

# Illumination-Free Gaze Estimation Method for First-Person Vision Wearable Device

Akihiro Tsukada<sup>1</sup>, Motoki Shino<sup>1,2</sup>, Michael Devyver<sup>1</sup> and Takeo Kanade<sup>1</sup>  
Carnegie Mellon University<sup>1</sup>, University of Tokyo<sup>2</sup>

{tsukada,devyver,tk}@cs.cmu.edu, motoki@sl.t.u-tokyo.ac.jp

## Abstract

*Gaze estimation is a key technology to understand a person's interests and intents, and it is becoming more popular in daily situations such as driving scenarios. Wearable gaze estimation devices are used for long periods of time, therefore non-active sources are not desirable from a safety point of view.*

*Gaze estimation that does not rely on active source, is performed by locating iris position. To estimate the iris position accurately, most studies use ellipse fitting in which the ellipse is defined by 5 parameters(position  $(x,y)$ , rotation angle, semi-major axis and semi-minor axis). We claim that, for iris position estimation, 5 parameters are redundant because they might be influenced by non-iris edges. Therefore, we propose to use 2 parameters(position) introducing a 3D eye model(the transformation between eye and camera coordinate and eyeball/iris size). Given 3D eye model, projected ellipse that represents iris shape can be specified only by position under weak-perspective approximation. We quantitatively evaluate our method on both iris position and gaze estimation. Our results show that our method outperforms other state-of-the-art's iris estimation and is competitive to commercial product that use infrared ray with respect to both accuracy and robustness.*

## 1. Introduction

*First Person Vision (FPV)[8] is a new concept that augments human cognitive functions by working side by side with the user. The ultimate goal of FPV is to work with people and understand their behavior and intent in order to improve their quality of life. To achieve that ultimate goal, FPV relies mostly on gaze information. By combining gaze ("Where am I looking?") and environment information, we understand not only people's interests but also their behaviors, and thus, we try to provide solutions to their demands[6]. As stated above, FPV can be applied to various fields, e.g. Human behavior analysis[9], Human-*

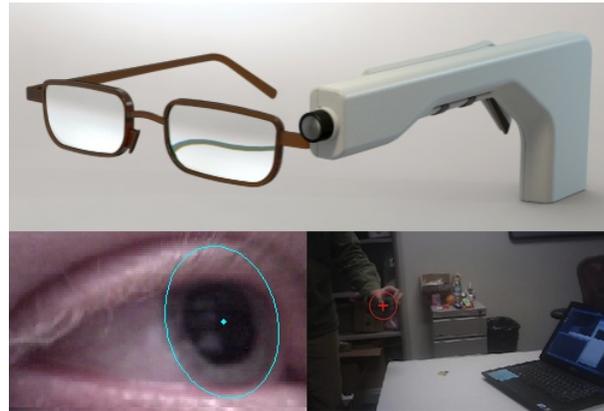


Figure 1: General overview of the device attached to eyeglasses (up) and gaze estimation example taken with the system, down left: Iris position estimation, down right: Gaze estimation.

computer interaction, rehabilitation and driver assistant.

When you think about driver assistant, Advanced driver-assistance systems (ADAS) have a potential to save people from traffic accidents. In ADAS, understanding environment around vehicle and driver's intent is indispensable to achieve robust and practical intelligent assistance. As is the case with FPV, the gaze plays quite important roles in ADAS[3],[14].

This paper aims to build a robust gaze estimation method. The required accuracy and robustness is similar to competitive commercial systems. Our device images and expected results are shown in Figure 1.

### 1.1. Related work and problems

FPV devices are designed to be worn for long periods of time (hours or even days) and it is preferable not to have an infrared light source for such a long time, for the safety of the user.

Gaze estimation technology has been available for many years using a variety of methods [18] [4]. In general, gaze estimation is achieved by analyzing a person's eyes and locating the iris/pupil position using an image sensor. Several

wearable systems are available today, i.e. i[16], g[13] and headgear[12]. They are using infrared (IR) as a light source toward the eye which makes them easier to extract the pupil, but have the disadvantage to introduce noise when used outdoors h daytime due to the ambient infrared illumination. The openEyes project [10], [11], has proposed both illumination free and using IR gaze estimation systems.

Iris/pupil position estimation methods are classified into two main categories: (a) Appearance based approach [1],[7],[15] and (b) Edge-based approach [5],[11],[14]. Tan et al. [15]. Edge-based approach employed an appearance manifold model to locate the iris position with simple feature filtering. An eye-appearance map was built with support vector regression[1]. They developed their image normalization method to improve robustness with respect to illumination change. Hansen and Pece [5] proposed an active-contour method. Their method use EM and RANSAC schemes to fit the ellipse, and work with a particle filter to detect iris and pupil position with multiple hypothesis. In Li et al. [11], main ideas of ray casting (locating two feature points per ray and filtering by distance) were introduced.

The main advantages of the appearance approach are processing time and robustness related to reflection and occlusion. The edge-based approach is more accurate than appearance, on the other hand the robustness still needs to be improved. So Wang et al. [17] adopted a combination of both. In most studies of edge-based, the ellipse is defined by 5 parameters that consist of cartesian positions (x,y), rotation angle, semi-major axis and semi-minor axis. Since various ellipse shapes can be expressed by 5 parameters, the ellipse tends to be influenced by non-iris edges that are caused by eyelid, eye-lush and reflection of light as mentioned in Li et al. [11]. There are still difficulties to locate the iris position robustly and efficiently.

## 1.2. Our Work

To overcome the problems mentionned above, we are introducing a 3D eye model. We assume that the transformation between the camera and the eye coordinates is fixed using, thanks to a wearable device. As the eyeball and iris size are constant to a subject, the projected iris shape is calculated using 3D eye model as an ellipse, under weak-perspective approximation. Basically, the iris position can be estimated by only 2 parameters, i.e. image coordinates. The advantage of our method is robust to outlier edges and occlusion.

We adopt a coarse to fine approach in iris extraction process to work robustly. Firstly, we initialize the iris position with a simple appearance filter. Based on this initial position, the projected iris shape is introduced as an ellipse by the 3D eye model. If there is a difference between the 3D eye model and the real eye structure, we fit the ellipse

with a minimizing cost function that consists of 'distance' and 'difference of shape' coefficients. The distance is calculated between the measured edge points to the estimated ellipse, and the difference of shape is calculated between the estimated ellipse fitting and the projected ellipse shape. This process is done iteratively while updating the ellipse shape.

We quantitatively evaluate our method on both iris position and gaze estimation compared to other state-of-the-art and commercial product that use infrared with respect to both accuracy and robustness.

## 2. 3D Eye-model for Gaze Estimation

First, we define our 3D eye model. As shown in Figure 2a, the eyeball coordinates are represented with X,Y,Z and the camera coordinates are x,y,z. We assume that the eyeball is a sphere with radius  $r_e$ , and iris contour is a circular ring of radius  $r_I$ . The distance from the center of the eye to the iris plane is defined as  $D$  as shown in Figure 2b. The relation between  $r_e$ ,  $r_I$  and  $D$  is  $D^2 = r_e^2 - r_I^2$ . Iris contour is expressed as  $\mathbf{X}_i = (X_i, Y_i, D)^T$ , where each contour point is satisfied with  $X_i^2 + Y_i^2 + D^2 = r_I^2$ , and the projected iris contour  $\mathbf{x} = (x, y)^T$  in image plane is expressed as follows:

$$\lambda \tilde{\mathbf{x}} = \mathbf{K} \mathbf{P} \mathbf{M} \tilde{\mathbf{X}} \quad (1)$$

where  $\tilde{\phantom{x}}$  means homogeneous coordinates, and  $\mathbf{K}, \mathbf{P}, \mathbf{M}$  are intrinsic parameter matrix, projection model matrix and extrinsic parameter matrix, respectively. We define the intrinsic matrix as follow

$$\mathbf{K} = \begin{bmatrix} f & 0 & u_0 \\ 0 & f & v_0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2)$$

where  $f$  denotes the camera focal length, and  $u_0, v_0$  is the image center. We use weak-perspective projection model because weak-perspective projection, in practice, gives a good approximation. Matrix  $\mathbf{P}$  is given by

$$\mathbf{P} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & z_p \end{bmatrix} \quad (3)$$

where we define a orthographic projection plane as  $z = z_p$  which indicates iris center,  $\mathbf{c}_I = (x_p, y_p, z_p)^T$ , in the camera coordinate as shown in Figure 2b. Extrinsic parameter matrix  $\mathbf{M}$  is defined as

$$\mathbf{M} = \begin{bmatrix} \mathbf{R}_0 & \mathbf{T}_0 \\ \mathbf{0}^T & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R}_I & \mathbf{0} \\ \mathbf{0}^T & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R}_0 \mathbf{R}_I & \mathbf{T}_0 \\ \mathbf{0}^T & 1 \end{bmatrix} \quad (4)$$

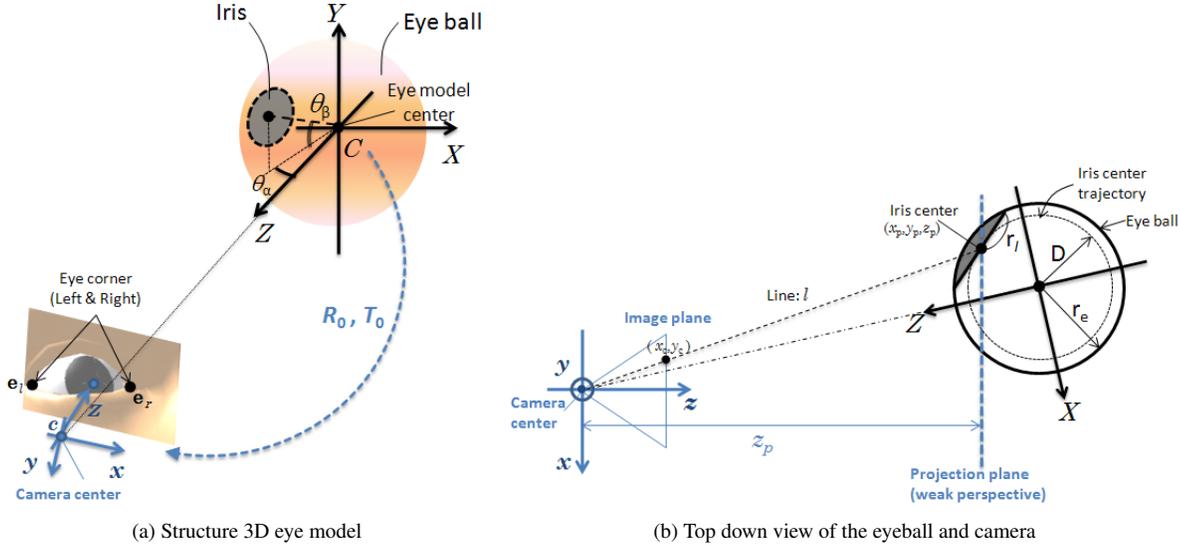


Figure 2: Our 3D eye model overview

where  $\mathbf{R}_0, \mathbf{T}_0$  are the rotation matrix and the translation matrix between the eyeball coordinates and the camera coordinates, we represent  $\mathbf{T}_0 = (t_x, t_y, t_z)^T$ , and  $\mathbf{0}$  indicates zero vector. The rotation matrix  $\mathbf{R}_0$  is defined according to the following procedures: (1) Z-axis must pass the camera center,  $v_z = -\mathbf{T}_0 / |\mathbf{T}_0|$ , and (2) X-axis is parallel to  $x$ - $z$  plane in camera coordinates,  $\mathbf{v}_x = (\sin \tau \ 0 \ -\cos \tau)^T$ , where  $\tau = \arctan t_z / t_x$ , as shown in Figure 2a. (3) Y-axis is cross product of  $\mathbf{v}_x$  and  $\mathbf{v}_z$ .  $\mathbf{R}_0$  is given by

$$\mathbf{R}_0 = \begin{bmatrix} \mathbf{v}_x & \mathbf{v}_x \times \mathbf{v}_z & \mathbf{v}_z \end{bmatrix} \quad (5)$$

$\mathbf{R}_I$  matrix represents the iris rotation and is expressed in equation (6), where  $\theta_\alpha, \theta_\beta$  indicate X-Y plane angle and Y-Z plane angle, respectively, as shown in Figure 2a.

$$\mathbf{R}_I = \begin{bmatrix} \cos \theta_\alpha & -\sin \theta_\alpha \sin \theta_\beta & \sin \theta_\alpha \cos \theta_\beta \\ 0 & \cos \theta_\beta & -\sin \theta_\beta \\ \sin \theta_\alpha & \cos \theta_\alpha \sin \theta_\beta & \cos \theta_\alpha \cos \theta_\beta \end{bmatrix}. \quad (6)$$

Since the iris contour is a circular ring, the projected iris contour on the image plane can be expressed by an ellipse under weak-perspective approximation. We define the standard ellipse form as  $\mathbf{x}^T \mathbf{Q} \mathbf{x} = 0$ , where the ellipse is expressed by 5 parameters:  $\mathcal{P} = \{x_c, y_c, a, b, \phi\}$ ,  $(x_c, y_c)$  indicates the ellipse center  $\mathbf{x}_c$  in image coordinate,  $a, b$  ( $a \geq b > 0$ ) indicates the radius length in the direction of  $x$  and  $y$  respectively, and  $\phi$  indicates the ellipse rotation angle as shown in Figure (3). Matrix  $\mathbf{Q}$  is represented as

$$\mathbf{Q} = \begin{pmatrix} q_0 & q_2 & -2x_c q_0 - y_c q_2 \\ q_2 & q_1 & -2y_c q_1 - x_c q_2 \\ -2x_c q_0 - y_c q_2 & -2y_c q_1 - x_c q_2 & q_0 x_c^2 + q_1 y_c^2 + x_c y_c q_2 - a^2 b^2 \end{pmatrix} \quad (7)$$

where  $q_i$  are given by

$$\begin{aligned} q_0 &= a^2 \sin^2 \phi + b^2 \cos^2 \phi \\ q_1 &= a^2 \cos^2 \phi + b^2 \sin^2 \phi \\ q_2 &= -2(a^2 - b^2) \sin \phi \cos \phi \end{aligned} \quad (8)$$

When the iris center  $(x_c, y_c)$  is given, other parameters  $\{a, b, \phi\}$  are described as follow

$$a = \sqrt{\frac{r_I^2 - \left(D^2 - \frac{x_c'^2 + y_c'^2}{D^2}\right)}{1 - \frac{x_c'^2 + y_c'^2}{D^2}}} \quad (9a)$$

$$b = \sqrt{\left(r_I^2 - \left(D^2 - \frac{x_c'^2 + y_c'^2}{D^2}\right)\right) \left(1 - \frac{x_c'^2 + y_c'^2}{D^2}\right)} \quad (9b)$$

$$\phi = \begin{cases} 0, & \text{for } \beta = 0 \text{ and } \alpha \leq \gamma \\ \pi/2, & \text{for } \beta = 0 \text{ and } \alpha > \gamma \\ 1/2 \cot^{-1} \left( \frac{a-\gamma}{2\beta} \right), & \text{for } \beta \neq 0 \text{ and } \alpha \leq \gamma \\ 1/2 \cot^{-1} \left( \frac{a-\gamma}{2\beta} \right) + \pi/2, & \text{for } \beta \neq 0 \text{ and } \alpha > \gamma \end{cases} \quad (9c)$$

where  $\alpha = 1 - \frac{y_c'^2}{D^2}$ ,  $\beta = \frac{x_c' y_c'}{D^2}$ ,  $\gamma = 1 - \frac{x_c'^2}{D^2}$  and  $(x_c', y_c')$  represent the position in camera coordinates of  $\mathbf{x}_c$ , is given by

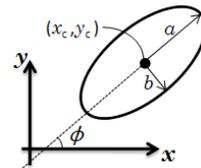


Figure 3: Ellipse standard form

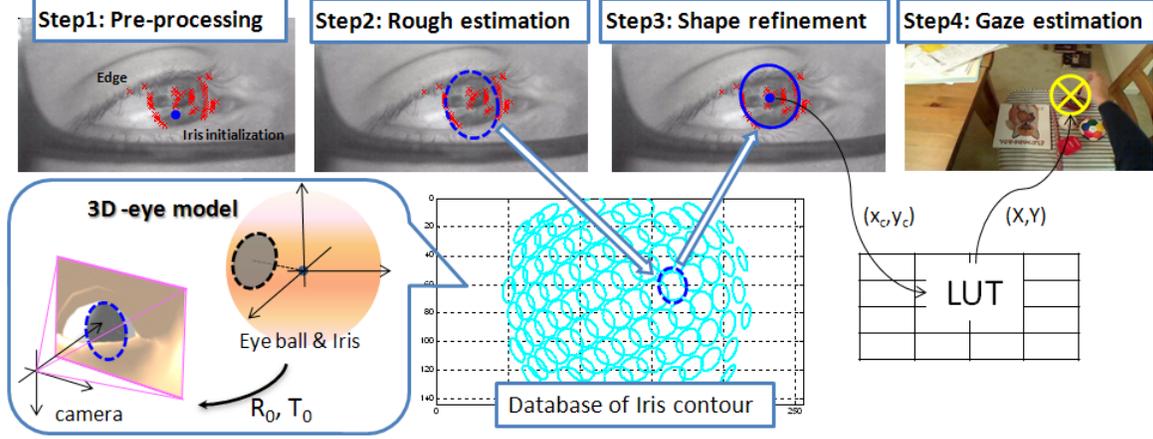


Figure 4: Overview of our gaze estimation method

$$\begin{pmatrix} x'_c \\ y'_c \end{pmatrix} = \frac{z_p}{f} \begin{pmatrix} x_c - u_0 \\ y_c - v_0 \end{pmatrix} - \begin{pmatrix} t_x \\ t_y \end{pmatrix} \quad (10)$$

where  $z_p$  is expressed by  $(x_p, y_p, z_p)^T = s(x_c, y_c, f)^T$  and it indicates a line that pass camera center and  $\mathbf{x}_c$  can be solved with the following equations:

$$(x_p - t_x)^2 + (y_p - t_y)^2 + (z_p - t_z)^2 = D^2 \quad (11)$$

, and equation (11) is a trajectory of iris center as shown in Figure 2b. So  $s$  is given as a solution of quadratic equation (12), and  $z_p$  equals  $z_p = s f$ .

$$(s x'_c - t_x)^2 + (s y'_c - t_y)^2 + (s f - t_z)^2 = D^2 \quad (12)$$

### 3. Gaze Estimation Implementation

Figure 4 shows an overview of our gaze estimation method. Our method consists of four steps. In Step 1, we extract the edges and initialize the iris position based on appearance. In Step 2, we search for a better iris position around the initial position. Using the current iris position, the projected iris shape can be calculated as an ellipse with the 3D eye model. We use this ellipse as a prior shape in the ellipse fitting process in Step3. Finally, we estimate the gaze point (in the outside image) using the iris position in Step 4.

#### 3.1. Pre-processing (Step 1)

After filtering the edges using a first derivative Gaussian filter, to extract the iris edges robustly, we use a non-maximum suppression (NMS) method with a threshold of edge magnitude. As vertical edges are most reliable for the iris images, we use one directional NMS. Edge's position and gradients are denoted as  $\mathbf{x}_i, \mathbf{g}_i, i = 1 \dots N_{pts}$ , where  $N_{pts}$  indicates a number of extracted edges.

For ellipse fitting, an initialization of iris position is needed. We use a simple filter to initialize the iris position [17], which showed methods that are using simple filter, such as step edge filter, to detect iris position efficiently. The initial iris center  $\mathbf{x}_c^* = (x_c^*, y_c^*)$  is calculated as  $\mathbf{x}_c^* = \text{argmax } \text{img}(\mathbf{x}) \star H$ , s.t.  $x_{e_l} < x_c^* < x_{e_r}$ , where  $\text{img}(\mathbf{x})$  is the image intensity of each pixel,  $H$  is simple filter and  $\star$  represent cross-correlation. To avoid false detection, we use both eye corners  $\mathbf{e}_l = (x_{e_l}, y_{e_l}), \mathbf{e}_r = (x_{e_r}, y_{e_r})$  (See Figure 2a) as boundaries for the iris position. An example of initialization result is shown in Figure 6.

#### 3.2. Rough Iris Position Estimation (Step 2)

##### 3.2.1 Edge point selection

Many non-iris edges are present on the image therefore, in order to reduce them, we introduce the edges reliabilities based on following ideas:

- According to  $x$ -coordinate of the iris center, edges are classified into two classes :  $w_L, w_R$
- Edges that exist close to the iris position are more reliable :  $w_d$
- Ocluded area of iris by eyelids is estimated by iris height position :  $w_o$

Reliabilities of  $w_L, w_R$  are expressed as follow

$$w_L(\mathbf{x}_i) = \begin{cases} 1, & \text{if } x_{e_l} < x_i < x_c^* \text{ and } \overrightarrow{\mathbf{x}_c^* \mathbf{x}_i} \cdot \mathbf{g}_i > 0 \\ 0, & \text{otherwise} \end{cases}$$

$$w_R(\mathbf{x}_i) = \begin{cases} 1, & \text{if } x_c^* < x_i < x_{e_r} \text{ and } \overrightarrow{\mathbf{x}_c^* \mathbf{x}_i} \cdot \mathbf{g}_i < 0 \\ 0, & \text{otherwise} \end{cases} \quad (13)$$

where  $\overrightarrow{\mathbf{x}_c^* \mathbf{x}_i}$  indicates a vector from  $\mathbf{x}_c^*$  to  $\mathbf{x}_i$ . And  $w_d$  is given by

$$w_d(\mathbf{x}_i) = \begin{cases} 1, & \text{if } \|\mathbf{x}_i - \mathbf{x}_c^*\| < d_{thresh} \\ 0, & \text{otherwise} \end{cases} \quad (14)$$

where  $d_{thresh}$  is a parameter being set manually according to the light conditions and an eye color.

Figure 5 shows how  $w_o$  is adjusted based on the iris position. The standard height  $\bar{y}$  is defined using the center of the eye corners, as  $\bar{y} = \left(\frac{y_{e_i} + y_{e_r}}{2}\right)$ . When the iris is located around  $\bar{y}$ , the upper and lower sides of the iris contours are not usually visible. When the iris position is located above  $\bar{y}$ , the lower side contour is visible but the upper side is not, and vice versa. Consequently, we need to set the corresponding angle to select the visible contour.  $w_o$  is defined as follow

$$w_o(\mathbf{x}_i) = \begin{cases} 1, & -B(\mathbf{x}_c^*) < \theta_i < B(\mathbf{x}_c^*) \\ 0 & \text{otherwise} \end{cases} \quad (15)$$

where  $\theta_i$  is the edge orientation given by  $\arctan(g_{y_i}/g_{x_i})$ . Since the corresponding angles are mainly related to the iris height, we set the height baseline as the middle of the eye corners, and define  $B(\mathbf{x}_c^*)$  as follow

$$B(\mathbf{x}_c^*) = \begin{cases} b_{bottom}(y_c^* - \bar{y}) & \text{if } (y_c^* - \bar{y}) < 0 \\ b_{upper}(y_c^* - \bar{y}) & \text{if } (y_c^* - \bar{y}) > 0 \end{cases} \quad (16)$$

and we define the function  $b(y_c^* - \bar{y})$  as sigmoid function, whose parameters were determined experimentally.

Finally, the edge reliability is calculated by multiplying every reliability

$$w(\mathbf{x}_i) = \{w_L(\mathbf{x}_i), w_R(\mathbf{x}_i)\} \cdot w_d(\mathbf{x}_i) w_o(\mathbf{x}_i) \quad (17)$$

### 3.2.2 Discrete Ellipse fitting

The accuracy of the initial iris position  $\mathbf{x}_c^*$  is not precise enough therefore, we are increasing the searching area around it. Since the projected ellipse  $\mathcal{P}_x$  is using only two parameters (see equations (9a),(9b),(9c)), we are able to use a discrete search method for run time efficiency. Discretized ellipse parameters  $\mathcal{P}_j = \{x_j, y_j, a_j, b_j, \theta_j\}, j = 1 \dots k$  are calculated in advance, and stored in a database. Using  $\mathbf{x}_c^*$ , we pick the  $n$  closest ellipse parameters,  $\mathcal{P}_k, k = 1 \dots n$ ,

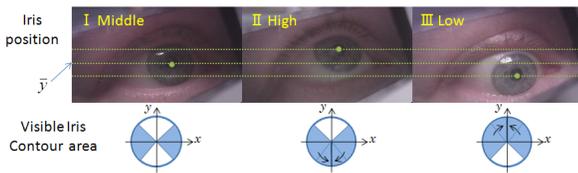


Figure 5: Visible contour part according to iris position

from the database. The ellipse center  $\mathcal{P}_{x_k}^*$  is then estimated using a cost function such as

$$\mathcal{P}_k^* = \underset{\mathcal{P}_k}{\operatorname{argmin}} \left\| \frac{\sum_{i=1}^{N_{pts}} w(\mathbf{x}_i) (|\mathbf{x}_i - \mathbf{x}_{\mathcal{P}_k}| + \lambda_0 (\mathbf{g}_i \cdot \nabla \mathbf{x}_{\mathcal{P}_k}))^2}{N_{in}} \right\| \quad (18)$$

where  $\mathbf{x}_{\mathcal{P}_k}$  indicates the minimum distance between an ellipse point and the edge  $\mathbf{x}_i$ .  $\nabla \mathbf{x}_{\mathcal{P}_k}$  indicates its gradient.  $N_{in}$  indicates the number of points  $\mathbf{x}_i$  which are satisfied with  $|\mathbf{x}_i - \mathbf{x}_{\mathcal{P}_k}| + \lambda_0 (\mathbf{g}_i \cdot \nabla \mathbf{x}_{\mathcal{P}_k}) < Thresh\_d_0$ . An example of discrete search points and fitted ellipse  $\mathcal{P}_k^*$  are shown in Figure 6.

### 3.3. Ellipse Shape Refinement (Step 3)

When  $\mathcal{P}_k^*$  becomes closer to the true iris shape, we are updating the reliability coefficient  $w(\mathbf{x}_i)$  using the edge point selection (in section 3.2.1). However, some undesirable edges remained so we used an ellipse  $\mathcal{P}_k^*$  into equation (18) as a shape constraint. The fine ellipse is calculated as follow

$$\underset{\mathcal{P}_x}{\operatorname{minimize}} \left\| \frac{\sum_{i=1}^{N_{pts}} w(\mathbf{x}_i) (|\mathbf{x}_i - \mathbf{x}_{\mathcal{P}_k}| + \lambda_0 (\mathbf{g}_i \cdot \nabla \mathbf{x}_{\mathcal{P}_k}))^2}{N'_{in}} \right\| + \lambda_1 S(\mathcal{P}_x, \mathcal{P}_k^*) \quad (19)$$

where the initial ellipse parameter is set as  $\mathcal{P}_k^*$ . The  $\lambda_1$  is determined by the accuracy of the 3D eye model. When  $\lambda_1$  is large value, a fitted ellipse is depend on the position(2 parameters). An example of result is shown in Figure 6.  $N'_{in}$  indicate the number of points  $\mathbf{x}_i$  that are satisfied with  $F(\mathbf{x}_i, \mathcal{P}_x) + \lambda_0 G(\mathbf{x}_i, \mathbf{g}_i, \mathbf{x}_e) < Thresh\_d_1$ .  $S$  function indicates an overlap error between two ellipses

$$S(\mathcal{P}_x, \mathcal{P}_k^*) = 1 - \frac{R(\mathcal{P}_x) \cap R(\mathcal{P}_k^*)}{R(\mathcal{P}_x) \cup R(\mathcal{P}_k^*)} \quad (20)$$

where  $R(\mathcal{P})$  represents the ellipse area defined by parameters  $\mathcal{P}$ , the union of the area is  $R(\mathcal{P}_x) \cup R(\mathcal{P}_k^*)$ , and  $R(\mathcal{P}_x) \cap R(\mathcal{P}_k^*)$  is their intersection. Parameters,  $\lambda_0, \lambda_1, Thresh\_d_0$  and  $Thresh\_d_1$ , are determined experimentally.

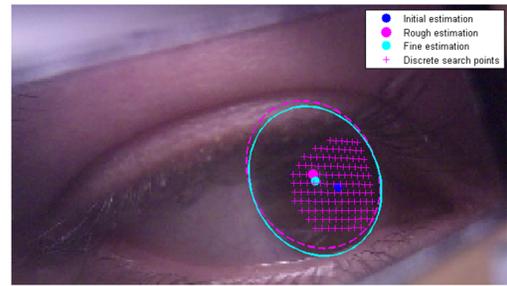


Figure 6: Example of ellipse fitting convergence

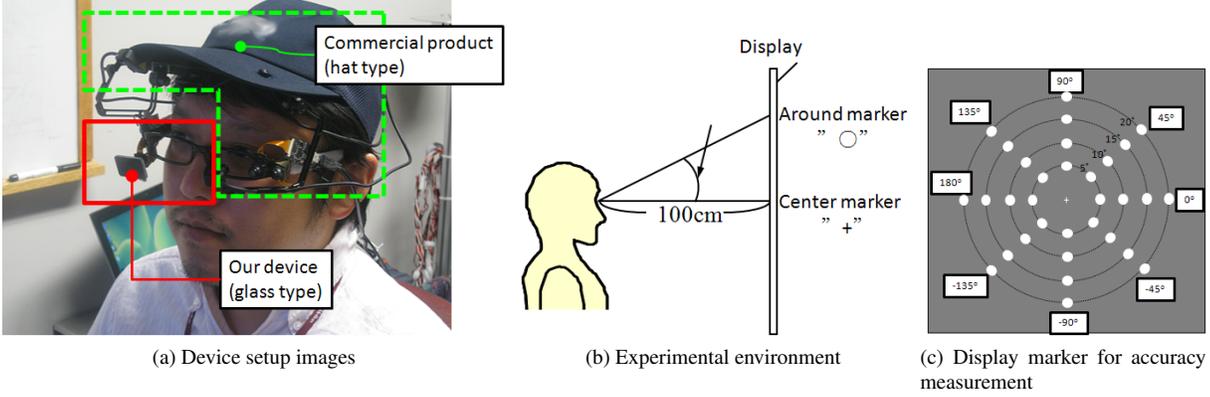


Figure 7: Experimental setup

### 3.4. Gaze estimation (Step 4)

Every position of the iris has one corresponding point in the outside image. In order to calculate the monotonic relationship between the two, we introduced a look-up-table (LUT) in which the gaze position is given by linear interpolation.

## 4. Experimental Evaluation of our method

To evaluate our method, we have implemented three experiments as follow

1. Ellipse fitting evaluation compared to openEyes [10].
2. Gaze accuracy compared to infrared-based commercial products (“IR product”).

We have tested our method using a Dell Laptop with a core-i5 2.4GHz CPU. The resolution of the eye-image is 256x144 [pixel] and of the outside-image is 1280x720 [pixel]. The camera frame rate is 30 frame/sec. Our device layout is represented on Figure 8. Details of our device are described in the [2]. The Eye-image was rectified using the

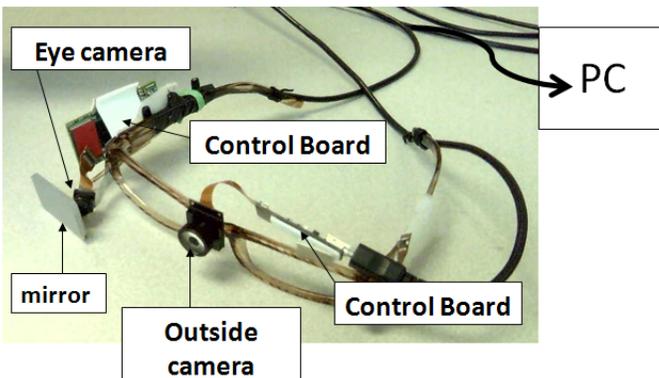


Figure 8: Our device of glass-mounted type

intrinsic parameters of the eye-camera. It took about 40ms to process every individual image.

We had to calibrate the system for every user in order to build the LUT and adjust some parameters. The database includes the ball radius  $r_e$ , the iris radius  $r_I$  and the eyeball center  $\mathbf{T}_0$  used to calculate the projected iris shape. In this experiment, the database had 600 ellipses that picked up discretely every 3 degree sampling of  $\theta_\alpha, \theta_\beta$ , and the LUT was built using about 40 sample points, while the 3D eye model coefficients  $r_e, r_I$  and  $\mathbf{T}_0$  were set manually. Through this experiment, parameters are set as follows:  $\lambda_0 = 0.5, \lambda_1 = 1, Thresh_{d_0} = 16[pixel], Thresh_{d_1} = 8[pixel]$ . Our 3D eye model works quite good, so we use only two parameters to fit ellipse.

### 4.1. Ellipse fitting evaluation

In this experiment, we compared the ellipse fitting performances of our device with the openEyes system (non-active source). Ground-truth ellipse data was defined using manual extraction. The overlap errors, defined in equation (20), was used for evaluation. Figure9 shows the comparison results. Some examples of overlap errors are displayed in Figure10. The gaze accuracy was directly affected when the overlap error was over 30%. Table 1 shows the average value and the standard deviation of each histogram. The average value of the distribution mainly corresponds to a measure of accuracy, and the deviation value to a measure of robustness. Our method shows better performances in both factors. In Figure 10, several comparable examples are shown.

### 4.2. Accuracy Evaluation of Gaze Estimation Method

Commercial gaze estimation systems that are using infrared are well known for their high accuracy ( $> \pm 0.5$  [degree]) as well as frequency (usually 60 frame/sec).

We placed our system on the subject’s head, together

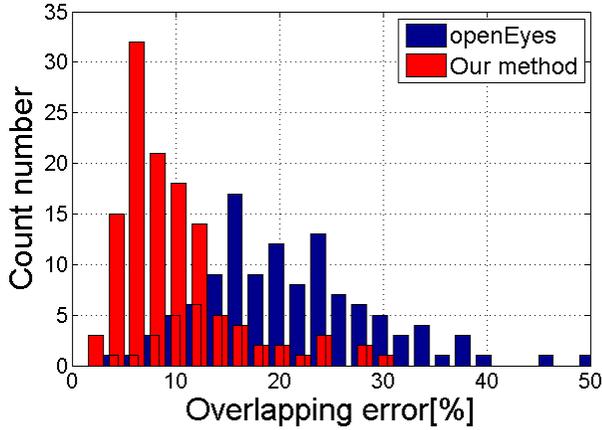


Figure 9: Accuracy and robustness of the iris extraction.

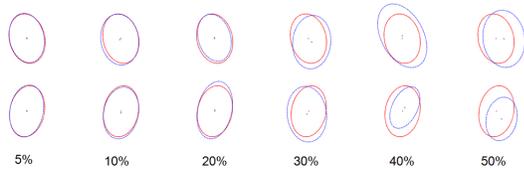


Figure 10: Overlap error examples

with the IR product (see Figure 7a). Both devices were attached to avoid interfering with each other and to capture data simultaneously. See Figure 7b. We presented a large

Table 1: Quantitative evaluation of our iris extraction compared to openEyes

	openEyes	Our method
Overlapping error average	29.31%	11.41%
Deviation (STD)	78.60%	49.88%



Figure 11: Gaze tracking samples comparison

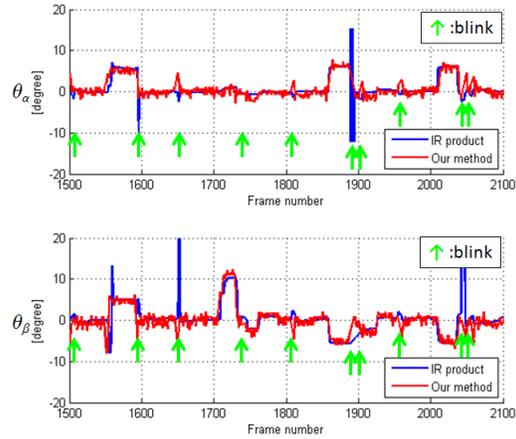


Figure 12: Transition of iris rotation. Upper graph show transition of  $\theta_\alpha$  and Bottom is  $\theta_\beta$ . Green Arrow indicates blink.

display with various markers at varying angles in front of the user. Two types of markers were presented: cross-hair and circle. The cross-hair is used as a default gaze position during the experiment, and the circles are displayed around the cross-hair (see Figure 7c).

We compared our method to the IR product with respect to the iris rotation angle ( $\theta_\alpha, \theta_\beta$ ). In order to compensate the different coordinates between the IR product and our system, we set ( $\theta_\alpha, \theta_\beta$ ) as (0,0) when a subject was gazing at a cross-hair as shown in Figure 7c. Figure 12 shows the transition of the iris rotation for 20 seconds.

Table 2 shows the average difference and the standard deviation for both systems. We decided not to display the data during blink times ( $\pm 7$  frames around blink frame were ignored).

Considering the time synchronization problem and a rotation error between the IR product and our device coordinates, these results prove that our method is competitive with the IR product.

We applied our method to driving scenario. In this scenario, we use wide view lens, 135[degree], and frame rate is 15fps and resolution is 1280x1024[pixel]. In figure 13, we showed one result of gaze estimation and iris extraction.

Table 2: Comparison our method to Product(N=365)

	Horizontal direction $\theta_\alpha$	Vertical direction $\theta_\beta$
Average mean error	0.66 [degree]	1.02 [degree]
Deviation(STD)	0.93 [degree]	1.84 [degree]



Figure 13: Result of Driving Scenario

## 5. Conclusion

In this paper, we proposed a gaze estimation method to overcome the difficult task of iris extraction. The ellipse shape can extract the iris correctly under occluded or reflecting conditions. Experimental results on actual data show that our method is accurate and competitive to commercial product that use infrared rays. Our method improves robustness compared to other methods with visible rays.

We have tested our method to various people under various lighting condition, and the gaze estimation showed high performance. Through our experiments, we kept parameter  $\lambda_1$  large value and never changed. It means that the 3D eye model has good effective to add constraint in ellipse fitting process, as a result the iris can be extracted accurately and robustly using only two parameters.

Finally, we applied our method to driving scenario, and we showed that our method must be available for driver assistant technology.

## References

- [1] Jean-Baptiste Keller Basilio Noris and Aude Billar. A wearable gaze tracking system for children in unconstrained environments. *Computer Vision and Image Understanding*, 2011. 2
- [2] Michael Devyver, Akihiro Tsukada, and Takeo Kanade. A wearable device for first person vision. In *Third International Symposium on Quality of Life Technology (isQoL2011). A Joint Event with RESNA 2011 Conference as a part of FICCDAT*, 2011. 6
- [3] Anup Doshi and Mohan M. Trivedi. On the roles of eye gaze and head dynamics in predicting driver's intent to change lanes. *IEEE Transactions on Intelligent Transportation Systems*, 10:453–462, 2009. 1
- [4] Heiko Drewes. *Eye Gaze Tracking for Human Computer Interaction*. PhD thesis, Media Informatics Group, LMU University of Munich, 2010. 1
- [5] Dan Witzner Hansen and Arthur E.C. Pece. Eye tracking in the wild. *Computer Vision and Image Understanding*, 98:155–181, 2005. 2
- [6] Mary Hayhoe and Dana Ballard. Eye movements in natural behavior. *TRENDS in Cognitive Sciences*, Vol.9 No.4:188–194, 2005. 1
- [7] Yen-wei Chen Hu-chuan Lu, Chao Wang. Gaze tracking by binocular vision and lbp features. In *International Conference on Pattern Recognition*, 2008. 2
- [8] Takeo Kanade. First person vision. In *First Workshop on Egocentric Vision(in conjunction with CVPR2009)*, 2009. 1
- [9] Yoichi Sato Kris M. Kitani, Takahiro Okabe and Akihiro Sugimoto. Fast unsupervised ego-action learning for first-person sports videos. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011. 1
- [10] Dongheng Li and Derrick Parkhurst. Open-source software for real-time visible- spectrum eye tracking. In *The 2nd Conference on Communication by Gaze Interaction (COGAIN)*, 2006. 2, 6
- [11] Dongheng Li, Jason Babcock, and Derrick J. Parkhurst. openeyes: a low-cost head-mounted eye-tracking solution. In *Vision for Human-Computer Interaction Workshop at CVPR*, 2006. 2
- [12] Jason S.Babcock and Jeff B. Pelz. Building a lightweight eyetracking headgear. In *ACM Eye tracking research and applications symposium*, 2004. 2
- [13] SMI. Sensomotoric instruments gmbh. In <http://www.smivision.com/>. 2
- [14] Iain Matthews Takahiro Ishikawa, Simon Baker and Takeo Kanade. Passive driver gaze tracking with active appearance models. Technical report, tech. report CMU-RI-TR-04-08, Robotics Institute, Carnegie Mellon University, February, 2004. 1, 2
- [15] Kar-Han Tan, David J Kriegman, and Narendra Ahuja. Appearance-based eye gaze estimation. In *Workshop on Applications of Computer Vision*, 2002. 2
- [16] Tobii. Technology ab stockholm, sweden. In [URL http://www.tobii.com/](http://www.tobii.com/). 2
- [17] Jian-Gang Wang, Eric Sung, and Ronda Venkateswarlu. Eye gaze estimation from a single image of one eye. In *iccv, vol. 1, pp.136, Ninth IEEE International Conference on Computer Vision (ICCV'03) - Volume 1, 2003*, 2003. 2, 4
- [18] L. Young and D. Sheena. Survey of eye movement recording methods. *Behav Res Methods Instrumentation*, 7:397–429, 1975. 1