actual treatment. However, their integration of the gaze tracking system into the treatment facility requires additional space-consuming and expensive hardware.

In contrast to the previous work, we strive for a whole new navigation scheme and a gaze tracking system that is completely integrated into the treatment. Our system is designed such, that it supports a fluent transition to a completely noninvasive treatment.

This paper is structured as follows: In Section II, we cover all aspects of our treatment reformation, the hardware, the algorithm, the treatment scheme, and the coregistration between coordinate systems (CS). In Section III, we present the accuracy of the overall integration, and finally, we summarize the benefits of our development and we present further work in Section IV.

II. METHODS

We present our contribution in a top-down manner. We start with the proposed treatment scheme, then we continue with the

3 Eye Tracking Supported Treatment

description of the gaze tracker integration into the treatment facility. Finally, we explain the developed gaze tracking hardware and software itself.

A. Treatment Scheme: Today and in the Future

Today, the workflow is basically as follows: After the patient has an intraocular tumor diagnosed, the ophthalmologist suggests a treatment with proton radiation at PSI and the patient has to undergo a planning phase. Amongst other ophthalmic examinations, different imaging modalities (mainly fundus and ultrasound imaging) are used to determine eye conditions, tumor location, and shape. The tumor is made visible for X-rays by suturing radio-opaque clips around the tumor periphery on the outer scleral surface (outer surface of eyeball). This is required to define the shape and to target the tumor during the actual treatment. After image acquisition and clip surgery, the medical physicist lays out a plan, based on an ellipsoidal eye model that includes the tumor shape and clips. With that plan, the ideal beam path through the eye is determined in a way, so that sensitive structures (e.g., anterior segment, macula, and optical nerve) are spared from radiation as much as possible.

The output of this planning stage is an estimated ideal gazing angle, assuming that the tumor within the eye is located at the isocenter, i.e., the reference point on the proton beam axis. For the actual radiation, the patient has to sit on the treatment chair and his or her head is fixated by a mask and a bite block. The planned gazing angle is controlled by asking the patient to fixate a shining diode that is appropriately positioned on the LED bar in front of his or her face. The patient has to be awake and be able to see and fixate the shining LED on the bar. Without the gaze tracker, the eye and tumor position of the patient cannot be measured directly. Therefore, the radio-opaque clips and two orthogonal X-ray systems are required to arrange the eye properly. This is done in an iterative way and usually requires several X-ray shots until the clips, visible on the X-ray projections, correspond to the planned clip positions. To preserve the adjusted gazing angle during radiation (gazing control), a medical physicist monitors the patient's eye motion over a camera. In case of sudden eye motion, a medical physicist manually interrupts the proton radiation.

Fig. 1 illustrates in its first column the scheme as it exists now and as it is described above. In the second column, we picture our proposed treatment scheme according to the steps below. The third column shows the scheme that we are working toward. The clip surgery would be kept for the transition period. In the future, it could be replaced by the production of a patient-specific eye model, which also contains the tumor [8], [9]. This eye model is based on a fusion of several image modalities, namely OCT, MRI, CT, and fundus imaging. For our purpose, the model could be registered with the 3-D model of the proposed gaze tracker. To do so, two distinct points and two angles would be used, which can be acquired in both models: center of pupil, center of corneal curvature, and two angles, describing the offset between optical and visual axis.

Fig. 2 illustrates the treatment hardware arrangement for a better understanding of the following scheme:
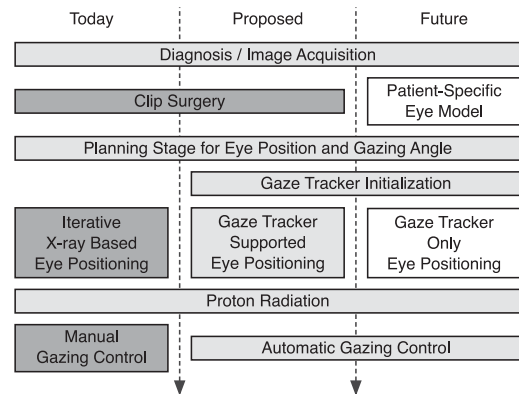

Fig. 1. Overview of treatment scheme (from top to bottom): Transition period from today (dark gray boxes) to the future (white boxes).

1) The medical physicist prepares the patient on the treatment seat and fixates its head ❿ with an appropriate mask and a bite block. The seat can be adjusted with six degrees of freedom in order to have the tumor roughly at the isocenter. This is required because the treatment device has a fixed horizontal beam line ❼.

2) The LED bar ❷ gets rotated and initialized with the appropriate LED turned ON, by means of the two polar coordinates coming from the treatment planning stage.

3) The technician rotates the entire gaze tracking hardware ❸, ❹ and ❻ to one of 36 possible discrete positions so that the gaze tracker is roughly aligned parallel to the LED bar ❷. The hot mirror ❻ is then roughly positioned in front of the shining LED. This gaze tracker alignment is a prerequisite for getting the eye into the field of view of the camera ❹.

4) Different eye positions along the beam axis might be needed for different patients, depending whether the tumor is more in the anterior or the posterior part of the eye. Accordingly, the optimal field of view and a good depth of focus of the camera ❹ has to be ensured. Slight adaptations to a new patient might therefore be required. The focus of the lens ❹ can be adjusted and the mirror ❻ can be tilted in order to get a sharp and appropriate field of view for the camera ❹. If no adaptations of the optical system (i.e., camera focus or mirror angle) are required, the camera calibration and homography can be reused from a previous session with the same gaze tracker position. If changes to the hardware were made, a gaze tracker calibration procedure is required to initialize the gaze tracking system (see below).

5) The exact position of our virtual camera ❺ in the world CS is known at this stage, thanks to the gaze tracker calibration procedure.

6) A subject-specific parameter optimization is performed by having the patient ❿ look at six calibration points given by the appropriate shining LED ❷ with known position
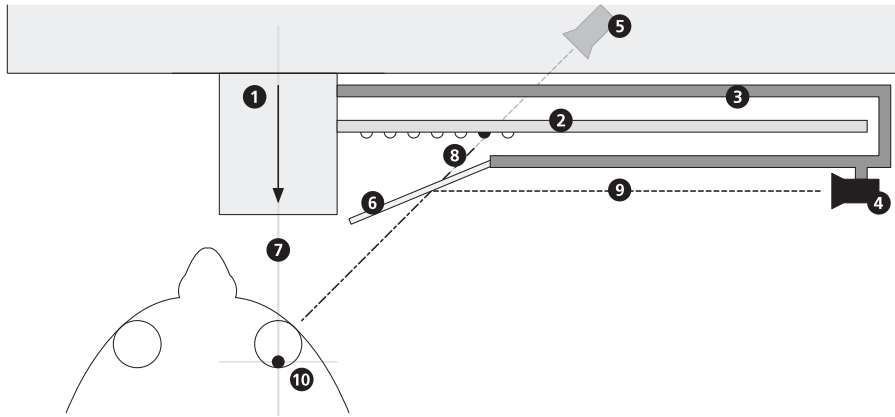
Fig. 2. Situation Plan (caudal view): ❶ Treatment device with proton beam; ❷ LED bar with activated fixation light, can be rotated around proton beam axis; ❸ Holder for gaze tracker, can be rotated around proton beam axis; ❹ IR camera and illumination (gaze tracker); ❺ Virtual gaze tracker position; ❻ Hot mirror reflecting IR wavelengths, while letting trough visible wavelength; ❼ Proton beam and rotation axis; ❽ Patient's line of sight, fixating activated LED; ❾ IR rays, imaging the eye; ❿ Patient with target eye tumor in focus at isocenter of treatment device.

in 3-D space. The algorithm has to be trained in order to fit the subject-specific parameters of the 3-D model to the appropriate patient getting treated.

7) Using the subject-specific parameters, the gaze tracker is able to estimate the point of gaze (PoG) and the position of certain anatomical landmarks of the eye (e.g., the pupil center) of the subject in the world CS. The PoG can additionally be used as fixation monitor to gate the proton beam in case of sudden eye motion. The determination of the pupil center and the nodal point of the eye (center of corneal curvature) enables us to initialize the planned position of the eye more precisely than by simply starting the iterative alignment process at an arbitrary position (simplified gaze tracker supported initialization, see Fig. 1). Since the initial estimate of the eye position is more accurate, fewer X-ray shots are required to align the tumor with the isocenter. In the future, the whole clip alignment with X-rays can be omitted, as soon as the 3-D eye model of the gaze tracker is coregistered with a patient-specific model [8], [9] showing the segmented tumor shape.

B. Gaze Tracker Calibration Procedure

Depending on the extent of adjustments made to the gaze tracker hardware since the last use, either a partial or a full calibration is required. If only the position of the gaze tracker was changed, a partial calibration is sufficient by following the instructions of steps 2 and 3. If either the focus or the aperture of the lens were changed, a full calibration is required by following the instructions of all of the following steps 1 to 3:

1) If the technicians made adjustments to the camera's lens ❹, they have to perform a camera calibration [10] in order to get the linear and nonlinear intrinsic camera parameters, namely, the camera matrix and the radial and tangential distortion, respectively. This camera calibration makes it possible to map between pixel distances of images and real-world distances in millimeter. Furthermore, it enables correcting for distortions in the acquired images. The calibration algorithm [10] requires a set of images showing a checkerboard pattern on a planar surface from different angles.

2) Once the camera ❹ is initialized (calibrated), its position relative to the world CS has to be determined. Instead of the head fixation mask for the patient, a calibration cube is mounted on the motorized chair. The calibration cube contains radio-opaque markers and can be aligned exactly to the world CS (isocenter of the proton beam) with the help of the two orthogonal X-rays.

3) Additionally, attached to the same cube is a checkerboard pattern at a known position, which is visible for the camera ❹ via the hot mirror ❻. The technician acquires a homography, based on a camera image showing the checkerboard. This enables to get the appropriate translation vector and rotation matrix from the camera CS to the checkerboard CS (finding extrinsic parameters). The optical rays between the camera ❹ and the checkerboard are deflected on the mirror ❻. Therefore the whole homography actually represents the transformation from the calibration cube to a virtual camera position ❺ behind the mirror ❻. All coordinates of the gaze tracker ❹ are transformed to the world CS by this transformation, thus the mirroring and the real camera position can be neglected.

At this stage the gaze tracker CS and the world CS have been successfully coregistered. The reason for coupling the gaze tracker CS with the world CS of the treatment device is twofold: First, to save physical space, we use the existing LED bar of the treatment device to calibrate and validate the gaze tracker. The LED bar of the treatment device itself is precalibrated and can be controlled by means of two polar angles. Thus, the treatment device itself becomes part of the gaze tracking system and vice versa. Second, we reuse the output of the gaze tracker, namely two coordinates of the eyeball, within the treatment device. For
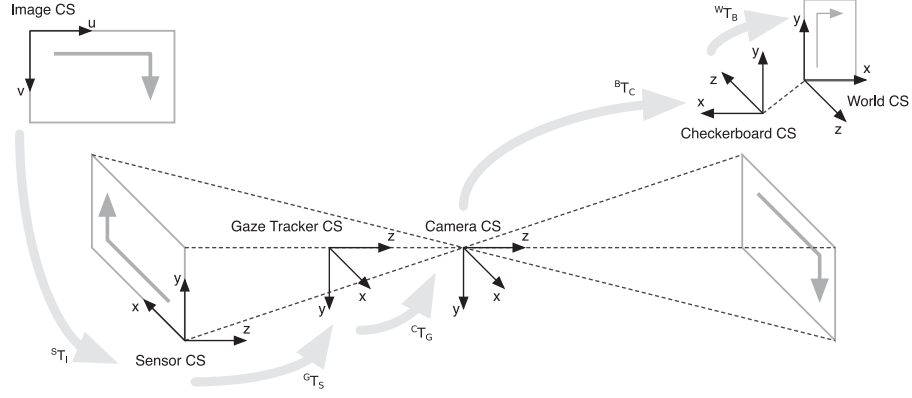
3 Eye Tracking Supported Treatment

Fig. 3. CSs and its transformations ($^{N}T_{M}$ = transformation from M CS to N CS, where M and N refer to the following CS abbreviations: I = image, S = sensor, G = gaze tracker, C = camera, B = checkerboard, W = world)—top left: image, top right: part of the treatment device, bottom: part of the camera, and the lens.

TABLE I
COREGISTRATION PIPELINE

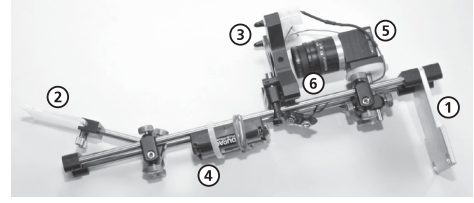|	
$^{I}\mathbf{p}$	Input: 2-D point in image CS.
$^{S}\mathbf{p} = {}^{S}T_{I} \cdot {}^{I}\mathbf{p}$	$^{S}T_{I}$ converting $[x, y]_{\text{px}}$ into $[x, y]_{\text{mm}}$ and adding third dimension to get $[x, y, z = 0]_{\text{mm}}$.
$^{G}\mathbf{p} = {}^{G}T_{S} \cdot {}^{S}\mathbf{p}$	$^{G}T_{S}$ translating to the principal point (center of sensor), rotating with $180°$ around the z-axis and translating along the z-axis with 17.52 mm (flange focal distance of c-mount cameras).
$^{C}\mathbf{p} = {}^{C}T_{G} \cdot {}^{G}\mathbf{p}$	$^{C}T_{G}$ depending on intrinsic camera parameter f (focal length). The translation along the z-axis consists of a removal of the flange focal distance and an addition of f.
$^{B}\mathbf{p} = {}^{B}T_{C} \cdot {}^{C}\mathbf{p}$	$^{B}T_{C}$ being an affine transformation coming from the extrinsic parameters of the camera calibration (homography).
$^{W}\mathbf{p} = {}^{W}T_{B} \cdot {}^{B}\mathbf{p}$	$^{W}T_{B}$ constant and given by dimensionality and location of used calibration cube.
$^{W}\mathbf{p}$	Output: 3-D point in world CS.

Fig. 4. Gaze tracking hardware for treatment facility: Consists of a frame ①, a hot mirror ②, two IR LEDs ③ with appropriate power supply ④, a camera ⑤ and a lens with an IR-pass filter ⑥.

those purposes, points in space must be transformed back and forth between the gaze tracker and the treatment device.

We propose a registration chain consisting of several CSs used as intermediate steps between the image CS on one side of the chain and the world CS (isocenter of the proton beam) on the other side.

Fig. 3 illustrates all involved CSs and shows how they are defined in space.

Table I illustrates the mentioned transformation chain, where $^{M}\mathbf{p}$ corresponds to a point vector in the M CS and $^{N}T_{M}$ represents a transformation matrix from M to N CS, where M and N refer to the following CS abbreviations: I = image, S = sensor, G = gaze tracker, C = camera, B = checkerboard, W = world.

The image, sensor, camera, and checkerboard CSs have to be used explicitly by the gaze tracking algorithm and the camera calibration. The gaze tracker CS origin is defined such, that we can measure (calibrate) the IR LEDs relative to it. We need at least one such "visible" CS, since all the others, close to the gaze tracker, are either hidden in the case (sensor CS) or virtual and not fixed relative to the hardware (image CS and camera CS). A good general overview of camera models and

calibration, coordinate transformations and its state-of-the-art nomenclature can be found in [11]. We do camera calibration and homography estimation with the OpenCV C++ library [12] by using the state-of-the-art algorithm described in [10].

C. Gaze Tracking Hardware

The present gaze tracker hardware consists of a frame, a hot mirror, an industry camera, two IR LEDs, and a battery pack. Fig. 2 illustrates the concept and Fig. 4 shows the prototype with all its components.

Especially the construction of the frame, which holds the remaining components, is challenging. The frame stability drastically impacts the reliability of the whole system. Slight displacements of hardware components may lead to drastic accuracy and repeatability loss. Position and angle changes of either the camera, the LEDs or the mirror could potentially occur during treatment because of vibrations of the treatment apparatus. Consequently, a very rigid frame construction is required to reach the desired mathematical stability in the 3-D model. However, optical engineering usually involves a lot of trial and adjustment with optical components for aspects which cannot be precalculated. That is why we finally ended up with optomechanical components to build the prototype for the treatment facility.

Those components enable us to build a rigid construction, while slight adjustments are still possible.

The hardware components are introduced briefly below: The industry camera (XIMEA MQ013MG-E2) has a C-mount for lenses, runs with 60 Hz at a resolution of 1280×1024 px.

The lens has a focal length of 35 mm and additionally an IR-pass filter, which blocks visible wavelengths below 650 nm.

Two IR LEDs (Osram Components SFH 4555), with a wavelength $\lambda = 860$ nm, 100 mA and 1.5 V each, are responsible for the ambient lighting of the eye as well as for the glints (specular reflections) on the subject's cornea used by the gaze tracker for spatial referencing. The two LEDs are connected in series with three 1.5-V batteries and a series resistor of 15 Ω.

The image quality is mainly given by the following parameters: aperture of the lens, exposure time of the camera, and amount of available light given by the LEDs.

Unfortunately, the human eye has no protection mechanism to avoid damage from IR light, since it is not visible. Therefore, it has to be ensured, that the radiation intensity limits, given by the safety norms, are not exceeded. In our case, the minimal distance between the eye and the LEDs has to be 250 mm, which is guaranteed by design.

A hot mirror is a coated piece of glass, which reflects IR light but is permeable for visible light. We use a hot mirror because we can save physical space by reflecting the optical rays of our 3-D gaze tracking model. Likewise, we do not want to hide a lit red LED on the LED bar, which could potentially be behind the mirror. Fig. 2 illustrates the principle and the physical space available.

D. Gaze Tracking Algorithm

The purpose of the gaze tracking algorithm is to estimate the PoG of the subject under observation. A good overview of existing gaze tracking algorithms is presented in [13]. We work with an extended version of a popular 3-D model described in [14]. The elegance of this model lies in its flexibility, since several different hardware configurations are possible. The configurations vary from one camera with one light source to multiple cameras with multiple light sources. The more cameras and lights used, the fewer the parameters that have to be optimized within the 3-D model. However, all the component positions have to be determined and registered with the gaze tracker CS. Our configuration with one camera and two light sources is a proper compromise between size of the apparatus and complexity of the model's solution space.

The algorithm can mainly be subdivided into the following parts: *Feature Detection, Model Fitting*, and *Subject-Specific Parameter Optimization*.

1) Feature Detection: As feature points on the eye, we use the coordinates of the pupil center and of both IR reflexions (glints). These features are detected in a semisupervised manner. By performing it this way, compared to a fully automatic manner, we reduce the complexity of the feature detection, and with that, the calculation time, whereas we increase the stability for different light conditions. Since we work in the IR range, the light conditions change mainly when sunlight is present or when camera gain, aperture, or exposure time is changed.

Let $\Omega = \{1 \ldots 1280\} \times \{1 \ldots 1024\}$ denote the discrete domain of the images provided by the camera, let $\Omega_R \subseteq \Omega$ denote a rectangular region of interest, and let $I : \Omega_R \to \{0 \ldots 255\}$ with $I = I(u, v)$ denote the image function that maps from pixel coordinates (u, v) within the region of interest to grayscale values. The region Ω_R is set manually to extract the eye. To detect the pupil and the glints, the supervisor also sets seed points $\{(u_i^p, v_i^p)\}_{i=1}^{N_p}$ for the pupil, $\{(u_i^i, v_i^i)\}_{i=1}^{N_i}$ for the iris, and $\{(u_i^g, v_i^g)\}_{i=1}^{N_g}$ for the glints. For the respective pixel intensities, we calculate the average values a.

$$a_p = \frac{1}{N_p} \sum_{i=1}^{N_p} I(u_i^p, v_i^p) \tag{1a}$$

$$a_i = \frac{1}{N_i} \sum_{i=1}^{N_i} I(u_i^i, v_i^i) \tag{1b}$$

$$a_g = \frac{1}{N_g} \sum_{i=1}^{N_g} I(u_i^g, v_i^g) \tag{1c}$$

then, we determine thresholds t_p and t_g for the pupil and glint, respectively, as

$$t_p = \frac{1}{2}(a_p + a_i) - s_p \quad \text{and} \quad t_g = a_g - s_g \tag{2}$$

where s_p and s_g are predefined tolerance values. Finally, we use the thresholds to calculate two binary images I_p and I_g as follows:

$$I_p(u, v) = \begin{cases} 1, & I(u, v) < t_p \\ 0, & I(u, v) \geq t_p \end{cases} \quad \text{and} \tag{3a}$$

$$I_g(u, v) = \begin{cases} 1, & I(u, v) > t_g \\ 0, & I(u, v) \leq t_g. \end{cases} \tag{3b}$$

Having the binary images, we label the individual blobs of I_p and I_g by applying a standard eight-connected component labeling algorithm. This results in $\hat{I}_p$ and $\hat{I}_g$, where the individual, separated blobs are labeled by an identifier number. The glint segmentation does not need any further processing, since the IR reflexions are by far the brightest regions in Ω_R. Since the pupil segmentation does not yet result in such a homogeneous result, we apply a closing morphological operator $\bullet$ to the labeled pupil image $\hat{I}_p$

$$\tilde{I}_p = \hat{I}_p \bullet S \tag{4}$$

where S is a structuring element with a circular shape and a radius of 9 px. This operator smoothes in particular the border region of the pupil.

In the case where several extracted pupil regions k occur, we have to distinguish the correct one from the wrong ones. Potentially wrong, dark regions, for example, come from mascara of women's eyes. For that, we calculate two different region properties, the equivalent diameter d_k and the eccentricity e_k for all k regions (pupil candidates). We first calculate the equivalent diameter d_k, where n_k is equal to the number of pixels

corresponding to the region k

$$d_k = \sqrt{\frac{4 \cdot n_k}{\pi}}. \tag{5}$$

To get the eccentricity of the regions, we have to get the second central moment to calculate the covariance matrices C_k. The two major eigenvalues λ_{a_k} and λ_{b_k} of each covariance matrix indicate the length of the major axis $\mathbf{a}$ and the minor axis $\mathbf{b}$. The eccentricity e_k can then be calculated for all regions k by

$$e_k = \sqrt{1 - \frac{\lambda_{\mathbf{b}_k}^2}{\lambda_{\mathbf{a}_k}^2}}. \tag{6}$$

Finally, the pupil candidate $\breve{I}_{\mathrm{p}}$ is selected by taking the label k of the biggest blob $\max(d_k)$ for all k, where the eccentricity $e_k < 0.7$

$$\breve{I}_{\mathrm{p}} : \Omega_{\mathrm{R}} \to \{0, 1\}, \breve{I}_{\mathrm{p}} := \begin{cases} 1 & \tilde{I}_{\mathrm{p}} = \arg\max_k(d_k) \\ 0 & \text{else.} \end{cases} \tag{7}$$

Having the pupil candidate $\breve{I}_{\mathrm{p}}$, we extract the edges $\breve{I}_{\mathrm{p}}$ of the region by applying the Sobel operator. To finally get the coordinates of the pupil center, we fit a circle into the edge image $\breve{I}_p$ with the Hough transform. The parametric form of a circle has three degrees of freedom (x and y coordinates and radius). In order to save calculation time, we limit the radius parameter to $\max(d_k) \pm 5\,\mathrm{px}$. Having the Hough accumulator H, we take the average of the four strongest peaks $\{\mathbf{h_1}, \ldots, \mathbf{h_4}\}$ of it

$$\mathbf{p} = \frac{1}{4} \cdot \sum_{n=1}^{4} \mathbf{h}_n. \tag{8}$$

The first vector component, $p_1 = p_x$ corresponds to the x coordinate, $p_2 = p_y$ to the y coordinate, and $p_3 = r$ to the radius of the detected pupil. Taking four peaks, rather than just one, enables us to take ellipsoidal shapes into account as well. This is in particular important, when the pupil, assumed to be a circle, is viewed from the side.

There may be cases where more than two glints are visible, two deflected by the mirror and two reflections coming directly from the LEDs. We calculate the centroids for all glint candidates in $\hat{I}_g$ and we compare them to the centroid of the pupil (p_x, p_y). The glint pair $(g_{x_1}, g_{y_1}, g_{x_2}, g_{y_2})$ closest to the pupil center is the pair we are looking for. Finally, we transform all three feature points to the world CS and pass them to the gaze estimation algorithm, i.e., the model fitting. Fig. 5 shows a typical eye image with its features.

2) Model Fitting: The basis of the 3-D gaze estimation model [14] is a set of equations describing relations in a 3-D vector space. The whole system of equations has to be solved repeatedly, for each re-estimation of a PoG, respectively, the estimation of the position of the eye. Solving the system is basically equivalent to estimating the unknown model parameters. The equations and the corresponding diagram can be found in the Appendix.

The system of equations can be subdivided into three subsystems, which get solved sequentially (see Fig. 7). The first system consists of 13 scalar equations with 12 scalar unknowns. The
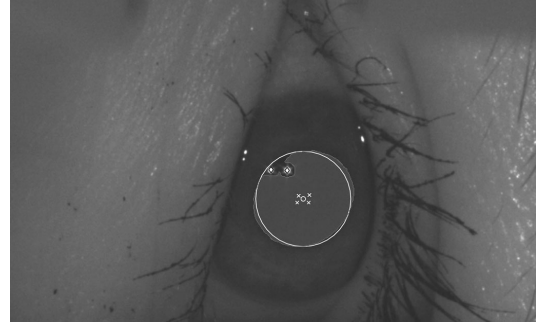

Fig. 5. Typical region of interest of the eye with features: Diamonds correspond to glints, the crosses correspond to Hough peaks resulting in the circle fit for the pupil.

individual equations bring certain points of the model in relation by using equations of lines, Euclidean distances between points and the law of reflection. The system is overdetermined and nonlinear because its unknowns are not separable (mixed unknown parameters). The respective output of the previous system is used to solve the next set of equations. The second system consists of a line equation and of a distance between points. Here, we finally end up with four scalar equations and four scalar unknowns, and the system is of quadratic order and can, therefore, be solved analytically. The third system, finally, consists of a Euclidean distance between two points, a triple product showing the coplanarity of three points and the law of refraction describing the situation on the cornea surface. This system contains three scalar equations with three scalar unknowns. It is nonlinear, again, because of mixed terms consisting of unknown parameters.

The model mainly consists of four different kinds of parameters.

1) Hardware related parameters have to be set once according to the gaze tracking hardware. This parameter set comprises of the position of the light sources, the nodal point of the camera, and the details about the camera sensor.
2) Subject-specific parameters are constant for a certain subject and cannot be measured directly: angles and lengths within the eye and the mean index of refraction of the aqueous humor and the cornea. These parameters get calibrated for each subject. As starting values for the optimization, tabulated average values are used.
3) Input parameters are the coordinates of the detected eye features (glints and pupil).
4) The rest of the parameters are only time dependent and describe the position of certain points within the subject's eye and the subject's PoG.

For more information on parameters, refer to [14].

Precise initial values are required for the last group of parameters, since the systems of equations are nonlinear and we are not using global optimization. The theoretical eye model of Gullstrand and Le Grand [15], and the hypothetical tumor position at the isocenter, enables us to estimate the required initial guess for the points in the eye.
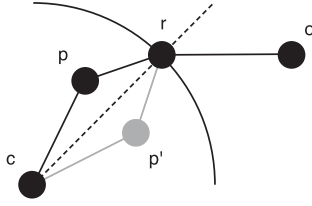
Fig. 6.   Points within the eye: $\mathbf{c}$ is the center of corneal curvature, $\mathbf{p}$ is the pupil, $\mathbf{r}$ is the point of refraction, $\mathbf{o}$ is the nodal point of the camera.

The parameter fit for the gaze estimation may find a numerically correct solution which is, however, not correct in a physical sense. This results in a wrong PoG estimation. Parameter $\mathbf{p}$ of the 3-D model [14] is especially affected. The 3-D model ensures the coplanarity of the corneal curvature $\mathbf{c}$, the pupil center $\mathbf{p}$, the point of refraction $\mathbf{r}$, and the nodal point $\mathbf{o}$ of the camera, but it cannot distinguish between $\mathbf{p}$ and $\mathbf{p}'$. Fig. 6 illustrates the situation.

In addition to the correct point $\mathbf{p}$, there is a numerically correct result $\mathbf{p}'$, which, however, is physically implausible because of the law of refraction. To overcome this, we introduce an additional equation, with which we compare the direction of two normals $\mathbf{n}_a$ and $\mathbf{n}_b$. The direction of the normal, given by the cross product of two vectors, is defined by the right-hand rule. This allows us to distinguish the two cases. The normal $\mathbf{n}_a$ on the plane defined by $\mathbf{c}$, $\mathbf{p}$, and $\mathbf{r}$ points in the same direction as the normal $\mathbf{n}_b$ on the plane defined by $\mathbf{p}$, $\mathbf{r}$, and $\mathbf{o}$

$$\mathbf{n}_a = [(\vec{\mathbf{c}} - \vec{\mathbf{p}}) \times (\vec{\mathbf{r}} - \vec{\mathbf{p}})] \tag{9}$$

$$\mathbf{n}_b = [(\vec{\mathbf{p}} - \vec{\mathbf{r}}) \times (\vec{\mathbf{o}} - \vec{\mathbf{r}})] \tag{10}$$

$$\cos^{-1}\left(\frac{\mathbf{n}_a \cdot \mathbf{n}_b}{||\mathbf{n}_a|| \cdot ||\mathbf{n}_b||}\right) = 0. \tag{11}$$

Additionally, we introduce a stochastic optimization by solving the third system again with randomly chosen initial values, when the system finds a wrong local minimum. This wrong local minimum can be detected when the residual of (11) is far from zero.

3) Subject-Specific Parameter Optimization: The previously mentioned parameter set consists of
1) $R =$ Radius of corneal curvature.
2) $K =$ Distance between the center of the pupil and the center of corneal curvature.
3) $n_1 =$ Effective index of refraction of the cornea and the aqueous humor combined.
4) $\alpha_{\text{eye}} =$ Horizontal angle between visual and optical axes of the eye.
5) $\beta_{\text{eye}} =$ Vertical angle between visual and optical axes of the eye.

All subjects have to go through a calibration procedure, where they have to look at six predefined points. The exact location of those points is known. Therefore, the average error between the PoG estimate and the true position of those six points is used to optimize the five subject-specific parameters. In the first step, we use tabulated values to get a first estimate of the PoG.

```
1:  procedure OptimizeParameters
2:      repeat
3:          R, K, n_1, α, β ← GetNewParameterGuess()
4:          procedure GazeTracking
5:              f ← DetectFeatures(i)        ▷ f = features
6:                                           ▷ i = image
7:              OptimizeSystemOne()
8:              SolveSystemTwo()
9:              while r ≉ 0 do               ▷ r = residual
10:                 ig ← GetNewInitialGuess()
11:                 r ← OptimizeSystemThree(ig)
12:             end while
13:             return p, c, α, β, PoG
14:                               ▷ optical & visual axis
15:                               ▷ PoG = point of gaze
16:         end procedure
17:     until PoG ≈ tCP    ▷ tCP = true calibration point
18: end procedure
```

Fig. 7.   Gaze tracking algorithm.

Fig. 7 illustrates the workflow of the algorithm.

III. Results

A. Experimental Setup

We performed our experiments with ten healthy volunteers to show the accuracy of the gaze tracking system integrated into the treatment facility. Each volunteer was placed on the treatment chair of the proton radiation system at PSI. Usually, a custom-made head mask is manufactured for a patient, which fixates the head and allows only eye rotation. Since the volunteers did not have their own custom-made head masks, we used the same demo mask for all of them. Because of that, all the volunteers were able to still slightly move their head, and with that, the position of the eye (roughly ± 3 mm in all directions). This, however, should not have had a negative influence on the results, as the model is capable of handling head movement. The motorized chair was moved to the initial position, which is normally used to treat the right eye. The cornea is thus positioned quite close to the collimator, ≈ 2 cm. The distance between cornea and LED bar is roughly 11 cm. We recorded 12 different predefined calibration points per volunteer. To do so, we reused the already existing LED bar. The LEDs on the bar have a diameter of 2 mm. Since they are so close to the eye, it is very hard for the volunteer on the chair to focus well on the LEDs. Hence, a halo effect appears around the lit LED. Usually, the calibration points get distributed over the whole field of view of the subject in order to realize the best result for the calibration of the subject-specific parameters. In our case, however, we were limited to a small area to place the calibration points. This is because of the possibilities to position the fixation light with the LED bar. Per volunteer, the 12 calibration points were recorded twice, one round after the other. For every calibration point, we recorded the coordinates of the calibration point together with an image of the eye, which was gazing at the corresponding calibration point. The data were divided into four
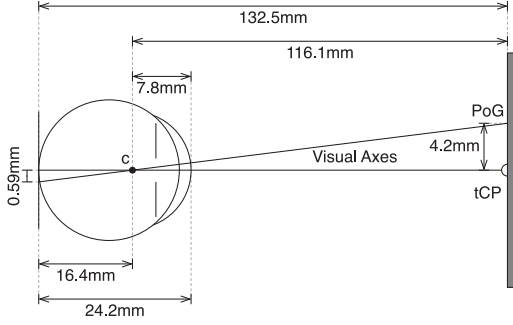
Fig. 8.   Measured error at LED bar and estimated error on retina.

Fig. 9.   Average PoG errors.

uniformly distributed subsets with six points each. On the first subset, we performed the calibration to get the subject-specific parameters. On the other three sets, we tested the accuracy of the PoG estimation with the previously optimized parameters (testing point sets).

For our experiments, we first calibrated the gaze tracker and acquired the images of the volunteers on-site at the treatment facility. Afterward, we performed the actual gaze tracking and evaluated the results retrospectively. Our algorithm is so far implemented in MATLAB and is not optimized for speed, yet. The duration for extracting the features and estimating one PoG is about 300 ms on a state-of-the-art laptop (2.8 GHz Intel Core i7). The duration for the optimization of the subject-specific parameters of one subject is about 80 s.

B. Experimental Results

The goal is to reach a sub-millimeter accuracy when determining the three dimensions of points in the subject's eye. Those points, coming from the 3-D model, are the point $\mathbf{c}$, referring to the center of corneal curvature and the point $\mathbf{p}$, the pupil center. Together with the angles α_{eye} and β_{eye}, both the optical and the visual axis can be determined exactly. However, since we cannot directly measure the accuracy at points $\mathbf{c}$ and $\mathbf{p}$, we first analyze the accuracy of our system by comparing the PoG estimation with the true calibration point (tCP) position. With that result, we calculate an error estimate on the isocenter (tumor position) by means of the intercept theorem.

The mean distance between the estimated points of gaze and the tCP is 4.20 mm. This is the overall average of ten volunteers with three point sets each and six calibration points per set ($6 \cdot 3 \cdot 10 = 180$ distances between estimated and true PoG). Those deviations can be measured directly. To show the accuracy independent of the distance to the calibration plane, the error is normalized and expressed in degrees measured at the nodal point of the eye $\mathbf{c}$. We use the theoretical eye model of Gullstrand and Le Grand [15] to get averaged eye length data (see Fig. 8).

The calibration bar with the LEDs is 132.5 mm away from the isocenter, and we assume that the tumor, and with this the retina of the eye, is at the isocenter. From the tabulated eye lengt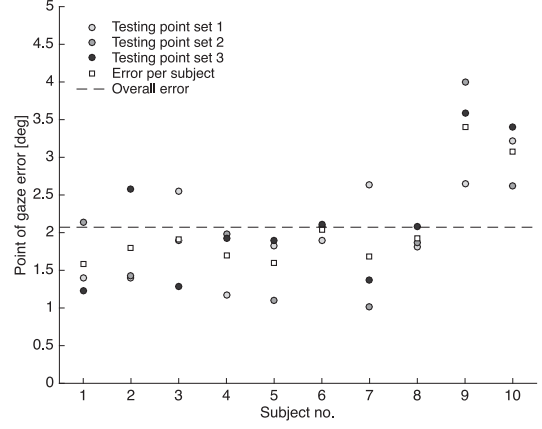h values [15], we estimate the hypothetical distance from the calibration plane to the nodal point of the eye $\mathbf{c}$ as

$$132.5 \text{ mm} - (24.2 \text{ mm} - 7.8 \text{ mm}) = 116.1 \text{ mm}. \quad (12)$$

To get the millimeter-to-degree conversion we calculate

$$\tan^{-1}(4.20/116.1) = 2.07°. \quad (13)$$

All measured data in millimeters are distance normalized and converted to degrees with the formula shown above and shown in Fig. 9. There is one column per subject and three circles per subject showing the averaged error per testing point set. The squares show the average error for one subject and the dashed line shows the overall error $2.07° \cong 4.20$ mm.

To get an error estimate on the retina's depth (and thus tumor position), we use the intercept theorem and calculate

$$\frac{4.2 \text{ mm} \cdot 16.4 \text{ mm}}{116.1 \text{ mm}} = 0.59 \text{ mm}. \quad (14)$$

Fig. 8 illustrates this error estimate. Fig. 10 shows the optimized subject-specific parameters (one parameter set per volunteer).

IV. DISCUSSION

We want to improve eye cancer treatment. For that reason, we built and presented a compact gaze tracker integrated into the OPTIS gantry of the Paul Scherrer Institute in Villigen, Switzerland. The development and the integration of our gaze tracking hardware and software enabled us to estimate the points of gaze of ten volunteers, with a mean accuracy of $2.07° \cong 4.20$ mm. This PoG error results in an error estimate of 0.59 mm on the retina. The retina is our region of interest, namely the potential tumor position, which we truly want to keep track of.

The result is encouraging, taking the complexity of the hardware and of the gaze tracking algorithm into account. The sub-millimeter accuracy is required for eye cancer treatment, and with our method we reach this requirement. The accuracy of our gaze tracking integration is expected in that range.
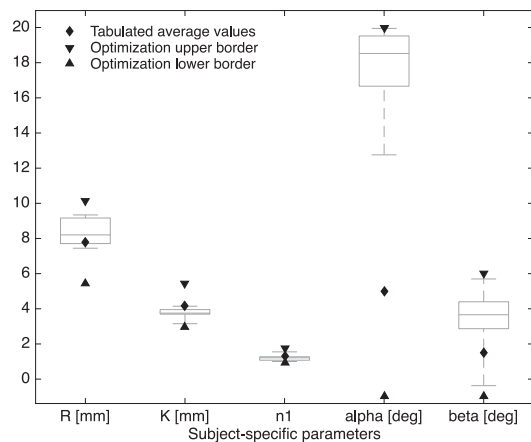
Fig. 10.   Optimized subject-specific eye parameters. For an explanation of the parameters, see Section II-D3.

When we compare our error in degrees with the error of the results shown in [14], then their result is more accurate roughly by a factor of two. This is not surprising, since the pipeline of the proposed system additionally contains the camera pose estimation (homography), which makes the system more complicated than those described in [5], [7], and [14]. Furthermore, the type and location of the calibration points (LED's), given by the treatment device, makes the subject-specific parameter optimization difficult. An accurate parameter optimization under those circumstances might, therefore, be more challenging than in a more general setup with less boundary conditions, as described in the mentioned papers. However, our results are promising when we look at the calibration points we used, which have, with the halo effect, roughly a diameter of 5 mm and we look at the rather small distribution of the calibration points.

The estimated accuracy of 0.59 mm on the depth of the retina is promising. However, optimization is necessary to fulfill the requirement of an overall accuracy below 1 mm.

The existing overall error is made up of several error sources. Two possible sources of error contribution are the resolution of the sensor (physical limitation) and the accuracy of the feature detection (algorithmic inaccuracies). A third error source is the calibration of the subject-specific parameters. The big calibration-fixation points, and the difficulty to focus on them, might be suboptimal. Additionally, we are dependent on the compliance of the subjects, how accurately they actually look at a calibration point. A fourth source of error is the homography estimation, which has an influence onto the transformation of points in space. And last but not least, we rely on the 3-D coordinates of the LEDs on the LED bar. These absolute coordinates are maybe inaccurate. One of the two outliers (subject 10) may be explained by the fact, that the corresponding volunteer was wearing contact lenses during the experiments.

When we look at the distribution of the individual subject-specific parameters (see Fig. 10) and compare them to the tabulated average values, we can see that the parameters α_{eye} and β_{eye} are comparatively far away from anatomically meaningful values. Those parameters may compensate for certain systematic errors mentioned earlier.

To get true accuracies rather than estimates at the depth where it is interesting for us, namely at the depth of the eye, we have to be able to measure the true points within the eye, which are estimated by the gaze tracker. To do so, a simplified eye phantom would be suitable, where the points in the eye (pupil, center of corneal curvature) can be calculated and compared to the estimated values. This will be subject of further research.

The integration of the proposed gaze tracking system enables us to cross check the proposed navigation scheme, which will replace the currently used initialization and navigation scheme of the treatment step-by-step.

The positive results allow us to continue working with this solution. Further hardware and software improvements are planned, which will especially lead to an even more user-friendly gaze tracker. However, accuracy improvements are also expected, since we plan, amongst other things, to use more precise and suitable optical components. We also plan to integrate an eye torsion monitoring mechanism, since the proposed gaze tracker cannot handle eye torsion so far.

Finally, our gaze tracking-based navigation scheme, together with a patient-specific eye model [8], [9], could potentially lead to a completely noninvasive proton therapy.

A major contribution of ours is the integration of the system into an already existing treatment facility. This makes the presented work quite application- and hardware-specific. However, the individual principles and the overall concept might be interesting for other facilities as well.

V. Conclusion

We developed and integrated a compact gaze tracker into the eye tumor treatment facility of the Paul Scherrer Institute in Villigen, Switzerland. With this gaze tracker, we want to accurately localize the eye in space, which supports the proper alignment of the head and the eye to the proton beam. With the proposed new treatment scheme, we aim at replacing the current tumor targeting system, which is accurate but also invasive. Our approach is completely noninvasive and we showed that the error estimate for the eye localization fulfills the required sub-millimeter accuracy. Further work, however, is required until we can treat the first patient completely noninvasively. The current results look promising and we are confident that, once the system is completed, the patients will benefit from the new treatment scheme.

Appendix
3-D Model Equations

Index i corresponds to the number of light sources, in our case, two: $i = \{1, 2\}$. Index j corresponds to the number of cameras. Since we have just one camera, this index can basically be neglected. Fig. 11 illustrates the ray-tracing diagram of the used gaze tracking model [14].
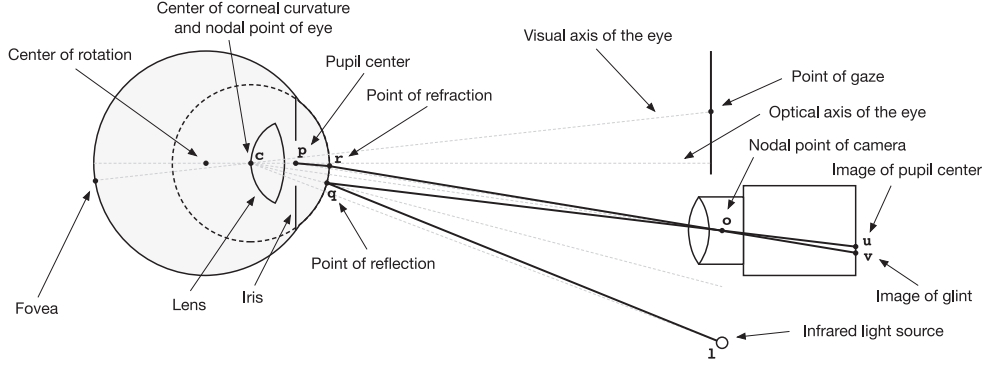
Fig. 11.   Ray-tracing diagram of the 3-D gaze tracking model based on to the scheme in [14].

Reconstruction of optical axis (System 1):

$$\mathbf{q}_{ij} = \mathbf{o}_j + k_{q,ij} \cdot (\mathbf{o}_j - \mathbf{u}_{ij}) \tag{15}$$

$$\|\mathbf{q}_{ij} - \mathbf{c}\| = R \tag{16}$$

$$(\mathbf{l}_i - \mathbf{q}_{ij}) \bullet (\mathbf{q}_{ij} - \mathbf{c}) \cdot \|\mathbf{o}_j - \mathbf{q}_{ij}\|$$
$$= (\mathbf{o}_j - \mathbf{q}_{ij}) \bullet (\mathbf{q}_{ij} - \mathbf{c}) \cdot \|\mathbf{l}_i - \mathbf{q}_{ij}\| \tag{17}$$

$$\mathbf{c} - \mathbf{o} = k_{c,b} \mathbf{b}_{\text{norm}} \tag{18}$$

$$\mathbf{b}_{\text{norm}} = \frac{\mathbf{b}}{\|\mathbf{b}\|} \tag{19}$$

$$\mathbf{b} = [(\mathbf{l}_1 - \mathbf{o}) \times (\mathbf{u}_1 - \mathbf{o})] \times [(\mathbf{l}_2 - \mathbf{o}) \times (\mathbf{u}_2 - \mathbf{o})]. \tag{20}$$

Reconstruction of optical axis (System 2):

$$\mathbf{r}_j = \mathbf{o}_j + k_{r,j} \cdot (\mathbf{o}_j - \mathbf{v}_j) \tag{21}$$

$$\|\mathbf{r}_j - \mathbf{c}\| = R. \tag{22}$$

Reconstruction of optical axis (System 3):

$$(\mathbf{r}_j - \mathbf{o}_j) \times (\mathbf{c} - \mathbf{o}_j) \bullet (\mathbf{p} - \mathbf{o}_j) = 0 \tag{23}$$

$$n_1 \cdot \|(\mathbf{r}_j - \mathbf{c}) \times (\mathbf{p} - \mathbf{r}_j)\| \cdot \|\mathbf{o}_j - \mathbf{r}_j\|$$
$$= n_2 \cdot \|(\mathbf{r}_j - \mathbf{c}) \times (\mathbf{o}_j - \mathbf{r}_j)\| \cdot \|\mathbf{p} - \mathbf{r}_j\| \tag{24}$$

$$\|\mathbf{p} - \mathbf{c}\| = K. \tag{25}$$

Reconstruction of visual axis

$$\frac{\mathbf{p} - \mathbf{c}}{\|\mathbf{p} - \mathbf{c}\|} = \begin{bmatrix} \cos(\varphi_{\text{eye}}) \cdot \sin(\theta_{\text{eye}}) \\ \sin(\varphi_{\text{eye}}) \\ -\cos(\varphi_{\text{eye}}) \cdot \cos(\theta_{\text{eye}}) \end{bmatrix} \tag{26}$$

$$\mathbf{g} = \mathbf{c} + k_g \cdot \begin{bmatrix} \cos(\varphi_{\text{eye}} + \beta_{\text{eye}}) \cdot \sin(\theta_{\text{eye}} + \alpha_{\text{eye}}) \\ \sin(\varphi_{\text{eye}} + \beta_{\text{eye}}) \\ -\cos(\varphi_{\text{eye}} + \beta_{\text{eye}}) \cdot \cos(\theta_{\text{eye}} + \alpha_{\text{eye}}) \end{bmatrix} \tag{27}$$

$$k_g = \frac{c_z}{\cos(\varphi_{\text{eye}} + \beta_{\text{eye}}) \cdot \cos(\theta_{\text{eye}} + \alpha_{\text{eye}})}. \tag{28}$$

REFERENCES

[1] E. Egger *et al.*, "Eye retention after proton beam radiotherapy for uveal melanoma," *Int. J. Radiation Oncol. Biol. Phys.*, vol. 55, no. 4, pp. 867–880, 2003.
[2] A. Ramasubramanian and C. Shields, *Retinoblastoma*. New Delhi, India: Jaypee Brothers, Medical Publishers, 2012.
[3] M. Goitein and T. Miller, "Planning proton therapy of the eye," *Med. Phys.*, vol. 10, no. 3, pp. 275–283, 1983.
[4] D. Shin *et al.*, "Eye tracking and gating system for proton therapy of orbital tumors," *Med. Phys.*, vol. 39, no. 7, pp. 4265–4273, 2012.
[5] M. B. Rüegsegger *et al.*, "Noninvasive referencing of intraocular tumors for external beam radiation therapy using optical coherence tomography: A proof of concept," *Med. Phys.*, vol. 41, no. 8, pp. 081704-1–081704-9, 2014.
[6] A. Fassi *et al.*, "Optical eye tracking system for noninvasive and automatic monitoring of eye position and movements in radiotherapy treatments of ocular tumors," *Appl. Opt.*, vol. 51, no. 13, pp. 2441–2450, May 2012.
[7] R. Via *et al.*, "Optical eye tracking system for real-time noninvasive tumor localization in external beam radiotherapy," *Med. Phys.*, vol. 42, no. 5, pp. 2194–2202, 2015.
[8] M. B. Rüegsegger *et al.*, "Statistical modeling of the eye for multimodal treatment planning for external beam radiation therapy of intraocular tumors," *Int. J. Radiat. Oncol. Biol. Phys.*, vol. 84, no. 4, pp. e541–e547, 2012.
[9] S. De Zanet *et al.*, "Landmark detection for fusion of fundus and MRI toward a patient-specific multimodal eye model," *IEEE Trans. Biomed. Eng.*, vol. 62, no. 2, pp. 532–540, Feb. 2015.
[10] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 11, pp. 1330–1334, Nov. 2000.
[11] J. Salvi *et al.*, "A comparative review of camera calibrating methods with accuracy evaluation," *Pattern Recognit.*, vol. 35, no. 7, pp. 1617–1635, 2002.
[12] G. Bradski, "The opencv library," *Doctor Dobbs J.*, vol. 25, no. 11, pp. 120–126, 2000.
[13] D. Hansen and Q. Ji, "In the eye of the beholder: A survey of models for eyes and gaze," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 3, pp. 478–500, Mar. 2010.
[14] E. Guestrin and E. Eizenman, "General theory of remote gaze estimation using the pupil center and corneal reflections," *IEEE Trans. Biomed. Eng.*, vol. 53, no. 6, pp. 1124–1133, Jun. 2006.
[15] W. Lotmar, "Theoretical eye model with aspherics," *J. Opt. Soc. Amer.*, vol. 61, no. 11, pp. 1522–1529, Nov. 1971.

Stephan Wyder was born in Bern, Switzerland, in 1982. He received the B.Sc. degree in computer science from the Bern University of Applied Sciences, Bern, Switzerland, in 2008, and the M.Sc. degree in biomedical engineering from the University of Bern, Bern, Switzerland, in 2012.

Furthermore, he has more than five years of work experience in the ophthalmic industry, where he worked on the development and integration of gaze tracking systems. He joined the Faculty of Medicine, Basel University in 2013 to work on the research project *Computer Assisted Proton Beam Radiotherapy for Intraocular Tumors*, funded by Swiss National Science Foundation, where he is responsible for the subproject *Gaze Tracking Based Intraocular Tumor Targeting for Proton Beam Radiotherapy*.

Fabian Hennings photograph and biography not available at the time of publication.

Simon Pezold was born in Ludwigsburg, Germany, in 1985. He received the *Diplom* degree (corresponding to a combined BS/MS) in medical informatics from the University of Heidelberg, Heidelberg, Germany, in 2010. He is currently pursuing the Ph.D. degree in biomedical engineering at the University of Basel, Basel, Switzerland.

His research interests include medical image segmentation and medical applications of computer-assisted navigation.

Jan Hrbacek photograph and biography not available at the time of publication.

Philippe C. Cattin was born in Switzerland in 1967. He received the B.Sc. degree from the University of Applied Science in Brugg/Windisch in 1991. In 1995 he received the M.Sc. degree in computer science and in 2003 the Ph.D. degree in robotics from ETH Zürich, Switzerland. From 2003 to 2007 he was a Postdoctoral Fellow with the Computer Vision Laboratory at ETH Zürich. In 2007 he became an Assistant Professor at the University of Basel and was promoted to Associate Professor in 2015. He is the founder of the Medical Image Analysis Center at the Medical Faculty of the University of Basel. He is currently the head of the recently founded Department of Biomedical Engineering at the University of Basel.

His research interests include medical image analysis, image-guided therapy and robotics-guided laser osteotomy. As a Principal Investigator he has finished many projects in these areas and published over 100 papers, patents and book chapter.

4 Improved Eye Tracking System

When we developed the first eye tracking prototype presented in Chapter 3, we mainly focused on compact hardware. This is because the available physical space in the treatment facility is very limited and therefore we only used one camera, since a stereo camera frame would have required a lot more space. Using only one camera enabled us to build a compact device, however, this setup had also a drawback: The mathematical eye tracking model was non-linear and had to be optimized for every point of gaze estimate. Additionally, the user calibration was quite complex and time consuming. When some parameters of the hardware calibration were not very accurately set, sudden instabilities occurred during the optimization for the point of gaze estimation.

The eye tracking model is much more stable for a stereo camera setup. This is why we were looking for a compact construction of a stereo camera frame. Figure 4.1 illustrates the idea behind our newly proposed setup. Instead of including a second space consuming camera to build a stereo setup, we just added a second mirror. This results in a setup with two virtual cameras, one part of the camera sensor represents the first virtual camera, the other part of the sensor the second virtual camera.
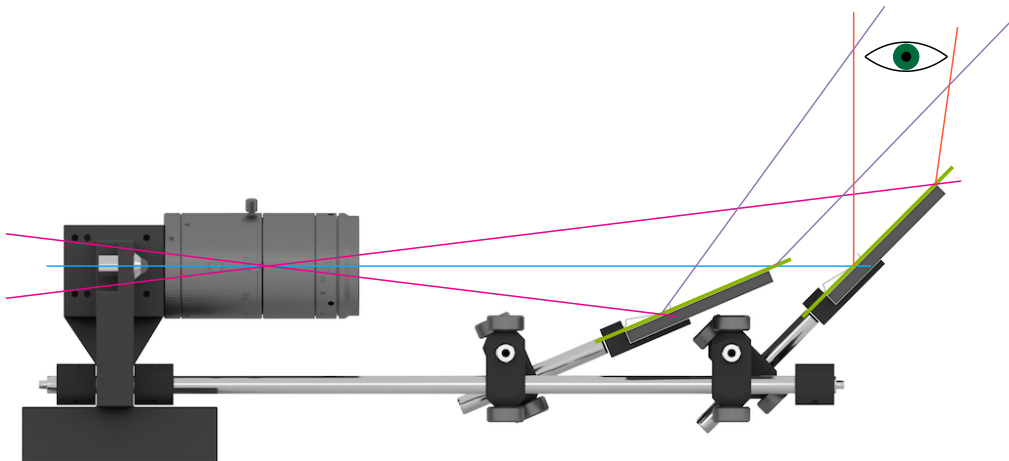

Figure 4.1: Stereo eye tracker with mirrors

Publication

The following publication was presented in 2016 in Athens at the *3rd MICCAI Workshop on Ophthalmic Medical Image Analysis, (OMIA)*:

S. Wyder and P. C. Cattin, "Stereo Eye Tracking with a Single Camera for Ocular Tumor Therapy," in *Proceedings of the Ophthalmic Medical Image Analysis International Workshop*, 2016, pp. 81–88

Stereo Eye Tracking with a Single Camera for Ocular Tumor Therapy

Stephan Wyder and Philippe C. Cattin

University of Basel, Department of Biomedical Engineering, Basel, Switzerland
{stephan.wyder,philippe.cattin}@unibas.ch

Abstract. We present a compact and accurate stereo eye tracking system using only one physical camera. The proposed eye tracking system is intended as a navigation system for ocular tumor therapy. There, the available physical space to mount an eye tracker is limited. Furthermore, high system accuracy is demanded. However, high eye tracker accuracy and system compactness often disagree. Current established eye trackers can live with that compromise, desktop devices focus more on accuracy whereas mobile devices focus on compactness. We combine a stereo eye tracking algorithm with a clever arrangement of two planar mirrors and a single camera to get high accuracy, precision and a compact design altogether. We developed an eye tracking prototype and tested the system with ten healthy volunteers. We show that the proposed eye tracker is more accurate and robust, while at the same time equally compact as a comparable eye tracking system containing one instead of two mirrors.

1 Introduction

Ocular tumors are a severe disease that may lead to blindness or even death if left untreated. Nowadays, specialists successfully treat the disease by radiating the patient's primary tumor with charged particles [3]. There is, however, a drawback: Although the tumor radiation itself is noninvasive, an invasive patient preparation is required. A surgeon thereby sutures radio-opaque clips on the outer scleral surface of the diseased eye. These clips are used to target the tumor during radiation therapy.

By introducing an eye tracker, also referred to as gaze trackers, into the current treatment workflow, clip surgery could be avoided [9, 10]. Eye trackers are devices able to estimate where a person is looking, i.e. the point of gaze [5], [7]. Certain eye trackers are based on a 3D model and therefore even have the ability to estimate the location of the eye in 3D space. This property enables us to use an eye tracker as a navigation system, namely to localize the eye and target the tumor during radiation therapy [9, 10].

The following eye tracker properties are important for this medical application: High accuracy and precision, computational speed and stability, and compact hardware design, enabling its integration into the radiation facility. However, eye tracker accuracy and compactness often disagree.

82   S. Wyder and P. C. Cattin

We present a new type of eye tracking system, extending an existing solution [10], where we are able to drastically increase accuracy and stability without having a significantly bigger device. The proposed eye tracker consists of one physical camera and we complement it with two planar mirrors. The integration into the treatment workflow remains unaffected and is done as described in [10].

By observing a scene (eye) over two mirrors, stereo images can be captured with a single camera (catadioptric stereo [2], [8]). This construction enables us to make the device compact and accurate at the same time. This is because we can optimally deflect different optical paths between the eye and the camera with the introduced mirrors. Furthermore, our point of gaze estimation is very accurate and the eye position estimation is very precise due to the virtual stereo camera frame (triangulation).

We tested our eye tracker with ten healthy volunteers and we show that our system is more accurate and reliable than a comparable eye tracking device [10], containing one instead of two planar mirrors. The proposed eye tracker is primarily designed and developed for ocular tumor therapy. However, we believe that this idea can easily be adapted for any other application, where compactness, high accuracy, and flexibility for the integration are demanded.

2   Methods

Hardware Setup and Calibration. The eye tracker (Fig. 1a) consists of an industry camera (XIMEA MQ022RG-CM), a 50 mm lens with an infrared-pass filter ($\lambda = 650$ nm), two infrared LEDs ($\lambda = 860$ nm), and two hot-mirrors (infrared reflection from 750 nm to 1125 nm). All components are mounted on rails with optomechanical holders. Using a mirror in general enables to optimally deflect the optical path between the eye and the camera. Therefore, we can place the camera where physical space is available (Fig. 1b). A hot-mirror, in our special case, is coated to reflect infrared waves and to transmit the visible part of the light-spectrum. It has the advantage that it does not obscure the view of the subject's eye, which might be only a couple of centimeters behind one of the hot-mirrors. Two infrared LEDs are used to illuminate the scene (eye) and to produce reflections on the cornea of the subject (glints). The LED positions are calibrated in advance and given by design.

To enable estimating the point of gaze and the eye position, we first need to know the absolute positions of the virtual cameras (nodal points / optical centers $\mathbf{o}_A$, $\mathbf{o}_B$) and the virtual LEDs ($\mathbf{l}_{A1}$, $\mathbf{l}_{A2}$, $\mathbf{l}_{B1}$, $\mathbf{l}_{B2}$). Therefore, the camera-mirror setup needs to be calibrated to get the intrinsic camera parameters.

The following steps describe the calibration procedure: (**i**) First, the operator arranges the hot-mirror positions and the tilting angles, in order to have a good camera view onto the target eye. (**ii**) The camera focus and aperture have to be adjusted properly. (**iii**) The operator then calibrates the camera once for a certain focus/aperture adjustment to get the intrinsic camera parameters [6], [11]. (**iv**) At this stage, all images coming from the physical camera $\mathbf{o}_j$ get undistorted to correct for lens errors. (**v**) Additionally, all images get flipped
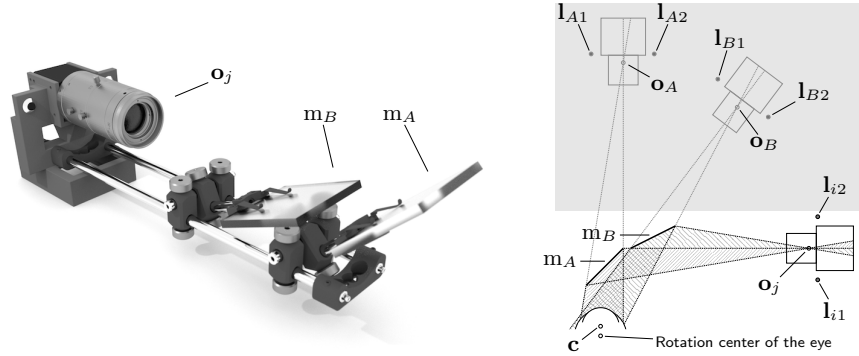
(a) Eye tracker consisting of an industry camera, a lens with an infrared-pass filter, two infrared LEDs, and two hot-mirrors (m_A,m_B). All parts are mounted with either a plastic holder or optomechanical components on rigid rails. The rails also serve as fixation point, such that the tracker can be attached to the treatment apparatus.

(b) Eye nodal point ($\mathbf{c}$), camera ($\mathbf{o}_j$), LEDs ($\mathbf{l}_{i1}$,$\mathbf{l}_{i2}$), hot-mirrors, virtual cameras ($\mathbf{o}_A$,$\mathbf{o}_B$), and virtual LEDs ($\mathbf{l}_{A1}$,$\mathbf{l}_{A2}$,$\mathbf{l}_{B1}$,$\mathbf{l}_{B2}$). The gray shaded box corresponds to physical space occupied by the treatment device.

Fig. 1: Eye tracking hardware (left) and optical arrangement (right)

horizontally in order to have image sections as they would have been made by the virtual cameras behind the mirrors [1]. The two mirrors are placed such, that after the flipping, the left part of the image looks like it would have been taken from $\mathbf{o}_B$, and the right half of the image, as it would have been taken from $\mathbf{o}_A$ (Fig. 1b). (**vi**) To get the absolute positions of the virtual cameras, the operator acquires the appropriate extrinsic camera matrices (homographies [6], [11]) using an image of a checkerboard. The checkerboard itself is co-registered with the world coordinate system of the treatment device. It has approximately the size of an eye and is placed at the same location. (**vii**) Having the two homographies H_A and H_B, the positions of the virtual LEDs ($\mathbf{l}_{A1}$, $\mathbf{l}_{A2}$, $\mathbf{l}_{B1}$. $\mathbf{l}_{B2}$) and the virtual cameras ($\mathbf{o}_A$, $\mathbf{o}_B$) can be determined.

Semi-Supervised Eye Feature Detection. Figure 2 shows a typical image (input) with the overlaid detected features (output) and the appropriate labels after undistortion and horizontal flipping.

To get all required eye features, (**i**) the operator sets two regions of interest (ROI_A, ROI_B) for the current eye tracking session. (**ii**) A couple of seed points have to be set for both ROIs within the pupil area, the iris area and the glints. (**iii**) The algorithm averages the individual seed point sets and defines from that an individual pupil- and glint-threshold for both ROIs: The threshold for the glints is defined by the average of the glint seed point values plus a certain tolerance ($\pm\frac{3}{256}$). The threshold for the pupil consists of the arithmetic mean between
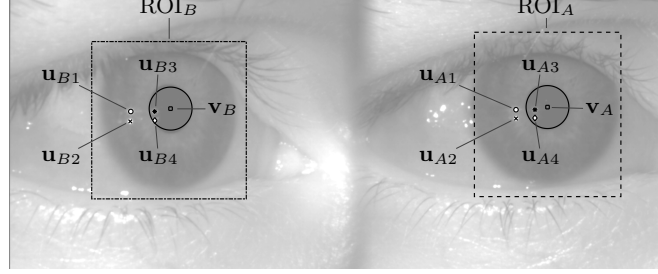
84   S. Wyder and P. C. Cattin

Fig. 2: Virtual stereo camera view onto the same eye with ROI_A, ROI_B, pupil centers ($\mathbf{v}_A$, $\mathbf{v}_B$), and glints ($\mathbf{u}_{B1}$, $\mathbf{u}_{B2}$, $\mathbf{u}_{B3}$, $\mathbf{u}_{B4}$, $\mathbf{u}_{A1}$, $\mathbf{u}_{A2}$, $\mathbf{u}_{A3}$, $\mathbf{u}_{A4}$)

the average of the iris seed points and the average of the pupil seed points, again plus the mentioned tolerance. Furthermore, (**iv**) the algorithm thresholds the image with the previously set threshold parameters. This results in four binary images, one with glint candidates and one with pupil candidates, both for ROI_A and ROI_B. After this, (**v**) a standard 8-connected-component labeling algorithm gets applied on all of the binary images to identify the individual blobs. (**vi**) All the pupil blobs get post-processed with a morphological closing operator to make their border regions more homogeneous. Next, (**vii**) the algorithm extracts the size and the eccentricity (measure of roundness of a certain area) for every blob. (**viii**) All pupil blobs that have an eccentricity above 0.75 are discarded (circle: 0, ellipse: < 1). (**ix**) From the remaining blobs, the algorithm takes the biggest one, builds the convex hull and calculates the centroid of it (pupil center). (**x**) To get the required glint centers, the algorithm extracts the centroids of the glint candidates. The four glints closest to the pupil center are taken and sorted such, that we have them for both ROIs in the order: top-left, bottom-left, top-right, bottom-right. Having this, the algorithm assigns the appropriate labels to the centroids of the glints (Fig. 2). (**xi**) All the coordinates of the extracted features (2D projections) get transformed to the camera coordinate system (3D points) [10]. Afterwards they get transformed with the appropriate homographies H_A and H_B from the camera coordinate system into the world coordinate system, depending on whether they were detected in ROI_A or in ROI_B.

Eye Position and Point of Gaze Estimation. The gaze tracking model is based on the method from Guestrin et al. [4], adapted for the catadioptric setup. At this stage of the eye tracking procedure, we have a couple of points in 3D space. Some of them are determined during the hardware calibration, some points were gathered during the eye feature detection. Using these known points we can calculate the eye position and the point of gaze.

All points and vectors are denoted with small bold letters and are $\in \mathbb{R}_3$. Known points are: $\mathbf{l}_i$ (light sources), $\mathbf{o}_j$ (nodal points of cameras), $\mathbf{u}_{ij}$ (images of glints on sensor), $\mathbf{v}_j$ (images of pupil on sensor), where i encodes the light sources and j encodes the cameras (Fig. 1b and Fig. 2).

First, the algorithm estimates the nodal point of the eye $\mathbf{c}$ by bringing some of the known points into relation. Coplanarity of points $\mathbf{l}_i$, $\mathbf{o}_j$, $\mathbf{u}_{ij}$, $\mathbf{c}$ can be described with the triple product:

$$\underbrace{(\mathbf{l}_i - \mathbf{o}_j) \times (\mathbf{u}_{ij} - \mathbf{o}_j)}_{\mathbf{w}_{ij}} \bullet (\mathbf{c} - \mathbf{o}_j) = 0 \quad \Leftrightarrow \quad \mathbf{w}_{ij} \bullet (\mathbf{c} - \mathbf{o}_j) = 0 \,. \tag{1}$$

Making use of the distributive property, we get:

$$\mathbf{w}_{ij} \bullet (\mathbf{c} - \mathbf{o}_j) = 0 \quad \Leftrightarrow \quad \mathbf{w}_{ij} \bullet \mathbf{c} - \mathbf{w}_{ij} \bullet \mathbf{o}_j = 0 \,. \tag{2}$$

Because $a \bullet b = a^T \cdot b$, we can write the above equation in matrix form:

$$\mathbf{w}_{ij}^T \cdot \mathbf{c} = \mathbf{w}_{ij} \bullet \mathbf{o}_j \,, \tag{3}$$

$$\underbrace{\begin{bmatrix} [(\mathbf{l}_{A1} - \mathbf{o}_A) \times (\mathbf{u}_{A4} - \mathbf{o}_A)]^T \\ [(\mathbf{l}_{A1} - \mathbf{o}_B) \times (\mathbf{u}_{B4} - \mathbf{o}_B)]^T \\ \vdots \\ [(\mathbf{l}_{B2} - \mathbf{o}_A) \times (\mathbf{u}_{A1} - \mathbf{o}_A)]^T \\ [(\mathbf{l}_{B2} - \mathbf{o}_B) \times (\mathbf{u}_{B2} - \mathbf{o}_B)]^T \end{bmatrix}}_{\mathrm{M}_2} \cdot \mathbf{c} = \underbrace{\begin{bmatrix} (\mathbf{l}_{A1} - \mathbf{o}_A) \times (\mathbf{u}_{A4} - \mathbf{o}_A) \bullet \mathbf{o_A} \\ (\mathbf{l}_{A1} - \mathbf{o}_B) \times (\mathbf{u}_{B4} - \mathbf{o}_B) \bullet \mathbf{o_B} \\ \vdots \\ (\mathbf{l}_{B2} - \mathbf{o}_A) \times (\mathbf{u}_{A1} - \mathbf{o}_A) \bullet \mathbf{o_A} \\ (\mathbf{l}_{B2} - \mathbf{o}_B) \times (\mathbf{u}_{B2} - \mathbf{o}_B) \bullet \mathbf{o_B} \end{bmatrix}}_{\mathbf{h}} \,. \tag{4}$$

Every row in the above system of equations represents one correspondence between a virtual light source and an image of a glint on one of the virtual camera sensors. This overdetermined system of linear equations can be solved with least squares:

$$\mathrm{M}_2 \cdot \mathbf{c} = \mathbf{h} \quad \Rightarrow \quad \mathbf{c} = (\mathrm{M}_2^{\mathrm{T}} \mathrm{M}_2)^{-1} \cdot \mathrm{M}_2^{\mathrm{T}} \mathbf{h} \,. \tag{5}$$

Having the point $\mathbf{c}$ (nodal point of the eye), the algorithm calculates $\overrightarrow{\mathbf{cp}}$, the geometrical axis of the eye, defined by the points $\mathbf{c}$ and $\mathbf{p}$ (pupil center). The geometrical axis of the eye $\overrightarrow{\mathbf{cp}}$ can also be seen as the line of intersection of two planes, defined by $\mathbf{o}_A$, $\mathbf{v}_A$, $\mathbf{c}$ and $\mathbf{o}_B$, $\mathbf{v}_B$, $\mathbf{c}$:

$$\overrightarrow{\mathbf{cp}} = [(\mathbf{o}_A - \mathbf{v}_A) \times (\mathbf{c} - \mathbf{o}_A)] \times [(\mathbf{o}_B - \mathbf{v}_B) \times (\mathbf{c} - \mathbf{o}_B)] \,, \tag{6}$$

$$\mathbf{s} := \overrightarrow{\mathbf{cp}} / \|\overrightarrow{\mathbf{cp}}\| \,. \tag{7}$$

The geometrical axis is only plausible (and accurate), when the mentioned planes have different orientation. Otherwise their normalized normals ($\hat{\mathbf{n}}_1$, $\hat{\mathbf{n}}_2$) are parallel or almost parallel and no (accurate) intersection can be calculated. We detect this special case by calculating the norm of the difference vector between $\hat{\mathbf{n}}_1$ and $\hat{\mathbf{n}}_2$. When this mentioned norm is below a certain tolerance (0.2), the result is marked as implausible (inaccurate).

The geometrical axis can also be expressed with a tilt and a shift angle (φ, θ), relative to the world coordinate system:

$$\varphi = \sin^{-1}(-\mathbf{s}_y), \quad \theta = \sin^{-1}(-\mathbf{s}_x / \cos(\varphi)) \,. \tag{8}$$

The visual axis of the eye (line-of-sight) is constructed by connecting the nodal point of the eye with the fovea, the point of sharpest vision on the retina. The subject-specific deviation between the geometrical axis and the visual axis can again be expressed with two angles α and β, for which the algorithm takes tabulated standard values as a first estimate ($\alpha = \pm 5°$, $\beta = 1.5°$). Consequently, the point of gaze is defined by the intersection of the visual axis with a certain plane in space, a computer screen or any other plane containing at least one calibration point. Assuming that the plane of intersection corresponds to the xy-plane ($z = 0$), the point of gaze is:

$$k = \mathbf{c}_z / \left[\cos(\varphi + \beta) \cdot \cos(\theta + \alpha) \right], \tag{9}$$

$$\textit{Point of gaze} = \mathbf{c} - k \cdot \begin{bmatrix} \cos(\varphi + \beta) \cdot \sin(\theta + \alpha) \\ \sin(\varphi + \beta) \\ \cos(\varphi + \beta) \cdot \cos(\theta + \alpha) \end{bmatrix} . \tag{10}$$

Having at least one calibration point with well known coordinates, α and β can be calculated, when comparing the true calibration point position with the estimated point of gaze position (calibration of subject-specific parameters) [4].

3   Results

We mounted the eye tracker on an optical table, equipped with 15 calibration points at well defined positions. Additionally, the optical table contained a calibration checkerboard, which defined the world coordinate system and was used to determine the absolute positions of the virtual cameras and the virtual LEDs. Our experimental setup is derived from the treatment facility and enables direct comparison. Having the system calibrated, which takes a few minutes, we let ten healthy volunteers fixate the predefined calibration points using an ophthalmic chin rest. The corresponding images were recorded with the camera. Afterwards, we calculated the point of gaze for every image i and decided whether the estimate is plausible or not with the criterion mentioned above. On average, we discarded five points per volunteer. If plausibility was given, we recorded on the one side the calculated α and β values and the deviation between the estimated point of gaze and the true calibration point location (point of gaze error). The recorded α_i and β_i were averaged per volunteer and used for a second evaluation with the new subject-specific parameters ($\overline{\alpha_i}$, $\overline{\beta_i}$). The resulting point of gaze errors of the second (calibrated) round of evaluation were transformed from millimeters to degrees (measured at point $\mathbf{c}$) and visualized in Fig. 3. This conversion makes the result independent of a specific geometrical setup. Consequently, it enables us to compare our result with the results from [10] and any other eye tracker accuracy. The average error over all volunteers measured at point $\mathbf{c}$ is below $0.96°$. Beside the good point of gaze accuracy, we observed a high precision in point $\mathbf{c}$ estimation. We looked at the bounding boxes containing all points $\mathbf{c}$ per volunteer (one estimation per calibration point). The mean bounding box over all volunteers had the dimensions: $\Delta x = 2.04\,\mathrm{mm}$, $\Delta y = 1.72\,\mathrm{mm}$,
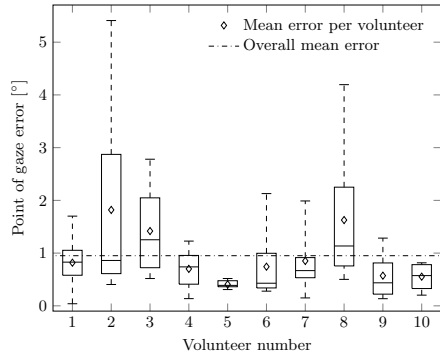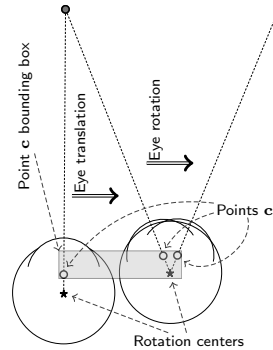
Fig. 3: Point of gaze error

Fig. 4: Eye translation & rotation

and $\Delta z = 1.52\,\text{mm}$. A displacement of point $\mathbf{c}$ in the xy-plane of about $0.8\,\text{mm}$ has to be expected, even when the eye is not translated but only rotated. This is because the nodal point $\mathbf{c}$ does not correspond to the rotation center of the eye (Fig. 4). Hence, one portion of the mentioned Δ-values is assumed to originate from slight eye translation.

The required time to process one image was on average $100\,\text{ms}$ for the feature extraction and below $1\,\text{ms}$ for calculating the eye position and the point of gaze. Our algorithm is implemented in MATLAB so far, intended for post-processing of already recorded images.

4   Discussion

The achieved point of gaze accuracy is more than $1°$ better as compared to [10], where only one instead of two mirrors was used. The precision of the eye position estimation is very high, and especially better in the depth compared to the reference system [10]. This can be explained by the triangulation angle, which is mainly responsible for the depth information and which is wider in the proposed solution due to the virtual stereo frame. Additionally, the proposed algorithm is faster and more robust, because no optimization of nonlinear systems is required. Furthermore, the patient specific calibration can be achieved with one calibration point and does not require a time-consuming procedure.

By using a setup with two hot-mirrors, we combine the benefits of a single camera system (only one camera to calibrate, no camera synchronization needed) with the advantages of a stereo camera system (simplified eye tracking model, fewer patient specific parameters, simplified calibration algorithm, better accuracy and precision).

The only limitation of our method, as mentioned above, is the fact that problematic geometrical constellations can occur when the geometrical axis $\overrightarrow{\mathbf{cp}}$ points towards the line connecting both virtual cameras (implausible result). This problematic constellation can be limited or even completely avoided by

88   S. Wyder and P. C. Cattin

tilting the individual mirrors or the whole eye tracker such, that the mentioned line connecting the virtual cameras does not cross the area, where high accuracy point of gaze estimation is required. This requirement is easy to fulfill in ocular tumor therapy and therefore no limitation.

5   Conclusion

Ocular tumor therapy can considerably be improved by integrating an eye tracking system. The whole tumor therapy can potentially be made noninvasive [10]. For this, we developed a novel stereo eye tracker with a single physical camera. Stereo eye tracking is more accurate and stable than an eye tracker based on a single camera. Our setup with two mirrors and one camera has several advantages as compared to a setup with two physical cameras: It is more compact, camera synchronization is not needed, and only one camera has to be calibrated. Our results show that we are much more accurate than with a conventional single camera setup. Therefore, our proposed eye tracker is eminently suited for ocular tumor navigation and other applications where both compactness and accuracy are needed.

References

1. Corballis, M.C.: Much ado about mirrors. Psychonomic Bulletin & Review 7(1), 163–169 (2000)
2. Gluckman, J., Nayar, S.K.: Catadioptric Stereo Using Planar Mirrors. International Journal of Computer Vision 44(1), 65–79 (2001)
3. Goitein, M., Miller, T.: Planning proton therapy of the eye. Medical Physics 10(3), 275–283 (May 1983)
4. Guestrin, E.D., Eizenman, M.: General theory of remote gaze estimation using the pupil center and corneal reflections. Biomedical Engineering, IEEE Transactions on 53(6), 1124–1133 (Jun 2006)
5. Hansen, D.W., Ji, Q.: In the Eye of the Beholder: A Survey of Models for Eyes and Gaze. Pattern Analysis and Machine Intelligence, IEEE Transactions on 32(3), 478–500 (Mar 2010)
6. Hartley, R., Zisserman, A.: Multiple view geometry in computer vision (2003)
7. Narcizo, F.B., Rangel de Queiroz, J.E., Gomes, H.M.: Remote Eye Tracking Systems: Technologies and Applications. In: 2013 26th Conference on Graphics, Patterns and Images Tutorials. pp. 15–22. IEEE (2013)
8. Nene, Sameer A, Nayar, Shree K: Stereo with mirrors. In: Computer Vision, 1998. Sixth International Conference on. pp. 1087–1094. IEEE (1998)
9. Via, R., Fassi, A., Fattori, G., Fontana, G., Pella, A., Tagaste, B., Riboldi, M., Ciocca, M., Orecchia, R., Baroni, G.: Optical eye tracking system for real-time noninvasive tumor localization in external beam radiotherapy. Medical Physics 42(5), 2194–2202 (May 2015)
10. Wyder, S., Hennings, F., Pezold, S., Hrbacek, J., Cattin, P.: With Gaze Tracking Towards Noninvasive Eye Cancer Treatment. Biomedical Engineering, IEEE Transactions on PP(99), 1–1 (2015)
11. Zhang, Z.: A flexible new technique for camera calibration. Pattern Analysis and Machine Intelligence, IEEE Transactions on 22(11), 1330–1334 (Nov 2000)

5 Eye Tracking System Evaluation

The stereo eye tracker presented in Chapter 4 is noticeably improved over the first eye tracking prototype in terms of algorithm stability and depth accuracy. Thanks to the special mirror arrangement the stereo eye tracker is not significantly bigger than the first single camera eye tracker.

One open question, however, persisted. So far, we were only able to make assumptions for the accuracy of the eye center estimate. This is because we evaluated the eye trackers in the conventional way with volunteers, and like this we did not have any ground truth data for the eye location.

To evaluate the eye center estimate of an eye tracker, we present a method with a movable and rotatable eye phantom and a corresponding kinematic model, which provides us with ground truth data. Figure 5.1 illustrates the eye tracker mounted on an optical table together with the testing stage.
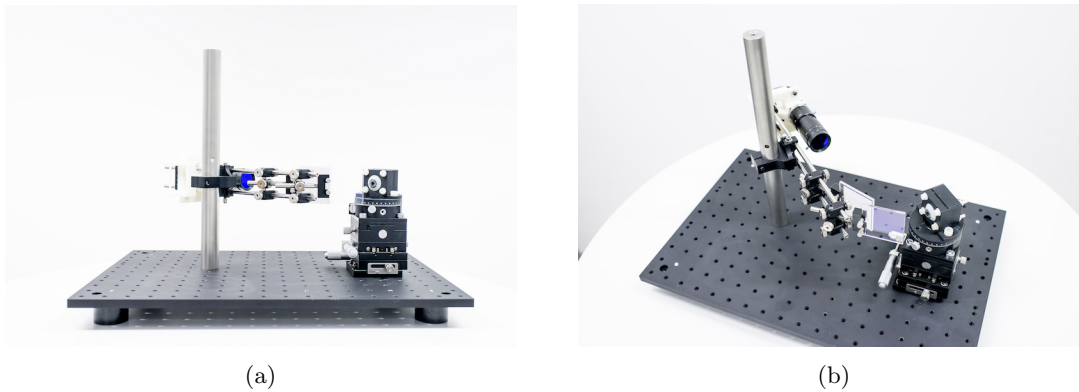

(a)   (b)

Figure 5.1: Eye tracker and testing stage on optical bench

Publication

The following publication was submitted to *Medical Physics, The International Journal of Medical Physics Research and Practice*:

S. Wyder and P. C. Cattin, "Eye Tracker Accuracy: Quantitative Evaluation of the Invisible Eye Center Location ," submitted

Eye Tracker Accuracy:
Quantitative Evaluation of the Invisible Eye Center Location

Stephan Wyder[*] and Philippe C. Cattin[†]

Department of Biomedical Engineering, University of Basel, Allschwil, Switzerland

(Dated: February 1, 2017)

Purpose. We present a new method to evaluate the accuracy of an eye tracker based eye localization system. Measuring the accuracy of an eye tracker's primary intention, the estimated point of gaze, is usually done with volunteers and a set of fixation points used as ground truth. However, verifying the accuracy of the location estimate of a volunteer's eye center in 3D space is not easily possible. This is because the eye center is an intangible point hidden by the iris.

Methods. We evaluate the eye location accuracy by using an eye phantom instead of eyes of volunteers. For this, we developed a testing stage with a realistic artificial eye and a corresponding kinematic model, which we trained with μCT data. This enables us to precisely evaluate the eye location estimate of an eye tracker.

Results. We show that the proposed testing stage with the corresponding kinematic model is suitable for such a validation. Further, we evaluate a particular eye tracker based navigation system and show that this system is able to successfully determine the eye center with sub-millimeter accuracy.

Conclusions. We show the suitability of the evaluated eye tracker for eye interventions, using the proposed testing stage and the corresponding kinematic model. The results further enable specific enhancement of the navigation system to potentially get even better results.

I. INTRODUCTION

Eye tracking devices, also known as eye- or gaze trackers are used to monitor eye movement. An eye tracker is usually used to determine a person's point of gaze. In market research, for instance, a wearable, video based eye tracking system can be used to uncover which product on which shelf is attracted by a test person. Certainly, there exist other constructions of eye trackers (e.g. desktop or embedded devices) and many other eye tracking applications (e.g. in usability testing or in automotive industry) [1, 2]. Different physical principles might be behind an eye tracker, depending on the application [3]. Video based eye trackers are the most widely used devices, because of their simplicity and the wide applicability.

In recent research, eye trackers are also used in navigation systems for computer assisted eye interventions [4–6]. In these cases, the eye tracker is used to estimate the 3D-location of the patient's eye, that is the eye center and orientation. We define the eye center as the center of corneal curvature. This can be useful to align an eye for an ophthalmic examination or treatment. Furthermore, the point of gaze, estimated by the eye tracker, is automatically monitored to interrupt an examination or treatment in case of sudden eye motion.

Using an eye tracker for medical interventions demands high system accuracy. This may decide between success or failure of an intervention because of the close proximity of critical structures within the eye. For instance, an eye localization accuracy below 1 mm is required, when an eye tracker is used to target intraocular tumors.

---

[*] stephan.wyder@unibas.ch

[†] philippe.cattin@unibas.ch

The demand for accurate eye tracking systems also raises the need for reliable accuracy measurement methods. Accuracy measurements are crucial for the development of an eye tracking system and also for the performance specification of the device.

Conventionally, eye tracker accuracy is evaluated with volunteers, who have to focus on certain fixation points located at well-known positions. The accuracy is then given by the deviations between the true fixation point locations and the point of gaze estimates of the eye tracker. As straightforward as this evaluation can be performed on the one hand, as difficult it is to see what parts of the system contribute to a certain error on the other hand. Testing this way does not enable us validating the accuracy of an intermediate product of the eye tracking pipeline, as for instance the eye center location. Furthermore, this validation method obviously depends on the cooperation of the volunteers. Hence, measuring the accuracy with an eye phantom seems to be the ideal complement for a thorough eye tracker evaluation.

Already Via et al. [4] used an eye phantom to asses the accuracy of an eye tracking system. However, details about the exact procedure remain partially unclear. Furthermore, it is not clear how realistic their eye phantom is. Also Świrski and Dodgson recognized the lack of a comprehensive evaluation method to test and improve the individual parts of an eye tracking system. They propose completely synthetic eye data [7] for accuracy and precision evaluation of eye tracking algorithms.

Compared to Via et al. [4], we build up our ground truth data using μCT-measurements to get highly accurate absolute eye center locations. In contrast to Świrski and Dodgson [7], we do not only evaluate the algorithm, but we validate the complete eye tracking system, including the whole optical path and the external referencing
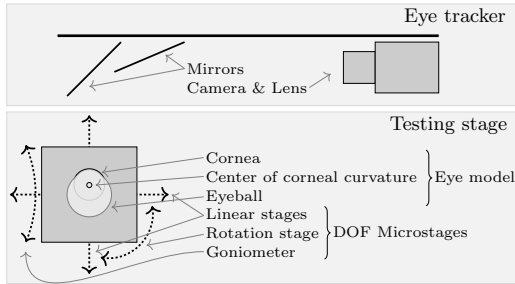
Figure 1: Eye tracker and testing stage (topdown view)

Figure 2: Typical eye model used by eye trackers

to a medical device. The evaluation of such an eye localization system involving eye tracker hardware and its environment cannot be done with rendered eye images. Neither can it be done with volunteer tests, because it is not possible to accurately measure the 3D location of a volunteers invisible eye center.

Accurate ground truth data is required for the accuracy evaluation of an eye localization system. We propose a procedure to fill this gap by providing accurate 3D-locations of the invisible and intangible eye center.

The basis is formed by a testing stage with four degrees of freedom (4 DOF), a mounted artificial glass eye, and an attached, black and white checkerboard pattern for external referencing. The testing stage enables us to move the whole eye forth and back and sidewards (by two linear stages). Additionally, the testing stage enables us to rotate the eye around two axes (by a rotation stage and a goniometer), in order to simulate an arbitrary line of sight. We built the testing stage and trained the parameters of its kinematic model with μCT-data (i.e. high precision 3D volumetric data) acquired of the testing stage in several different configurations (i.e. eye positions and orientations). The μCT-data provides us with accurate information about the location and the geometry of the eye and the checkerboard. Figure 1 illustrates the two involved parts, the eye tracker we want to evaluate and the proposed testing stage to accomplish the evaluation.

Having the testing stage ready and the kinematic model trained, we position and orient the artificial eye in known locations and compare this against the eye center locations predicted by the eye tracker. The eye center locations are given by the centers of corneal curvature (i.e. center of a cornea best fit sphere). The trained kinematic model provides us with exactly the same point in a common coordinate system (CS), which is also accessible by the eye tracker. This enables us to compare the eye location estimates of the eye tracker with the ground truth data, given by the testing stage model.

With the proposed testing stage, it is possible to quantitatively evaluate the performance of any 3D model based eye tracker. Using this method, we show that a particular eye tracking system [6] estimates the eye cen-
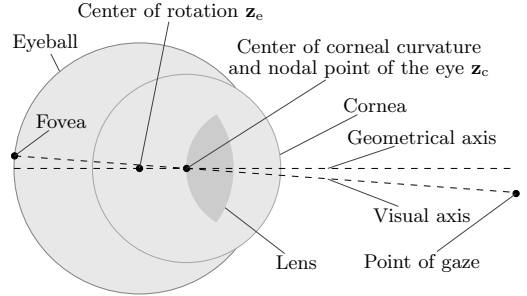
ter location with sub-millimeter accuracy.

We describe in the following sections our proposed method for the accuracy evaluation and the results achieved when testing a particular eye tracker [6].

II.   METHODS

We propose a custom-built hardware testing stage and an appropriate kinematic model with its calibration, to evaluate the accuracy of the center location of the corneal curvature, estimated by an eye tracker. This section consists of three parts. First, we give an insight into a typical eye tracking model based on 3D ray tracing. Second, we present the testing stage hardware with its components. The testing stage hardware enables us to position and orient the embedded artificial eye such that the eye tracker can perform its intended measurements. The testing stage hardware basically replaces the testing volunteer, with the advantage of having the exact position of the eye (i.e. ground truth). Third, we present the corresponding testing stage model, which we parametrize, train and validate with μCT data. The kinematic model enables us to determine the exact glass eye position in every possible testing stage configuration with sub-millimeter accuracy.

Consequently, this enables us to test an eye tracker on the artificial eye prothesis with several different eye positions and orientations. The 3D eye location estimate of the eye tracker can then be compared to the ground truth data of the testing stage model, which is configured according the status of the testing stage.

A.   Eye Tracker Model

The complexity of existing eye tracking models vary. The eye is often modeled with two spheres. Figure 2 illustrates such a typical eye model. One sphere represents the eyeball, its center consequently corresponds to the rotation center of the eye. The second sphere, the sphere cap respectively, represents the cornea. The center of the corneal curvature corresponds to the nodal point of
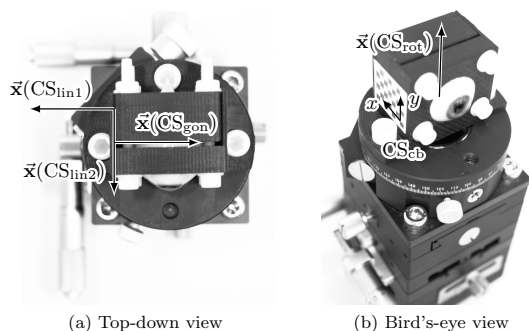
(a) Top-down view (b) Bird's-eye view

Figure 3: Testing stage with the glass eye, its holder with the checkerboard and the microstages

Figure 4: Concept of testing stage: Comparison of the eye center location $\mathbf{z}_c^\star$ estimated by the eye tracker with the ground truth $\mathbf{z}_c$

the eye, where the optical rays cross, before they hit the retina [8].

The two sphere centers define the geometrical axis of the eye. Hence, the orientation of an eye in space can be determined by the geometrical axis. The location of the eye (i.e. eye center) is given by the center of the corneal curvature, which lies on the mentioned, geometrical axis and is an integral part of most of the 3D model based eye trackers.

The fovea (point of sharpest vision) is located on the retina (backside of the eyeball) but is not in line with the geometrical axis. A point we focus on with our eye gets imaged on the fovea. That is why also the visual axis plays an important role in such a model. The visual axis connects the fovea with the nodal point of the eye and the point of gaze. The angle between visual axis and geometrical axis has to be calibrated per patient.

The eye tracker [5, 6] which we test with the proposed testing stage is based on the model of E. D. Guestrin and M. Eizenman [8].

B. Testing Stage Hardware

The testing stage we developed consists of a translation stage with two axes with parameters P_1 and P_2 (*OptoSigma TADC-652WS25-M6*), a goniometer stage with parameter P_3 (*OptoSigma GOH-65A50-M6*), and a rotation stage with parameter P_4 (*OptoSigma KSW-656-M6*). The variables P_1, P_2, P_3, and P_4 represent the values, which are set for the corresponding microstages. The linear stages have a vernier scale included enabling to measure with a precision of $10\,\mu$m. The rotation stage and goniometer also contain a vernier scale enabling us to measure with a precision of angular minutes. To simulate the human eye, we use a handcrafted eye prosthesis made from glass by the *Swiss Institute For Artificial Eyes, Lucerne, Switzerland*. To interface the artificial eye with the stages we designed a rigid and robust eye holder. Since the eye prosthesis does a priori not have an
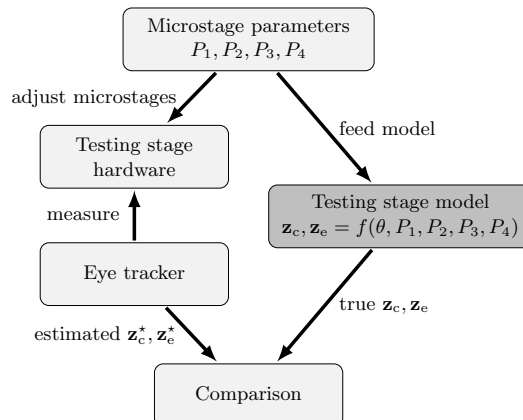
exactly known geometry, we made a 3D scan of it with a μCT device (*GE phoenix nanotom m*). We segmented the surface of the eye with *Fiji*, an image processing package [9]. We afterwards used *Blender*, an open source 3D creation suite (http://www.blender.org), to design a holder accurately interfacing the eye with the stages. The holder additionally contains a black and white checkerboard stuck on its side. The checkerboard is printed with an off-the-shelf laser printer, which contains toner visible in the μCT. The stages are serially mounted and on top of them is the eye holder, which was printed on a *Stratasys Fortus 250mc* 3D printer. The testing stage is shown in Figure 3.

C. Testing Stage Kinematic Model

The aim of the kinematic model is to determine the exact center location of the corneal curvature $\mathbf{z}_c$ for a certain testing stage configuration (P_1, P_2, P_3, P_4) and to transform the coordinates to a common coordinate system.

The internal model parameters θ, that have to be trained, basically consist of six right-handed coordinate systems (CS): CS_{vol} is the common CS for all μCT - volumes, CS_{lin1}, CS_{lin2}, CS_{gon}, and CS_{rot} correspond to their appropriate microstage and CS_{cb} is the checkerboard CS. CS_{vol} can be seen as the CS for model input data, whereas CS_{cb} is the CS for the output data. CS_{cb} is accessible by the eye tracker and the testing stage. Additionally, θ contains $\hat{\mathbf{z}}_c$ and $\hat{\mathbf{z}}_e$, the center locations and the radii of the cornea and the eyeball, yet unaffected by P_1, P_2, P_3, P_4 (neutral position).

Figure 4 illustrates the role of the testing stage model within our contribution.

The origins of the CSs and the corresponding orienta-

tions are defined based on the acquired μCT data. We adjust a few positions of each individual microstage and acquire a μCT volume for each configuration. This enables us to train the internal kinematic model parameters θ.
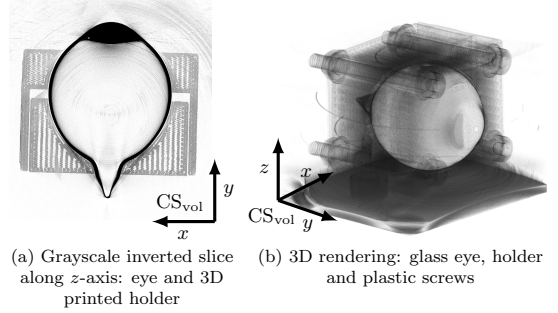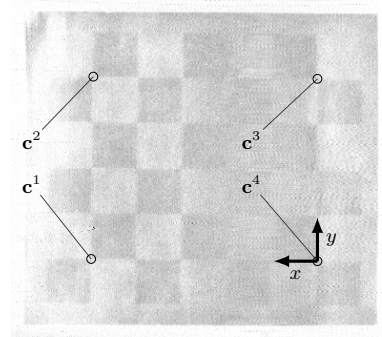
μCT Data Acuisition. As seen in Tab. I, we acquired 15 μCT-volumes, which help us to define the mentioned internal model parameters θ. Furthermore, we used some μCT measurements to test the integrity of our kinematic model. The table shows the number (identifier) of the measurement (#), the state of the individual microstages during a certain scan and the type of the measurement ($\star$).

Table I: μCT-data acquisition plan

	Stages				
	P_1	P_2	P_3	P_4	
#	linear 1	linear 2	gonio.	rotation	$\star$
1	0 mm	0 mm	0°	0°	1,3
2	−7.5 mm	0 mm	0°	0°	3
3	7.5 mm	0 mm	0°	0°	5
4	0 mm	0 mm	0°	0°	1,3
5	0 mm	−7.5 mm	0°	0°	3
6	0 mm	7.5 mm	0°	0°	5
7	0 mm	0 mm	0°	0°	1,4
8	0 mm	0 mm	−15°	0°	4
9	0 mm	0 mm	8°	0°	5
10	0 mm	0 mm	15°	0°	4
11	0 mm	0 mm	0°	0°	2,4
12	0 mm	0 mm	0°	−30°	4
13	0 mm	0 mm	0°	15°	5
14	0 mm	0 mm	0°	30°	4
15	0 mm	0 mm	0°	0°	2

$\star_1$ corresponds to training scans, where the microstages are in neutral position. For testing, we use $\star_2$ scans, which also correspond to neutral position scans. $\star_3$ scans are used to train the linear stages. $\star_4$ scans are used to train the rotation and goniometer stages. $\star_5$ scans are used to test the kinematic model accuracy of the individual degrees of freedom.

To acquire the required data, we use the *GE phoenix nanotom m* μCT device. In order to get a good contrast for the glass eye surface as well as for the checkerboard pattern in the acquired μCT data, we set the voltage to 50 kV and the current to 310 μA. To limit the required overall acquisition time for the 15 scans, we used a so called *fast scan mode*, for which the specimen in the nanotom rotates continuously 360° during a defined time (in our case 20 min). These settings result in 1599 projections (3072 px $\times$ 2400 px), exposed with 750 ms each. The isotropic voxels have the side length of 25 μm. The resulting reconstructions (3D volumes) of the projections are cropped to the content of importance and have the

(a) Grayscale inverted slice along z-axis: eye and 3D printed holder

(b) 3D rendering: glass eye, holder and plastic screws

Figure 5: Visualized μCT data (CS_{vol}) acquired with *GE phoenix nanotom m*

Figure 6: Checkerboard corners ($\mathbf{c}^k$) and CS_{cb} as seen in the μCT data

size of 2100 px $\times$ 1900 px $\times$ 1700 px. Additionally, we reduce the grayscale depth from 16 bit to 8 bit by linearly mapping the grayscale-values between 23'000 and 35'000 to the range between 0 and 255, such that both, the eye surfaces as well as the checkerboards are well visible. The whole process of reducing the volume dimensions and the grayscale depth is mainly required to reduce the amount of data for further processing. The size of one final volume is still 6.8 GB.

Figure 5 illustrates the data acquired with the μCT. Figure 5a shows one slice perpendicular to the z-axis and Figure 5b shows a volume rendering of a μCT scan. Both figures illustrate also the location and orientation of the CS_{vol}.

In order to be able to train our kinematic model with the acquired data, we first need to segment the required features.

μCT Data Segemention. We extract two different types of features from the acquired volumes, four checkerboard corners ($\mathbf{c}^k$, where $k \in \{1, 2, 3, 4\}$), as they are visible in Figure 6, and the surface of the glass eye (the black contour visible in Figure 5a).

To train the kinematic model we need to have the cor-

ner point coordinates as they are visualized in Figure 6 for all 15 data volumes. We extract the coordinates of $\mathbf{c}^k$ by hand using *Fiji's* "Big Data Viewer". This plugin enables to visualize a slice with an arbitrary orientation and to show the 3D coordinates of a given voxel. This results in $15 * 4$ 3D coordinates in CS_{vol} coordinate system.

The following procedure describes the extraction of the glass eye for all volumes in neutral configuration ($\star 1$ and $\star 2$, see Tab. I). We process the volumes (thresholding and surface extraction) again by using *Fiji* [9]. The edge of the eye is segmented by applying a threshold of 115, which is an experimentally found value. Afterwards we extract the surface from the segmented eye using the marching cubes method (using the "3D Viewer" plugin). The surface mesh can be exported as STL file directly with this plugin. This results in a mesh basically consisting of an outer and an inner surface of the glass eye along with some unwanted holes and additional artifacts.

To clean up the geometry we import the mesh into *Blender*. Within *Blender* we first create several objects by separating the imported mesh by loose parts. All but the biggest part (the eye) can be deleted. To save later processing time, we apply a mesh decimation. We extract the cornea and the eyeball separately to individually fit a sphere afterwards. The cleaned cornea- and eyeball-mesh are exported again as STL for all 5 mentioned volumes.

After μCT data acquisition and segmentation we end up with four 3D coordinates each ($\mathbf{c}^k$, $k \in \{1, 2, 3, 4\}$) for all 15 volumes. In addition we have an extracted cornea and an eyeball mesh for five of the 15 volumes (where $\star 1$ and $\star 2$).

Kinematic Model Calibration. All data used as input (checkerboard corner points, cornea mesh, eyeball mesh) to train the internal model parameters θ are in the right-handed CS_{vol} coordinate system and are given in voxel. We also express the other coordinate systems relative to CS_{vol}.

Let $\mathbf{c}_j^k \in \mathbb{R}^3$ be a 3D vector in CS_{vol} representing a checkerboard corner point, where $k \in \{1, 2, 3, 4\}$ encodes the checkerboard corner point number and $j \in \{1, 2, 3, ..., 15\}$ encodes the number of the measurement (#).

Let G_p^k be a group of $\mathbf{c}_j^k$, where $k \in \{1, 2, 3, 4\}$ encodes the checkerboard corner point number and $p \in \{1, 2, 3, 4, 5\}$ encodes the type ($\star$) of the scan group (Tab. I).

A coordinate system is defined using four position vectors expressed in CS_{vol}. The first column vector represents the origin $\vec{\mathbf{o}}$ of the corresponding CS expressed in CS_{vol}. The remaining three column vectors represent the positions where the unit vectors (basis vectors) of the corresponding CS point to:

$$\text{CS} = \underbrace{\begin{pmatrix} o_x & x_x & y_x & z_x \\ o_y & x_y & y_y & z_y \\ o_z & x_z & y_z & z_z \\ 1 & 1 & 1 & 1 \end{pmatrix}}_{\text{Homogeneous coordinates in } \text{CS}_{\text{vol}}}.$$

Usually a CS is represented with a rigid 4×4-transformation matrix (isometry) consisting of a rotation and a translation. Our slightly different CS definition has the advantage, that the unit vectors can directly be extracted after a transformation is applied to the CS.

First, we define CS_{lin1} and CS_{lin2}, which represent the linear stage 1 and 2, the two stages at the bottom of the microstage stack. The origins $\vec{\mathbf{o}}$ of CS_{lin1} and CS_{lin2} are given by the median ($\tilde{}$) of three corner points, where $k = 1$. These three corner points come from volumes, where the stages were in neutral position during the scan ($\star 1$ volumes):

$$\vec{\mathbf{o}}(\text{CS}_{\text{lin1}}) = \vec{\mathbf{o}}(\text{CS}_{\text{lin2}}) = \widetilde{G_1^1}.$$

The x-axes of CS_{lin1} and CS_{lin2} are pointing in the positive direction of the corresponding translational axis of the appropriate microstage. They are defined using the median ($\tilde{}$) of all four translation vectors

$$\vec{\mathbf{x}}(\text{CS}_{\text{lin1}}) = \vec{\mathbf{o}}(\text{CS}_{\text{lin1}}) + \frac{\vec{\mathbf{x}_1}}{\|\vec{\mathbf{x}_1}\|},$$

where $\vec{\mathbf{x}_1} = \widetilde{\{\mathbf{c}_1^k - \mathbf{c}_2^k | k \in \{1, 2, 3, 4\}\}}$ and

$$\vec{\mathbf{x}}(\text{CS}_{\text{lin2}}) = \vec{\mathbf{o}}(\text{CS}_{\text{lin2}}) + \frac{\vec{\mathbf{x}_2}}{\|\vec{\mathbf{x}_2}\|},$$

where $\vec{\mathbf{x}_2} = \widetilde{\{\mathbf{c}_4^k - \mathbf{c}_5^k | k \in \{1, 2, 3, 4\}\}}$.

The y-axis $\vec{\mathbf{y}}$ and z-axis $\vec{\mathbf{z}}$ of both systems are defined in an arbitrary way using the cross product, such that we get well defined right handed CSs with orthogonal axes. Particular orientations of $\vec{\mathbf{y}}$ and $\vec{\mathbf{z}}$ are not important, since we use these two CSs only for translation along the x-axis.

Second, we define CS_{gon} and CS_{rot}, which represent the goniometer and the rotation stages, the two topmost stages of the microstage stack. The origins $\vec{\mathbf{o}}$ of CS_{gon} and CS_{rot} are given by best fit circle centers. Because all checkerboard corners k of the particular measurements lie in a plane perpendicular to the rotation axes of the stages, we take the median of the found circle centers. To find the appropriate circle centers we fit for all four corner points k a circle using three measurements per fit. The best fit circle-function (BFC) [10] returns the center of the fitted circle:

$$\vec{\mathbf{o}}(\text{CS}_{\text{gon}}) = \widetilde{\{\text{BFC}(\mathbf{c}_7^k, \mathbf{c}_8^k, \mathbf{c}_{10}^k) | k \in \{1, 2, 3, 4\}\}},$$

$$\vec{\mathbf{o}}(\text{CS}_{\text{rot}}) = \widetilde{\{\text{BFC}(\mathbf{c}_{11}^k, \mathbf{c}_{12}^k, \mathbf{c}_{14}^k) | k \in \{1, 2, 3, 4\}\}}.$$

To define the x-axes (rotation axes) of the two topmost stages of the microstage stack, we take the normal vector perpendicular to the plane given by the appropriate corner points:

$$\vec{\mathbf{x}}(\text{CS}_{\text{gon}}) = \widetilde{\{(\mathbf{c}_7^k - \mathbf{c}_8^k) \times (\mathbf{c}_7^k - \mathbf{c}_{10}^k) | k \in \{1, 2, 3, 4\}\}},$$

$$\vec{\mathbf{x}}(\mathrm{CS_{rot}}) = \overline{\{(\mathbf{c}_{11}^k - \mathbf{c}_{12}^k) \times (\mathbf{c}_{11}^k - \mathbf{c}_{14}^k) | k \in \{1,2,3,4\}\}},$$

where $\times$ denotes the cross-product. The y-axis $\vec{\mathbf{y}}$ and z-axis $\vec{\mathbf{z}}$ of both systems are again defined in an arbitrary way using the cross product, such that we get well defined right handed CSs with orthogonal axes.

Third, we determine the center of the cornea best fit sphere, as well as the center of the eyeball best fit sphere based on the prepared mesh from measurement #1. To do so, we use the segmented and cleaned meshes and we fit a sphere in a least-square-sense [10]. We first rearrange the general equation of a sphere,

$$(x_i - x_0)^2 + (y_i - y_0)^2 + (z_i - z_0)^2 = r^2,$$

such that we can write the expression in matrix notation and solve for the unknowns x_0, y_0, z_0, and r, which represent the center coordinates and the radius of the sphere. The variables x_i, y_i, and z_i are the coordinates of any point lying on the surface of the particular sphere. This results in two vectors $\mathbf{z}_c$ for the cornea center and $\mathbf{z}_e$ for the eyeball center containing the best fit sphere center coordinates and the appropriate radius.

Figure 7 illustrates $\mathbf{z}_c$, $\mathbf{z}_e$, and the vertices of the mesh (gray dots) with the corresponding best fit spheres (BFS). The visualized checkerboard corners ($\mathbf{c}^k$) represent the median of the corners, where the stages are in neutral position($\star_1$ containing #1, #4, and #7).

The kinematic model is at this stage characterized such that we have defined four CSs corresponding to a microstage each and the centers and radii of the cornea and the eyeball. All these position vectors are expressed in $\mathrm{CS_{vol}}$. In order to get the true position of the sphere centers (cornea or eyeball), we just have to translate $\mathbf{z}_c$ or $\mathbf{z}_e$ along the x-axis of linear stage CSs or rotate around the x-axis of the goniometer or rotation stage according to what is adjusted at the testing stage hardware (i.e. the microstages).

Using the Kinematic Model. The trained testing stage model takes four parameters (P_1, P_2, P_3, P_4). These are the four individual microstage position settings which are set on the testing stage hardware while the eye tracker estimates the cornea center for the corresponding eye position. P_1 and P_2 are in millimeters (mm). P_3 and P_4 are in angular degrees (°). Processing these parameters, the trained kinematic model is able to return (expressed in the common $\mathrm{CS_{cb}}$) the position of the cornea center. This position acts as ground truth for the eye tracker validation (Figure 4). If we are adjusting a certain microstage position (e.g. $P_1 = +6\,\mathrm{mm}$ on the linear stage 1), then this affects not only the position of $\mathbf{z}_c$ and $\mathbf{z}_e$, but also the microstages (their CSs, respectively) above the microstage which gets adjusted. The microstage stack is as follows (from bottom to top): $\mathrm{CS_{lin1}}$, $\mathrm{CS_{lin2}}$, $\mathrm{CS_{gon}}$, and $\mathrm{CS_{rot}}$. And on top of the stack is the eye with $\mathbf{z}_c$ and $\mathbf{z}_e$.

The workflow is as follows:

1. Hardware adjustment of a microstage a ($a \in \{\mathrm{lin1, lin2, gon, rot}\}$)

2. Basis change from $\mathrm{CS_{vol}}$ to the corresponding CS_a of all remaining CSs, which are above the current CS_a in the stack

3. Basis change to CS_a of the sphere centers ($\mathbf{z}_c$ and $\mathbf{z}_e$)

4. Application of the transformation matrix T_a (e.g. rotation of $+3°$) to all the remaining CSs and the sphere centers

5. Basis change of the CSs and the sphere centers back to $\mathrm{CS_{vol}}$

The workflow is repeated for all microstages (for all four parameters, respectively) beginning with the lowest one.

The individual rigid transformations T_a, which are applied on the corresponding local CS look as follows (translation along or rotation around x-axis):

$$\mathrm{T_{lin1}} = \begin{bmatrix} 1 & 0 & 0 & P_1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \mathrm{T_{lin2}} = \begin{bmatrix} 1 & 0 & 0 & P_2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$\mathrm{T_{gon}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(P_3) & -\sin(P_3) & 0 \\ 0 & \sin(P_3) & \cos(P_3) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$\mathrm{T_{rot}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(P_4) & -\sin(P_4) & 0 \\ 0 & \sin(P_4) & \cos(P_4) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

The rigid transformations ${}^a\mathrm{T_{vol}}$ to change the basis from $\mathrm{CS_{vol}}$ to CS_a and back are defined as follows. For this, we use a method based on singular value decomposition (SVD), which is robust in terms of noise [11]. The method returns a rigid transformation ${}^a\mathrm{T_{vol}}$ (rotation and translation) when passing CS_a-matrix (expressed in $\mathrm{CS_{vol}}$) and the $\mathrm{CS_{vol}}$-matrix (expressen in $\mathrm{CS_{vol}}$):

$$\mathrm{CS_{vol}} = \underbrace{\begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix}}_{\text{Homogeneous coordinates in } \mathrm{CS_{vol}}}.$$

Having ${}^a\mathrm{T_{vol}}$, we change the basis of the remaining CSs (CSs above the current one in the microstage stack), $\mathbf{z}_c$, and $\mathbf{z}_e$. Afterwards, we apply the transformation T_a and change the basis back to $\mathrm{CS_{vol}}$ for all CSs b, which are above CS_a:

$$\mathrm{CS}_b' = ({}^a\mathrm{T_{vol}})^{-1} \cdot (\mathrm{T}_a \cdot ({}^a\mathrm{T_{vol}} \cdot \mathrm{CS}_b)).$$

where a represents the CS, which we adjust (e.g. $\mathrm{CS_{lin1}}$). The sphere centers are adjusted as well for each parameter P_1, P_2, P_3, P_4:

$$\mathbf{z}_e' = ({}^a\mathrm{T_{vol}})^{-1} \cdot (\mathrm{T}_a \cdot ({}^a\mathrm{T_{vol}} \cdot \mathbf{z}_e)),$$

(a) Top-down view

(b) Lateral view
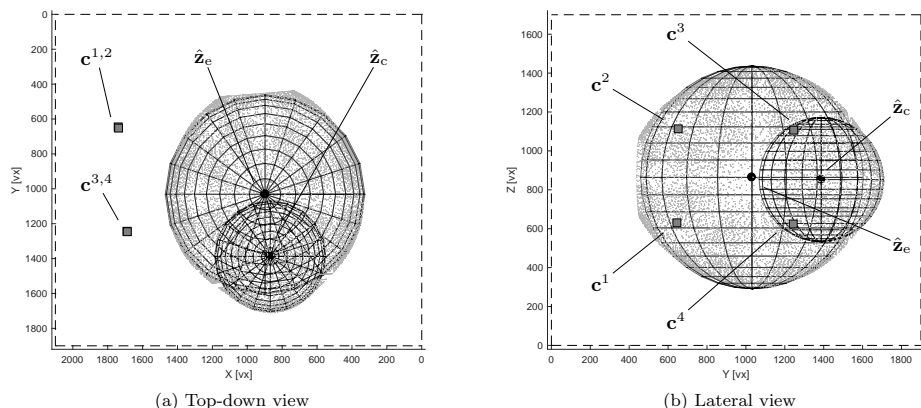
Figure 7: Testing stage model visualization

$$\mathbf{z}'_c = (\ ^{a}\mathrm{T}_{\mathrm{vol}}\ )^{-1} \cdot (\ \mathrm{T}_a \cdot (\ ^{a}\mathrm{T}_{\mathrm{vol}} \cdot \mathbf{z}_c\ )).$$

Step-by-step, we apply all transformations for a certain testing stage configuration, until we have the position $\mathbf{z}_c$ and $\mathbf{z}_e$ for the current microstage configuration expressed in CS_{vol}. The last step is to change the basis of the sphere centers from CS_{vol} to CS_{cb}, our common CS.

For the eye tracker tests, the tracker is rigidly mounted to a certain position, such that the checkerboard pattern (also attached to the eye holder) is completely visible by the eye tracker camera. For the external referencing of the eye tracker (here with the testing stage) we perform a homography estimation [12] based on a checkerboard pattern [5, 6]. This enables the eye tracker to express its guess about the sphere centers in CS_{cb}. We configure the testing stage (adjusting linear, rotation, and goniometer stages) such that the visibility of the checkerboard pattern from the eye tracker is well (sharp and complete pattern). This particular stage configuration enables us to access CS_{cb} from our kinematic model. The origin lies on the corner point 4, the x-axis points towards corner point 1 and the y-axis points towards corner point 3 (Figure 6). This CS_{cb} definition holds for both the eye tracker and the testing stage model.

The workflow described above is applied again at the very end to transform the sphere centers to CS_{cb} according to the microstage configuration (P_1, P_2, P_3, P_4) at the time of external referencing.

III. EXPERIMENTS

A. Kinematic Model consistency

To make sure that we trained our testing stage model sufficiently accurate, we used the μCT measurements of type $\star 5$ and $\star 2$ (see Tab. I) to validate the integrity of the trained x-axes of the individual CSs. We used

the median checkerboard corner points of the measurements $\star 2$ ($\{\widetilde{G_2^1}, \widetilde{G_2^2}, \widetilde{G_2^3}, \widetilde{G_2^4}\}$) to predict with our testing stage model the new checkerboard corner locations under four certain configurations. We used one configuration (P_1, P_2, P_3, P_4) for each DOF. For this, we took the four different configuration sets from the measurements $\star 5$. Having the new checkerboard corner locations calculated, we compared the model estimates (based on measurements $\star 2$) with the checkerboard corners, which we extracted manually (measurements $\star 5$). The mean error (corner-reprojection-error) of the four $\star 5$-measurements times four checkerboard corners (16 points) was 31 µm.

Additionally, we analyzed the angles between the x-axes of the trained coordinate systems (CS_{lin1}, CS_{lin2}, CS_{gon}, and CS_{rot}). Assuming the microstages are ideally mounted and aligned on top of each other, we would have to expect angles of $90\,°$ between the x-axes. We found out that we have a $89.5\,°$ angle between the linear stages, $90.9\,°$ between the linear stage 2 and the goniometer rotation axis and $90.2\,°$ between the rotation axes of the goniometer and the rotation stage.

We also performed cornea-fit-refit experiments, where we fitted a new sphere to all of the scans $\star 5$. The mean deviation between the five sphere centers was $\pm\,36$ µm.

B. Eye tracker accuracy

Setup. We tested a video based stereo eye tracker [6] with the proposed testing stage hardware and the corresponding kinematic model. For this, we rigidly mounted both the testing stage and the eye tracker on an optical bench and aligned the eye tracker such that a good visibility on to the artificial eye of the testing stage was given. We adjusted the focus and the aperture of the lens (part of the eye tracker) and performed a camera calibration [12] to get the intrinsic camera pa-

rameters (focal length, distortions). Having the camera calibrated, we adjusted the testing stage such that the holder's checkerboard was visible by the eye tracker ($P_1 = +8\,\text{mm}$, $P_2 = +7\,\text{mm}$, $P_3 = 8\,°$, $P_4 = +56\,°$). With the eye tracker we performed a homography estimation (based on an image snapshot of the checkerboard) in order to be able to transform the eye tracker output, the center of the corneal curvature, to the common checkerboard coordinate system CS_{cb} [5]. The camera calibration and the referencing to an external system (testing stage or a medical device) is part of the eye tracker calibration procedure.

For the actual validation, we set 20 different eye positions and orientations with the testing stage to mimic snapshots of a natural eye movement. To get a better impression of the results we only adjusted one microstage at the same time, while the three other stages were in neutral position. The microstages were set to $P1 = \{7.5, 10, 12.5, 15, 17.5\}[\text{mm}]$, then $P2 = \{7.5, 10, 12.5, 15, 17.5\}[\text{mm}]$, $P3 = \{-10, -5, 0, 5, 10\}[°]$, and $P4 = \{290, 298, 307, 316, 324\}[°]$. This resulted in five positions per microstage and with that in 20 eye tracker estimates of the corneal curvature location $\mathbf{z}_c^\star$. We set the same parameters on our kinematic model and generated the ground truth of the center location of the corneal curvature. Figure 4 illustrates this workflow.

Results. We compared the 20 different center locations of corneal curvature from the eye tracker with the ground truth data from the testing stage. The mean deviation between two 3D points, the accuracy a respectively is as follows: The mean accuracy $\mu(a) = 0.68\,\text{mm}$, the median accuracy $\widetilde{a} = 0.67\,\text{mm}$. Subdivided into the individual orientation components: The mean accuracy $\mu(a_x) = 0.32\,\text{mm}$, the median accuracy $\widetilde{a_x} = 0.33\,\text{mm}$. The mean accuracy $\mu(a_y) = -0.09\,\text{mm}$, the median accuracy $\widetilde{a_y} = -0.09\,\text{mm}$. The mean accuracy $\mu(a_z) = -0.54\,\text{mm}$, the median accuracy $\widetilde{a_z} = -0.55\,\text{mm}$. Figure 8 and Figure 9 illustrate the distribution of the error.

Thanks to the proposed method we were able to analyze the nature of the error and unveil a slight bias of a yet unknown source. For this, we removed the average error vector from our eye tracker estimates and compared the result again with the ground truth, then we got a mean relative error $\mu(a_{\text{rel}}) = 0.32\,\text{mm}$. By eliminating this error, the overall eye tracker accuracy can even be increased.

We also evaluated the accuracy of the eye orientation. For this, we calculated the geometrical axes for the eye tracker estimate by using the pupil center and the center of corneal curvature $\mathbf{z}_c^\star$ and for the kinematic model by using the centers of both spheres $\mathbf{z}_c$, $\mathbf{z}_e$. In theory, all four points lie on the geometrical axis, however, it is not the case for our eye phantom. That is why we calculated the relative angle between the geometrical axes from one measurement to the next and then we compared these relative angles between the ground truth and the eye tracker estimates. The mean relative angle error is $0.50°$, which indicates high angular precision.
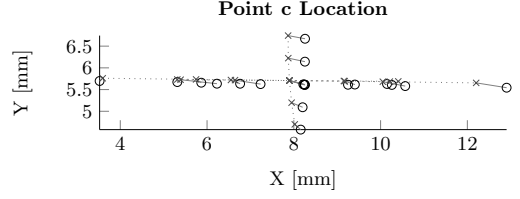

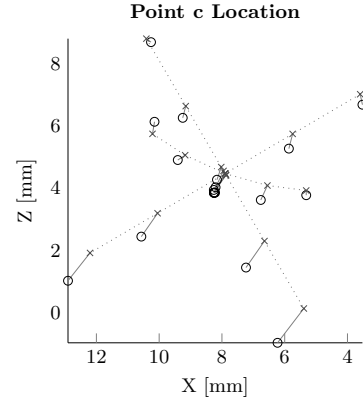Figure 8: Accuracy in the X/Y plane (o = eye tracker estimate, x = ground truth, ... = DOF)

Figure 9: Accuracy in the X/Z plane (o = eye tracker estimate, x = ground truth, ... = DOF)

IV. DISCUSSION

We were able to successfully validate the eye tracker of interest with our testing stage hardware and the corresponding kinematic model. The tests showed that the eye tracker can determine the eye location (center of corneal curvature) with an accuracy below $0.7\,\text{mm}$. The accuracy of the validated navigation system for proton radiotherapy hence fulfills the requirements of sub-millimeter accuracy. The mean relative error $\mu(a_{\text{rel}})$ is smaller by roughly a factor of two compared to the mean error $\mu(a)$, which is a strong indication for high precision but also for a slight bias of a yet unknown source. Our system helped to detect and quantify this bias.

Figure 8 and Figure 9 show this slight systematic error along the longitudinal axis of the eye tracker.

It is difficult to compare the results to any other similar validation method, because to our best knowledge, no one did so far such a comprehensive validation of the eye location accuracy. Having for instance a closer look at [4], it is not clear how exactly the ground truth was generated.

A. Testing stage hardware and kinematic model

The systematic error from the eye tracking tests may be explained by an imprecise cornea best fit sphere. We prepared the cornea mesh in a way, where we only had limited influence on the vertex distribution. Fitting a sphere with another method than with a least-square method might be more accurate.

Maybe the most important error source is the manual segmentation of the checkerboard corner points. To improve this, we suggest exchanging the checkerboard pattern, which is used on the one hand for the external referencing of the eye tracker (homography) and on the other hand to train and validate the whole testing stage model. Hence, the pattern, its segmentation respectively, is central for the validation. A better pattern might be dots in a certain arrangement (similar to the squares in the checkerboard pattern). This pattern could easily be segmented automatically, by choosing the center of mass of the circles or the ellipsoids, respectively, taking the thickness of the ink into account.

B. Eye tracker

Depending on the application different levels of accuracy are required. Our achieved sub-millimeter accuracy in determining the eye location is sufficient for our medical application with especially high demands. If there should be higher demands, the detailed validation results, for instance the distribution of the error, might provide helpful information for eye tracker improvement.

V. CONCLUSION

Using an eye tracker to localize the eye in space can potentially improve today's eye interventions. For instance, when treating eye tumors with protons, our non-invasive eye tracker based solution might some day replace the state-of-the-art invasive navigation method.

We proposed a quantitative evaluation method with which we showed that our eye tracker is able to fulfill the requirements, namely, to determine the location of the eye with sub-millimeter accuracy. Our proposed evaluation method does not replace the eye tracker tests with volunteers that are used nowadays, but it complements the validation, enabling new eye tracking applications: eye localization.

We are sure, that in the future more and more applications, especially in ophthalmology, will benefit from an eye localization system.

ACKNOWLEDGMENTS

[1] Narcizo Fabricio Batista, Queiroz Jose Eustaquio, Gomes Herman Martins. Remote Eye Tracking Systems: Technologies and Applications in *2013 26th Conference on Graphics, Patterns and Images Tutorials*:15–22IEEE 2013.

[2] Hansen Dan Witzner, Ji Qiang. In the Eye of the Beholder: A Survey of Models for Eyes and Gaze *Pattern Analysis and Machine Intelligence, IEEE Transactions on.* 2010;32:478–500.

[3] Duchowski Andrew T. *Eye tracking methodology - theory and practice (2. ed.).* London: Springer London 2007.

[4] Via Riccardo, Fassi Aurora, Fattori Giovanni, et al. Optical eye tracking system for real-time noninvasive tumor localization in external beam radiotherapy *Medical Physics.* 2015;42:2194–2202.

[5] Wyder Stephan, Hennings Fabian, Pezold Simon, Hrbacek Jan, Cattin Philippe C. With Gaze Tracking Toward Noninvasive Eye Cancer Treatment *Biomedical Engineering, IEEE Transactions on.* 2016;63:1914–1924.

[6] Wyder Stephan, Cattin Philippe C. Stereo Eye Tracking with a Single Camera for Ocular Tumor Therapy in *Proceedings of the Ophthalmic Medical Image Analysis International Workshop*:81–88 2016.

[7] Świrski Lech, Dodgson Neil. Rendering synthetic ground truth images for eye tracker evaluation in *Proceedings of the 2014 Symposium on Eye-Tracking Research & Applications*(New York, New York, USA):219–222ACM 2014.

[8] Guestrin Elias Daniel, Eizenman Moshe. General theory of remote gaze estimation using the pupil center and corneal reflections *Biomedical Engineering, IEEE Transactions on.* 2006;53:1124–1133.

[9] Schindelin Johannes, Arganda-Carreras Ignacio, Frise Erwin, et al. Fiji: an open-source platform for biological-image analysis *Nature Methods.* 2012;9:676–682.

[10] Leon Steven J. Linear algebra with applications Macmillan New York 1980.

[11] Besl P J, McKay H D. A method for registration of 3-D shapes *IEEE Transactions on Pattern Analysis and Machine Intelligence.* 1992;14:239–256.

[12] Zhang Zhengyou. A flexible new technique for camera calibration *Pattern Analysis and Machine Intelligence, IEEE Transactions on.* 2000;22:1330–1334.

6 Discussion and Conclusion

Eye melanoma is slightly more common in men than in women.

(cancer.org)

It was our goal to find a possibility to replace today's invasive navigation method for the treatment of eye tumors with protons. For this we developed several eye tracking prototypes and we integrated them into the proton treatment facility of the PSI, the so-called OPTIS2. We made classical eye tracking accuracy tests with healthy volunteers. The corresponding publications can be found in Chapters 3 and 4. Furthermore, we developed a method to evaluate the location estimate of a volunteer's eye center. This evaluation was not possible with classical volunteer tests, because the eye center is an intangible point hidden by the iris. For this, we used an eye phantom on microstages and a corresponding kinematic model. To get the required accuracy for our ground truth data, we calibrated the kinematic model with data from a μCT-system. With our eye tracker we achieve a sub-millimeter accuracy when estimating the 3D eye location in space. The corresponding publication can be found in Chapter 5.

Discussion

Our investigations presented in this thesis led to many new insights, most importantly:

- Today's 3D eye tracking models are surprisingly accurate in terms of both point of gaze estimation and eye localization. This has not necessarily to be expected, particularly because a real corneal surface sometimes is quite far away from the modeled spherical shape.

- The whole system can only be as good as the weakest link in the chain. All the requirements for an accurate eye localization system can be fulfilled with: carefully chosen hardware, a suitable 3D eye tracking model, and an accurate homography estimate for the co-registration with an external system. The link in the chain with most potential for improvement might be a reliable and accurate feature detection.

- The proposed eye localization system is designed and customized for the proton therapy of eye tumors. However, other ophthalmic examination or intervention devices might benefit as well from such a system. Eye movement is in these cases almost always an issue and our eye tracking system could be the ideal tool to monitor and react on it.

Research Questions

> **Research question 1:** *Is it possible to replace today's invasive navigation system with a novel, non-invasive method?*

The next step in the project should be the side-by-side comparison of the new treatment scheme with the state-of-the-art method. It would be a huge success, if our new treatment scheme would reach the same accuracy as the state of the art. The advantages are obvious: The whole treatment would become non-invasive.

However, up to now, we can only make a limited prediction, since the whole pipeline is not yet ready for such a test.

> **Research question 2:** *Is the potential replacement method able to localize the eye in space with sub-millimeter accuracy?*

Our third publication provides the answer to this research question. We were able to show that the accuracy is below a millimeter and the system still has potential for improvement, mainly in terms of the feature detection. Consequently, the research question can clearly be answered with 'yes'.

Outlook

There is still some work to do until the eye localization system is market-ready. The main focus would have to be set on:

- The *feature detection*, the process of extracting the glints and the pupil center from an input image, can still be improved. The accuracy of the detected features makes up a big part of the overall system accuracy. Further would it be helpful to detect states, which make a reliable eye tracking impossible: eyelashes or eye lid covers eye, eye out of focus, no eye at all in the scene, glints found outside of spherical area of cornea, amongst others.

- The *eye tracking model* proposed in the second publication in Chapter 4 has the weakness, that a certain geometrical constellation can occur, which makes it difficult to determine an accurate point of gaze. The eye localization, however, is not affected by this, it is only the point of gaze estimation. This problem should anyway be solved.

- The *device calibration* of the stereo eye tracker with the two mirrors might be difficult for unexperienced operators. This is because the mirrors can be slid on the rails and they can be tilted. These many degrees of freedom obviously influence the common filed of view, which should be maximized. An additional degree of freedom is coming from the focus settings of the camera. An optimal setting for the mirror arrangement and ideal focus settings of the lens might exist and should be elaborated. This would enable us to build a case with fixated mirrors and consequently the system calibration could be performed much easier.

- The development of a *realtime software package* with a user friendly interface would finally enable us to create a product not only usable by engineers and researchers.

Conclusion

In general, we are sure that in the future more and more applications, especially in ophthalmology, will benefit from an eye localization system. In particular, we hope that our research and development leads to an ocular tumor treatment, which will be safer and more pleasant for patients suffering from this serious disease.

7 Bibliography

[1] University of Chicago Press Staff, *The Chicago Manual of Style.* University of Chicago Press, 2003.

[2] M. B. Rüegsegger, M. Bach Cuadra, A. Pica, C. A. Amstutz, T. Rudolph, D. Aebersold, and J. H. Kowal, "Statistical Modeling of the Eye for Multimodal Treatment Planning for External Beam Radiation Therapy of Intraocular Tumors," *International Journal of Radiation Oncology ● Biology ● Physics*, vol. 84, no. 4, pp. e541–e547, Nov. 2012.

[3] S. I. De Zanet, C. Ciller, T. Rudolph, P. Maeder, F. Munier, A. Balmer, M. B. Cuadra, and J. H. Kowal, "Landmark Detection for Fusion of Fundus and MRI Toward a Patient-Specific Multimodal Eye Model," *Biomedical Engineering, IEEE Transactions on*, vol. 62, no. 2, pp. 532–540, 2015.

[4] C. Ciller, S. I. De Zanet, M. B. Rüegsegger, A. Pica, R. Sznitman, J.-P. Thiran, P. Maeder, F. L. Munier, J. H. Kowal, and M. B. Cuadra, "Automatic Segmentation of the Eye in 3D Magnetic Resonance Imaging: A Novel Statistical Shape Model for Treatment Planning of Retinoblastoma," *International Journal of Radiation Oncology ● Biology ● Physics*, vol. 92, no. 4, pp. 794–802, Jul. 2015.

[5] S. Wyder, F. Hennings, S. Pezold, J. Hrbacek, and P. C. Cattin, "With Gaze Tracking Toward Noninvasive Eye Cancer Treatment," *Biomedical Engineering, IEEE Transactions on*, vol. 63, no. 9, pp. 1914–1924, 2016.

[6] S. Wyder and P. C. Cattin, "Stereo Eye Tracking with a Single Camera for Ocular Tumor Therapy," in *Proceedings of the Ophthalmic Medical Image Analysis International Workshop*, 2016, pp. 81–88.

[7] P. Jovanovic, M. Mihajlovic, and J. Djordjevic-Jocic, "Ocular melanoma: an overview of the current status," *International Journal of Clinical and Experimental Pathology*, vol. 6, no. 7, pp. 1230–1244, 2013.

[8] F. L. Munier, J. Verwey, A. Pica, A. Balmer, L. Zografos, H. Abouzeid, B. Timmerman, G. Goitein, and R. Moeckli, "New developments in external beam radiotherapy for retinoblastoma: from lens to normal tissue-sparing techniques," *Clinical and Experimental Ophthalmology*, vol. 36, no. 1, pp. 78–89, Jan. 2008.

[9] B. Damato, A. Kacperek, D. Errington, and H. Heimann, "Proton beam radiotherapy of uveal melanoma," *Saudi Journal of Ophthalmology*, vol. 27, no. 3, pp. 151–157, Jul. 2013.

[10] A. M. Allen, T. Pawlicki, L. Dong, E. Fourkal, M. Buyyounouski, K. Cengel, J. Plastaras, M. K. Bucci, T. I. Yock, L. Bonilla, R. Price, E. E. Harris, and A. A. Konski, "An evidence based review of proton beam therapy: The report of ASTRO's emerging technology committee," *Radiotherapy and Oncology*, vol. 103, no. 1, pp. 8–11, Apr. 2012.

7 Bibliography

[11] E. Egger, L. Zografos, A. Schalenbourg, D. Beati, T. Bhringer, L. Chamot, and G. Goitein, "Eye retention after proton beam radiotherapy for uveal melanoma," *International Journal of Radiation Oncology • Biology • Physics*, vol. 55, no. 4, pp. 867–880, Mar. 2003.

[12] M. Goitein and T. Miller, "Planning proton therapy of the eye," *Medical Physics*, vol. 10, no. 3, pp. 275–283, May 1983.

[13] D. W. Hansen and Q. Ji, "In the Eye of the Beholder: A Survey of Models for Eyes and Gaze," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 32, no. 3, pp. 478–500, Mar. 2010.

[14] F. B. Narcizo, J. E. Rangel de Queiroz, and H. M. Gomes, "Remote Eye Tracking Systems: Technologies and Applications," in *2013 26th Conference on Graphics, Patterns and Images Tutorials*. IEEE, 2013, pp. 15–22.

[15] A. T. Duchowski, "A breadth-first survey of eye-tracking applications," *Behavior Research Methods, Instruments, & Computers*, vol. 34, no. 4, pp. 455–470, 2002.

[16] A. T. Duchowski, *Eye tracking methodology - theory and practice (2. ed.)*. London: Springer London, 2007.

[17] C. H. Morimoto and M. R. M. Mimica, "Eye gaze tracking techniques for interactive applications." *Computer Vision and Image Understanding*, vol. 98, no. 1, pp. 4–24, 2005.

[18] A. Fitzgibbon, M. Pilu, and R. B. Fisher, "Direct least square fitting of ellipses," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 5, pp. 478–480, 1999.

[19] D. Zhu, S. T. Moore, and T. Raphan, "Robust pupil center detection using a curvature algorithm," *Computer Methods and Programs in Biomedicine*, vol. 59, no. 3, pp. 145–157, Jun. 1999.

[20] J. C. Barry, U. M. Pongs, and W. Hillen, "Algorithm for Purkinje images I and IV and limbus centre localization," *Computers in Biology and Medicine*, vol. 27, no. 6, pp. 515–531, Nov. 1997.

[21] C. H. Morimoto, D. Koons, A. Amir, and M. Flickner, "Pupil detection and tracking using multiple light sources," *Image and vision computing*, vol. 18, no. 4, pp. 331–335, Mar. 2000.

[22] D. W. Hansen and A. E. C. Pece, "Eye tracking in the wild," *Computer Vision and Image Understanding*, vol. 98, no. 1, pp. 155–181, Apr. 2005.

[23] E. D. Guestrin and M. Eizenman, "Remote Point-of-Gaze Estimation with Free Head Movements Requiring a Single-Point Calibration," in *2007 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. IEEE, 2007, pp. 4556–4560.

[24] D. W. Hansen, J. S. Agustin, and A. Villanueva, "Homography normalization for robust gaze estimation in uncalibrated setups," in *Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications*. New York, New York, USA: ACM, Mar. 2010, pp. 13–20.

[25] E. D. Guestrin and M. Eizenman, "Remote point-of-gaze estimation with single-point personal calibration based on the pupil boundary and corneal reflections," in *2011 24th IEEE Canadian Conference on Electrical and Computer Engineering (CCECE)*. IEEE, 2011, pp. 000 971–000 976.

[26] Z. Zhu and Q. Ji, "Novel Eye Gaze Tracking Techniques Under Natural Head Movement," *Biomedical Engineering, IEEE Transactions on*, vol. 54, no. 12, pp. 2246–2260, 2007.

[27] J. J. Cerrolaza, A. Villanueva, M. Villanueva, and R. Cabeza, "Error characterization and compensation in eye tracking systems," in *Proceedings of the 2012 Symposium on Eye-Tracking Research & Applications*. New York, New York, USA: ACM, Mar. 2012, pp. 205–208.

[28] W. Lotmar, "Theoretical Eye Model with Aspherics*," *Journal of the Optical Society of America*, vol. 61, no. 11, pp. 1522–1529, Nov. 1971.

[29] E. D. Guestrin and M. Eizenman, "General theory of remote gaze estimation using the pupil center and corneal reflections," *Biomedical Engineering, IEEE Transactions on*, vol. 53, no. 6, pp. 1124–1133, Jun. 2006.

[30] D. Model and M. Eizenman, "An automatic personal calibration procedure for advanced gaze estimation systems," *Biomedical Engineering, IEEE Transactions on*, vol. 57, no. 5, pp. 1031–1039, 2010.

[31] L. R. Young and D. Sheena, "Survey of eye movement recording methods," *Behavior Research Methods & Instrumentation*, vol. 7, no. 5, pp. 397–429, 1975.

[32] Z. Zhang, "A flexible new technique for camera calibration," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 22, no. 11, pp. 1330–1334, Nov. 2000.

[33] R. Hartley and A. Zisserman, "Multiple view geometry in computer vision," *Robotica*, 2005.

[34] J. Heikkilä and O. Silvén, "A Four-step Camera Calibration Procedure with Implicit Image Correction." in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE Comput. Soc, 1997, pp. 1106–1112.

[35] E. Dubrofsky, "Homography estimation," Master's thesis, The University of British Columbia, Vancouver, 2009.

[36] S. Wyder and P. C. Cattin, "Eye Tracker Accuracy: Quantitative Evaluation of the Invisible Eye Center Location ," submitted.

Stephan Wyder

BSc Computer Science

MSc Biomedical Engineering

Schosshaldenstrasse 16a
CH-3006 Bern
📞 *+41 78 823 15 65*
✉ *stephan.wyder@pixelwise.ch*

Personal Details

Date of Birth	13th January 1982
Nationality	Swiss

Date of Birth: 13th January 1982

Nationality: Swiss

Education

2013–2017   **PhD in Biomedical Engineering**, *University of Basel*, Basel.
Topic: Eye Tracking Based Navigation for Proton Therapy.
Supervisor: Prof. Dr. Philippe C. Cattin, Dept. of Biomedical Engineering.

2009–2012   **MSc in Biomedical Engineering**, *University of Bern*, Bern.
Focus: Image - Guided Therapy.

2004–2008   **BSc in Computer Science**, *Bern University of Applied Sciences*, Bern.
Focus: Computer Perception and Virtual Reality.

2002–2003   **Berufsmatura**, *Gewerblich Industrielle Berufsschule Bern*, Bern.

1998–2002   **Apprenticeship in Information Technology**, *Ascom AG*, Bern.

Work Experience

2013–2017   **Scientific coworker (100%)**, *MIAC (University of Basel)*, Basel.
Research in the field of eye tracking and medical navigation systems.

2012–2013   **Biomedical engineer (100%)**, *ARTORG (University of Bern)*, Bern.
Civil service. Research and development in the field of ophthalmic slit lamps and camera systems. Enhancements of optical autofocus systems.

2008–2012   **Software engineer (40% - 100%)**, *ISS AG*, Port.
Research and development in the field of medical technology (ophthalmology). Innovation projects in a small, interdisciplinary team.

2004–2007   **Software developer (60% - 100%)**, *Bucher & Suter AG*, Worblaufen.
Development of software with PHP, Java and C++ in the field of contact centers. Integration of customized solutions into customer's IT environment.

Accepted Peer Reviewed Publications

[1] S. Wyder *et al.*, "With Gaze Tracking Toward Noninvasive Eye Cancer Treatment," *Biomedical Engineering, IEEE Transactions on*, vol. 63, no. 9, pp. 1914–1924, 2016.

[2] S. Wyder and P. C. Cattin, "Stereo Eye Tracking with a Single Camera for Ocular Tumor Therapy," in *Proceedings of the Ophthalmic Medical Image Analysis International Workshop*, 2016, pp. 81–88.