NAIST-IS-DD1461207

Doctoral Dissertation

STARE: Real-Time, Wearable, Simultaneous Gaze Tracking and Object Recognition from Eye Images

Lotfi El Hafi

September 15, 2017

Graduate School of Information Science
Nara Institute of Science and Technology

A Doctoral Dissertation
submitted to Graduate School of Information Science,
Nara Institute of Science and Technology
in partial fulfillment of the requirements for the degree of
Doctor of ENGINEERING

Lotfi El Hafi

Thesis Committee:

Professor Tsukasa Ogasawara	(Supervisor)
Professor Hirokazu Kato	(Co-supervisor)
Associate Professor Jun Takamatsu	(Co-supervisor)
Assistant Professor Ming Ding	(Co-supervisor)
Associate Professor Kentaro Takemura	(Tokai University)

STARE: Real-Time, Wearable, Simultaneous Gaze Tracking and Object Recognition from Eye Images[*]

Lotfi El Hafi

Abstract

This thesis proposes STARE, a wearable system to perform real-time, simultaneous eye tracking and focused object recognition for daily-life applications in varied illumination environments. The proposed system extracts both the gaze direction and scene information using eye images captured by a single RGB camera facing the user's eye. In particular, the method requires neither infrared sensors nor a front-facing camera to capture the scene, making it more socially acceptable when embedded in a wearable device. This approach is made possible by recent technological advances in increased resolution and reduced size of camera sensors, as well as significantly more powerful image treatment techniques based on deep learning.

First, a model-based approach is used to estimate the gaze direction using RGB eye images. A 3D eye model is constructed from an image of the eye by fitting an ellipse onto the iris. The gaze direction is then continuously track by rotating the model to simulate projections of the iris area for different eye poses and matching the iris area of the subsequent images with the corresponding projections obtained from the model. By using an additional one-time calibration, the point of regard (POR) is computed, which allows to identify where a user is looking in the scene image reflected on the cornea.

Next, objects in the scene reflected on the cornea are recognized in real time using the gaze direction information. Deep learning algorithms are applied to

[*] Doctoral Dissertation, Graduate School of Information Science, Nara Institute of Science and Technology, NAIST-IS-DD1461207, September 15, 2017.

classify and then recognize the focused object in the area surrounding the reflected POR on the eye image. Additional processes using High Dynamic Range (HDR) demonstrate that the proposed method can perform in varied illumination conditions.

Finally, the validity of the approach is verified experimentally with a 3D-printable prototype of a wearable device equipped with dual cameras, and a high-sensitivity camera in extreme illumination conditions. Further, a proof-of-concept implementation of a state-of-the-art neural network shows that the focused object recognition can be performed in real time.

To summarize, the proposed method and prototype contribute a novel, complete framework to 1) simultaneously perform eye tracking and focused object analysis in real time, 2) automatically generate datasets of focused objects by using the reflected POR, 3) reduce the number of sensors in current gaze trackers to a single RGB camera, and 4) enable daily-life applications in all kinds of illumination. The combination of these features makes it an attractive choice for eye-based human behavior analysis, as well as for creating large datasets of objects focused by the user during daily tasks.

Keywords:

Eye model, Corneal image, Gaze tracking, Object recognition, Wearable device, Real time

ＳＴＡＲＥ：眼球画像を用いた実時間処理可能な装着型デバイスによる視線追跡と物体認識*

エル ハフィ ロトフィ

内容梗概

本論文では，日常生活の様々な照明環境下で利用可能であり，装着型デバイスを用いて視線追跡と注目物体の認識を同時に実時間で実行できるシステム「ＳＴＡＲＥ」を提案する．本システムは，単眼ＲＧＢカメラをユーザの眼球が見えるように取り付け，撮影された眼球画像を用いて視線方向とシーン情報を抽出する．シーンを取得するためにフロントカメラを必要としないため，常時装着型であるにもかかわらず社会的（例：肖像権・プライバシー）に受け入れられやすい．深層学習に基づく強力な画像処理技術だけでなく，カメラセンサの高解像度化および小型化などの近年の技術進歩により，本アプローチは実現可能となった．

本手法では，はじめにモデルベースの手法を用いて，眼球ＲＧＢ画像から視線方向を推定する．まず，眼球画像に映る虹彩の輪郭に楕円を当てはめることによって，眼球の３Ｄモデルを構築する．推定の際，構築された眼球の３Ｄモデルを回転させることにより，異なる眼球姿勢における虹彩領域を計算し，計測により得られた画像に投影し照合することで，視線方向を推定する．時系列で得られる眼球ＲＧＢ画像に適用することで，視線方向を連続的に追跡する．さらに，キャリブレーションを１回おこなうことで，視線方向と角膜上で反射した画像中にユーザが注目した物体が映り込む領域との関係をモデル化することができる．

次に，角膜での反射を介して得られるシーン画像（角膜反射画像）と視線方向情報とを組み合わせることで，注目物体を実時間で識別する．角膜反射画像上で注目物体が映るであろう領域に深層学習アルゴリムを適用することで，物体を分類し認識する．ハイダイナミックレンジ（ＨＤＲ）カメラを代わりに使用することで，提案手法が様々な照明条件で実行できることを示す．

*奈良先端科学技術大学院大学 情報科学研究科 博士論文, NAIST-IS-DD1461207, 2017 年 9 月 15 日.

最後に，カメラを２台装備し３Ｄプリンタで製造可能な装着型デバイスのプロトタイプを作成すること，および極端な照明条件下でハイダイナミックレンジカメラを用いることで，本アプローチの妥当性を多面的に検証する．さらに，本アプローチの概念実証をおこなうため，実際にニューラルネットワークを用いて実装し，注目物体の認識が実時間で実行できることを示した．

まとめると，本手法と装着型デバイスのプロトタイプは斬新で完全なフレームワークを提供するものであり，以下のような特徴を持つ．１）視線追跡と注目物体の認識を実時間で同時に実行できる．２）角膜反射画像中の注目領域を用いて，注目物体のデータセットを自動的に生成できる．３）単眼ＲＧＢカメラのみで計測可能なシステムである．４）様々な照明環境下で利用可能であるため，提案する装着型デバイスは日常生活で利用可能である．これらの特徴を実現できたことで，視線に基づいた人間の行動分析や，日常業務中にユーザが注視している物体に関する膨大なデータセットの生成に適用できると考えている．

キーワード

眼球モデル，角膜画像，視線追跡，物体認識，装着型デバイス，実時間処理

Contents

List of Figures

List of Tables

Stare: look fixedly or vacantly at someone or something with one's eyes wide open. ”He stared at her in amazement.” (Oxford Dictionary)

1. Introduction

Tracking the gaze and movements of the eye using a wearable device is a way to obtain natural data that can be used to analyze users' interests, behaviors and state of mind, Nitschke *et al.* [1]. Further, eye tracking can be used as an intuitive augmented reality (AR) input, or used to reduce motion sickness induced by ill-calibrated virtual reality (VR) devices, Geuss [2]. The growing interest in the maturing fields of AR, VR and smart wearable devices has created new momentum for eye tracking.

Today, most commercially available eye-tracking solutions are based on active infrared (IR) illumination and the so-called Purkinje images: an IR light source illuminates the eye and one or multiple IR camera sensors capture its reflections on different eye layers, called Purkinje images, to compute the gaze direction. Current state-of-the-art wearable eye trackers, such as the Tobii Pro Glasses 2, advertise below 1° error of gaze tracking using this technique.

However, IR-based methods have drawbacks. First, IR active illumination is mostly limited to indoor environments as sunlight interferences occur in direct daylight. Second, as the reflected scene on the cornea is not visible in the IR spectrum, an additional front-facing RGB camera is required to match the computed point of regard (POR) in the scene. Such front-facing cameras pose significant social concerns which have proved to be an obstacle for smart eyewear efforts such as the Google Glass.

To solve these problems, this thesis proposes STARE[1], a new method to extract both the gaze direction and scene information using eye images captured by a single RGB camera facing the user's eye. In particular, the method does not require a front-facing camera to capture the scene, making it more socially acceptable when embedded in a wearable device. This approach is made possible by recent technological advances: increased resolution and reduced size of camera sensors, as well as significantly more powerful image treatment techniques based on deep learning.

[1] Simultaneous Tracking and Attention Recognition from Eyes.

1.1 Method Overview

To estimate the gaze direction, a model-based approach is used. First, a 3D eye model is constructed from an image of the eye by fitting an ellipse onto the iris. Then, the gaze direction is continuously tracked by rotating the model to simulate projections of the iris area for different eye poses and matching the iris area of the subsequent images with the corresponding projections obtained from the model. By using an additional one-time calibration, the reflected POR on the cornea can also be computed, which allows me to identify where a user is looking in the scene image reflected on the cornea. Next, using the gaze direction information to automatically build large datasets, deep learning algorithms are applied to classify and then recognize the focused object in the area surrounding the reflected POR on the eye image. Finally, experiments using a High Dynamic Range (HDR) sensor demonstrate that the proposed method can perform in varied illumination conditions.

In order to validate the method in daily-life situations, several experiments have been conducted using different hardware such as a wearable prototype equipped with dual cameras, and a high-sensitivity camera in extreme illumination conditions. Further, a proof-of-concept implementation of a state-of-the-art neural network showed that the focused object recognition can be performed in real time.

1.2 Contributions

Several contributions of this thesis have been published in peer-reviewed journals and conference proceedings. In particular, the model approach for gaze tracking based on eye images was proposed in [3], HDR processing of corneal images was explored in [4, 5], and the method was further refined with preliminary results of deep learning-based object recognition from eye images in [6]. This thesis summarizes and extends the aforementioned previous works by proposing simultaneous eye gaze extraction and focused object analysis in real time using deep learning, as well as the design for the 3D printable prototype. The main results have been published in [7].

To summarize, the proposed method and prototype contribute a novel, complete framework to:

1. Simultaneously perform eye tracking and focused object analysis in real time.

2. Automatically generate datasets of focused objects by using the reflected POR.

3. Reduce the number of sensors in current gaze trackers to a single RGB camera.

4. Enable daily-life applications in all kinds of illumination.

The combination of these features makes it an attractive choice for eye-based human behavior analysis, as well as for creating large datasets of objects focused by the user during daily tasks. Finally, the simultaneity in the gaze tracking and the focus analysis is an important step forward to provide an end-to-end solution that includes the capture device, the gaze estimation, and the object recognition in a single unified system and interface, without requiring additional post-processing of gaze data.

1.3 Thesis Outline

The remainder of the thesis is structured as follows. Section 2 introduces the related work and Section 3 the wearable prototype. Section 4 describes the geometric eye model derived from the main characteristics of the human eye, and how to reconstruct a 3D eye model from an image of the eye to estimate both its location and orientation relative to the camera. Section 5 proposes a method to continuously track the gaze direction using the previously built model. Section 6 presents the deep learning techniques applied for real-time recognition of the focused object. Finally, Section 7 explores HDR imaging and Section 8 concludes with an overview of potential future work and applications.

All the resources used in this project are available for download at `http://robotics.naist.jp/stare/`.

2. Related Work

Gaze tracking has been studied over the past 30 years. Early efforts using RGB eye images and pupil tracking were rapidly discarded in favor of more precise and less computational-consuming IR imaging, based on the Purkinje images, as initially proposed by Guestrin and Eizenman [8]: an IR light source illuminates the eye and multiple IR camera sensors capture its reflections on different eye layers to compute the gaze direction, as illustrated by Figure 1.

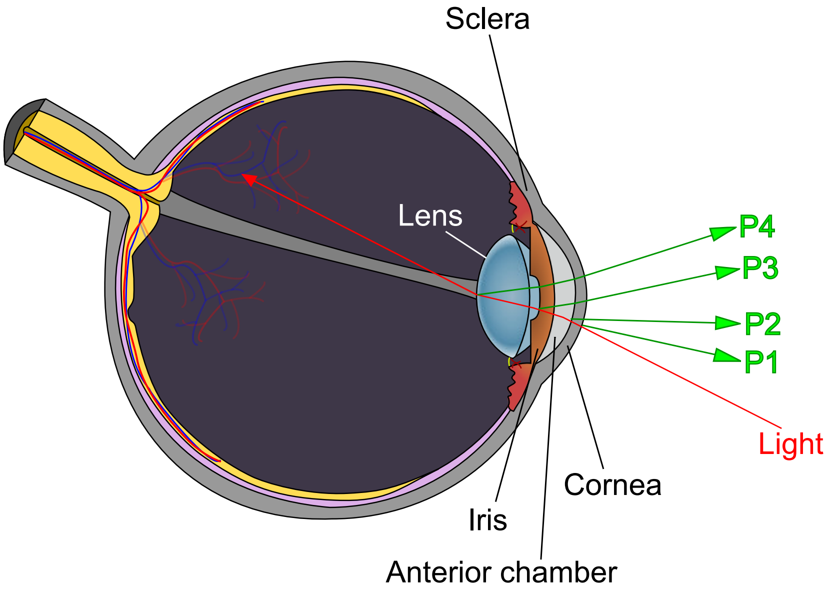


Figure 1. Illustration of the four Purkinje images illuminated by an active IR light source. Authorized reproduction and modification under the Creative Commons Attribution-Share Alike 4.0 International license, courtesy of Z22.

However, the advances in camera sensor technology and computational power have renewed the interest for the analysis of corneal reflections, also known as corneal imaging. A comparison between RGB and IR eye images is given by Figure 2.

Corneal imaging was first explored by Nishino and Nayar [9]. This early work analyzed the information contained in the eye images and suggested to use corneal reflections to obtain information about the surrounding environments. In particular, they showed that the combination of the cornea of an eye and the camera

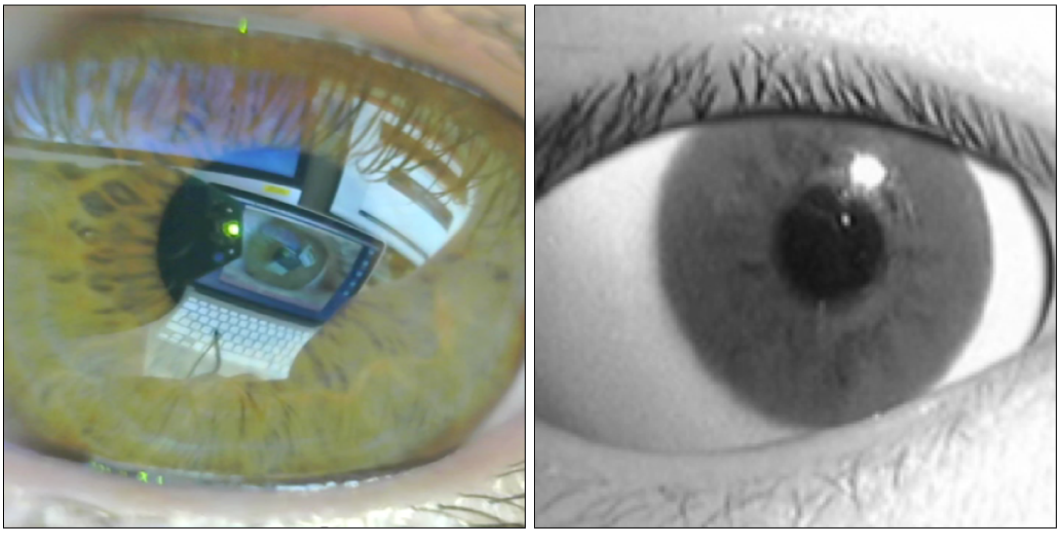


Figure 2. Comparison between RGB (left) and IR (right) eye images.

viewing the eye form a catadioptric imaging system. They also demonstrated that the structure of an object in the scene can be computed from the two eyes of face picture. They concluded by suggesting to exploit the visual information embedded in the appearance of eyes, which convey rich information about the person's intent and circumstances.

In a following effort, Nitschke *et al.* [1] conducted an extensive survey of the potential applications for corneal imaging, including visual recognition, computer graphics and vision, human-computer interaction and diagnostic studies. They concluded that analyzing and exploiting the corneal reflections from eye images can be beneficial to accomplish a wide range of tasks that involve information about the environment and the observer as advancement in computational systems, devices and architectures demands for novel interfaces and forms of interaction.

More recent research by Nitschke *et al.* [10], Takemura *et al.* [11, 12], and Plopski *et al.* [13] successively proposed new methods to calibrate the eye pose relative to a camera under various circumstances.

This lead to the first practical use of corneal imaging for a wearable gaze tracker by Takemura *et al.* [14]. They proposed a method for estimating the POR using the corneal images captured from an eye camera, and the scene images captured from a front-facing camera. Although they demonstrated that the POR could be estimated continuously, their method was limited by the quality of the corneal images which were dependent on ambient illumination, and therefore the method could not be applied in low-light conditions. They concluded that the POR should not be estimated using corneal images alone, reason why they used an additional front-facing camera to capture the scene.

However, recent technological advances in size and resolution of the camera sensors, as well as in machine learning, have made possible to estimate both the POR and the focused object using corneal images alone, eliminating the need for additional sensors such as a front-facing camera. A brief summary between RGB and IR technologies for eye image analysis is given by Table 1.

Table 1. Comparison between RGB and IR technologies for eye image analysis.

RGB eye images	IR eye images
Advantages	Advantages
Feature-rich images.	Purkinje images/reflections.
Scene reflection, no additional front camera is required for scene analysis.	Lightweight processing mostly based on geometry properties.
	Below 1° error is commonly achieved.
	+30 years of research, commercially available.
Disadvantages	Disadvantages
Inoperable in low-light conditions.	Requires active IR illumination.
Heavy processing based on computer vision algorithms.	No scene reflection, often requires an additional front camera for scene analysis.
Iris color contamination.	Sunlight interferences in outdoor environments.
Spherical distortions.	
Eyelid occlusions.	

3. Wearable Prototype

To validate the research, an experimental eye tracker is required. Based on the main requirements of the proposed method, the prototype has to:

1. Be wearable and suited for regular use in daily-life activities.
2. Use RGB cameras that can capture sharp and feature-rich eye images at close range.

To achieve these, the prototype uses a pair of JINS MEME glasses, Kunze *et al.* [15], as a base. On these, a 3D-printed frame is mounted to fix two Logitech C310 camera sensors, as shown in Figure 3. This prototype is the result of the several iterations shown in Figure 6. The lens of the cameras has been customized to achieve a sharp focus at very close range. Eye images from both eyes are captured with a resolution of 1280 × 960 pixels at 30 fps. The two video streams are transmitted to a workstation using two USB 2.0 cables via an USB Video Class (UVC) 1.1 standard interface. The total weight (excluding cables) is 100 g. The JINS MEME glasses provide additional sensors: an electrooculograph, an accelerometer, and a gyroscope for six-axis head movement tracking. No front-facing camera is present as the scene is extracted from the reflections captured by the eye cameras.

Note that although the prototype has two eye cameras, the proposed method only requires one. The extra eye camera is mainly used to compare results from both eyes. Also, the extra sensors provided by the JINS MEME are mainly used for debug purpose when developing the solution. In particular, the electrooculograph can be used to quickly validate the results of the proposed gaze tracking method.

3.1 Software

All the modules related to vision are implemented using OpenCV 3.2 C++ functions wrapped by a Python 3 frontend. This offers a good compromise between the speed of C++ and the convenience of Python 3. Also, the OpenCV CUDA modules are called whenever possible to benefit from GPU acceleration.

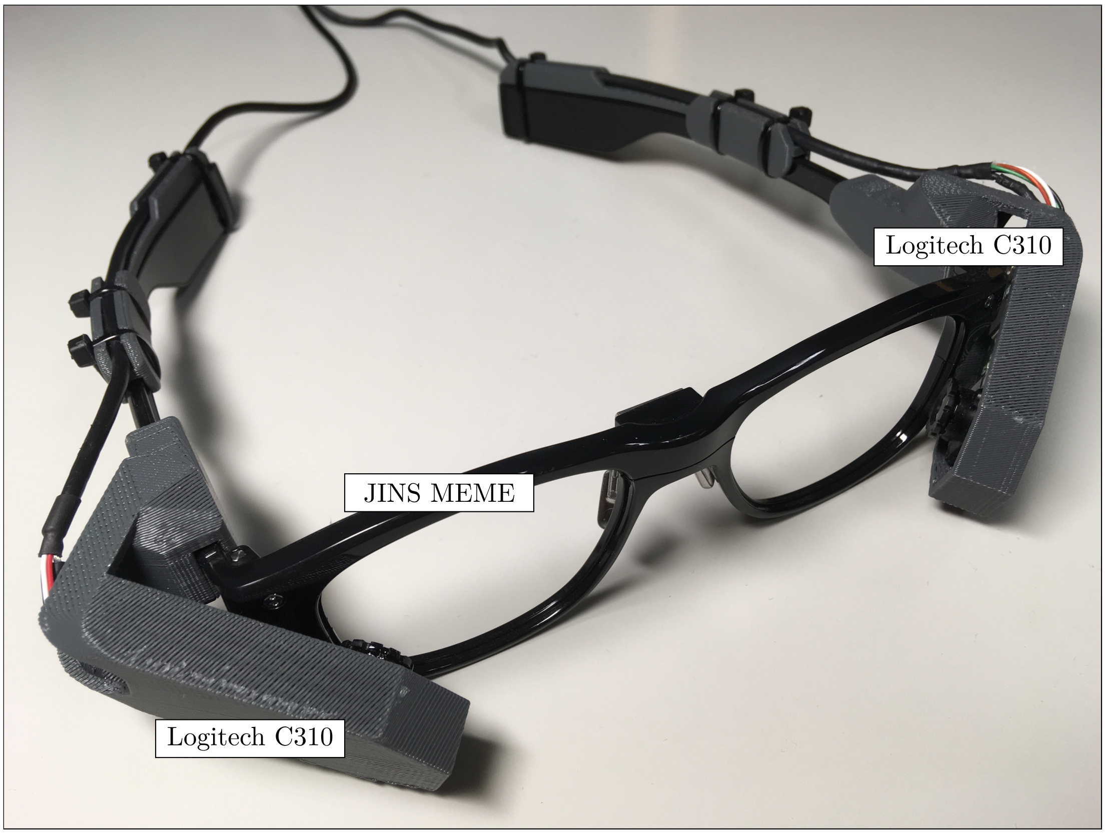


Figure 3. 3D-printable wearable prototype built with a pair of JINS MEME and two Logitech C310 cameras.

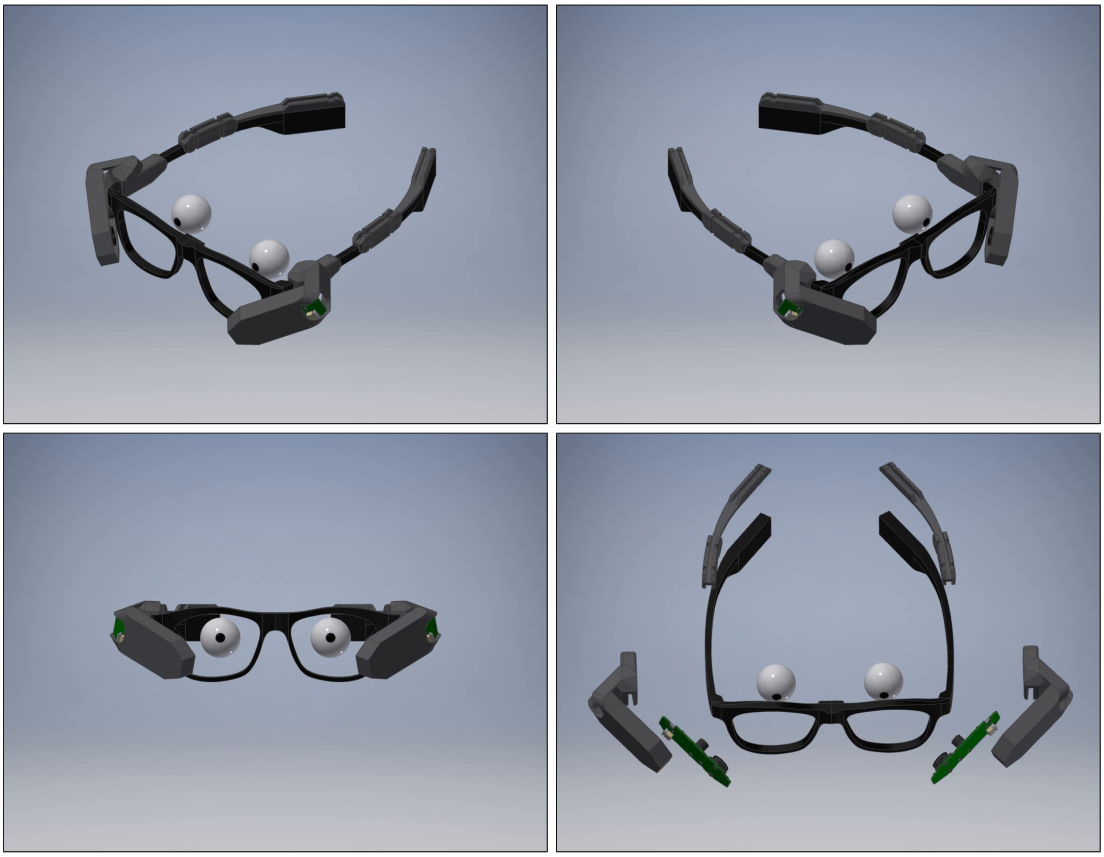


Figure 4. Split views of the 3D-printable CAD resources.

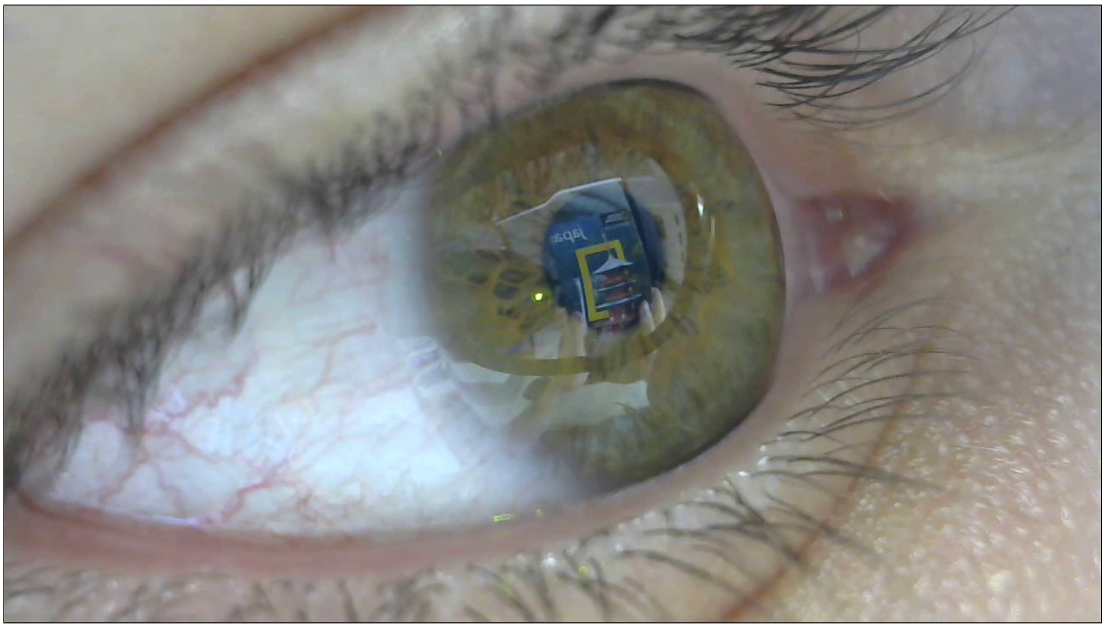


Figure 5. Example of feature-rich eye image obtained from the wearable prototype.

The framework used to implement the modules related to deep learning is Google TensorFlow, developed by Abadi *et al.* [16] at Google Brain. This framework is also implemented in Python 3 and offers a high level of abstraction to work on multiple GPUs/CPUs and scale the solution on different hardware.

All the libraries used in this project are open-source.

3.2 Specifications

The full specifications of the wearable prototype are given in Table 2.

3.3 Deployment

The current version of the prototype still requires to be connected to a powerful workstation equipped with a CUDA-enable GPU to achieve real-time object recognition. Therefore, although the prototype is wearable, it is not yet ready for daily use and commercial deployment.

The next version will consider using a NVIDIA Jetson TX2 module to embed

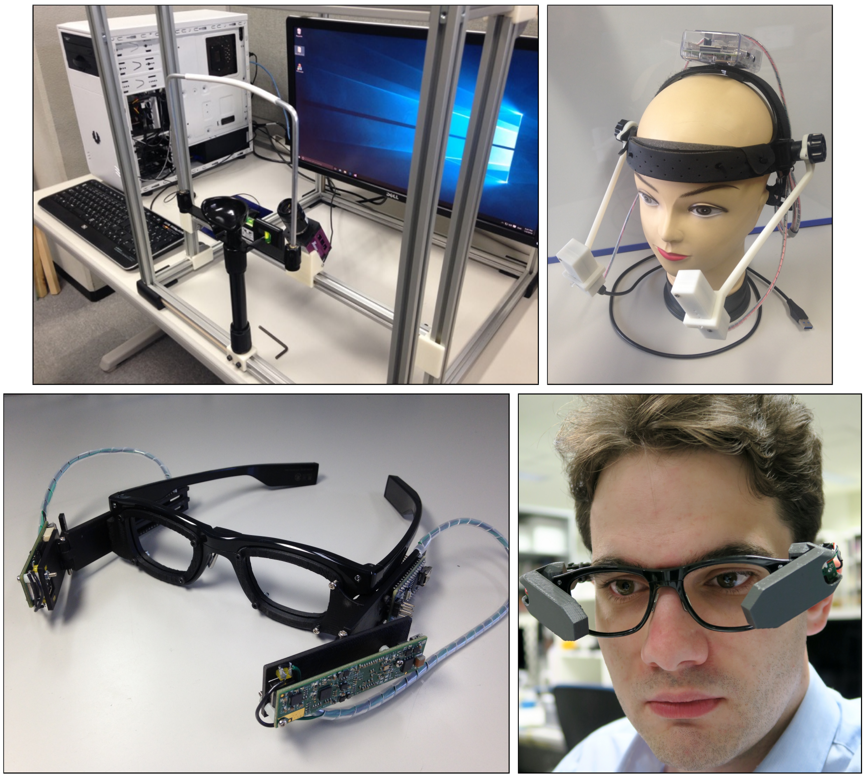


Figure 6. Evolution of the prototype from a static setup to a wearable device.

Table 2. Detailed specifications of the wearable prototype.

Components	Specifications	Values	Units
Frame: JINS MEME	Sensors	Accelerometer, gyroscope, electrooculograph	1
Cameras: Logitech C310	Spectrum	RGB	2
	Resolution	1280 × 960	
	Rate	30 fps	
	Minimum distance	15 mm	
	Interface	USB 2.0	
CPU: Intel Core i5-7600	Architecture	Quad-core	1
	Clock	3.5 GHz	
GPU: NVIDIA GeForce GTX 1080	CUDA cores	2560	1
	Memory	8 GB	

power-efficient computing into the device, making it truly wearable. The main challenge will be to provide the same real-time experience with 10 times less CUDA cores than in the NVIDIA GeForce GTX 1080 used currently. However, the computing speed of the recognition achieved in Section 6 will most certainly allow to do so.

Finally, at the time of the writing, the JINS MEME and its sensors are not strictly required outside development, and can be replaced by a passive 3D-printable frame. Although never used during the development, this passive frame has been completely designed and can be printed at any time.

4. Eye Model

This section describes the main characteristics of the human eye and how to derive a geometric model from them.

4.1 Human Eye Anatomy

Figure 7 shows a cross-section of the human eye. When observing an eye from the outside, the most distinctive parts are the colored iris, the pupil at its center, and the white sclera that surrounds it, as described by Nitschke *et al.* [10]. The outer layer of the eye is the cornea, which is more difficult to observe. It covers the iris and fades into the sclera at the limbus. The cornea focuses images onto the retina, or more precisely, onto the fovea which is the most sensitive part of the eye. Important properties of the cornea are its transparency and its specular reflection characteristics due to the film of tears that coats its surface. This mirror-like characteristic will be particularly relevant for extracting information about the scene and the POR.

4.2 Eye Geometric Model

The human eye can be subdivided into two overlapping spheres of different sizes: a smaller cornea sphere that includes the cornea, the iris, the pupil and the lens, and a bigger sclera sphere that includes the sclera, the vitreous humor and the retina with its fovea. The two spheres intersect at the limbus which defines a circle. This model is described in Figure 8:

- Points **L**, **C** and **S** are respectively the limbus, cornea and sclera centers. A priori unknown.
- The vector **g** is the optical axis of the eye, crossing all the aforementioned centers. It intersects the cornea sphere at the cornea apex designated by **A**.
- The vector **v** is the visual axis that goes from the fovea to the actual POR. The visual axis corresponds to the gaze direction and its estimation is the purpose of any gaze tracking system.

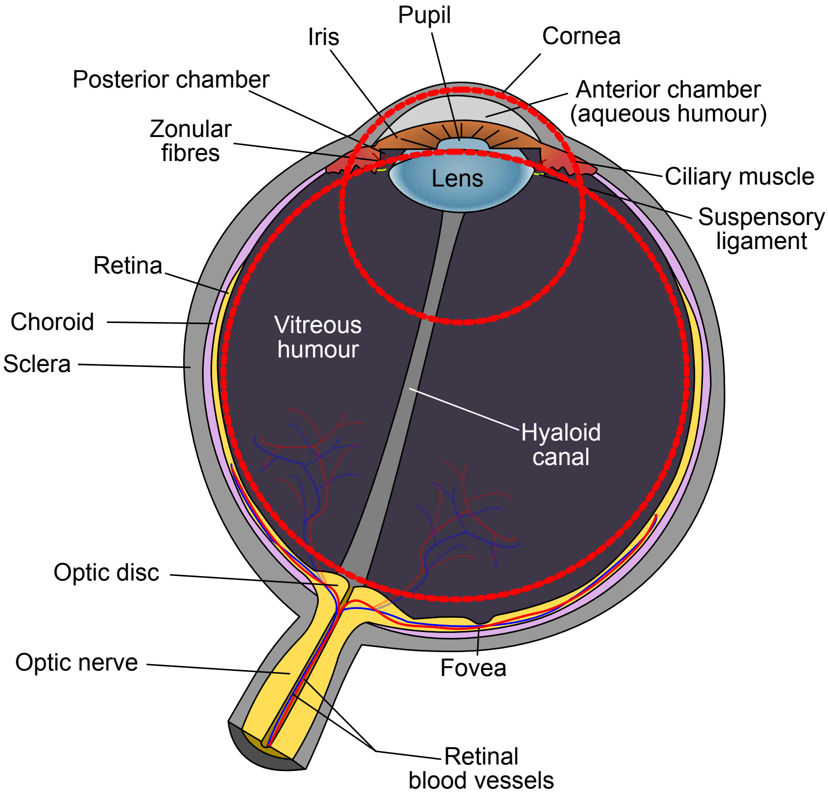


Figure 7. Cross-section of the human eye with the cornea and sclera spheres indicated by the two dashed circles in red. Authorized reproduction and modification under the Creative Commons Attribution-Share Alike 4.0 International license, courtesy of Z22.

- Distances r_L, r_C and r_S are respectively the limbus, cornea and sclera radii. All are known anatomical parameters.

- Distances d_{AL}, d_{LC} and d_{CS} separate the different components of the model. All are known anatomical parameters.

The anatomical parameters used in this thesis have been extensively studied by Wandell [17]. His work provides average values of eye parameters between multiple individuals. In the following, these values are used when referring to the known parameters.

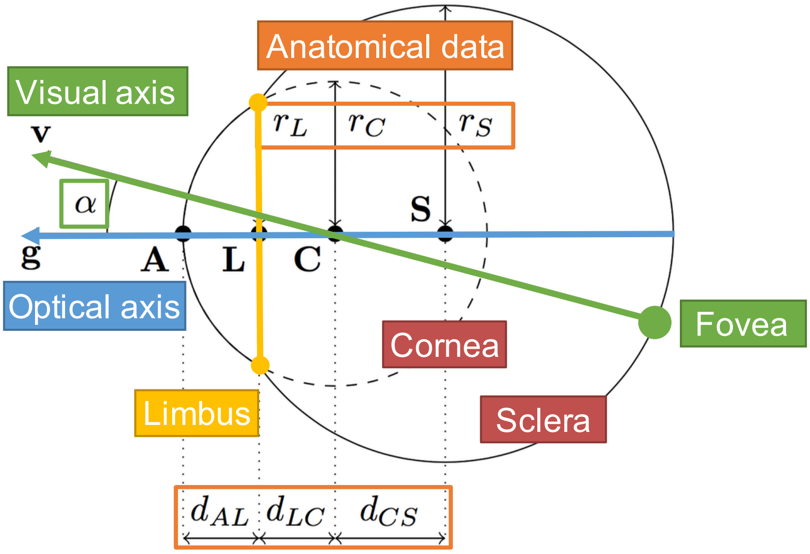


Figure 8. Geometric model of the human eye.

The optical axis is easy to estimate from the geometric properties of the eye but the visual axis is not. However, even though the visual axis, not the optical axis, corresponds to the direction of the POR, the optical axis can be used as an approximation of the visual axis. The angle formed by the two axes is denoted by α and assumed to be constant.

This geometric model will be applied throughout this paper to estimate the pose of the eye from an image. It will also be used to describe interactions between the incident light and the cornea surface. By nature, such a model can only approximate the reality: the actual shape of the eye is more complex than the one described by the model and its anatomical parameters vary between individuals. However, the variation of parameters among individuals is assumed to be sufficiently small.

4.3 Eye Model Construction

Both the location and orientation relative to the camera of the 3D eye model are estimated from a single eye image following the method of Nitschke *et al.* [10]. An overview is given by Figure 9.

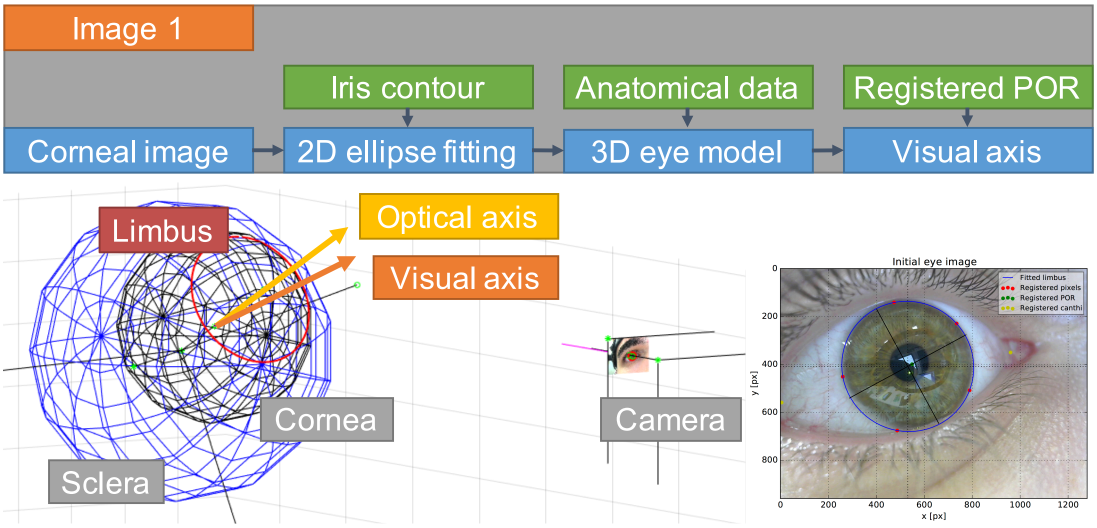


Figure 9. Overview of the 3D eye model construction.

First, weak perspective projection is assumed since the depth of the tilted limbus is much smaller than the distance between the eye and the camera. Thus, the almost circular limbus projects to an ellipse described by five parameters in the image coordinates: the center coordinates (c_u, c_v), the radii r_{max} and r_{min}, and the tilt ϕ as shown in Figure 10. Their values are estimated by fitting an ellipse on a set of limbus points using Least Squares. Figure 11 shows an example of the result.

Now that the limbus has been fully described on the image plane, a 3D model of the eye can be constructed. Its pose in the world coordinate frame, *i.e.* the coordinates of the limbus center $\mathbf{L}$ and the direction of the optical axis $\mathbf{g}$, is

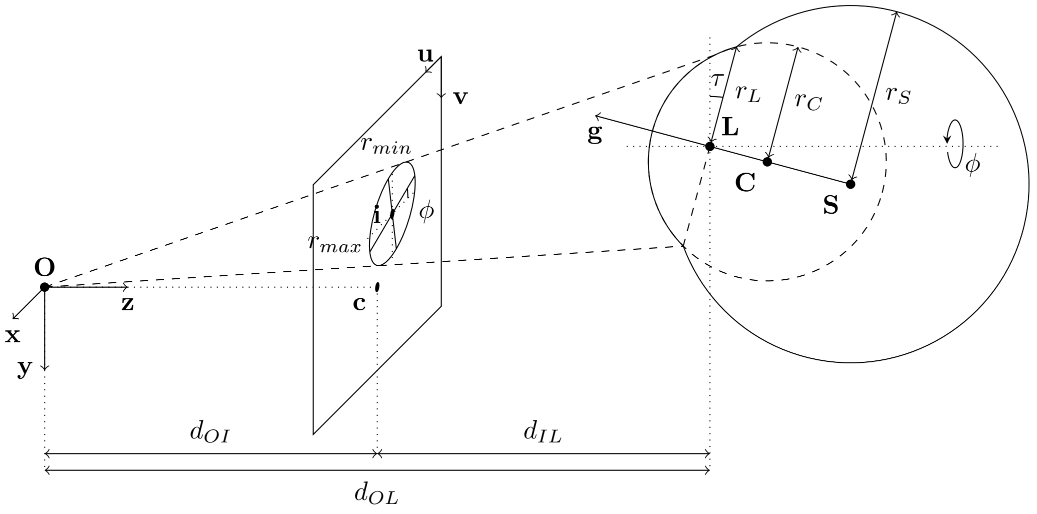


Figure 10. 3D eye model construction from an eye image.

computed following the geometric construction originally proposed by Nitschke *et al.* [10]. The origin $\mathbf{O} = (0,\ 0,\ 0)^\intercal$ is at the center of the camera lens as shown in Figure 10. When the camera focus is assumed to be at infinity[2], d_{OL} can be expressed as:

$$d_{OL} = f\frac{r_L}{r_{max}},$$

where r_L, r_{max} and the focal length of the camera f are known. If the limbus center is defined as $\mathbf{L} = (L_x,\ L_y,\ L_z)^\intercal$, by similarity:

$$\frac{L_x}{(i_u - c_u)\, s_x} = \frac{d_{OL}}{f}, \quad \frac{L_y}{(i_v - c_v)\, s_y} = \frac{d_{OL}}{f},$$

where s_x and s_y are the pixel-to-world-unit scaling coefficients, obtained from camera calibration, respectively along the x and y directions. By combining

[2] Which means $f = d_{OI}$. At close range, this sometimes cannot be assumed depending on the sizes of the sensor and the lens as described in preliminary work to this thesis [3]. To solve this problem, the thin lens model is used:

$$\frac{1}{f} = \frac{1}{d_{OI}} + \frac{1}{d_{OL}},$$

where $f \neq d_{OI}$ and d_{OL} is required to compute d_{OI}. In the case of a head-mounted device, d_{OL} is assumed to be known and constant.

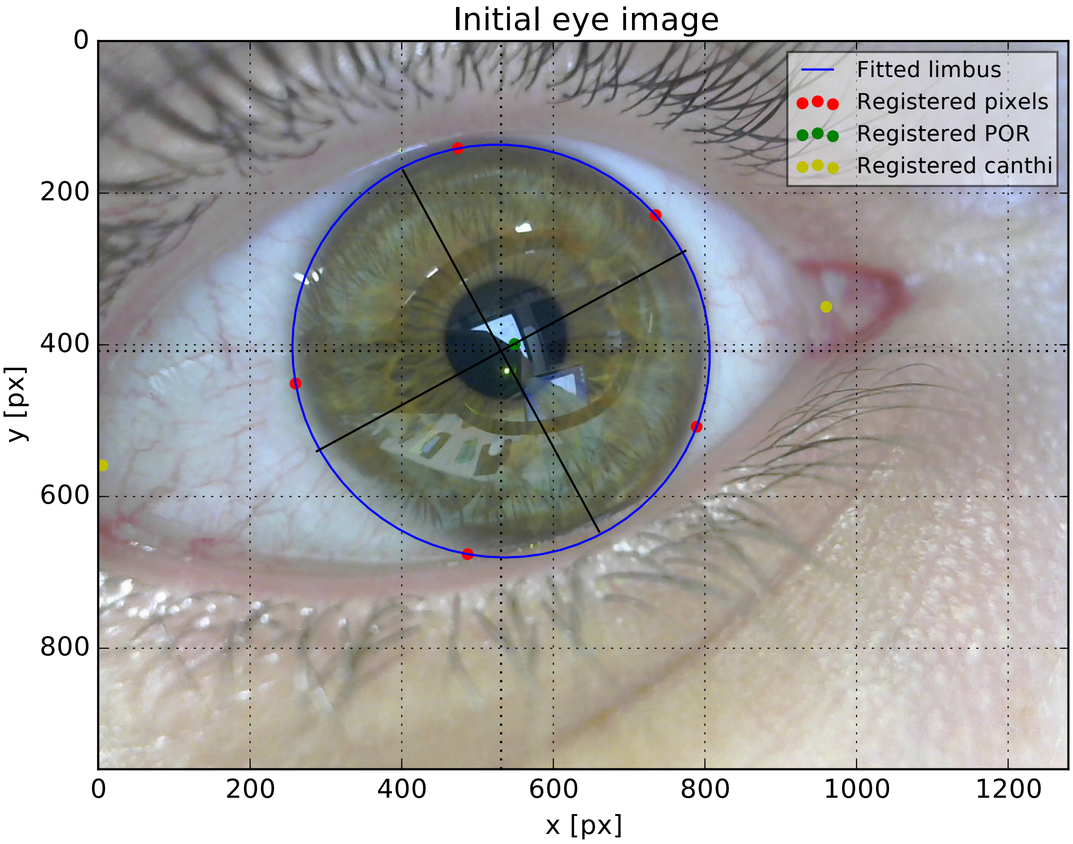


Figure 11. Initial eye image used for model construction.

these equations:

$$\mathbf{L} = \left(\frac{d_{OL}\,(i_u - c_u)\,s_x}{f},\ \frac{d_{OL}\,(i_v - c_v)\,s_y}{f},\ d_{OL} \right)^{\mathsf{T}}.$$

The tilt τ of the limbus plane with respect to the image plane is estimated from the shape of the ellipse up to a sign ambiguity:

$$\tau = \pm \arccos\left(\frac{r_{min}}{r_{max}}\right).$$

Indeed, two different limbus poses are possible from the projection alone: one looking in the direction of positive values of y, and another looking in the direction of negative values of y. In the case of a head-mounted camera, the ambiguity can be easily solved by knowing the relative pose of the camera to the eye, which is usually fixed and sufficiently tilted to avoid any ambiguity.

If not, this ambiguity can also be solved by attaching two LEDs to the camera, as proposed in preliminary work of this thesis [18]. A line between the camera origin **O** and the cornea center **C** intersects at a point where the camera origin is reflected on the cornea, as shown in Figure 12. Therefore, the projected cornea center location is obtained by finding the mean point of the two LEDs reflected in the image and use this information to resolve the sign ambiguity of τ.

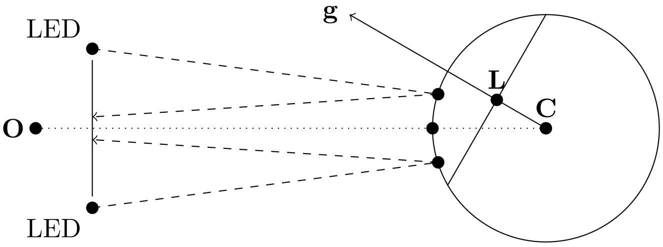


Figure 12. Solving the tilt ambiguity of the eye pose relative to the camera using LEDs.

The optical axis **g** is then given by:

$$\mathbf{g} = (\sin\tau \sin\phi,\ -\sin\tau\cos\phi,\ -\cos\tau)^{\mathsf{T}},$$

where ϕ is already known as the rotation angle of the ellipse fitted on the limbus in the image plane. Finally, the cornea center **C** and the sclera center **S** are given by:

$$\mathbf{C} = \mathbf{L} - d_{LC}\mathbf{g}, \quad \mathbf{S} = \mathbf{L} - \left(d_{LC} + d_{CS}\right)\mathbf{g},$$

and the limbus is computed as the intersection between the cornea and sclera spheres.

4.4 Visual Axis Calibration

To evaluate the gaze direction, *i.e.* the visual axis $\mathbf{v}$, an additional one-time calibration step is required to compute the angle α from the optical axis $\mathbf{g}$. In the current implementation, the user is asked to manually register the reflected POR on the image.

Figure 13 describes the relationship between the visual axis and the incident light coming from the POR, where $\mathbf{P}$ is the POR and $\mathbf{R}$ the reflected POR, *i.e.* the POR in the scene reflected on the corneal image. $\mathbf{n} = x\mathbf{s} + y\mathbf{l}$ is the normal at the reflected POR with $\mathbf{l}$ and $\mathbf{s}$ respectively the directions to the POR and to the camera optical center. The normal parameters x and y are unknown.

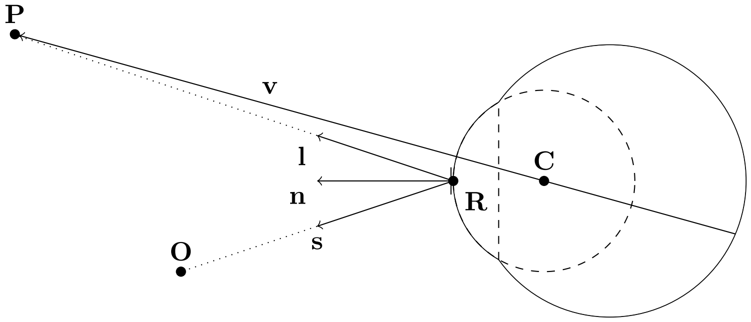


Figure 13. One-time visual axis calibration.

Assuming that the distance from the reflected POR to the POR is known, which can be expected from the user during a calibration process, the direction of the visual axis can be computed using a specular model of a sphere, Eberly [19]. The first step is to compute the normal $\mathbf{n}$ in order to find $\mathbf{R}$. This consists of solving the following biquadratic equation:

$$4cdy^4 - 4dy^3 + \left(a + 2b + c - 4ac\right)y^2 + 2\left(a - b\right)y + a - 1 = 0,$$

where $a = \mathbf{s} \cdot \mathbf{s}$, $b = \mathbf{s} \cdot \mathbf{l}$, $c = \mathbf{l} \cdot \mathbf{l}$, $d = \|\mathbf{s} \times \mathbf{l}\|^2$ are the coefficients. When $x = \left(-2y^2 + y + 1\right) / \left(2by + 1\right)$ is defined, the normal $\mathbf{n}$ is computed from the solution in $x > 0$ and $y > 0$. The reflected POR $\mathbf{R}$ and the visual axis $\mathbf{v}$ are then computed by straightforward vector geometry.

Although the calibration is required to compute the visual axis, it appears in the last results presented in Section 6 that the calibration of the visual axis does not influence much the focused object recognition. This means that the visual axis calibration can be skipped, relying exclusively on the optical axis as a first approximation of the gaze, if the focused object analysis is ultimately more important than the gaze direction precision alone.

5. Gaze Tracking

From the model built previously using a single eye image, the pose of the eye continuously tracked in the subsequent images, thus enabling real-time gaze tracking. An overview of the proposed algorithm is given by Figure 14.

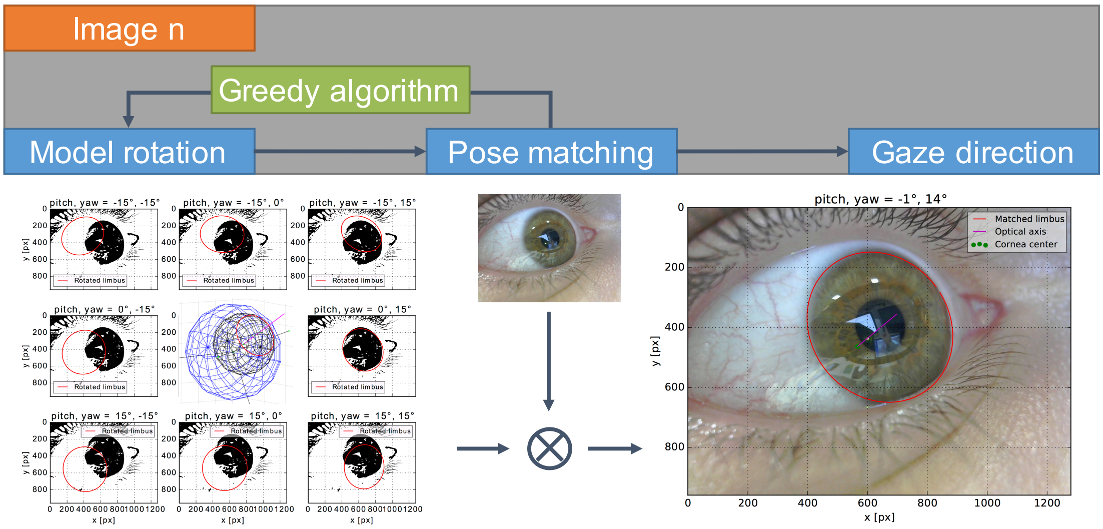


Figure 14. Overview of the model-based gaze tracking.

5.1 Pose Matching

In order to track the optical axis direction from the current image of the eye, first a pitch-yaw-roll reference frame is attached to the sclera center **S** of the 3D model built previously. The yaw axis is aligned with the eye corners for convenience. The pitch and yaw angles are respectively denoted by ϕ and ψ. By rotating the model around the pitch and yaw axis, several limbus projections are simulated for different eye poses, as shown in Figure 15. Note that the roll angle is not considered, assuming that the human eye is not capable of such a rotation.

The limbus of the rotated model is then projected into a binary image to serve as a mask. The projected area is set as binary 1s. The projection that matches the current limbus pose is then detected by summing the logical products of the inverted binary image of the current frame and the mask for each pitch and yaw

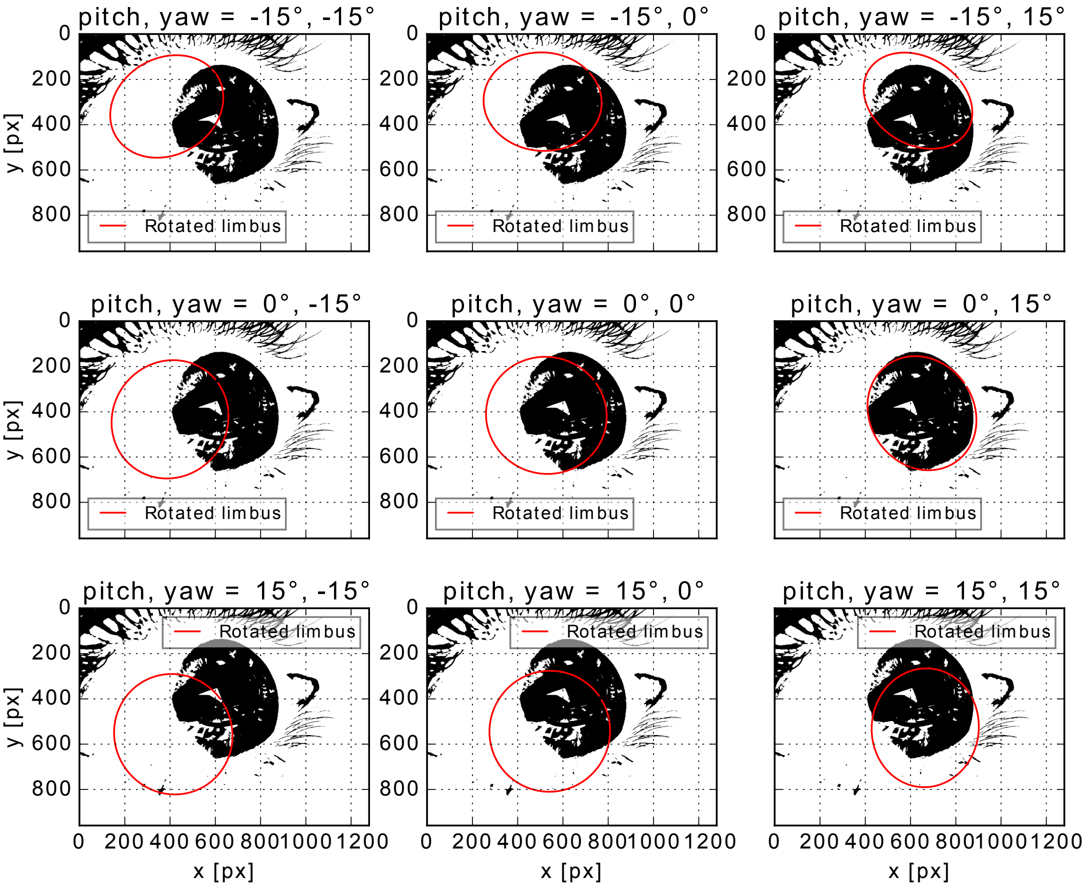


Figure 15. Simulation of several limbus projections for different eye poses.

values. The maximum value among the summed logical products corresponds to the current pose of the eye, as shown in Figure 16.

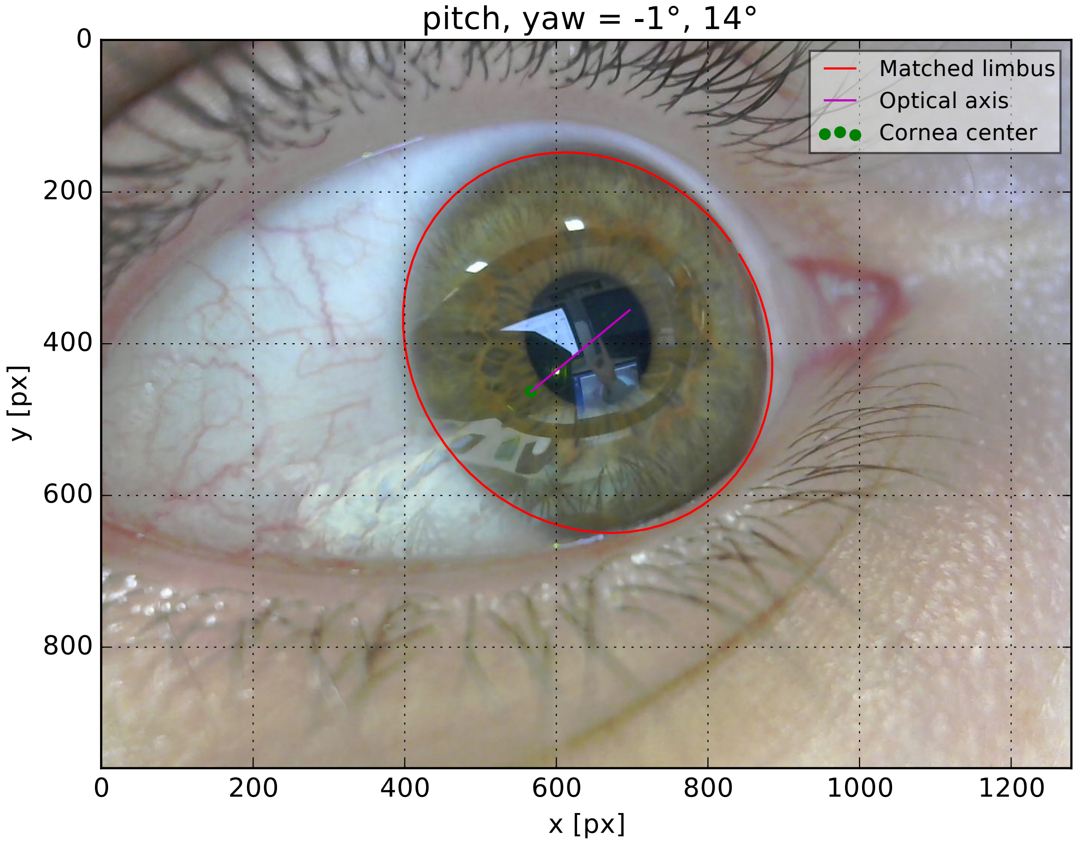


Figure 16. Matching the projection of the rotated limbus into the image plane.

This method was initially proposed by Takemura *et al.* [12, 14]. The current implementation moves toward a local minimum using a greedy algorithm, as shown by Figure 17. The aim is to accelerate the matching by avoiding generating unnecessary candidate poses.

From an initial guessed pose, the algorithm computes the poses of the adjacent pitch and yaw values. The newly computed pose that matches the best the current eye pose serves as the initial pose in the next iteration. The algorithm loops until no more significant change is observed.

Although the greedy algorithm is relatively simple and cannot guarantee that the result is a global minimum, it is nonetheless very fast and suited to this particular use case for the following reasons:

- Avoiding generating unnecessary projections, thus accelerating the overall pose matching.
- As the eye shapes and features are well known, the local minimum is most likely to be global.
- Since humans look straight most of the time, the pose corresponding to zero-values of pitch and yaw is a strong initial guess, thus only a few iterations are required to match the current eye pose and the matching is fast.

Note that the reflections on the iris area can result in white spots on the binary image that may introduce errors when computing the sum of the logical products.

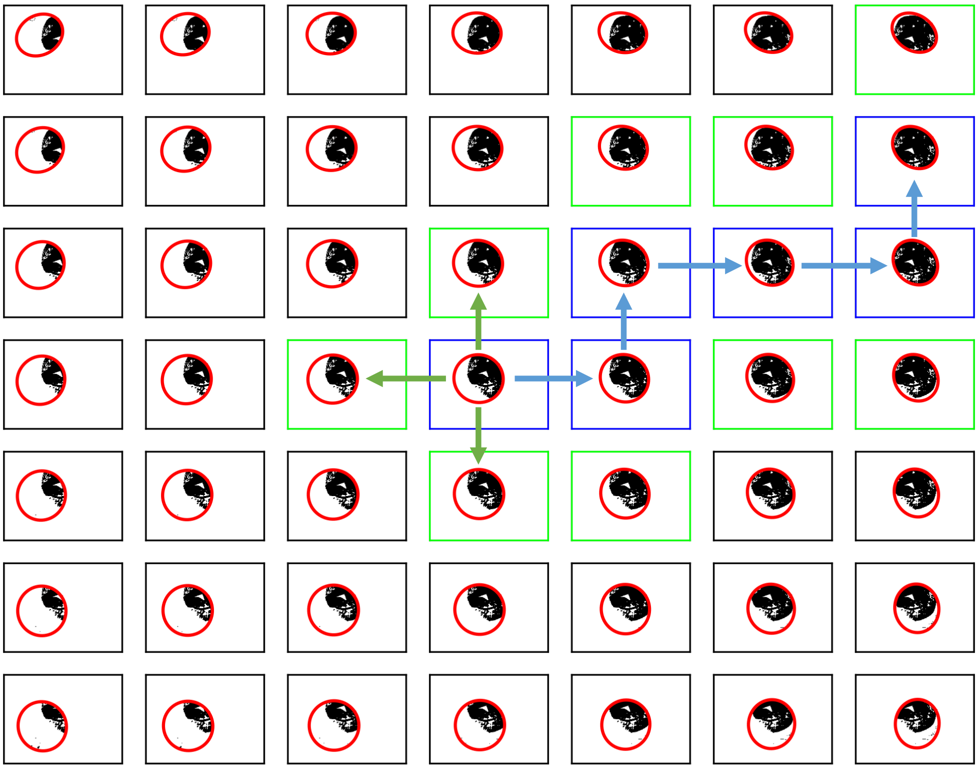


Figure 17. Greedy algorithm moving toward a local minimum. Blue: selected, green: computed, black: ignored.

5.2 Tracking Results

To assess the precision of the method, a user is asked to sit in front of a computer screen and look at 9 different targets displayed at each corner while staying still, as shown in Figure 18. The camera position relative to the screen is measured and assumed to be constant. The visual axis direction is computed from the measurements and compared with the one given by the implementation of the proposed method.

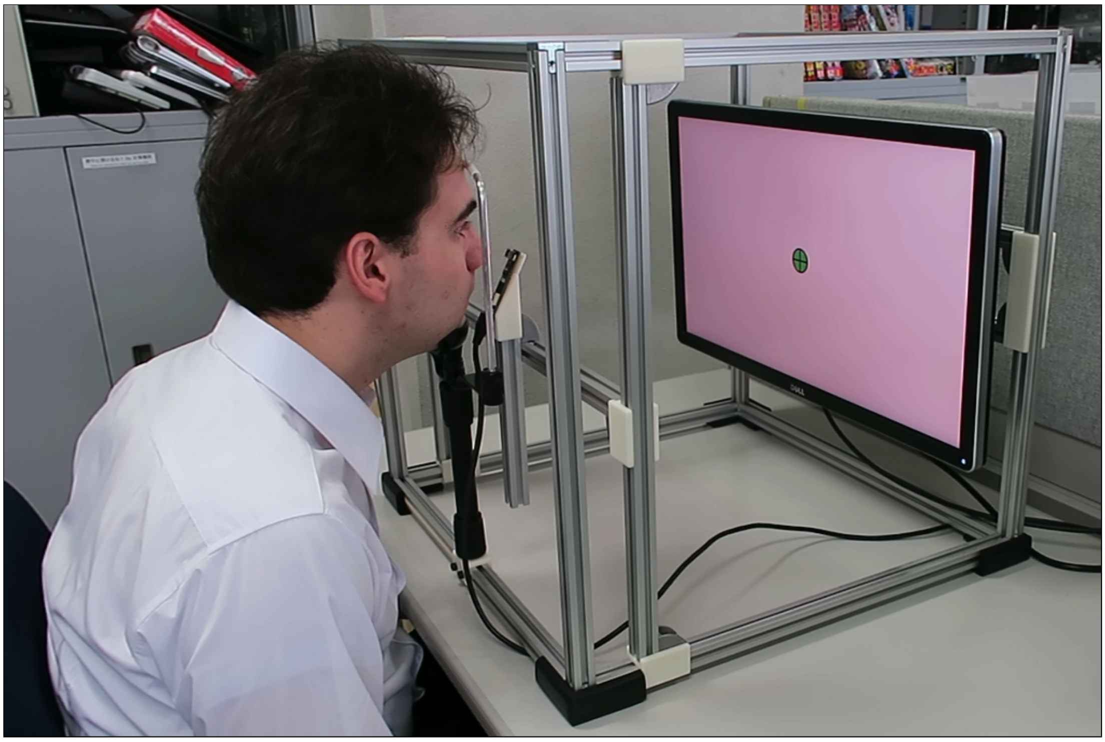


Figure 18. Experiment setup used to evaluate the precision of the gaze tracking.

Results obtained with a screen placed at 500 mm indicate a mean angular error of 2.5° and a maximum angular error of 5°, as shown in Figure 19. A drop of accuracy is observed as the eye looks toward the outside of the screen. This drop was expected as the occlusions from the eyelids are more prevalent when the eye approaches its maximal range of motion. However, although this drop is significant, this situation does not often happen as humans usually compensate reachability by moving their head while keeping a straight gaze with their eyes resting around the centered position.

It is important to remember that most commercial IR-based solutions advertise higher precision, usually below 1° of angular error. However, they exclusively focus on gaze direction estimation and require additional sensors to extract the focused object from the scene. The proposed method makes up for the lower precision of the RGB-based tracking by simultaneously extracting the focused object on the corneal reflection. Ultimately, this information is regarded as more important than gaze precision alone.

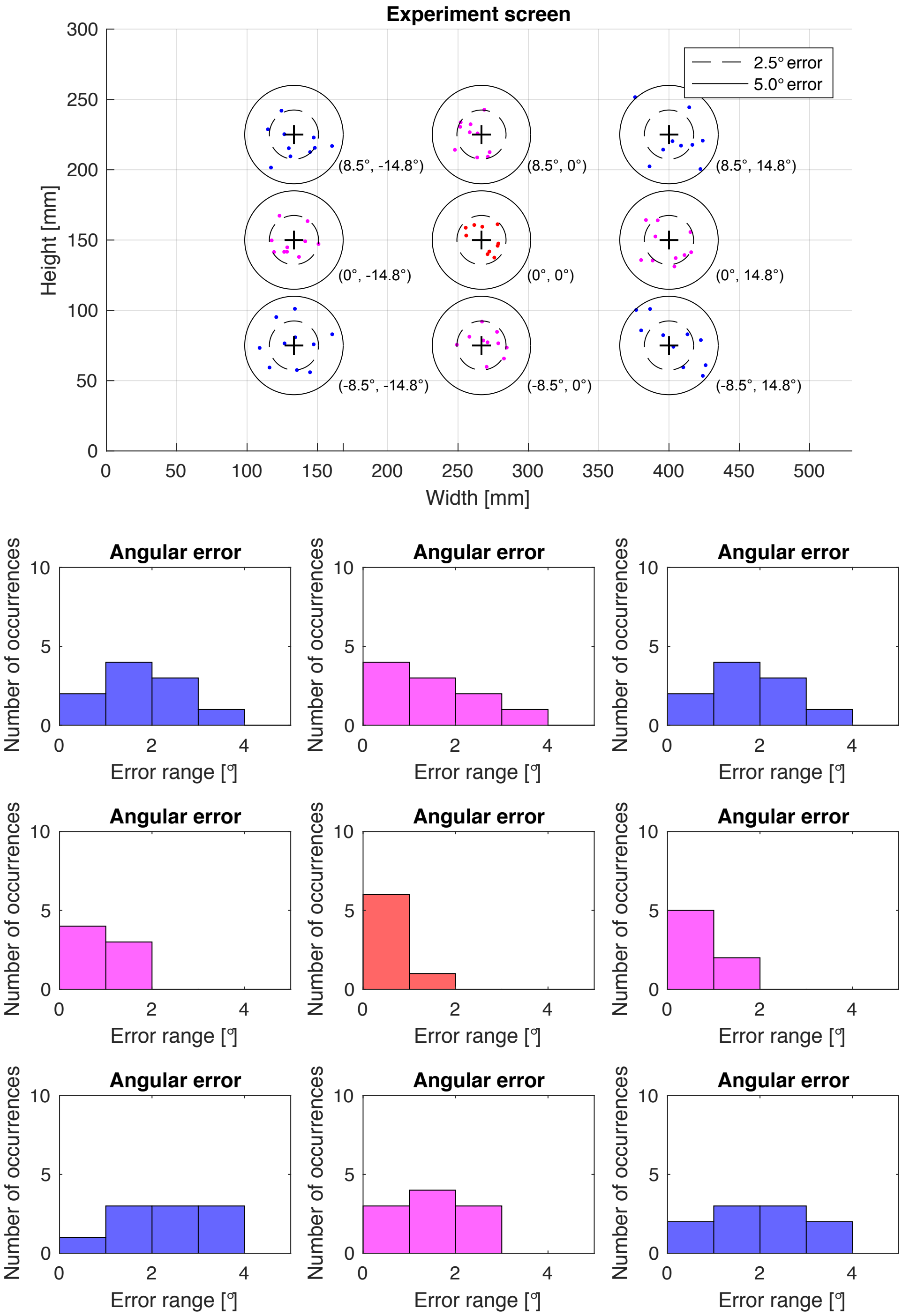


Figure 19. Gaze tracking results for 9 targets displayed on a screen with user sitting still at 500 mm. Pitch and yaw values are shown inside the parentheses.

6. Object Recognition

Preliminary works of this thesis [4, 5] proposed a method to match the 2D features between the corneal images and a reference image using a combination of feature-based algorithms. Unfortunately, the recognition suffered from noise due to iris contamination and spherical distortions, could barely run in real time, and results were prone to mismatches. Figure 20 illustrates these challenges. It was concluded that directly using 2D features on highly-distorted eye images did not yield useful results.

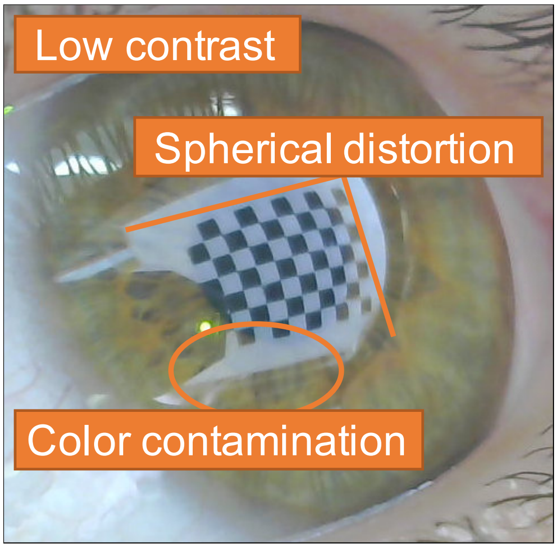


Figure 20. Typical eye image environment with overall low contrast, noise due to iris contamination, and spherical distortions.

From this previous experience, further investigations were conducted in [6] while moving toward deep learning strategies to address both the contamination and distortion issues. Based on the framework proposed by Donahue *et al.* [20], Transfer Learning was used to retrain Google Inception-v3 with new datasets obtained from eye images. Although the training was fast and the recognition accurate, the solution could not be used in real time.

This section first introduces more context by summarizing the previous unsuccessful trials mentioned above. Then, both the accuracy and real-time issues are addressed by training a new model based on You Only Look Once (YOLO), a state-of-the-art convolutional neural network (CNN) initially proposed by Redmon *et al.* [21] and improved by Redmon and Farhadi [22].

6.1 Feature Matching

The first explored idea consisted of matching 2D features between the corneal images and a reference image as follows:

1. First, detecting features and extract descriptors using Speeded-Up Robust Features (SURF), as described by Bay *et al.* [23].

2. Next, matching the descriptor vectors using Fast Library for Approximate Nearest Neighbors (FLANN), as proposed by Muja and Lowe [24].

3. Then, removing outliers using both Random Sample Consensus (RANSAC), Fischler and Bolles [25], and the gaze information obtained with the previously proposed method: incorrect matches outside an arbitrary-delimited area around the reflected POR on the cornea are eliminated.

4. Finally, a bounding box is returned by computing a homography from the filtered matches.

An overview is given by Figure 21.

Figure 22 shows a typical result obtained with this method while Table 3 details the average detection rates for 10 benchmark sequences, of 100 frames each, containing the same reference object as Figure 22. The resulted detection rate is low and the feature-based recognition suffers from multiple issues:

- The noise due to iris contamination and the distortions of the reflection limit the number of correct matches. Even though incorrect matches can be filtered out knowing the POR, further strategies should be applied to increase the matching rate.

- The current implementation can barely run in real time using only one reference image. Increasing the number of reference images, thus the number of potentially recognized objects, will prevent the system to achieve real time in any daily-life situations were multiple-object recognition is required.

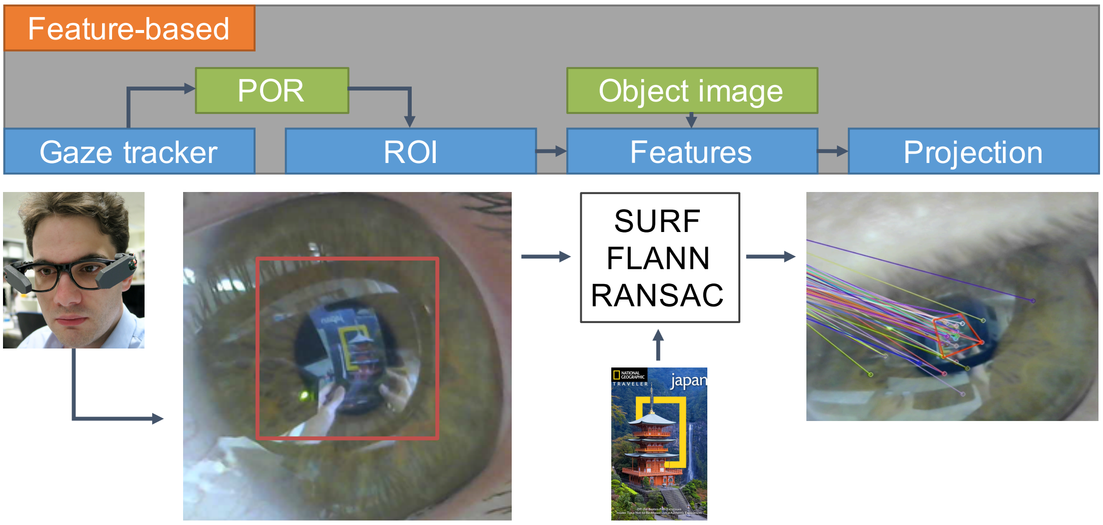


Figure 21. Overview of the object recognition using 2D features.

- One potential way to improve the detection rate would be to use the 3D model to unwrap/flatten the cornea reflections before applying the feature-matching algorithms. However, this operation is computationally intensive and will further prevent achieving real time.

It was concluded that directly using 2D features on highly-distorted eye images did not yield useful results.

Table 3. Average feature-based recognition rate with 100-frame benchmark sequences.

Benchmark sequence	True detection (avg)	No detection (avg)	False detection (avg)	Recognition rate (avg)
Reference book	15.4	5.7	78.9	15.4%

6.2 Transfer Learning

From the previous feature-matching experience, moving toward deep learning has been proven essential. There are two main reasons behind this move:

1. First, the image reflected on the cornea is highly distorted by its shape and contaminated by the iris. Therefore, it is hard to determine which

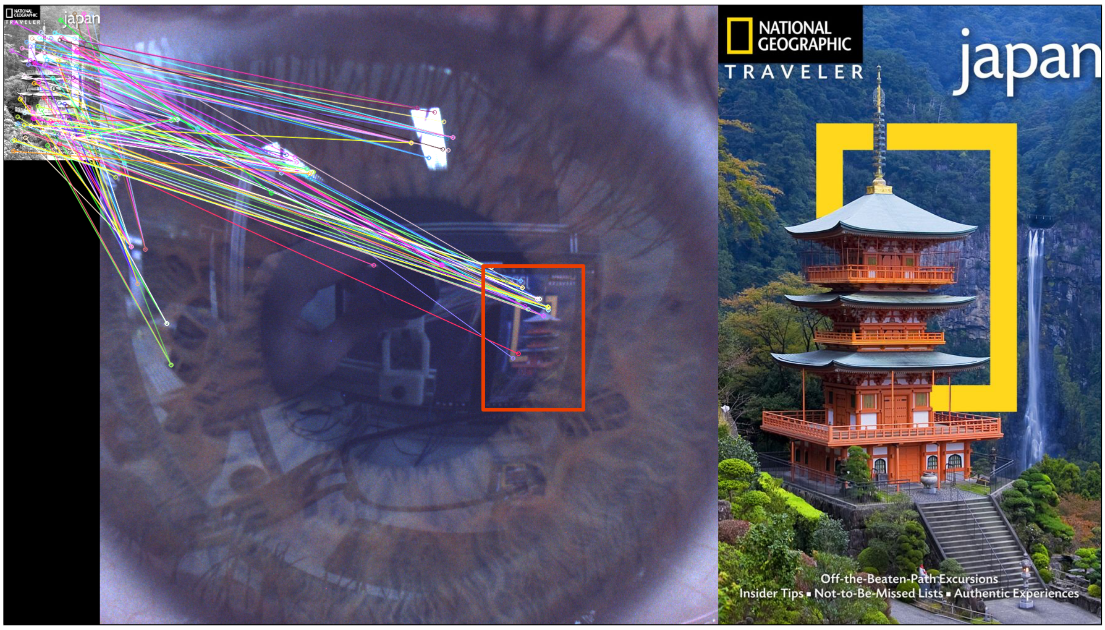

NATIONAL GEOGRAPHIC
TRAVELER
japan
Off-the-Beaten-Path Excursions
Insider Tips ▪ Not-to-Be-Missed Lists ▪ Authentic Experiences

Figure 22. Matching 2D features between the corneal image and a reference image with SURF, FLANN, and RANSAC. Incorrect matches outside the red box are filtered out using the gaze information.

2D features are important to achieve a successful recognition. Using deep learning, the machine will select the most relevant features by itself during the learning phase, if provided with enough corneal image data.

2. Second, although deep learning requires tremendous amounts of data to obtain good prediction results from the training, the gaze information obtained previously can be leveraged to automatically generate annotated data using the wearable prototype.

Instead of training a new model from scratch, it was decided first to use Transfer Learning to retrain Google Inception-v3, Szegedy *et al.* [26], with new datasets obtained from eye images. This open-source model was initially trained for the ImageNet Large Visual Recognition Challenge, Russakovsky *et al.* [27], to classify entire image contents into 1000 classes. By using Transfer Learning, the training process can be greatly accelerated by taking the fully-trained Inception-v3 for a set of common objects and retrain from the existing weights for new classes of eye images. Following this idea, only the final layer is retrained from scratch, while leaving all the others untouched, as suggested by [20]. An overview is given by Figure 23.

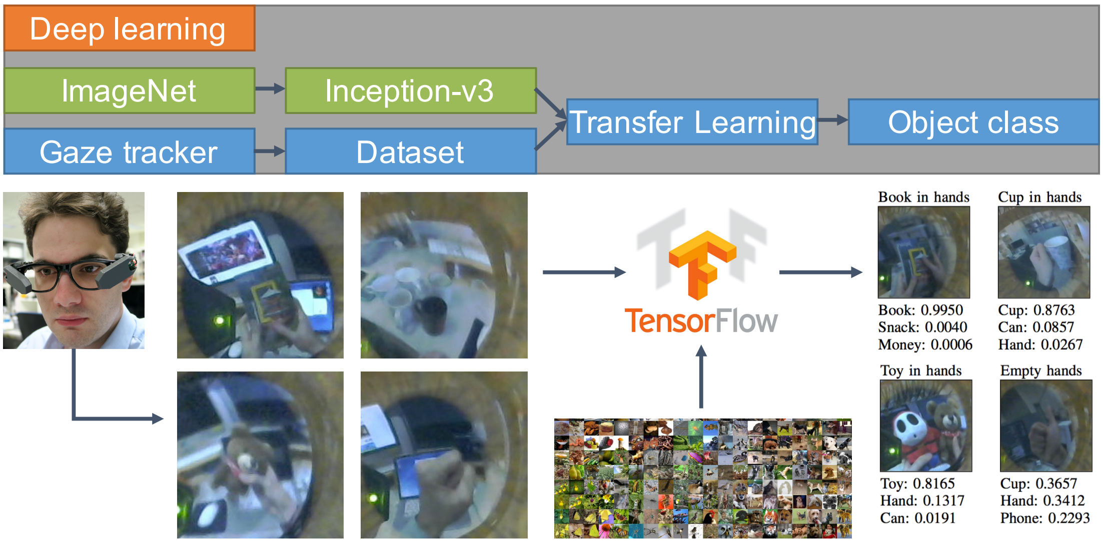


Figure 23. Overview of the object recognition using Transfer Learning.

6.2.1 Dataset Generation and Training

Deep learning requires large amounts of data to obtain good prediction results from the training, but creating and/or gathering these data is usually very time-consuming. However, gaze information can be leveraged to make the whole process more efficient.

Using the wearable prototype, a user is asked to look at several types of daily-life objects while manipulating them during an experiment. Cornea images are then automatically cropped with a 200×200 pixel area around the reflected POR using the proposed gaze tracking method. During this experiment, at least 1000 images for each object of 10 different classes were generated: books, cups, toys, screens, pens, money, phones, cans, snacks, and hands. Some data samples are shown in Figure 24. Since the objects were manipulated one by one, it is very easy to assign the whole cropped images of a sequence to their corresponding labels, *i.e.* annotate the images, among the 10 different ones.

The proposed solution is implemented with Google TensorFlow, Abadi *et al.* [16], and runs on a machine equipped with a CUDA-enabled GPU. The learning process takes about 30 min for a dataset of 10 000 images (1000 per object type) with 4000 training steps. Each step chooses 10 images at random from the training dataset and feeds them into the final layer to get predictions. Those predictions are then compared against the actual labels to update the final layer weights through the back-propagation process. A final accuracy evaluation is run on a set of images kept separate from the training and validation images.

6.2.2 Results and Discussion

Table 4 shows the results obtained with the four object types described in Figure 24. For each test image, the three highest predictions are returned. The accuracy exceeds 80% in the cases of object manipulation, but dramatically falls around 35% for empty hands. However, since hands are present on every images, this case is the most challenging and low accuracy was expected.

Transfer Learning improved the accuracy of the recognition compared to the feature-based method, but the solution still suffers from issues that make it impractical for daily-life applications:

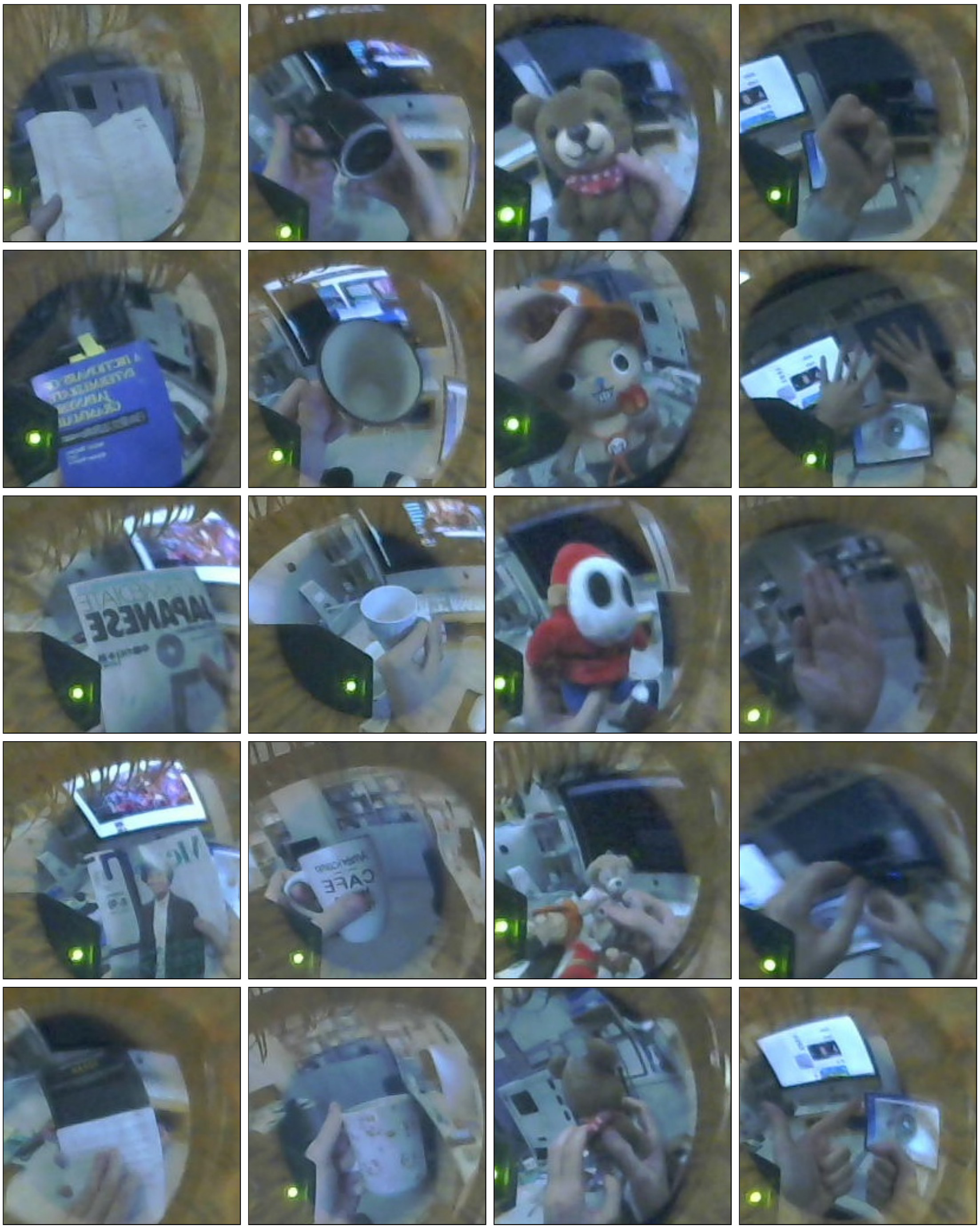
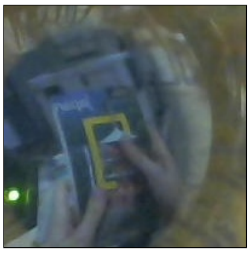
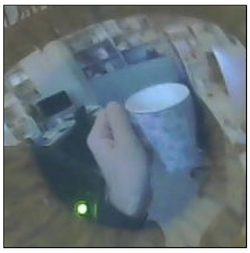
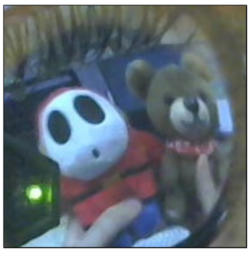
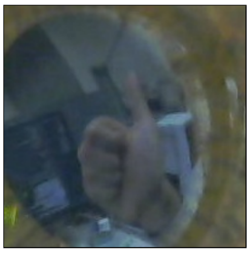


Figure 24. Data samples for books, cups, toys and hands (cropped and centered around the POR).

Table 4. Transfer Learning results with 4 benchmark images, each of a different object type.

Book in hands	Cup in hands	Toy in hands	Empty hands
			
Book: 0.9950	Cup: 0.8763	Toy: 0.8165	Cup: 0.3657
Snack: 0.0040	Can: 0.0857	Hand: 0.1317	Hand: 0.3412
Money: 0.0006	Hand: 0.0267	Can: 0.0191	Phone: 0.2293

- The whole cornea image is recognized a one item of the training datasets, thus multiple-object detection with precise bounding boxes around the focused objects is impossible.

- The detection runs below 0.4 fps and is far from being real time. However, contrary to the previous feature-based method, multiplying the number of trained objects does not affect the performance at run time.

- Although adjusting the final layers of a model trained against ImageNet using Transfer Learning enables fast and convenient retraining, it is questionable to not use cornea images from the start to train a new model more adapted to eye images.

From these conclusions, it appeared that training a new model from scratch using a state-of-the-art CNN could address both the accuracy and real-time issues.

6.3 Real-Time Recognition

To tackle the real-time issues of the previous methods, a new model is trained based on the state-of-the-art CNN called You Only Look Once (YOLO), initially proposed by Redmon *et al.* [21] and improved by Redmon and Farhadi [22]. An overview is given by Figure 25.

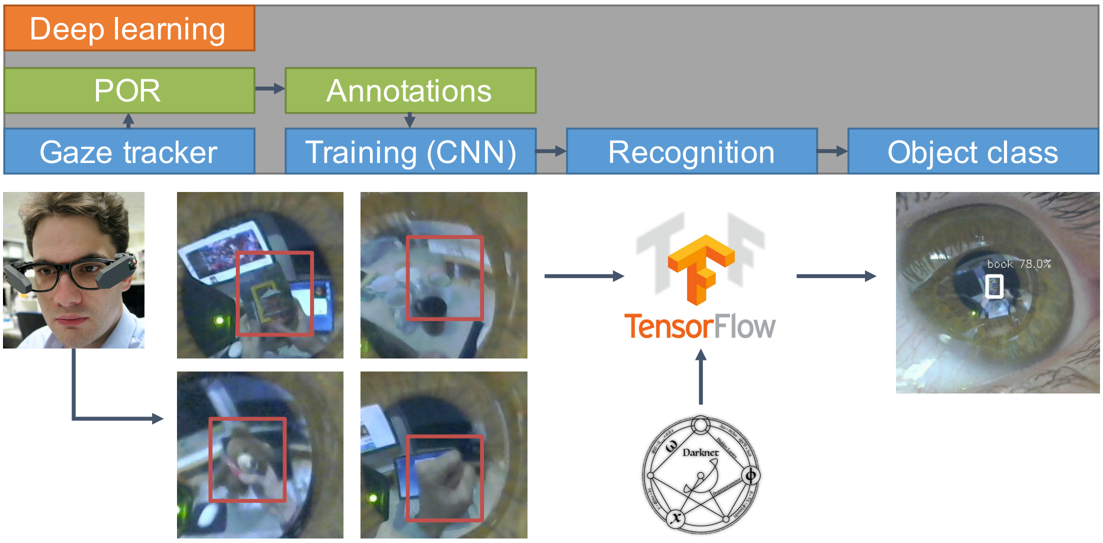


Figure 25. Overview of the object recognition using YOLO.

By using YOLO, only a single neural network is applied to the full eye image. The network divides the image into regions and predicts bounding boxes and probabilities for each region. These bounding boxes are weighted by the predicted probabilities for each class of objects.

At run time, the model looks at the whole image and makes predictions with a single-network evaluation. This makes it extremely fast, more than 100× faster [22] than the more conventional Fast R-CNN, Girshick [28], which requires multiple networks for a single image. In the proposed system, the YOLO detection runs above 70 fps, much faster than the 30 fps camera input.

6.3.1 Dataset Generation and Training

Using the wearable prototype, a user is asked to look at several variations of daily-life objects while manipulating them. Each object belongs to one of the following 10 classes: books, cups, toys, screens, pens, money, phones, cans, snacks, and hands. By using the proposed gaze tracking method, a 100 × 100 pixel area around the reflected POR is then automatically annotated in all the corneal images to generate the training datasets, as shown in Figure 26 with books. Contrary to the previous method using Transfer Learning, only the bounding box around the focused object at the POR is annotated, thus avoiding annotating whole cropped images that includes irrelevant background information.

The model is then trained to perform real-time, in-hands object recognition. The graph of the complete network is shown in Figure 27. Using the wearable prototype, at least 1000 images for each object variation, belonging to one of the 10 classes, were generated in 5 different locations to reduce the influence of the illumination conditions during the training. The proposed method using YOLO is then run against at least 5 benchmark videos, of 100 frames each, where a user manipulates the objects. In order to validate the robustness of the system, several experiments have been conducted in various environments and including objects that, although belonging to one of the 10 experimental classes, were not used to generate data and thus not trained.

6.3.2 Results and Discussion

The following section discusses the results obtained by training new models using YOLO. Several experiments have been conducted to validate different aspects of the recognition. In particular, the discussion compares the successful recognitions between:

- Trained and untrained objects in a same environment for a specific class.
- Trained and untrained environments for the same objects of a specific class.
- Learning-based and feature-based methods for a trained object in a same environment.
- Manual, human-made annotations and automatic, POR-based annotations for a trained object in a same environment.
- Calibrated, visual axis-based annotations and uncalibrated, optical axis-based annotations for a trained object in a same environment.
- Finally, 10 classes of trained and untrained objects in a same environment.

In all experiments, at least 5 trials (or benchmark videos), of minimum 100 frames, were used for each environment or object. The results presented hereafter are the average successful recognitions across these trials.

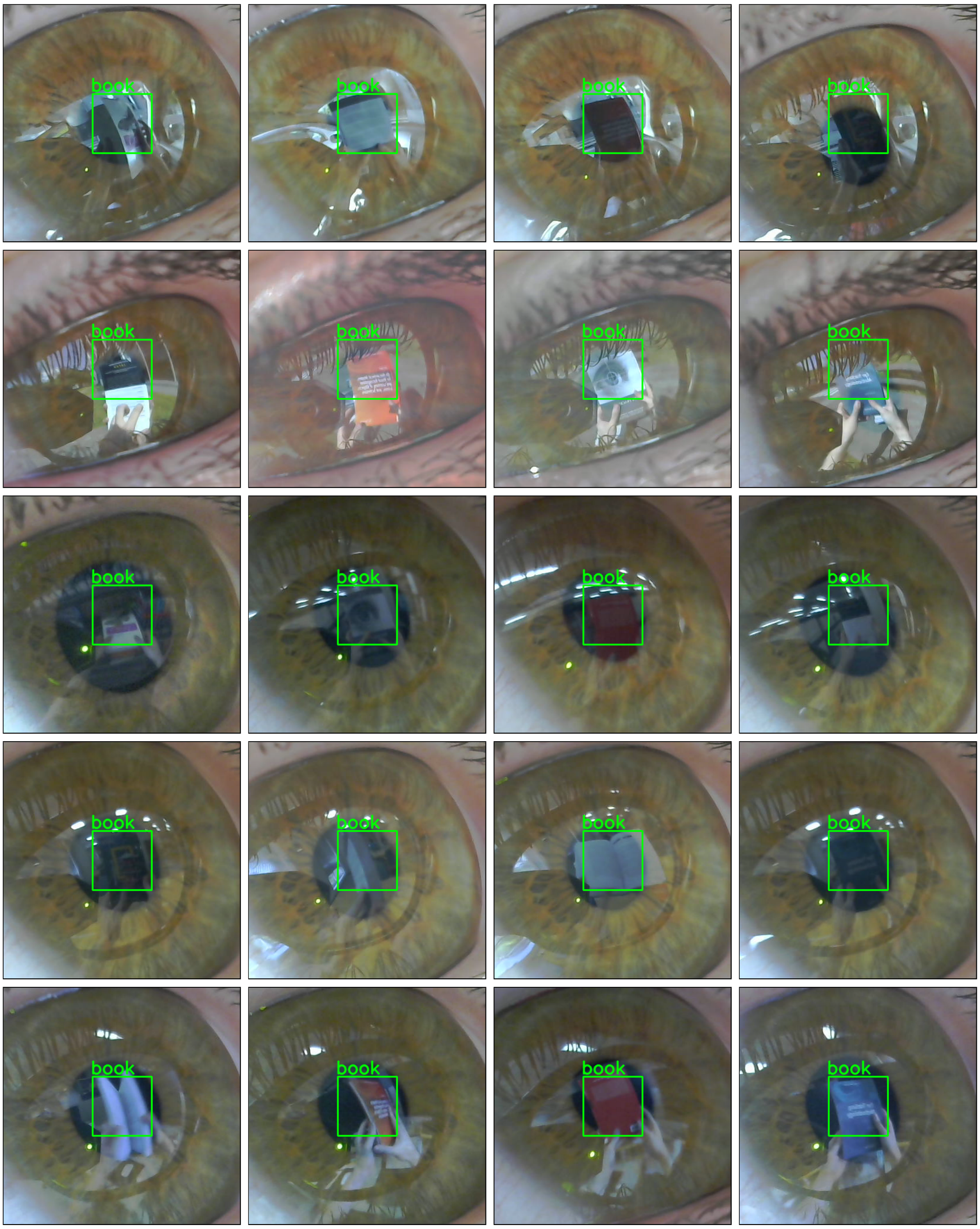


Figure 26. Data samples for book annotation (cropped and centered around the POR).

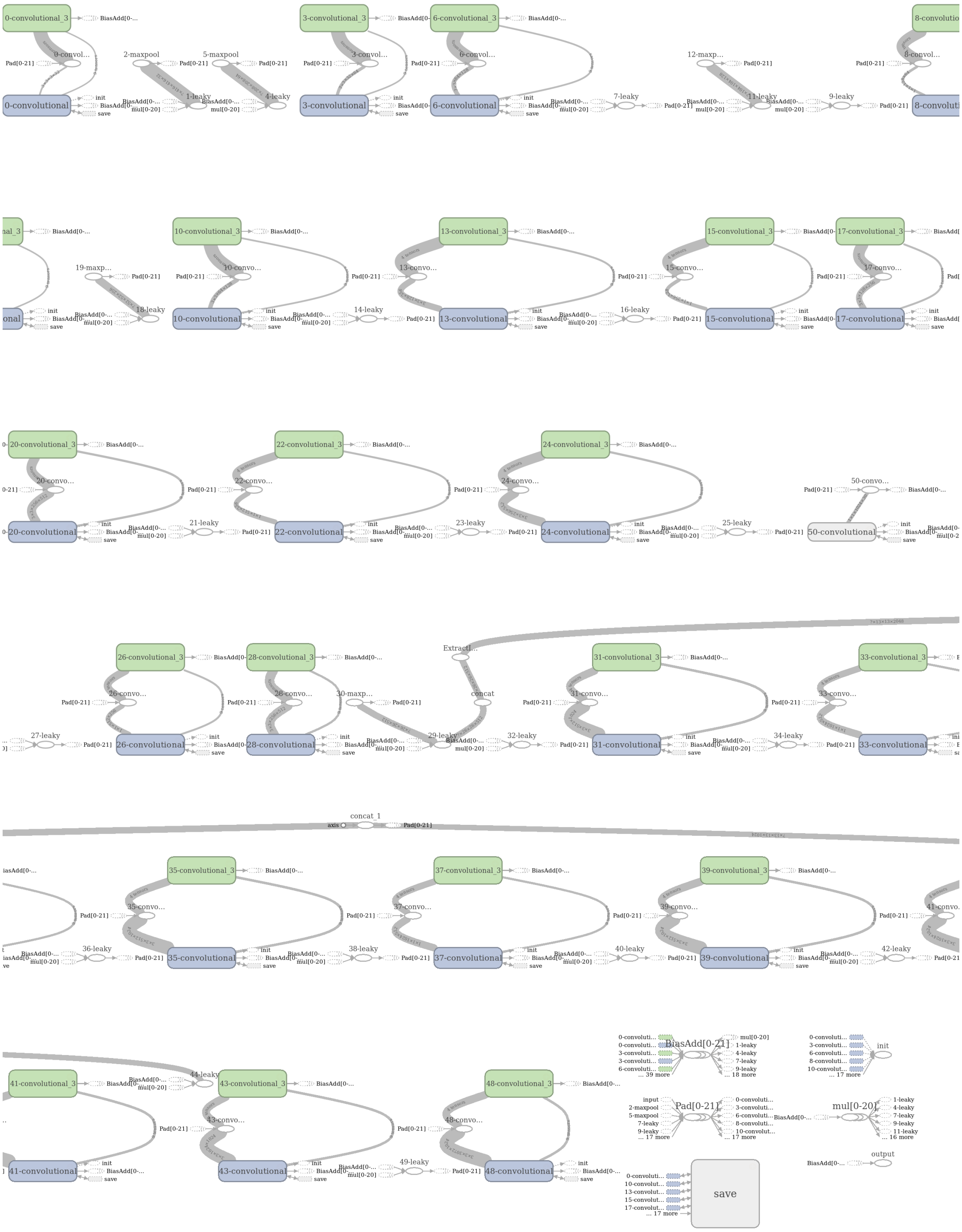


Figure 27. Graph of the complete YOLO network used for training.

Trained versus Untrained Objects This experiment compares the recognition rates of trained and untrained objects for a specific class, books, in a same environment. The aim is to validate the robustness of the system against not only trained objects of a specific class, but also against untrained objects of the same class.

In the experiment, 10 variations of books were used to train the model. Then, a user was presented with both a subset of the trained books and a set new books, not used for training. For each object, at least 5 trials were performed. An example of trial results for a trained object is given in Figure 29, where one out of every 10 frames are shown for readability. The average successful recognitions are summarized in Figure 28.

The results show strong recognition rates for the trained objects. In the case of untrained objects, a significant drop is observed. However, untrained objects can still be recognized in more than half the total number of frames in average.

Trained versus Untrained Environments This experiment compares the recognition rates in trained and untrained environments with trained objects of specific class: books. The aim is to validate the robustness of the system across different environments and illumination conditions, and especially untrained ones.

In the experiment, 10 variations of books were used to train the model, but all the data were captured in the same indoor environment: a laboratory. Then, a user was asked to manipulate the objects in 4 different locations under various illuminations, such as artificial lighting, direct and indirect sunlight. For each environment, at least 5 trials were performed. An example of trial results for 3 different locations is given in Figure 31, where one out of every 10 frames are shown for readability. The average successful recognitions are summarized in Figure 30.

The results show a strong recognition rate for the trained environment. Although a drop is observed in the case of untrained locations and illuminations, the system performs robustly and the objects are successfully recognized in more than two thirds of the total number of frames used during the trials in average.

Learning-Based versus Feature-Based Methods Figure 32 shows the comparison between the previously described feature-based method, using SURF, and

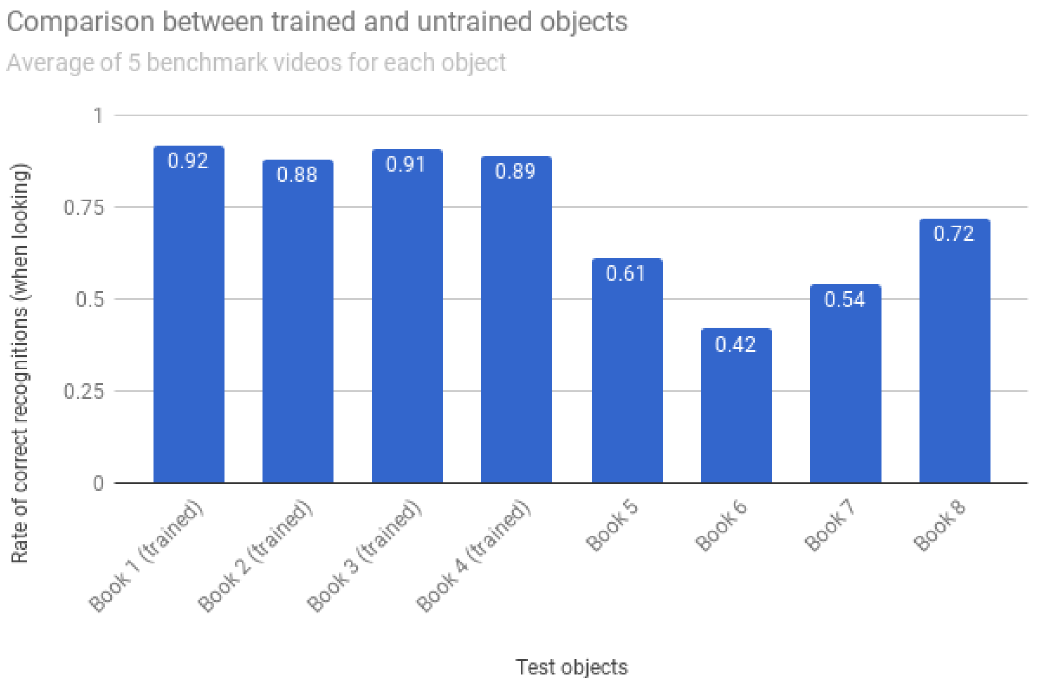


Figure 28. Comparison of successful recognitions between trained and untrained objects in a same environment for a specific class (books). Average of 5 trials for each object.

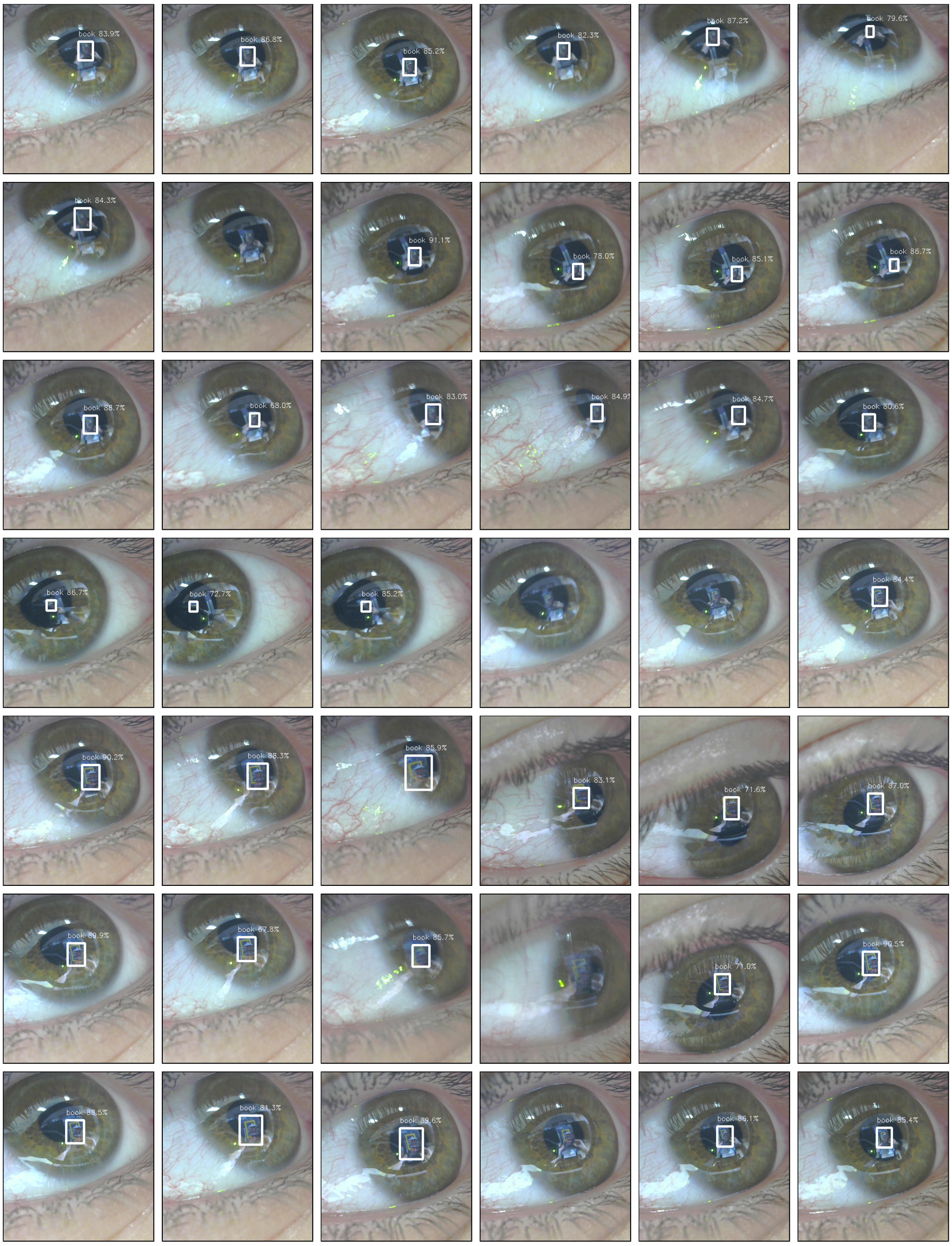


Figure 29. Example of trial results for a trained object. One out of every 10 frames are shown for readability.

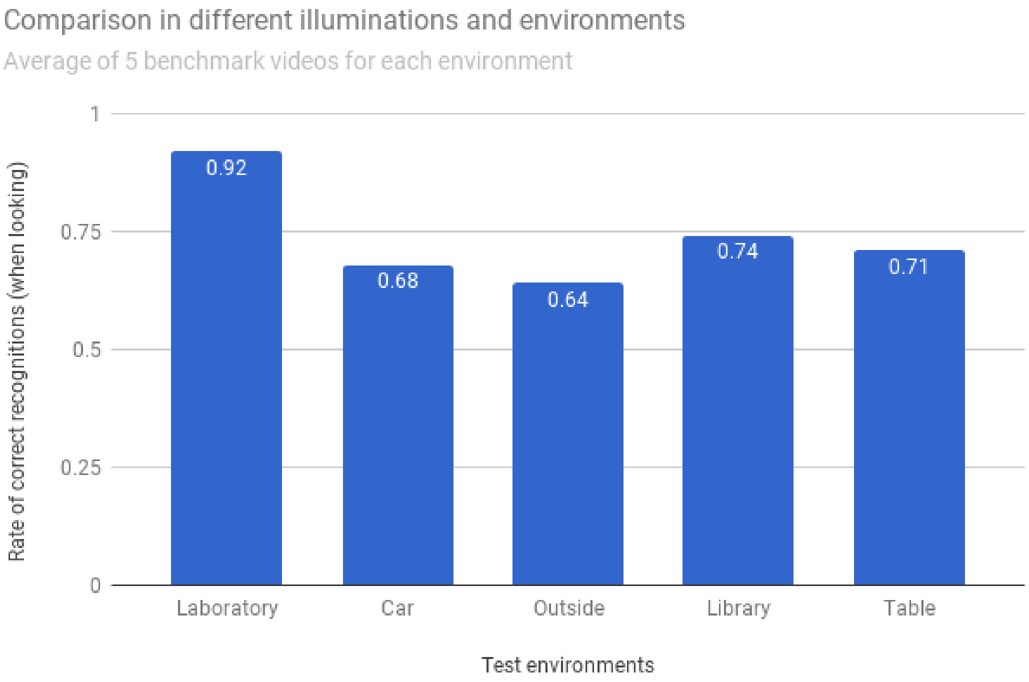


Figure 30. Comparison of successful recognitions between trained and untrained environments for the same objects of a specific class (books). Average of 5 trials for each environment.

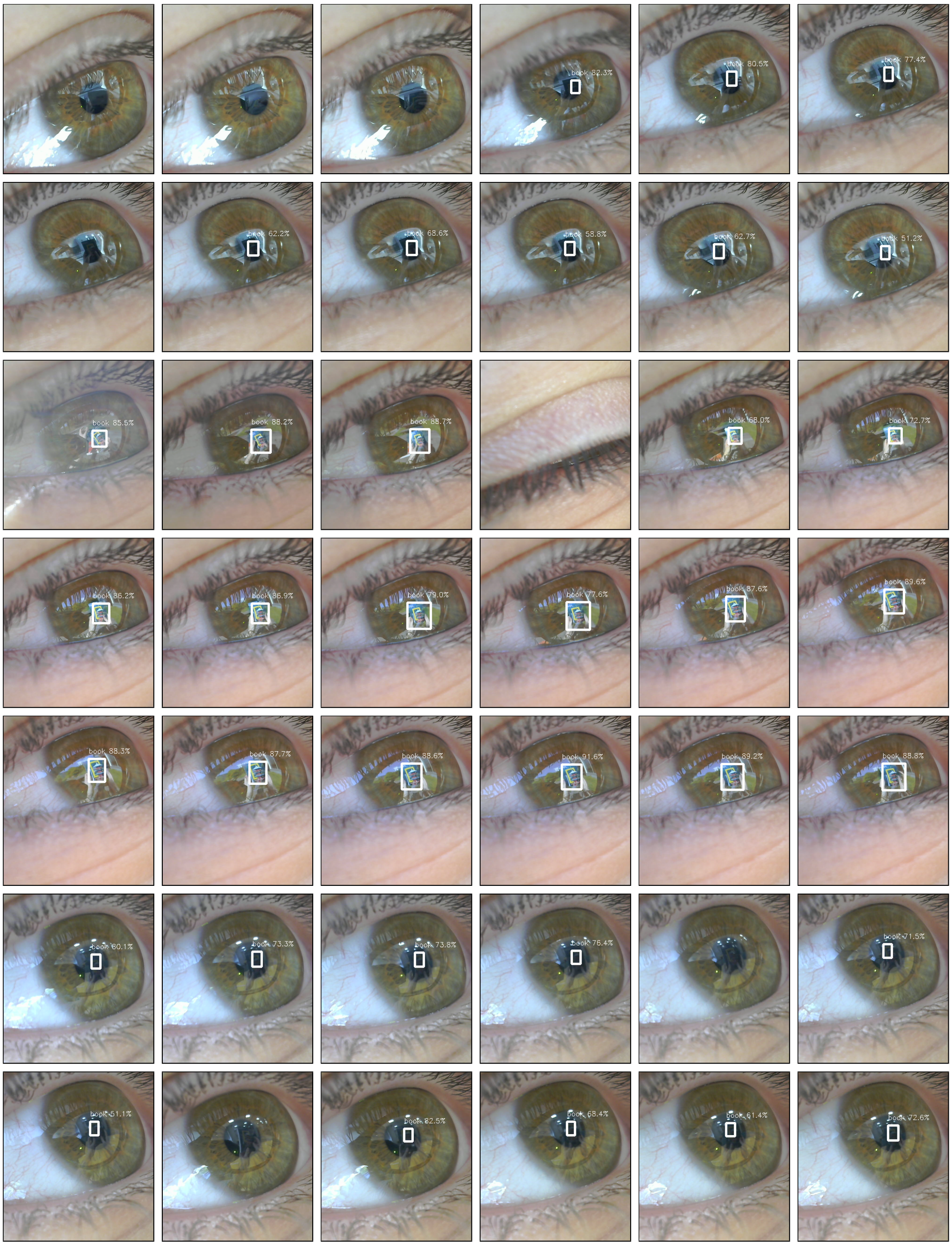


Figure 31. Example of trial results in untrained environments. One out of every 10 frames are shown for readability.

the proposed learning-based method using YOLO. The average results obtained with 5 trial using the same object, a book, show that the learning-based method presents 5 times more successful recognitions than the feature-based one, thus making deep learning essential in object recognition from corneal imaging.

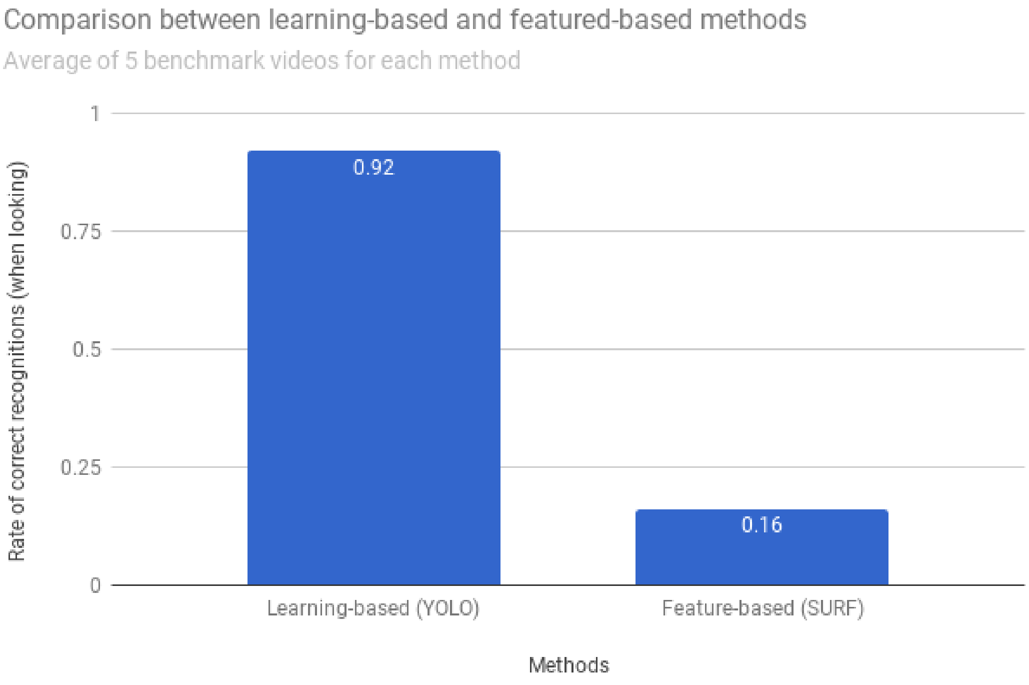


Figure 32. Comparison of successful recognitions between learning-based and feature-based methods. Average of 5 trials for each method.

Automatic versus Manual Annotations Figure 33 shows the comparison between the proposed POR-based annotations and manually annotated data to generate the datasets. The manually annotated data, more time-consuming but also more precise than the proposed automatic POR-based method, can be seen as an upper boundary of the expected recognition rates using YOLO.

The results obtained using the same trained object, a book, during the same benchmark trial using both the automatic and manual annotated datasets for the training, show that the automatic POR-based annotation method experiences less than 10% drop in accuracy. Moreover, the model using manual annotations achieved almost 100% of successful recognitions. This important result means that the proposed method can potentially approach perfect recognition rates by further refining the automatic POR-based annotation method.

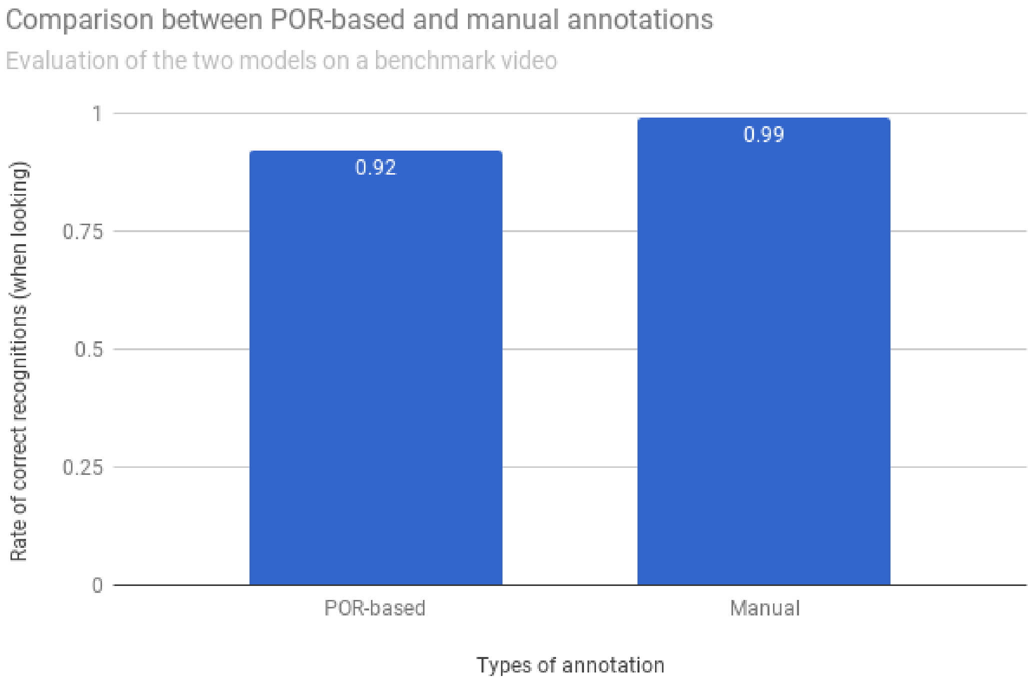


Figure 33. Comparison of successful recognitions between POR-based and manual annotations. Evaluation of the two models in a single identical trial.

Calibrated versus Uncalibrated Annotations Figure 34 shows the comparison between a calibrated system using the visual axis to compute the POR for automatic annotation and dataset generation, and an uncalibrated system using the optical axis as a first approximation of the visual axis.

The results obtained using the same trained object, a book, during the same benchmark trial, show that there is very little difference between the two methods. It appears that the calibration of the visual axis does not influence the focused object recognition. This interesting result means that the visual axis calibration can be skipped if the focused object analysis is ultimately more important than the gaze direction precision alone.

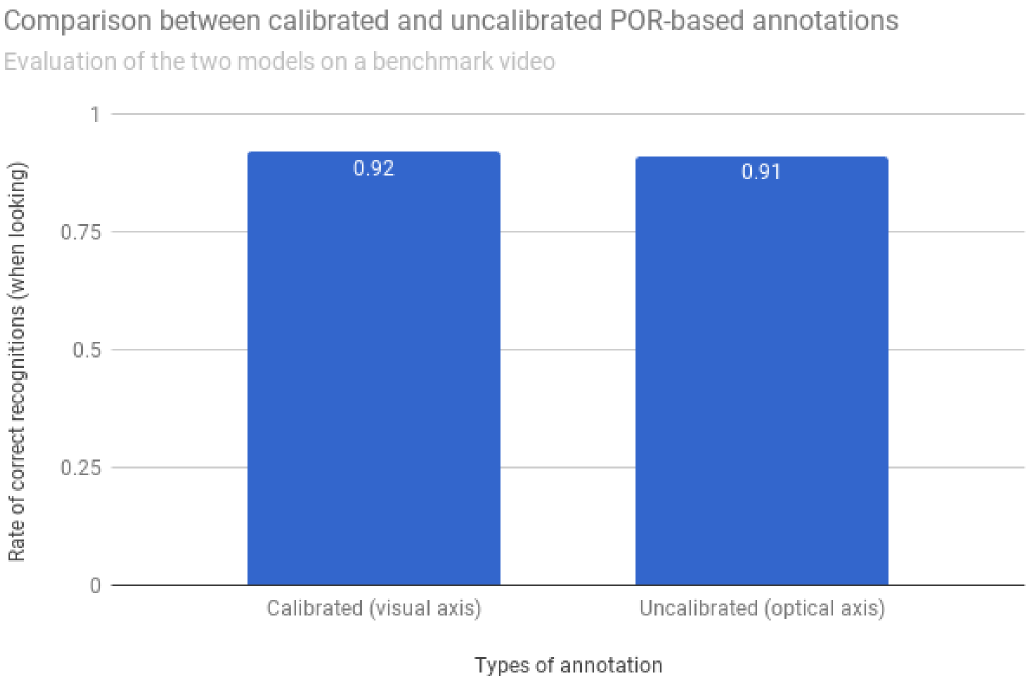


Figure 34. Comparison of successful recognitions between caloibrated (visual axis) and uncalibrated (optical axis) annotations. Evaluation of the two models in a single identical trial.

Comparison Between 10 Classes Finally, Figure 35 and Figure 36 show the average results obtained with 10 classes of trained and untrained objects, respectively, in a same environment. At least 10 variations of each class of objects, books, cups, toys, screens, pens, money, phones, cans, snacks, and hands, were used to train the model, and 5 trials were performed for each class.

All classes of objects experience a similar drop in successful recognitions when tested against untrained objects. However, untrained objects can still be recognized in more than half of the total number of frames in average, thus demonstrating the robustness of the system with several kinds of daily-life objects.

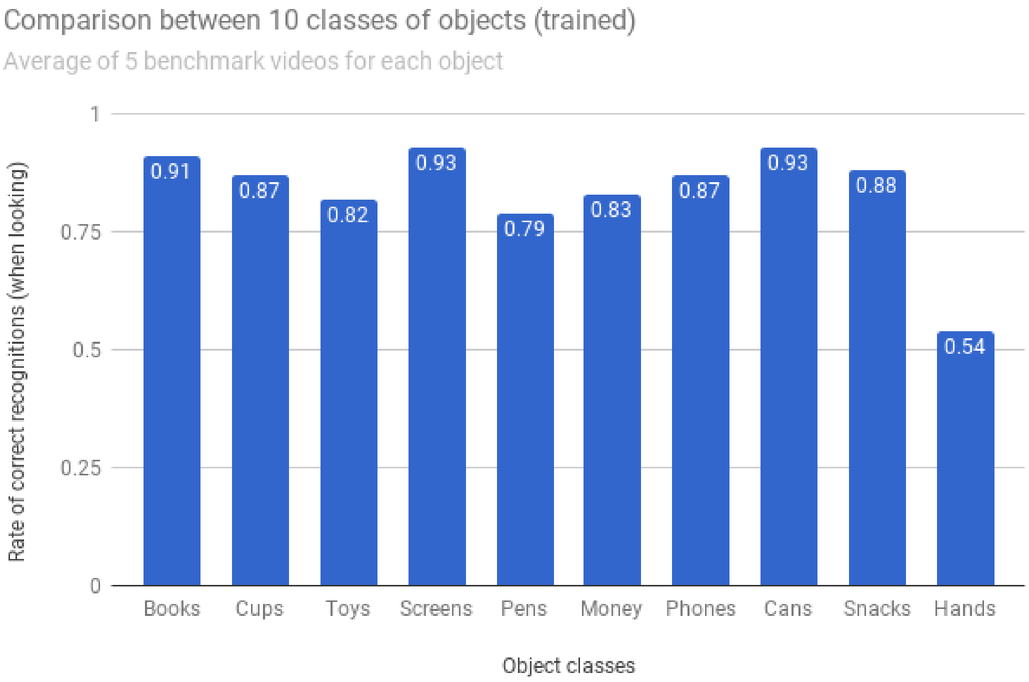


Figure 35. Comparison of successful recognitions between 10 classes of trained objects in a same environment. Average of 5 trials for each object.

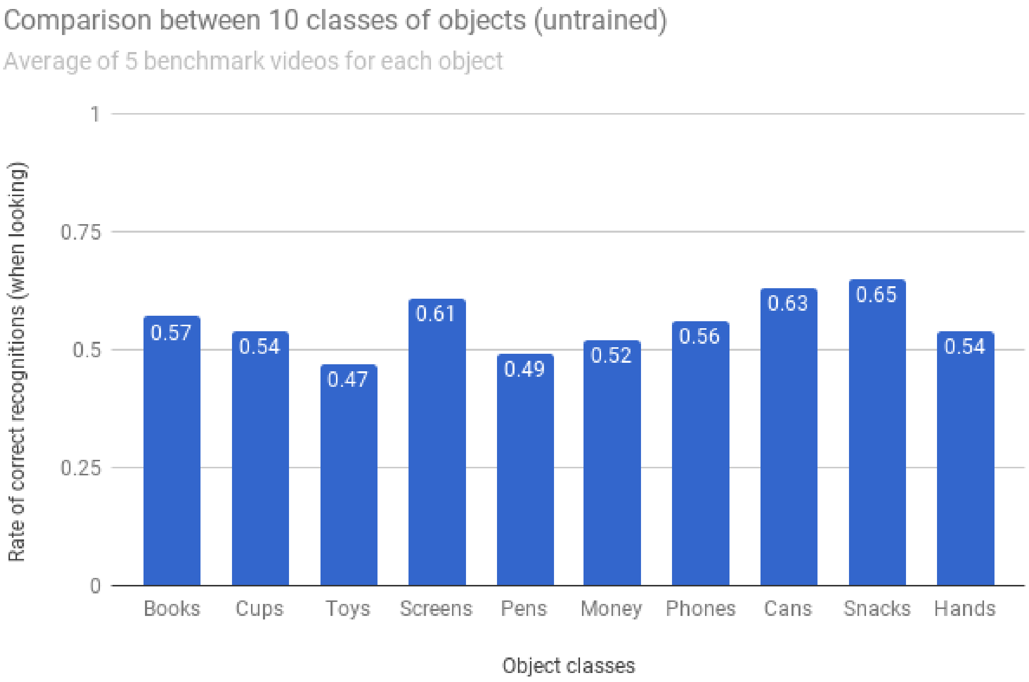


Figure 36. Comparison of successful recognitions between 10 classes of untrained objects in a same environment. Average of 5 trials for each object.

Although the recognition rates are in overall lower for untrained objects and environments, these results show that gaze tracking and focused object recognition can be performed in real time with sufficient precision for human behavior analysis in daily life. Plus, increasing the size of the datasets overtime can greatly improve the results for both trained and untrained objects. Finally, deep learning as been proven essential over feature-based methods, and the model trained using YOLO can potentially achieve almost perfect recognition if given high-quality annotated data.

7. High Dynamic Range

Corneal images are highly sensitive to lighting conditions. Direct sunlight and low-light conditions are particularly challenging for standard RGB cameras:

- On the one hand, low-light conditions require a high-sensitivity camera to capture clear eye images which rapidly saturates under normal illumination.
- On the other hand, corneal reflections present a high dynamic under daylight with some reflected sunlight that can also saturate the sensor, thus obscuring the scene features.

An example of high dynamic is given in Figure 37.

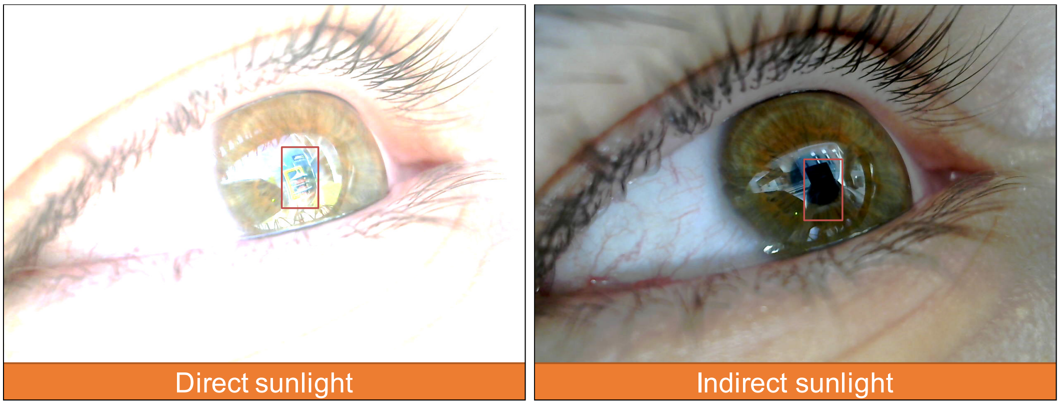


Figure 37. Example of high dynamic: direct sunlight while reading outside (left), indirect sunlight while reading inside a car (right). Details of the focused object are lost in both case.

To address these issues, experiments using a ViewPLUS Xviii high-sensitivity camera and HDR processing were conducted. The Xviii camera captures 11 8-bit RGB images at increasing levels of sensitivity over an 18-bit dynamic range. These images are combined using a HDR algorithm, Exposure Fusion by Mertens *et al.* [29], to reveal the features reflected on the cornea in varied illumination conditions. Figure 38 shows an example at night time.

To evaluate the gain offered by using HDR in very low-light conditions, the YOLO-based recognition rate of two 100-frame benchmark sequences are compared in Table 5. The two sequences were captured simultaneously using the

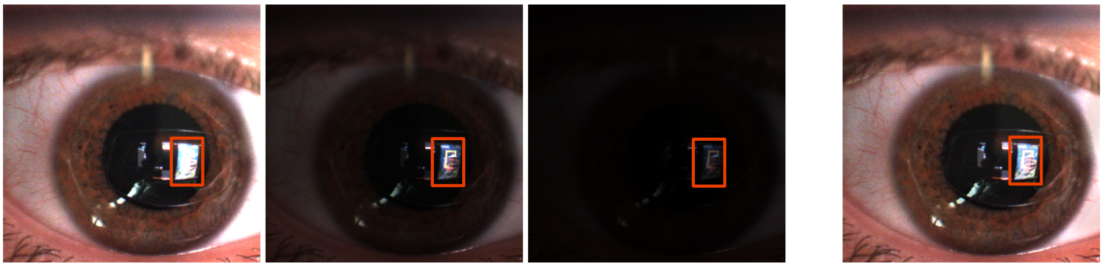


Figure 38. Combining 3 frames obtained from the HDR camera (left) with Exposure Fusion to reveal the full dynamic of the corneal image at night time (right).

Xviii camera at night time while manipulating a book: one sequence have been processed with HDR to reveal the full dynamic of the cornea while the other is extracted from the first 8 bits of the Xviii that correspond to the lowest sensitivity of the 18-bit dynamic of the camera, thus closely simulating the kind of images obtained using a non-HDR camera at night.

Table 5. Comparison between HDR and non-HDR recognition rate at night time with two 100-frame benchmark sequences.

Benchmark sequence	True detection	No detection	False detection	Recognition rate
non-HDR	0	100	0	0%
HDR	24	71	5	24%

Without HDR processing, the corneal images are simply too dark to yield any results. It is also important to notice that the recognition rate using HDR is lower than the one using a non-HDR camera in a normal illumination environment. However, augmenting the training datasets with eye images processed by HDR could improve the results.

Using a high-sensitivity camera combined with HDR techniques allows to apply the proposed method independently of the illumination conditions, thus making the solution suited for application in daily-life activities. While the current size of the Xviii camera is too large for embedding it in a wearable prototype, the pace of progress in the miniaturization of both cameras and image sensors makes me hopeful that a wearable HDR camera will be available in the not too distant future.

8. Conclusion

This thesis proposed a wearable system to perform real-time, simultaneous eye tracking and focused object recognition for daily-life applications in varied illumination environments. For this, a model-based approach was described to estimate the gaze direction using a single RGB camera, and a method to recognize objects in the scene images reflected on the cornea in real time, without any additional sensors such as a front-facing camera. The automatic annotations using the reflected region of interest and the POR dramatically reduce the effort to create very large datasets. As a result, deep learning approaches could be easily applied to the corneal images to recognize the focused object. The experimental results showed that gaze tracking and focused object recognition can be performed in real time with sufficient precision in daily-life object manipulation, for both trained and untrained objects, in both trained and untrained environments. Finally, deep learning was proven essential over feature-based methods, and the model trained using YOLO can achieve very high recognition rates if given enough data.

8.1 Future Work

This thesis explored the two important aspects of gaze tracking and focused object recognition from eye images. However, these first steps are part of a broader vision for the future of corneal imaging: proposing a low-complexity, single-sensor approach to human attention, behavior and emotion analysis by combining all the features contained in eye images.

To reach this ultimate goal, it is necessary to investigate how to combine multiple eye information obtained from a single camera, such as the gaze direction, the corneal reflections, the pupil size variations, the peripheral vision, the eyelid movements and the eye saccades to create a full framework of human behavior analysis from eye images.

Achieving such enterprise requires further developments in many different areas, including but not limited to:

- Estimating human behavior and psychological state by tracking features such as the pupil size variations that indicate changes in illumination and/or concentration, the eye saccades that express unconscious behaviors, and the eyelid movements that give an estimation of the fatigue.

- Evaluating not only the direction of the gaze, but also the distance to the POR. This could be achieved with stereo reconstruction from left and right corneal images.

- Extracting the anatomical parameters of the user to improve accuracy, and removing the extra step of visual axis calibration.

- Improving further the wearable prototype by embedding HDR-enabled cameras as well as more computational power on the device.

- Scaling the method to non-wearable high-resolution cameras, such as dashboard cameras, web cameras or surveillance cameras (CCTV), which could remotely extract eye information.

8.2 Potential Applications

Potential applications are wide-ranging: daily-life support in attention activities, new AR/VR interfaces, intuitive driving assistance, eye disease or neurological disorder diagnosis, surveillance and forensics, and more. With the rapid development of imaging sensor technology, particularly in terms of reduced size and increased resolution, the same proposed method could also be applied to non-wearable high-resolution cameras which could remotely extract eye information. Taking all these aspects into consideration, the proposed method has wide application prospects with potentially important impact, as illustrated by the following examples:

- Developing the next generation of smart eyewear that could analyze the user's gaze and level of attention during daily-life activities, such as driving or manipulating machines. By using a single sensor, low production cost and power consumption are expected to drive rapid adoption in smart wearables. In addition, by eliminating the need for a front-facing camera, the proposed system will avoid the social concerns that were an obstacle for previous efforts in smart eyewear like the Google Glass.

- In the context of the next-generation AI-powered factories, the proposed research could be integrated into the workers' protective eyewear to alert them in the case of immediate danger due to inattention or fatigue, thus increasing efficiency and safety.

- The proposed method and device could also be used in human-robot interaction studies by assessing the human and robot visual self-experience in collaborative tasks or joint attention scenarios, thus enabling new research on human-robot safety.

- The ease with which eye data and visual feedback can be collected using the proposed device will enable researchers to build the large datasets that are required to further develop AI-based object recognition and human behavior analysis. In particular, eye movements could be analyzed to diagnose eye diseases or neurological disorders early.

By contributing with STARE to the first steps toward an integrated, single-sensor approach to human behavior analysis through eye images, the research conducted in this thesis will hopefully enable a new generation of smart eyewear and interfaces using natural eye-based controls.

Acknowledgements

First and foremost, I would like to deeply thank the Ministry of Education, Culture, Sports, Science and Technology of Japan for having granted me with the MEXT Scholarship to help me achieve my goals and realize my dreams.

Next, I would like to express my sincere gratitude to my supervisor, Professor Tsukasa Ogasawara, for his invitation to join the Robotics Laboratory of the Nara Institute of Science and Technology (NAIST) and his continuous support of my research. None of this would have been possible without his kind guidance, financial support, research facilities, and immense knowledge.

Besides Professor Tsukasa Ogasawara, I would like to thank the rest of my thesis committee: Professor Hirokazu Kato (NAIST), Associate Professor Jun Takamatsu (NAIST), Associate Professor Kentaro Takemura (Tokai University) and Assistant Professor Ming Ding (NAIST), not only for their encouragement, but also for their insightful comments which incented me to widen my research from various perspectives. In particular, I would like to express a special thank to Associate Professor Jun Takamatsu for his day-to-day supervision and mentorship.

I also gratefully acknowledge the support of Lecturer Atsutoshi Ikeda (Kindai University) who helped me starting this project, and Felix von Drigalski for his helpful corrections and proofreading during the redaction of this thesis.

Finally, I would like extend my thanks to my parents and closest ones: Merci d'avoir fait de moi l'homme que je suis devenu aujourd'hui.

References

[1] C. Nitschke, A. Nakazawa, and H. Takemura, “Corneal Imaging Revisited: An Overview of Corneal Reflection Analysis and Applications,” *IPSJ Transactions on Computer Vision and Applications*, vol. 5, pp. 1–18, 2013.

[2] M. Geuss, “Why Eye Tracking Could Make VR Displays Like the Oculus Rift Consumer-Ready,” *Ars Technica*, jun 2014.

[3] L. El Hafi, K. Takemura, J. Takamatsu, and T. Ogasawara, “Model-Based Approach for Gaze Estimation from Corneal Imaging Using a Single Camera,” in *IEEE/SICE International Symposium on System Integration (SII)*, Nagoya, dec 2015, pp. 88–93.

[4] L. El Hafi, M. Ding, J. Takamatsu, and T. Ogasawara, “Gaze Tracking Using Corneal Images Captured by a Single High-Sensitivity Camera,” in *International Broadcasting Convention (IBC)*, Amsterdam, sep 2016, pp. 33–43.

[5] L. El Hafi, M. Ding, J. Takamatsu, and T. Ogasawara, “Gaze Tracking Using Corneal Images Captured by a Single High-Sensitivity Camera,” *The Best of IET and IBC 2016-2017*, vol. 8, pp. 19–24, 2016.

[6] L. El Hafi, M. Ding, J. Takamatsu, and T. Ogasawara, “Gaze Tracking and Object Recognition from Eye Images,” in *IEEE International Conference on Robotic Computing (IRC)*, Taichung, apr 2017, pp. 310–315.

[7] L. El Hafi, M. Ding, J. Takamatsu, and T. Ogasawara, “STARE: Real-Time, Wearable, Simultaneous Gaze Tracking and Object Recognition from Eye Images,” *SMPTE Motion Imaging Journal*, vol. 126, no. 6, 2017.

[8] E. D. Guestrin and M. Eizenman, “General Theory of Remote Gaze Estimation Using the Pupil Center and Corneal Reflections,” *IEEE Transactions on Biomedical Engineering*, vol. 53, no. 6, pp. 1124–1133, jun 2006.

[9] K. Nishino and S. K. Nayar, “Corneal Imaging System: Environment from Eyes,” *International Journal of Computer Vision*, vol. 70, no. 1, pp. 23–40, 2006.

[10] C. Nitschke, A. Nakazawa, and H. Takemura, ‘‘Display-Camera Calibration Using Eye Reflections and Geometry Constraints,’’ *Computer Vision and Image Understanding*, vol. 115, no. 6, pp. 835–853, 2011.

[11] K. Takemura, K. Takahashi, J. Takamatsu, and T. Ogasawara, ‘‘Estimating 3-D Point-of-Regard in a Real Environment Using a Head-Mounted Eye-Tracking System,’’ *IEEE Transactions on Human-Machine Systems*, vol. 44, no. 4, pp. 531–536, 2014.

[12] K. Takemura, T. Yamakawa, J. Takamatsu, and T. Ogasawara, ‘‘Estimation of a Focused Object Using a Corneal Surface Image for Eye-Based Interaction,’’ *Journal of Eye Movement Research*, vol. 7, no. 3, pp. 1–9, 2014.

[13] A. Plopski, Y. Itoh, C. Nitschke, K. Kiyokawa, G. Klinker, and H. Takemura, ‘‘Corneal-Imaging Calibration for Optical See-Through Head-Mounted Displays,’’ *IEEE Transactions on Visualization and Computer Graphics*, vol. 21, no. 4, pp. 481–490, 2015.

[14] K. Takemura, S. Kimura, and S. Suda, ‘‘Estimating Point-of-Regard Using Corneal Surface Image,’’ in *Symposium on Eye Tracking Research and Applications (ETRA)*, Safety Harbor, mar 2014, pp. 251–254.

[15] K. Kunze, K. Inoue, K. Masai, Y. Uema, S. S.-A. Tsai, S. Ishimaru, K. Tanaka, K. Kise, and M. Inami, ‘‘MEME: Smart Glasses to Promote Healthy Habits for Knowledge Workers,’’ in *ACM Special Interest Group on Computer Graphics and Interactive Techniques (SIGGRAPH)*, Los Angeles, aug 2015, pp. 1–1.

[16] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R. Monga, S. Moore, D. G. Murray, B. Steiner, P. Tucker, V. Vasudevan, P. Warden, M. Wicke, Y. Yu, and X. Zheng, ‘‘TensorFlow: A System for Large-Scale Machine Learning,’’ in *USENIX Symposium on Operating Systems Design and Implementation (OSDI)*, Savannah, nov 2016, pp. 265–283.

[17] B. A. Wandell, ‘‘Foundations of Vision,’’ *Stanford University*, 1995.

[18] L. El Hafi, P. M. Uriguen Eljuri, M. Ding, J. Takamatsu, and T. Ogasawara, ‘‘Wearable Device for Camera-Based Eye Tracking: Model Approach Using Cornea Images,’’ in *JSME Robotics and Mechatronics Conference (ROBOMECH)*, Yokohama, jun 2016.

[19] D. H. Eberly, ‘‘Computing a Point of Reflection on a Sphere,’’ *The Morgan Kaufmann Series in Interactive 3D Technology*, pp. 1–4, 2008.

[20] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell, ‘‘DeCAF: A Deep Convolutional Activation Feature for Generic Visual Recognition,’’ in *International Conference on Machine Learning (ICML)*, vol. 32, Beijing, jun 2014, pp. 647–655.

[21] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, ‘‘You Only Look Once: Unified, Real-Time Object Detection,’’ in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, jun 2016, pp. 779–788.

[22] J. Redmon and A. Farhadi, ‘‘YOLO9000: Better, Faster, Stronger,’’ *arXiv Computer Vision and Pattern Recognition*, pp. 1–9, 2016.

[23] H. Bay, T. Tuytelaars, and L. Van Gool, ‘‘SURF: Speeded Up Robust Features,’’ in *European Conference on Computer Vision (ECCV)*, vol. 1, Graz, may 2006, pp. 404–417.

[24] M. Muja and D. G. Lowe, ‘‘Fast Approximate Nearest Neighbors with Automatic Algorithm Configuration,’’ in *International Conference on Computer Vision Theory and Applications (VISAPP)*, Lisboa, feb 2009, pp. 331–340.

[25] M. A. Fischler and R. C. Bolles, ‘‘Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography,’’ *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[26] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, ‘‘Rethinking the Inception Architecture for Computer Vision,’’ in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, jun 2016, pp. 2818–2826.

[27] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, ‘‘ImageNet Large Scale Visual Recognition Challenge,’’ *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[28] R. Girshick, ‘‘Fast R-CNN,’’ in *IEEE International Conference on Computer Vision (ICCV)*, Santiago, dec 2015, pp. 1440–1448.

[29] T. Mertens, J. Kautz, and F. V. Reeth, ‘‘Exposure Fusion,’’ in *Pacific Conference on Computer Graphics and Applications (PG)*, Maui, oct 2007, pp. 382–390.