Automatic Acquisition of a 3D Eye Model for a Wearable First-Person Vision Device

Akihiro Tsukada* and Takeo Kanade†
The Robotics Institute, Carnegie Mellon University

Abstract

A wearable gaze tracking device can work with users in daily-life. For long time of use, a non-active method that does not employ an infrared illumination system is desirable from safety standpoint. It is well known that the eye model constraints substantially improve the accuracy and robustness of gaze estimation. However, the eye model needs to be calibrated for each person and each device. We propose a method to automatically build the eye model for a wearable gaze tracking device. The key idea is that the eye model, which includes the eye structure and eye-camera relationship, impose constraints on image analysis even when it is incomplete, so we adopt an iterative eye model building process with gradually increasing eye model constraints. Performance of the proposed method is evaluated in various situations, including different eye colors of users and camera configurations. We have confirmed that the gaze tracking system using our eye model works well under general situations: indoor, outdoor and driving scene.

CR Categories:

Keywords: Gaze Tracking, eye tracking, 3D eye model, Wearable, First Person Vision

1 Introduction

First Person Vision (FPV) [Kanade 2009] is a new concept that augments human cognitive functions by working side by side with the user. The goal of FPV is to work with people to understand their behavior and intent for the purpose of improving their quality of life. Gaze has an important role in the FPV concept, especially when wearable gaze tracking is required [Hayhoe and Ballar 2005].

Earlier gaze tracking systems used an intrusive method(using contact lens with a hole for pupil), but more recent ones use computer-vision based non-intrusive methods (video-oculography) for eye-position detection. Video-oculography is classified into two categories: appearance-based and feature-based approaches:

- *An appearance-based approach* uses an entire eye image as the feature descriptor and maps the feature descriptor to gaze position [Tan et al. 2002]. It works well in both indoors and outdoors, but is sensitive to illumination change and less accurate compared with the next feature-based approach.

- *A feature-based approach* uses corneal reflection [Ohno and Mukawa 2004], pupil contour [Pérez et al. 2003], and iris contour [Hansen and Pece 2005] to estimate eye position. Corneal
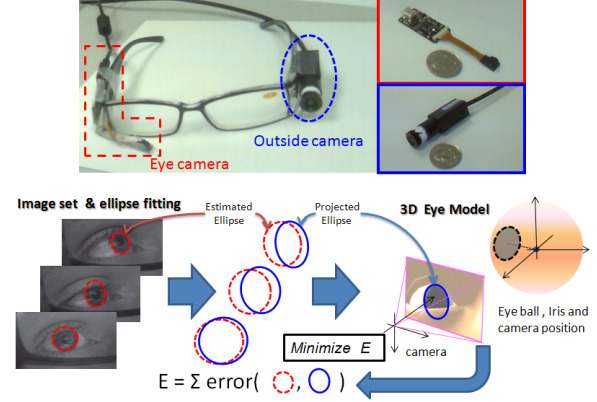
---
*e-mail: aki10tsukada@gmail.com
†e-mail: tk@cs.cmu.edu

Figure 1: *Overview of our device and method: (up) devices of eye and outside cameras that can attached glass easily , (down) proposed method for itelative3D eye model building automatically.*

reflection and pupil contour features need Infrared Ray(IR) reflection, while iris contour feature can be detected under natural lighting. When with IR-illumination, good accuracy (less than 1 degree) is achieved. Comprehensive surveys can be found in [Hansen and Ji 2010; Morimoto and Mimica 2005].

As a device to be used daily, it is undesirable for the eye to illuminate by IR for a long period, especially for older people and children [Basilio Noris and Billar 2011], even if intensity of IR meets the safety standards. Not using IR illumination requires a vision technique to locate the iris or pupil. Li and Parkhurst [2006] proposed ellipse fitting with RANSAC scheme, and Vester-Christensen et al. [2005] employed an active-contour method to track the iris or pupil based on a combination of a particle filter and the EM algorithm. These methods still lack robustness in illumination change and handling occlusion.

Recently, Wu et al. [2007] improved robustness to occlusion by introducing 3D eye models that consist of 3D-eyeballs, iris and eyelids in order to allow the device to work under natural light. However, their method needs to measure, at run-time, as many as seven parameters (eyeball size, iris size and eyelids positions) in every frame, which made the eye position detection difficult. Tsukada et al. [2011] showed that at run-time, only two parameters are actually to be measured, and achieved accuracy and robustness that are competitive to commercial products [NAC -EMR-9] that use IR. Their method, however, still required to build an accurate eye model and camera position by a time consuming and cumbersome manual process.

We propose a method to build an accurate 3D eye model automatically by extending the Tsukada's approach. The model specifies the size of the eyeball and iris, and the external relationship between the eye camera and eye. As shown in Figure 1, our model building method consists of feature extraction (ellipse fitting) and an iterative eye model parameter estimation process. We adopt a coarse-to-fine approach in adding constraints from eye model adaptively.

2 Eye Model Description

Referring to Figure 2, our eye model description includes an eyeball sphere with radius r_e, and the iris circle with radius r_I. The center C is located at $T = [t_x\, t_y\, t_z]^\top$ from camera center c. Our goal is to estimate $P = (r_e, r_I, t_x, t_y, t_z)$ automatically from a training data set.
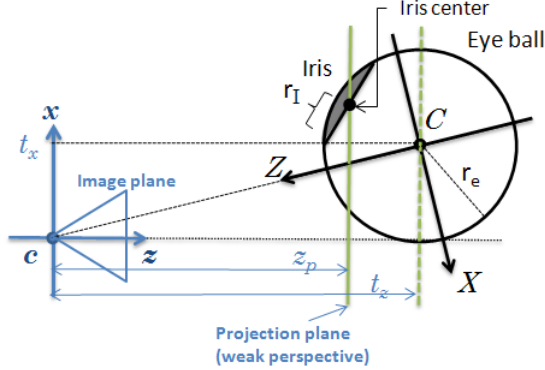

Figure 2: *Description of our eye model*

We set the Z-axis of the eye coordinate to pass camera center. The intrinsic camera parameters are: focal length f and image center (u_0, v_0). We assume that these intrinsic camera parameters have been calibrated in advance. When the projected position of iris center (u_c, v_c) is given, the projected iris shape under weak-perspective approximation is an ellipse whose major/minor axes a, b, the rotation angle ϕ are expressed as follows:

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

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

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

where $D = \sqrt{r_e^2 - r_I^2}$, $\alpha = 1 - \frac{y_c^2}{D^2}$, $\beta = \frac{x_c y_c}{D^2}$ and $\gamma = 1 - \frac{x_c^2}{D^2}$. The location (x_c, y_c) is the ellipse center position in the camera coordinate, given by

$$\begin{pmatrix} x_c \\ y_c \end{pmatrix} = \frac{z_p}{f}\begin{pmatrix} u_c - u_0 \\ v_c - v_0 \end{pmatrix} - \begin{pmatrix} t_x \\ t_y \end{pmatrix} \tag{2}$$

where z_p is the distance along z-axis between the camera center and the iris center as shown in Figure 2.

3 Eye Model Building Method

We describe how to estimate model parameters P from training data $\{I_1, I_2, \cdots, I\}$, each of which is an image of the eye at various positions. The method proceeds in four steps, as shown in Figure 3. First, we obtain an initial ellipse fitting to each image in the training data set, and then remove outliers using grid-based
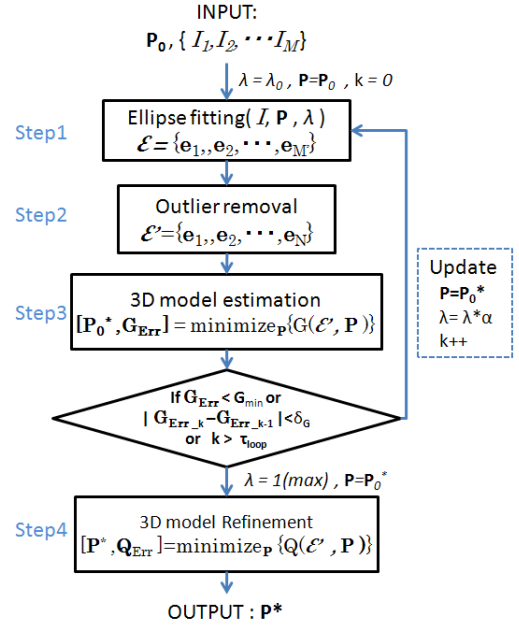

Figure 3: *Flowchart of Eye Model Building*

clustering. Next, the 3D eye model is estimated by repeatedly imposing ellipse fitting. Last, we perform the final refinement of the 3D eye model.

3.1 Feature extraction: Ellipse Fitting (Step1)

We extract ellipse features from training data by Tsukada method starting with the initial eye model parameters: $P_0 = [r_{e0}, r_{I0}, t_{x0}, t_{y0}, t_{z0}]$. Obviously, the eye model imposes on the location and shape of the ellipse that appear in the image. The more reliable the eye model is, the stronger constraints can be imposed. Since the initial eye model P_0 may include some errors, the Tsukada approach uses a parameter λ that controls the degree of imposing model constraints in ellipse fitting[1]. The λ takes range from 0 to 1; $\lambda = 0$ means no constraint, and $\lambda = 1$ means fitting the projected ellipse from the model with no deformation allowed. This way, from the i-th training data set, an ellipse e_i is extracted, $e_i = [a_i, b_i, x_{ei}, y_{ei}, \phi_i]$ as defined by Equation (1a),(1b),(1c) and (2). Images that contain eye blinking or blurring due to quick saccades are excluded by using the number of fitted edges and fitting score. Then, ellipse-set $\mathcal{E} = \{e_1, e_2, \cdots, e_{M'}\}$ are produced.

3.2 Outlier Removal (Step2)

There are ill-fitted ellipses in ellipse set $\mathcal{E}$. We introduce three steps how to remove ill-fitted ellipses as follows:

- Set grids $S_{(h,w)}$, $h = 1, \cdots, H$, $w = 1, \cdots, W$, to cover whole eye with reference to eye corner, and cluster each ellipse using the nearest neighbor distance method that is from ellipse center (x_{ei}, y_{ei}) to grid center as shown in Figure 4 (up).

- Each grid $S_{(h,w)}$ has some ellipses, e.g. $\{e_1, e_3, e_{10}\}$, and the average ellipse is represented as $\bar{e}_{S_{h,w}}$. We find ill-fitted

---

[1] λ in this paper is represented by λ_1 in equation (19) in [Tsukada et al. 2011].

ellipses using an overlapped area $g\left(\boldsymbol{e}_j, \bar{\boldsymbol{e}}_{S_{h,w}}\right)$ between average ellipse $\bar{\boldsymbol{e}}_{S_{h,w}}$ and ellipse $\boldsymbol{e}_j$, where j represents the index of ellipses in same grid. The overlapped area is defined by

$$g\left(\boldsymbol{e}_j, \bar{\boldsymbol{e}}_{S_{h,w}}\right) = 1 - \frac{R(\boldsymbol{e}_j) \cap R\left(\bar{\boldsymbol{e}}_{S_{h,w}}\right)}{R(\boldsymbol{e}_j) \cup R\left(\bar{\boldsymbol{e}}_{S_{h,w}}\right)} \qquad (3)$$

where $R(\boldsymbol{e})$ represents the ellipse $\boldsymbol{e}$ surface. $R(\boldsymbol{e}_j) \cup R\left(\bar{\boldsymbol{e}}_{S_{h,w}}\right)$ indicates the union of the area, and $R(\boldsymbol{e}_j) \cap R\left(\bar{\boldsymbol{e}}_{S_{h,w}}\right)$ indicates their intersection. The g has a value from 0 to 1: $g = 0$ means perfectly overlapped, and $g = 1$ means non-overlapped.

- Ellipses that have a higher overlapped-score than threshold τ_G are removed. Then, we remove the grids where the number of ellipses are smaller than threshold τ_N due to low reliability.

Finally, we pick up a constant number of ellipses, τ_{NP}, from each grid for equalization, and set a new ellipse-set $\mathcal{E}' = \{\boldsymbol{e}_1, \boldsymbol{e}_2, \cdots, \boldsymbol{e}_N\}, N \leqq M'$.

3.3 Rough 3D Eye Model Building(Step3)

Given ellipse-set $\mathcal{E}' = \{\boldsymbol{e}_1, \boldsymbol{e}_2, \cdots, \boldsymbol{e}_N\}$, a projected ellipse $\boldsymbol{ep}_i$ is estimated using parameters $\boldsymbol{P}$ and Equation (1a),(1b) and (1c). The 3D eye model parameter $\boldsymbol{P}_0^*$ is estimated by solving the equation:

$$\boldsymbol{P}_0^* = \arg \underset{\boldsymbol{P}}{\text{minimize}}\, G\left(\boldsymbol{e}_i, \boldsymbol{ep}_i\right) \qquad (4)$$

where the cost function G is expressed as $G = \underset{i=1\,to\,N}{\sum} g(\boldsymbol{e}_i, \boldsymbol{ep}_i)/N$. The function g was introduced in Equation (3) for similarity calculation. In order to be robust against outliers, we set a maximum value of g_{MAX}.

Initial ellipse-set $\{\boldsymbol{e}_1, \boldsymbol{e}_2, \cdots, \boldsymbol{e}_N\}$ are not accurate enough to estimate the precise 3D eye model, so we repeat Step1~3 until they converge. During this iterative process, we increase the constraint λ gradually. The convergence criteria defined as the number of iterative reaches predefined parameter τ_{Loop}, $|G_{current} - G_{previous}| \leq \delta_G$ or minimum value G_{Min}.

3.4 3D Eye Model Refinement (Step4)

We refine parameter $\boldsymbol{P}_0^*$ using a cost function of Q. The function Q is introduced by Tsukada and consists of two terms: (1) distance between projected ellipse and edges $d_{i,j}$ ($j = 1, \cdots K_i$), and K_i is the number of edges of i-th image; and (2) the difference of angle between ellipse's normal direction and gradient $\nabla d_{i,j}$ on each edge point. The projected ellipse $\boldsymbol{ep}_i$ is estimated by the Tsukada method where parameter λ is set as $\lambda = 1$, and the cost function Q is expressed by

$$Q\left(d_i, \boldsymbol{ep}_i\right) = \frac{\underset{j=1\,to\,K_i}{\sum}\left(dsitance(d_{i,j}, \boldsymbol{ep}_i) + \beta \cdot angle(\nabla d_{i,j}, \boldsymbol{ep}_i)\right)}{K_i} \qquad (5)$$

where β is a coefficient of angle difference, this is a predefined parameters. Final 3D eye model and camera position are obtained by

$$\boldsymbol{P}^* = \arg \underset{p}{\text{minimize}} \sum_{i=1\,to\,N'} Q\left(d_i, \boldsymbol{ep}_i\right) \qquad (6)$$

In practice, the statistics data say that eye ball size of an adult is almost the same radius 11.5~12.5[mm], and there is limit of iris size, with radius 5~7[mm], so we add these limitations in Equation (4) and (6).
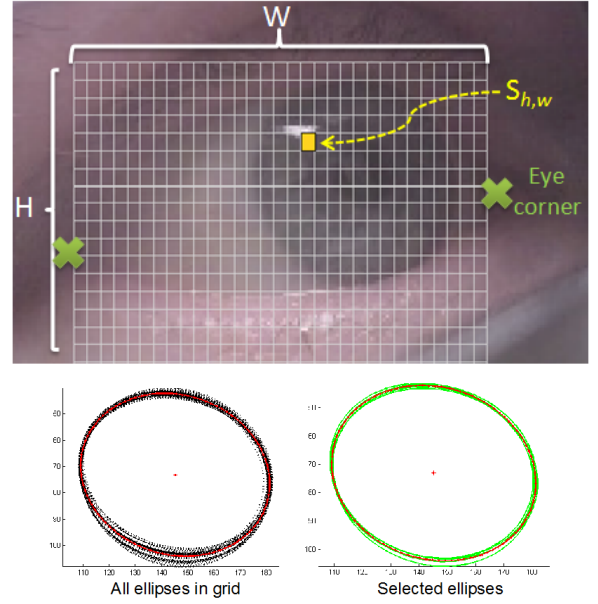

Figure 4: *Outlier removal:(up): grid images for clustering. "Yellow grid" means target grid. (down left): All ellipses in the target grid. Red ellipse is an average ellipse. (down right): Green are selected ellipses in step2. Red is the average ellipse. Large difference ellipses are removed.*

4 Evaluation of Our Method

### 4.1 The Wearable Gaze Tracking Device tested

The device has two cameras: one is for capturing the eye and another is for out-side. The resolution of eye camera is 256×144 (color) and the outside camera is 1280×720 (color). A unit containing cameras and an associated control board with USB interface can be attached on a pair of glasses. The unit connected to a PC for capturing images with synchronization works at ~30 fps.

4.2 Quantitative Evaluation

We evaluate the performance of our proposed method using the error score of ellipse fitting. In the experiments, we applied our method to five people who have various eye color and different eye-camera position. To capture various eye positions for training images, we use monitor and control marker positions flexibly. Throughout our experiments, the number of training data set is 400 and test data set is 100 images, which do not include blink or blurred images. The grid sizes are $H = 36$ and $W = 64$, and predefined parameters are set as follows: $\{\lambda_{initial} = 0.05, \beta = 1.0, \tau_G = 0, \tau_N = 5, \tau_{NP} = 7, \tau_{Loop} = 3, g_{MAX} = 8, \delta_G = 0.1, G_{min} = 30\}$. It takes about 2 hours to build a 3D eye model in Matlab.

Table 1 shows the experimental result: Q score and parameter (eye size and camera position), i.e.$(r_e, r_I, t_x, t_y, t_z)$. The Q score is defined as cost function Q Equation (5). Our method is more accurate than the manual model building. Also robustness, (i.e.standard deviation) is far superior. You can see that the Q score of person D is worse than others, this is because the reflection of lighting was so strong that edges had many outliers in training data set.

Table 1: *Compared our method to manual building model using average and standard deviation of ellipse fitting score.*

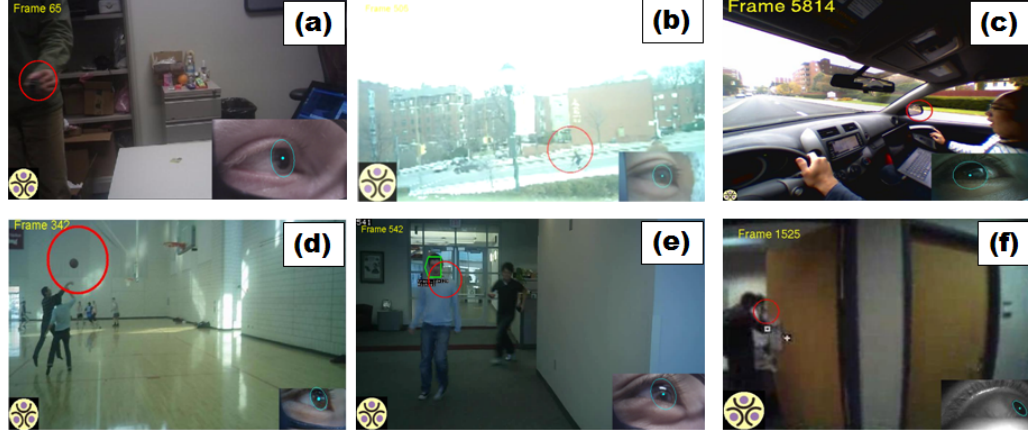|	person A Brown eye	person B dark Brown with contact lens	person C Brown-gray eye	person D Blue-gray eye	person E Blue eye
Q score: Manual[avg,std]	31.02 ± 16.91	32.33 ± 8.63	33.47 ± 19.71	65.80 ± 32.98	50.91 ± 44.18
Q score: Our method[avg,std]	24.22 ± 14.07	27.43 ± 7.60	30.74 ± 8.34	42.35 ± 22.12	35.49 ± 13.61
Camera Position Eyeball, iris size	[−6.25 1.95 50.60] [5.11 11.85]	[−1.99 − 2.30 54.39] [5.40 11.96]	[−3.96 − 4.51 63.04] [5.03 11.61]	[0.20 2.81 54.27] [5.24 11.99]	[−3.20 − 4.81 64.67] [5.21 11.88]

Figure 5: *Examples of gaze tracking results using our method. (a) Apply to blue eye. (b) Outdoor scene. (c) Driving scene. (d) Sports scene. (e)Combination with Face Detector. (f) Apply our method to IR images that are captured by commercial gaze tracking product.*

4.3 Examples of Gaze Tracking using our Eye Model

We apply our eye model building method to the Tsukada's approach. As shown in Figure 5, test data includes several different situations: a) different eye color, b) outdoor, c) driving, d) sports , e) combination with Face Detector and f) apply to IR images (for pupil detection). The gaze tracker that used our model building method works robustly even with different eye color, existing light reflection and occlusion (Please see supplementary materials). In test data f), we apply the exact same method, and we did not add any process such as corneal reflection removal, but it works robustly.

5 Conclusion

We propose a method to build a 3D eye model automatically for wearable gaze tracking. Our method adopts coarse-to-fine approach by adding eye model constraints iteratively, and achieves more accurate and robust ellipse fitting.

References

BASILIO NORIS, J.-B. K., AND BILLAR, A. 2011. A wearable gaze tracking system for children in unconstrained environments. *Computer Vision and Image Understanding*.

HANSEN, D. W., AND JI, Q. 2010. In the eye of the beholder: A survey of models for eyes and gaze. *IEEE Transactions on pattern analysis and machine intelligence 32*, 478–500.

HANSEN, D. W., AND PECE, A. E. 2005. Eye tracking in the wild. *Computer Vision and Image Understanding 98*, 155–181.

HAYHOE, M., AND BALLAR, D. 2005. Eye movements in natural behavior. *TRENDS in Cognitive Sciences Vol.9 No.4*, 188–194.

KANADE, T. 2009. First person vision. In *First Workshop on Egocentric Vision(in conjunction with CVPR2009)*.

LI, D., AND PARKHURST, D. 2006. Open-source software for real-time visible- spectrum eye tracking. In *The 2nd Conference on Communication by Gaze Interaction (COGAIN)*.

MORIMOTO, C. H., AND MIMICA, M. R. 2005. Eye gaze tracking techniques for interactive applications. *Computer Vision and Image Understanding 98 Issue 1*, 4–24.

NAC. -EMR-9. Nac-emr-9 eye tracking products. In *http://www.nacinc.com/products/Eye-Tracking-Products/EMR-9/*.

OHNO, T., AND MUKAWA, N. 2004. A free-head, simple calibration, gaze tracking system that enables gaze-based interaction. In *ETRA: eye tracking research & application symposium*.

PÉREZ, A., CÓRDOBA, M. L., GARCÍA, A., MÉNDEZ, R., MUÑOZ, M. L., PEDRAZA, J. L., AND SÁNCHEZ, F. 2003. A precise eye-gaze detection and tracking system. In *WSCG*.

TAN, K.-H., KRIEGMAN, D. J., AND AHUJA, N. 2002. Appearance-based eye gaze estimation. In *Workshop on Applications of Computer Vision*.

TSUKADA, A., SHINO, M., DEVYVER, M., AND KANADE, T. 2011. Illumination-free gaze estimation method for first-person vision wearable device. In *Computer Vision in Vehicle Technology:From Earth to Mars(In conjunction with ICCV 2011)*.

VESTER-CHRISTENSEN, M., LEIMBERG, D., ERSBØLL, B. K., AND HANSEN, L. K. 2005. Deformable models for eye tracking. In *Den 14. Danske Konference i Mønstergenkendelse og Billedanalyse*.

WU, H., KITAGAWA, Y., WADA, T., KATO, T., AND CHEN, Q. 2007. Tracking iris contour with a 3d eye-model for gaze estimation. In *Asian Conference on Computer Vision*.