Two-Eye Model-Based Gaze Estimation from A Kinect Sensor

Xiaolong Zhou[1], Haibin Cai[2], Youfu Li[3], and Honghai Liu[2,4]

***Abstract*— In this paper, we present an effective and accurate gaze estimation method based on two-eye model of a subject with the tolerance of free head movement from a Kinect sensor. To accurately and efficiently determine the point of gaze, i) we employ two-eye model to improve the estimation accuracy; ii) we propose an improved convolution-based means of gradients method to localize the iris center in 3D space; iii) we present a new personal calibration method that only needs one calibration point. The method approximates the visual axis as a line from the iris center to the gaze point to determine the eyeball centers and the *Kappa* angles. The final point of gaze can be calculated by using the calibrated personal eye parameters. We experimentally evaluate the proposed gaze estimation method on eleven subjects. Experimental results demonstrate that our gaze estimation method has an average estimation accuracy around 1.99°, which outperforms many leading methods in the state-of-the-art.**

I. Introduction

Gaze estimation is to determine the point of regard of a person, which plays an important role in understanding human attention, feelings, and desires. It has been widely explored in many intelligent systems for virtual reality, human-computer interaction, human-robot interaction, human behavior analysis and so on. Some gaze estimation researchers concentrated on using the pupil center corneal reflection technique. This kind of technique normally requires one or multiple infrared lights and high-quality cameras, which limits the system's potential for broader applications. Moreover, most of the existing gaze estimation systems have low tolerance toward head movement, which hinders them from being widely used.

Recently, Kinect-based 3D gaze estimation [1], [2], [3], [4], [5], [6], [7], [8] has attracted increasing attention since it is low-cost, non-intrusive, simple-setup and it allows free head movements. Generally, Kinect-based gaze estimation methods can be roughly classified into non-eye model-based methods and eye model-based methods. Non-eye model-based methods are typically appearance-based or regression-based. For example, Mora and Odobez [1] estimated 3D gaze from multimodal Kinect data and achieved an estimation accuracy with average error around 7.6°−12.6°. Furthermore, they proposed a geometric generative 3D gaze estimation method [2] based on an appearance generative process that modeled head-pose rectified eye images recovered by using of RGB-D cameras, which improved the estimation accuracy to 6.3°. Cazzato et al. [3] incorporated the 3D head pose to estimate the final gaze direction according to the geometric relations among the sensor, observer and target. They reported the estimation errors for unaware users with 6.9° while for informed users with 3.6°. The main benefit of non-eye model-based methods are specific personal calibration free. However, the estimation accuracy of this kind of method is low (generally above 6°).

Different from the non-eye model-based methods that estimate the gaze using appearance or regression technique, 3D eye model-based methods directly determine the gaze using the geometric relationship among human eyes, sensors and gazing points. For example, J. Li and S. Li [4] proposed an eye-model-based 3D gaze estimation method from a Kinect sensor. They built a head model based on the Kinect sensor and calibrated the eyeball center by gazing at a target in 3D space. The gaze direction was estimated after the calibration and the reported average error of estimation was around 6°. Recently, they estimated the gaze from color image based on an eye model with known head pose [5]. They first determined the 3D eyeball center in calibration manner by gazing at the center of the color image camera, and then estimated the 3D iris center using the information of its contour and projection. They reported the average estimation errors for seven subjects with 5.9° vertically and 4.4° horizontally. Sun et al. [6] estimated the gaze direction based on a 3D geometric eye model by considering the head movement and deviation of the visual axis from the optical axis. They reported a high estimation accuracy of 1.4°-2.7°. However, the proposed method involved many calibration procedures like screen-camera calibration and personal calibration with multiple calibration points.

Although eye model-based gaze estimation methods can achieve a higher accuracy (below 6°), this kind of method normally require specific personal calibration, which involves human interactions. Moreover, the estimation accuracy greatly relies on the number of calibration points. Generally, more calibration points will lead to higher estimation accuracy while at the same time require more human interactions.

Besides the personal calibration, the 3D location of human's iris is another key technique that affects the final gaze estimation accuracy. Currently, a large number of iris center

This work was supported in part by the National Natural Science Foundation of China (61403342, 61673329, U1509207, 61325019, 51575338) and Hubei Key Laboratory of Intelligent Vision Based Monitoring for Hydroelectric Engineering (2014KLA09).

[1]Xiaolong Zhou is with the College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou, China. zxl@zjut.edu.cn

[2]Haibin Cai and Honghai Liu are with the School of Computing, University of Portsmouth, Portsmouth, UK.

[3]Youfu Li is with the Department of Mechanical and Biomedical Engineering, City University of Hong Kong, Hong Kong, China.

[4]Honghai Liu is with the State Key Laboratory of Mechanical Systems and Vibration, School of Mechanical Engineering, Shanghai Jiao Tong University, Shanghai, China.

localization methods has been explored. Despite significant progress, fast and accurate iris center localization is still a challenging problem due to the individuality of eyes, occlusion, dynamics and illumination. Moreover, although high accurate iris center location can be obtained through high quality eye tracking systems, the intrusive or expensive devices make these methods unattractive [9], [10].

To leverage the accuracy and automation, we propose an eye model-based gaze estimation method with one calibration point. Moreover, we present a non-intrusive and fast iris center localization method for low resolution images from a Kinect sensor. In our system, only one Kinect sensor is employed and the subject's gaze can be estimated with free head movements. Our method can be used for various vision-based applications and hence has significant practical impact. The main contributions follow.

1) An effective two-eye model-based gaze estimation method that can achieve a relative low average estimation error (about 1.99°) with free head movements from a Kinect sensor is proposed. Different from the conventional single eye model-based gaze estimation methods, the proposed method averages the gazes of both eyes for a final gaze estimation. The experimental results demonstrate that the proposed method outperforms state-of-the-art Kinect-based gaze estimation methods.

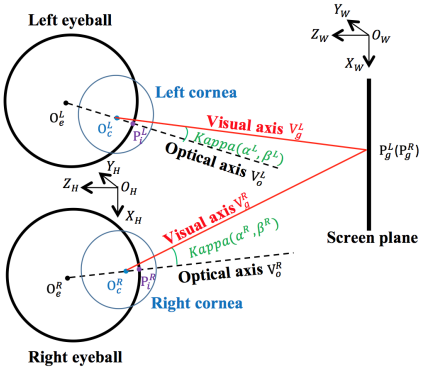
2) An improved convolution-based means of gradients iris center localization method is presented. Compared with the conventional means of gradients method, the improved method either improves the accuracy or dramatically reduces the computational cost.

3) A new personal calibration method by approximating the visual axis as a line from the iris center to the gaze point is proposed to estimate the eye parameters with only one calibration point.

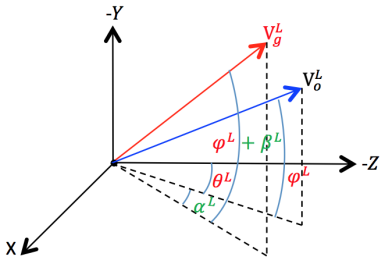
The paper is organized as follows. Section II introduces the overview of proposed two-eye model-based gaze estimation method. Section III details the 3D iris center localization method and eyeball centers and *Kappa* angles estimation method. Some experimental results are discussed in Section IV, and followed by concluding remarks in Section V.

II. Two-Eye Model-Based Gaze Estimation

Fig. 1a illustrates the proposed 3D model of two eyes of a subject. $\mathbf{O}_e^L$ (or $\mathbf{O}_e^R$), $\mathbf{O}_c^L$ (or $\mathbf{O}_c^R$), and $\mathbf{P}_i^L$ (or $\mathbf{P}_i^R$) denote the centers of eyeball, cornea, and iris of left eye (or right eye), respectively. The dash lines through the centers of eyeball, cornea and iris represent the optical axes for both eyes. $\mathbf{V}_o^L$ (or $\mathbf{V}_o^R$) is a unit vector of optical axis of left eye (or right eye). The red lines from the corneal center to the point of gaze on the screen plane denote the visual axes for both eyes. $\mathbf{V}_g^L$ (or $\mathbf{V}_g^R$) is a unit vector of visual axis of left eye (or right eye). The angle of deviation of the visual axis from the optical axis is known as the *Kappa*, which almost keeps constant for a subject. As shown in Fig. 1b, α^L (or α^R) and β^L (or β^R) are the horizontal and vertical components of the *Kappa* angle, respectively. $X_W Y_W Z_W$ and $X_H Y_H Z_H$ represent for the world coordinate system and the head coordinate



(a)



(b)

Fig. 1: An illustration of proposed 3D eye model and *Kappa* angle. (a) Top view of two eyes and screen plane. (b) Space relationship between the optical axis and visual axis.

system, respectively. In our experiment, the world coordinate system is built in the camera center of the Kinect sensor while the head coordinate system is in the center of two inner eye corners of a subject. When a subject looks at a point on the screen plane, the points of gaze of left and right eyes are represented as $\mathbf{P}_g^L$ and $\mathbf{P}_g^R$, respectively. In this paper, both eyes are modeled to effectively estimate the point of gaze. Without explicit description, all the parameters involved are relative to the world coordinate system and all the vectors are column vectors.

Take left eye as an example, the point of gaze can be calculated by

$$\mathbf{P}_g^L = \mathbf{O}_e^L + c \cdot \mathbf{V}_o^L + \lambda^L \cdot \mathbf{V}_g^L \tag{1}$$

where $c = \left\|\mathbf{O}_e^L \mathbf{O}_c^L\right\|_2$ is a constant, and normally it is approximately 5.3mm [11]. $\lambda^L = \left\|\mathbf{O}_c^L \mathbf{P}_g^L\right\|_2$ can be obtained by

$$\lambda^L = -\frac{(\mathbf{O}_e^L + c \cdot \mathbf{V}_o^L)^T \cdot \mathbf{V}_s + n}{(\mathbf{V}_g^L)^T \cdot \mathbf{V}_s} \tag{2}$$

where $(\cdot)^T$ is the transpose of a vector. $\mathbf{V}_s$ and n are parameters of screen plane function and can be determined from the camera-screen calibration [6]. For any point $\mathbf{P}_g$ on the screen plane, we have $\mathbf{P}_g \cdot \mathbf{V}_s = -n$.

The eyeball center $\mathbf{O}_e^L$ is variable for different head pose in the world coordinate system, but it keeps constant related to the head coordinate system. So, we can first estimate the eyeball center $\mathbf{O}_e^{L,H}$ in the head coordinate system and then obtain the eyeball center $\mathbf{O}_e^L$ in the world coordinate system

by rotating and translating $\mathbf{O}_e^{L,H}$ with the estimated head pose $\{\mathbf{R}_t, \mathbf{t}_t\}$. $\mathbf{R}_t$ and $\mathbf{t}_t$ are the rotation matrix and translation matrix of head pose at time t, respectively. The details of head pose estimation are stated in section IV. A.

$$\mathbf{O}_e^L = \mathbf{R}_t \cdot (\mathbf{O}_e^{L,H})^T + \mathbf{t}_t \tag{3}$$

The unit vector of optical axis $\mathbf{V}_o^L$ is calculated according to the eyeball center $\mathbf{O}_e^L$ and iris center $\mathbf{P}_i^L$.

$$\mathbf{V}_o^L = \frac{\mathbf{P}_i^L - \mathbf{O}_e^L}{r_e} \tag{4}$$

where $r_e = \left\|\mathbf{P}_i^L \mathbf{O}_e^L\right\|_2$ is the radius of eyeball.

As shown in Fig. 1b, once the unit vector of optical axis $\mathbf{V}_o^L$ has been determined, its horizontal angle θ^L and vertical angle φ^L can be calculated.

$$\mathbf{V}_o^L = \begin{bmatrix} cos(\varphi^L)sin(\theta^L) \\ sin(\varphi^L) \\ -cos(\varphi^L)cos(\theta^L) \end{bmatrix} \tag{5}$$

Then, the unit vector of visual axis $\mathbf{V}_g^L$ can be calculated by rotating the optical axis with the *Kappa* angle.

$$\mathbf{V}_g^L = \begin{bmatrix} cos(\varphi^L + \beta^L)sin(\theta^L + \alpha^L) \\ sin(\varphi^L + \beta^L) \\ -cos(\varphi^L + \beta^L)cos(\theta^L + \alpha^L) \end{bmatrix} \tag{6}$$

where α^L and β^L denote the horizontal angle and vertical angle of the *Kappa* angle, respectively.

So far, we can estimate the point of gaze of left eye $\mathbf{P}_g^L$ according to the aforementioned equations. However, eye model parameters, $\mathbf{R}_t$, $\mathbf{t}_t$, $\mathbf{P}_i^L$, $\mathbf{O}_e^{L,H}$, α^L and β^L, should be determined beforehand. We will detail the proposed methods for estimating these parameters in section III.

Similarly, the point of gaze of right eye $\mathbf{P}_g^R$ can be estimated once the eye model parameters of right eye are determined.

To further improve the estimation accuracy, we calculate the final point of gaze $\mathbf{P}_g$ by averaging the estimated gazes of left and right eyes based on the fact that both eyes are gazing at a same point on the screen.

$$\mathbf{P}_g = \frac{1}{2}(\mathbf{P}_g^L + \mathbf{P}_g^R) \tag{7}$$

In our experiment, two stages including calibration stage and test stage are implemented. The calibration stage aims to estimate the personal eye model parameters $(\mathbf{O}_e^{L,H}, \mathbf{O}_e^{R,H}, \alpha^L, \beta^L, \alpha^R, \beta^R)$ with a given calibration point, while the test stage is to estimate the actual gaze point of the subject based on the estimated eye model parameters.

III. Eye Model Parameters Estimation

A. Head Pose Estimation

We first detect the face region in the RGB image using an appearance-based boosted cascade face detector [12] with default parameters. After the face region has been identified, we employ a fast and accurate supervised descent method (SDM) [13] to detect and track the facial features. After the facial features of a subject at time t have been detected, the corresponding 3D coordinates of the features can be obtained by the calibrated Kinect sensor. We then model the 3D face of this subject at time t using the obtained 3D features as $\mathbf{X}_t$. Typically, the face model of each subject is person-specific.

The goal of head pose estimation is to determine the head rotation matrix $\mathbf{R}_t$ in terms of yaw, pitch and roll, and translation vector $\mathbf{t}_t$, at time t. To calculate the head pose, a reference face model of the subject should be built first. We require each subject to keep frontal to the Kinect sensor for a certain time and calculate the average to form a reference model $\mathbf{X}^r$. Then, the head pose of a subject at time t can be determined by minimizing the following equation.

$$\arg\min_{\mathbf{R}_t, \mathbf{t}_t} \|\mathbf{R}_t \mathbf{X}^r + \mathbf{1}_{1\times n} \otimes \mathbf{t}_t - \mathbf{X}_t\| \tag{8}$$

where $\mathbf{1}_{1\times n}$ denotes a row vector of ones of size n (n is the number of feature points), $\otimes$ represents the Kronecker produce. We can solve Eq. (8) using Singular Value Decomposition [14] and then obtain the head pose $\{\mathbf{R}_t, \mathbf{t}_t\}$ of the subject at time t.

B. Improved Means of Gradients Iris Center Localization

The means of gradients method [15] has attracted considerable attention due to its easy implementation and high accuracy, which has been reported to have the best average performance for eye center (or iris center) localization. It makes use of the relationship between a possible iris center and the vector field of all the image gradients.

$$\begin{aligned} c' &= \arg\max_c \left\{ \frac{1}{N} \sum_{i=1}^{N} (\mathbf{d}_i^T \cdot \mathbf{g}_i)^2 \right\} \\ \mathbf{d}_i &= \frac{x_i - c}{\|x_i - c\|_2} \qquad \forall i : \|\mathbf{g}_i\|_2 = 1 \end{aligned} \tag{9}$$

where c' denote the located iris center position and c is the possible iris center. The dot product will reach the biggest if the displacement vector d_i and the gradient vector g_i have the same orientation which will happen if the point x_i lies on the boundary of the circle whose center point is c. The displacement vector d_i and gradient vector g_i are scaled to unit length to obtain an equal weight for all pixel positions. N is the number of pixels of the image. The algorithm calculates dot products of the normalised displacement vectors and the gradient vectors for every possible iris center. Each pixel in the image is a potential iris center. The pixel that has the maximum value of mean of dot products is regarded as the final iris center.

Although the means of gradients method can locate iris center accurately, the heavy computational cost hampers its real time applications. The computational complexity of this method is $O(N^2)$, where N stands for the number of pixels of the eye area. The algorithm calculates the dot product of all the displacement vectors d_i and the gradient vectors g_i. Thus for a possible iris center, all the pixels in the eye image are used for the dot product. Although the computational complexity can be decreased by considering only the same

orientation displacement vectors and the gradient vectors that have a significant magnitude, the accuracy will drop dramatically. To remedy this, we propose a convolution-based means of gradients method which is capable of reducing the computational cost while at the same time improving the accuracy. In the proposed method only the pixels on the circular boundary of a possible iris center are used to calculate the dot product. So the computational complexity can be greatly reduced. Meanwhile, the negative influence of other points such as eyelids and eye corners in the dot product can also be eliminated. Different sizes of masks are built to convolute the eye images. Each mask contains a circle whose center point is at the center of the masks and the pixels value on the boundary of the circle are normalised. Assuming the radius of the circle is r, then both the width and height of the built mask will be $2r+1$.

We propose to apply convolution to the dot product and the corresponding equation can be further extended to the following.

$$\begin{aligned}
&\sum_{i=1}^{n} \mathbf{d}_i^T \cdot \mathbf{g}_i \\
&= \sum_{i=1}^{n} (x_{di} x_{gi} + y_{di} y_{gi}) \\
&= \sum_{i=1}^{n} \big((x_i - x_c) x_{gi} + (y_i - y_c) y_{gi}\big) \\
&= \sum_{i=1}^{n} (x_i x_{gi}) - x_c \sum_{i=1}^{n} x_{gi} + \sum_{i}^{n} (y_i y_{gi}) - y_c \sum_{i}^{n} y_{gi}
\end{aligned} \tag{10}$$

where (x_{di}, y_{di}) is the coordinate of d_i, which can be calculated by the difference between the circular boundary point (x_i, y_i) and the possible iris center (x_c, y_c). (x_{gi}, y_{gi}) is the coordinate of g_i, which can be calculated through partial derivatives or other methods by computing image gradients. We build two position images I_{px} and I_{py} of pixels for x and y positions (shown as Fig. 2), respectively. The size of the position image is the same as the size of eye region image. Similarly, two gradient images I_{gx} and I_{gy} of pixels for x and y directions also can be obtained. By doing so, the $\sum_{i=1}^{n}(x_i x_{gi})$ can be calculated by firstly multiplying position image I_{px} with gradient image I_{gx} and then using the former designed mask to convolute the result. The $x_c \sum_{i=1}^{n} x_{gi}$, similarly, can be calculated by firstly convoluting the designed mask with the gradient image I_{gx} and then multiplying the result with the position image I_{py}.

Since only the pixels on the boundary of the circles are used to calculate the dot products, the other pixels of the eye image cannot affect the result. Thus we propose to directly use the sum of the dot products rather than the square of the dot products. The final position of iris center is determined by searching the maximum of the following equation.

$$\max_{(r,x_0,y_0)} \left(\frac{1}{r} \sum_{i=1}^{n} \mathbf{d}_i^T \cdot \mathbf{g}_i\right) \tag{11}$$

1	2	3	…	x
1	2	3	…	x
1	2	3	…	x
1	2	3	…	x
1	2	3	…	x

(a)

1	2	3	…	y
1	2	3	…	y
1	2	3	…	y
1	2	3	…	y
1	2	3	…	y

(b)

Fig. 2: Illustration of two position images. (a) The position image I_{px} for x position, x is equal to the width of the eye image (b) The position image I_{py} for y position, y is equal to the height of the eye image

where (x_0, y_0) represents the coordinate of iris center. To locate the iris center, the proposed method searches the maximum of Eq. (11) by changing the values of radius and center points. The FFT is employed in the realisation of convolution where only 2 cycles DFT and 1 cycle IDFT are performed. So the computational complexity of the convolution-based means of gradients is $O(P\log_2(P)N)$, where P satisfies $P \leq X+Y+C$. X, Y is the number of rows and columns of the center coordinate and C is a constant number. Compared to the computational complexity $O(N^2)$ of the conventional means of gradients method, the proposed method significantly improves the processing speed.

So far, iris center coordinate $\mathbf{p}_i^L$ (or $\mathbf{p}_i^R$) in the image plane can be effectively and efficiently determined according to the proposed method. Then, its coordinate in 3D space $\mathbf{P}_i^L$ (or $\mathbf{P}_i^R$) can be obtained by incorporating the depth information captured by the Kinect sensor.

C. Personal Calibration for Eyeball Center and Kappa Angle Calculation

Conventionally, the line from the corneal center to the point of gaze on the screen plane is defined as the visual axis for an eye [16], [4], [5], [6], shown as dash lines $\mathbf{O}_c^L\mathbf{P}_g$ and $\mathbf{O}_c^R\mathbf{P}_g$ in Fig. 3. To estimate the eye model parameters with the conventional visual axis, the eyeball center is assumed to be known [16] or many calibration points are required [4], [5], [6]. To automatically estimate the eyeball center as well as the *Kappa* angle with less calibration points, we approximate the visual axis $\mathbf{V}_g^{L'}(\mathbf{V}_g^{R'})$ as a line from the iris center to the point of gaze on the screen plane (shown as lines $\mathbf{P}_i^L\mathbf{P}_g$ and $\mathbf{P}_i^R\mathbf{P}_g$ in Fig. 3). It is reasonable to make such a approximation since the difference between the new *Kappa* angle and the original *Kappa* angle is too little to be negligible. Take left eye for instance, the new *Kappa* angle $\{\alpha^{L'}, \beta^{L'}\}$ and the original *Kappa* angle $\{\alpha^L, \beta^L\}$ have the following relationship.

$$\angle Kappa\{\alpha^{L'}, \beta^{L'}\} = \angle Kappa\{\alpha^L, \beta^L\} + \angle \mathbf{O}_c^L\mathbf{P}_g\mathbf{P}_i^L \tag{12}$$

where $\angle \mathbf{O}_c^L\mathbf{P}_g\mathbf{P}_i^L$ is very small because the distance from the corneal center to the iris center is far smaller than the distance from the iris center to the gaze point.

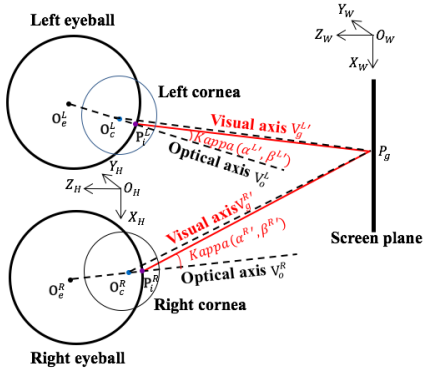


Fig. 3: An illustration of the simplified eye model.

With this simplified model, we can simultaneously estimate the eyeball center $\mathbf{O}_e^{L,H}$ and *Kappa* angle $\{\alpha^{L'}, \beta^{L'}\}$ with only one calibration point $\mathbf{P}_g$. The following six steps details the proposed method.

Step1: Asking the subject to look at a calibration point $\mathbf{P}_g$ with known coordinate (x_g, y_g, z_g). Estimating the subject's head pose $\{\mathbf{R}_t, \mathbf{t}_t\}$ and 3D iris center $\mathbf{P}_i^L$ according to the methods in Sections III.A and III.B.

Step 2: Calculating the eyeball center $\mathbf{O}_e^L$ in the world coordinate system based on Eq. (3) and then calculating the unit vector of optical axis $\mathbf{V}_o^L$ based on Eq. (4). According to Eq. (3)-(5), we have:

$$\frac{\mathbf{P}_i^L - \mathbf{R}_t \cdot (\mathbf{O}_e^{L,H})^T - \mathbf{t}_t}{r_e} = \begin{bmatrix} cos(\varphi^L) sin(\theta^L) \\ sin(\varphi^L) \\ -cos(\varphi^L) cos(\theta^L) \end{bmatrix} \tag{13}$$

$$r_e = \left\| \mathbf{P}_i^L - \mathbf{R}_t \cdot (\mathbf{O}_e^{L,H})^T - \mathbf{t}_t \right\|_2 \tag{14}$$

where r_e is the radius of eyeball and it is approximately 12.4mm (axial) [17].

Step 3: Calculating the unit vector of visual axis $\mathbf{V}_g^{L'}$ based on Eq. (6) by rotating the optical axis with the *Kappa* angle $\{\alpha^{L'}, \beta^{L'}\}$. Also, the $\mathbf{V}_g^{L'}$ can be obtained according to the given calibration point $\mathbf{P}_g$ and estimated iris center point $\mathbf{P}_i^L$. Then we have:

$$\begin{bmatrix} cos(\varphi^L + \beta^{L'}) sin(\theta^L + \alpha^{L'}) \\ sin(\varphi^L + \beta^{L'}) \\ -cos(\varphi^L + \beta^{L'}) cos(\theta^L + \alpha^{L'}) \end{bmatrix} = \frac{\mathbf{P}_g - \mathbf{P}_i^L}{\left\| \mathbf{P}_g \mathbf{P}_i^L \right\|_2} \tag{15}$$

Step 4: Estimating the eyeball center coordinate $\mathbf{O}_e^{L,H}(x, y, z)$, optical axis angles $\{\theta^L, \varphi^L\}$ and *Kappa* angle $\{\alpha^{L'}, \beta^{L'}\}$ according to Eq. (13)-(15) by gazing a calibration point with multiple head poses.

So far, we have estimated the eyeball center coordinate, optical axis angles and *Kappa* angle according to the simplified eye model. Based on these estimations, we can then further update the *Kappa* angle according to the actual visual axis in the normal eye model.

Step 5: Assume that the optical axis is fixed after the simplified model-based estimation, which means the estimated $\mathbf{O}_e^{L,H}$ and $\{\theta^L, \varphi^L\}$ are constant. We can then calculate the corneal center as:

$$\mathbf{O}_c^L = \mathbf{O}_e^L + c \cdot \mathbf{V}_o^L = \mathbf{R}_t \cdot (\mathbf{O}_e^{L,H})^T + \mathbf{t}_t + c \cdot \mathbf{V}_o^L \tag{16}$$

$$\mathbf{V}_o^L = \frac{\mathbf{P}_i^L - \mathbf{R}_t \cdot (\mathbf{O}_e^{L,H})^T - \mathbf{t}_t}{\left\| \mathbf{P}_i^L \mathbf{O}_e^L \right\|_2} \tag{17}$$

Step 6: Based on the calculated corneal center $\mathbf{O}_c^L$ and the calibration point $\mathbf{P}_g$, we can determine the final *Kappa* angle $\{\alpha^L, \beta^L\}$ by solving the following equation.

$$\begin{bmatrix} cos(\varphi^L + \beta^L) sin(\theta^L + \alpha^L) \\ sin(\varphi^L + \beta^L) \\ -cos(\varphi^L + \beta^L) cos(\theta^L + \alpha^L) \end{bmatrix} = \frac{\mathbf{P}_g - \mathbf{O}_c^L}{\left\| \mathbf{P}_g \mathbf{P}_c^L \right\|_2} \tag{18}$$

Similarly and simultaneously, we can estimate the $\mathbf{O}_e^{R,H}$ and $\{\alpha^R, \beta^R\}$ of right eye with the same calibration point. Note that once the $\mathbf{O}_e^{L,H}$ and $\mathbf{O}_e^{R,H}$ have been determined, they remain constant no matter head moves or gaze changes because they are fixed with respect to the head coordinate system.

In our experiment, the stage for estimating the eyeball center and *Kappa* angle is so called calibration stage. Before gaze estimation, each subject is first required to look at a given calibration point with multiple head poses to estimate his/her intrinsic eye parameters ($\mathbf{O}_e^{L,H}, \mathbf{O}_e^{R,H}, \alpha^L, \beta^L, \alpha^R, \beta^R$). After that, the estimated parameters will be employed to estimate any point of gaze of the subject.

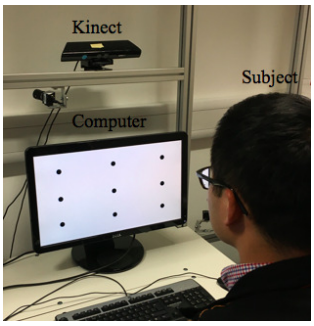


Fig. 4: System setup of our gaze estimation method with a Kinect sensor.

IV. Experimental Evaluation

Our system only uses a Kinect sensor that is mounted above the computer monitor, as shown in Fig. 4. Each subject is required to sit in front of the monitor and to keep his/her head in the field of view of the Kinect. We test the proposed two-eye model-based gaze estimation method on eleven subjects.

There are two stages, calibration stage and test stage, for each subject to gaze estimation. The calibration stage is to estimate the eye parameters of each subject. These estimated parameters are then used in the test stage for determining the gaze of a subject. Due to the limited measure range (80cm-400cm in default mode) of the Kinect sensor, the

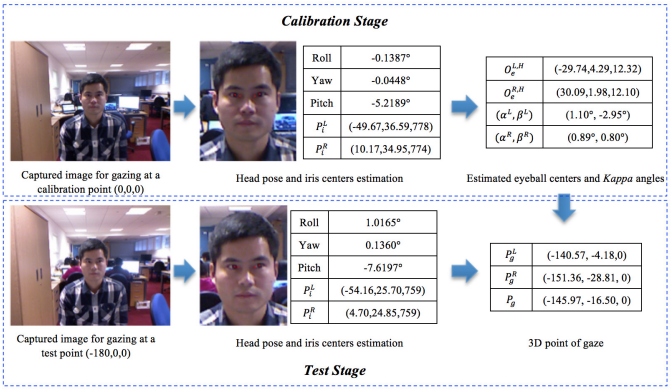


Fig. 5: An example of estimating one subject's 3D point of gaze.

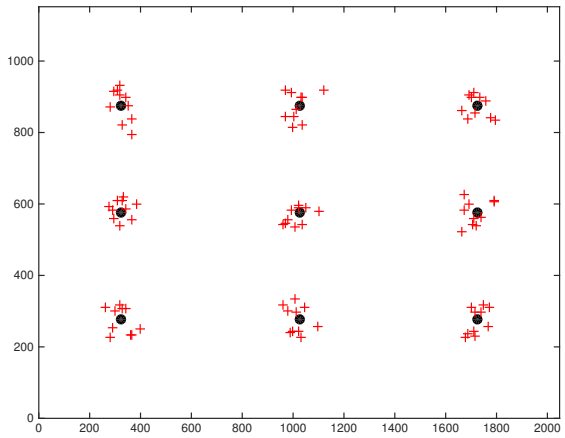


Fig. 6: Estimated points of gaze of subject 1 on the screen plane. The black dots and red crosses represent the ground truth points and the estimated points of gaze, respectively.

distance between each subject and the Kinect sensor should not exceed the limit in the experiment.

In the calibration stage, each subject is required to look at a given calibration point with two different head poses. First, the head poses and 3D iris centers of both eyes are obtained according to Sections III.A and III.B, respectively. The eyeball centers and *Kappa* angles are then estimated according to Section III.C. Note that the subject can freely gaze with any head positions and poses as long as the head is within the field of view of the Kinect sensor.

Take one subject as an example (shown as Fig. 5). The subject is first asked to gaze at a point (0,0,0), and then his head pose can be estimated as roll -0.1387°, yaw -0.0448°, and pitch -5.2189°. Also, 3D iris centers of left and right eyes can be obtained as coordinates (-49.67,36.59,778) and (10.17,34.95,774), respectively. Finally, we can estimate the eyeball centers ((-29.74,4.29,12.32) for left eyeball and (30.09,1.98,12.10) for right eyeball) in the head coordinate system and *Kappa* angles ((1.10°,-2.95°) for left eye and (0.89°,0.80°) for right eye) of both eyes based on the proposed method.

Similarly, we can estimate the eye model parameters of other ten subjects and the results are listed in Table I.

In the test stage, the subject is required to gaze at a ground truth point and the proposed method is tested to estimate the actual point of gaze. Note that the subject can freely move his head and change his gaze direction as he/she wants. The only constraint is that his/her head should not be out of view of the field of the Kinect sensor. Otherwise, the sensor could fail to capture the eye images and thus result in failure of gaze estimation.

As shown in Fig. 5, the subject is gazing at a test point (-180,0,0) with estimated head pose (roll 1.0165°, yaw 0.1360°, and pitch -7.6197°) and iris centers ((-54.16,25.70,759) for left iris and (4.70,24.85,759) for right iris). By incorporating the eye model parameters obtained in the calibration stage, the points of gaze of both eyes ((-140.57,-4.18,0) for left eye gaze and (-151.36,-28.81,0) for right eye gaze) can be calculated. The final point of gaze (-145.97,-16.50,0) is then determined by averaging the estimated gazes of two eyes.

A. Evaluation of gazing multiple ground truth points at a same position

To quantitatively analyze the accuracy of estimated point of gaze, an angular degree α_g [18] is calculated.

$$\alpha_g = arctan\,(D_{gg}/D_{ss}) \tag{19}$$

where D_{gg} denotes the distance between the estimated point of gaze and the ground truth point, and D_{ss} is the distance between the subject and the screen plane. The smaller α_g is, the higher of estimation accuracy is.

By using the eye parameters estimated at calibration stage (shown in Table I), the angular degree of each gaze of each subject can be determined according to the proposed method at test stage. In our experiments, nine evenly distributed ground truth points on the screen (as shown in Fig. 4) are used to evaluate the proposed gaze estimation method. When gazing at each point, the subject is asked to rotate his/her head step-by-step in roll, pitch, and yaw, respectively, untill he/she cannot see the point.

Fig. 6 illustrates the distribution of estimated points of gaze of one subject (subject 1) when gazing at nine ground truth points with a distance of D_{ss}=1000mm.

Table. II shows the average accuracy (angular degree) and tolerance of head movements of gazing nine ground truth points of each subject at a same position (D_{ss}=1000mm). The head movement tolerance involved in the table denotes the maximum rotation angles for roll, pitch, and yaw, respectively. The results show that our method can achieve a good gaze estimation performance with an average accuracy is 1.99° under an average head pose with roll 11.63°, pitch 13.44° and yaw 9.43°.

TABLE I: Estimated Eye Parameters of Eleven Subjects.

Subjects	$\mathbf{O}_e^{L,H}$	$\mathbf{O}_e^{R,H}$	$\{\alpha^L, \beta^L\}$	$\{\alpha^R, \beta^R\}$
1	$[-29.74, 4.29, 12.32]^T$	$[30.09, 1.98, 12.10]^T$	$\{1.10°, -2.95°\}$	$\{0.89°, 0.80°\}$
2	$[-32.29, 4.76, 12.78]^T$	$[31.03, -2.54, 12.45]^T$	$\{-2.63°, -3.27°\}$	$\{1.45°, 2.24°\}$
3	$[-28.18, 1.46, 15.93]^T$	$[32.49, -0.50, 14.94]^T$	$\{-0.10°, -3.07°\}$	$\{0.23°, -2.01°\}$
4	$[-33.50, 2.99, 13.74]^T$	$[35.58, -0.74, 13.83]^T$	$\{-0.41°, 0.10°\}$	$\{3.21°, -1.60°\}$
5	$[-30.32, 3.05, 15.43]^T$	$[32.19, -0.01, 15.66]^T$	$\{-1.31°, -3.64°\}$	$\{-1.79°, -3.31°\}$
6	$[-30.67, 0.73, 12.16]^T$	$[29.25, 3.09, 12.04]^T$	$\{-1.45°, -1.75°\}$	$\{-2.49°, -3.14°\}$
7	$[-30.94, 3.28, 10.36]^T$	$[29.99, 0.55, 10.89]^T$	$\{1.95°, 1.58°\}$	$\{-1.84°, 2.49°\}$
8	$[-30.68, 0.92, 12.07]^T$	$[32.45, 1.02, 13.23]^T$	$\{-0.28°, 0.47°\}$	$\{-0.25°, 0.93°\}$
9	$[-31.97, -5.64, 12.40]^T$	$[32.25, 2.14, 13.25]^T$	$\{-2.47°, 0.94°\}$	$\{-2.50°, 1.78°\}$
10	$[-28.58, 3.87, 15.37]^T$	$[29.90, -3.81, 12.67]^T$	$\{0.22°, 0.61°\}$	$\{-1.07°, -0.42°\}$
11	$[-30.22, 1.36, 12.37]^T$	$[30.03, 2.84, 12.85]^T$	$\{-0.97°, -1.26°\}$	$\{-1.56°, -2.19°\}$

TABLE II: Average Estimated Gaze Accuracy and Tolerance of Head Movements.

Subjects	Average accuracy	Head movements tolerance
1	2.12°	$12.8° \times 14.7° \times 9.6°$
2	2.04°	$11.6° \times 13.9° \times 9.3°$
3	1.90°	$13.2° \times 14.5° \times 10.1°$
4	1.87°	$11.7° \times 13.3° \times 9.4°$
5	2.07°	$10.3° \times 11.8° \times 8.2°$
6	1.96°	$9.8° \times 11.2° \times 8.7°$
7	1.89°	$13.5° \times 14.6° \times 10.5°$
8	1.92°	$11.4° \times 12.3° \times 9.1°$
9	2.01°	$10.9° \times 13.1° \times 9.2°$
10	1.98°	$10.6° \times 13.5° \times 9.7°$
11	2.11°	$12.1° \times 14.9° \times 9.9°$
Average	1.99°	$11.63° \times 13.44° \times 9.43°$

B. Evaluation of gazing a ground truth point at different positions

To demonstrate the sensitiveness of the gaze accuracy against the distance D_{ss}, we devise an experiment that a subject gazes at a ground truth point with eight different D_{ss}. At each distance, the subject is also asked to rotate his/her head step-by-step in roll, pitch, and yaw, respectively, untill he/she cannot see the point. The average accuracy of estimated gazes is determined as the estimation accuracy for each distance. Fig. 7 demonstrates the relationship between the gaze estimation accuracy and the distance D_{ss}. From the results, we can conclude that the best gaze distance is approximately 925mm-1135mm. Less than 925mm or larger than 1135mm will degrade the accuracy. Normally, the smaller the D_{ss} is, it is easier to obtain the accurate 3D iris center and thus more possible to obtain a higher estimation accuracy. However, when D_{ss} is too small to exceed the limit measure range of the Kinect, the accuracy will be greatly reduced (shown as D_{ss}=630mm in Fig. 7) since the 3D information captured by the Kinect under this distance is unreliable.

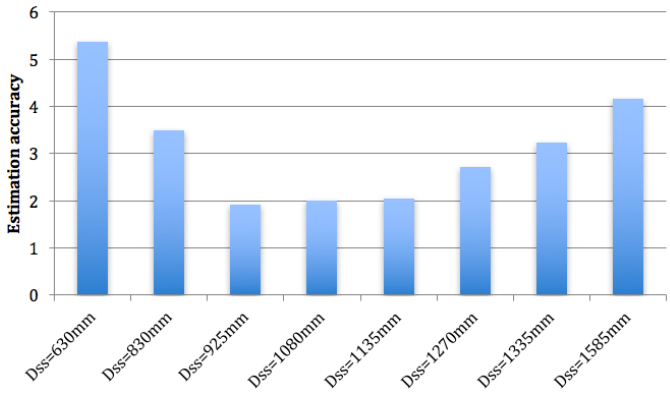


Fig. 7: Gaze estimation accuracy with different D_{ss}.

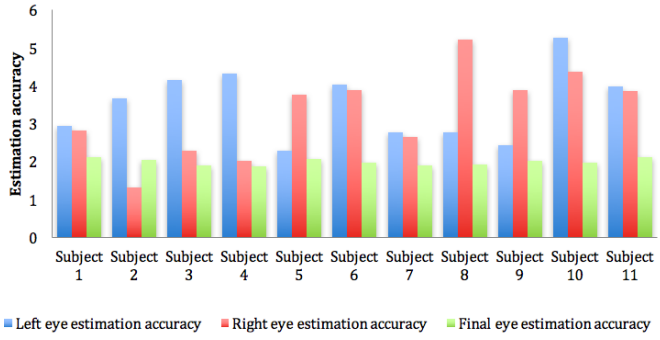


Fig. 8: Estimation accuracy of left eye gaze, right eye gaze, and final gaze.

C. Evaluation of two-eye model-based gaze estimation

To show the merit of averaging gazes of both eyes as a final gaze, we experimentally compare the accuracy of left eye gaze estimation, right eye gaze estimation, and final eye gaze estimation for each subject. As shown in Fig. 8, the comparison result validates that the averaging gaze is reliable and acceptable.

TABLE III: Comparison with the State-of-the-art Kinect-based Gaze Estimation Methods.

Methods	Reported accuracy	Features
Mora and Odobez[1]	7.6°-12.6°	Non eye model
Jafari and Ziou[7]	Above 10°	Non eye model
Jafari and Ziou[8]	7.9°	Non eye model
Mora and Odobez[2]	6.3°	Eye model
Cazzato et al.[3]	6.9°	Eye model
Li and Li[4]	6°	Eye model
Li and Li[5]	Vertical 5.9°, horizontal 4.4°	Eye model
Sun et al.[6]	1.4°-2.7°	Eye model
Ours	1.99°	Eye model

D. Comparison with the state-of-the-art

To further demonstrate the superior performance of the proposed method, we compare the estimation accuracy of our method with the accuracy of the state-of-the-art Kinect-based gaze estimation methods. Results in Table III indicate that our method outperforms all the regression-based methods as well as most of the model-based methods. In addition, the number of test subjects in our experiments is also comparable. Although Sun's method [6] can achieve a little more accurate estimation, it requires more calibration points. The main reason for lower accuracy of our method is the approximation of the visual axis which to some extent enlarges the *Kappa* angles.

V. Conclusion

In this paper, an effective gaze estimation method based on 3D eye model was presented. The proposed method was capable of estimating human's gaze directly from a Kinect sensor with free head movement. A convolution-based means of gradients iris center localization method was developed, which significantly improved the accuracy and speed of the conventional means of gradients method. A simplified eye model was proposed, which approximated the visual axis as a line from the iris center to the gaze point, to effectively estimate the eyeball centers and *Kappa* angles. Different from the conventional eye model-based methods that used many calibration points to calculate the parameters, only one point was utilized based on the simplified eye model. The human's gaze was then directly calculated according to the estimated head pose, iris centers and eye model parameters. The conventional methods used one single eye for gaze estimation, while the proposed method averaged the estimated gazes of both eyes for a final gaze. Experiments conducted on eleven subjects demonstrated the good performance of the proposed gaze estimation methods. Moreover, the estimation accuracy of the proposed method outperformed many leading methods in the state-of-the-art.

Our method allows subject to freely move his/her head with average roll 11.63°, pitch 13.44° and yaw 9.43°. However, when the head movement is large, the performance of our method will degrade. Since we estimate the final gaze using estimations of both eyes, large estimation error of any one eye caused by the large head movement will result in a low accuracy of the final estimated gaze. To remedy this, multi-view gaze estimation [19] will be investigated in our future work. Moreover, we will focus on incorporating the human's gaze with visual tracking methods [20] to recognize human's activity in human-computer or human-human interaction.

References

[1] K. A. F. Mora and J. M. Odobez, "Gaze estimation from multimodal Kinect data," *Proc. IEEE Conf. Comput. Vis. Pattern Recog. Workshop*, pp. 25–30, 2012.

[2] K. A. F. Mora and J. M. Odobez, "Geometric generative gaze estimation (G^3E) for remote RGB-D cameras," *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, pp. 1773–1780, 2014.

[3] D. Cazzato, M. Leo, and C. Distante, "An investigation on the feasibility of uncalibrated and unconstrained gaze tracking for human assistive applications by using head pose estimation," *Sensors*, vol. 2014, no. 14, pp. 8363–8379, 2014.

[4] J. Li and S. Li, "Eye-model-based gaze estimation by RGB-D camera," *Proc. IEEE Conf. Comput. Vis. Pattern Recog. Workshop*, pp. 606–610, 2014.

[5] J. Li and S. Li, "Gaze estimation from color image based on the eye model with known head pose," *IEEE Trans. Human-Machine Systems*, pp. 1–10, 2015, available online.

[6] L. Sun, Z. Liu, and M.-T. Sun, "Real time gaze estimation with a consumer depth camera," *Information Sciences*, vol. 320, pp. 346–360, 2015.

[7] R. Jafari and D. Ziou, "Gaze estimation using Kinect/PTZ camera," *Proc. IEEE Int. Symp. Robot. Sensors Environ.*, pp. 13–18, 2012.

[8] R. Jafari and D. Ziou, "Eye-gaze estimation under various head positions and iris states," *Expert Systems with Applications*, vol. 42, no. 1, pp. 510–518, 2015.

[9] R. Valenti and T. Gevers, "Accurate eye center location through invariant isocentric patterns," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 9, pp. 1785–1798, 2012.

[10] Z. Ramdane-Cherif and A. Nalt-AliNait-Ali, "An adaptive algorithm for eye-gaze-tracking-device calibration," *IEEE Transactions on Instrumentation and Measurement*, vol. 57, no. 4, pp. 716–723, 2008.

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

[12] P. Viola and M. J. Jones, "Robust real-time face detection," *Int. J. Computer Vision*, vol. 57, no. 2, pp. 137–154, 2004.

[13] X. Xiong and F. De la Torre, "Supervised descent method and its applications to face alignment," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 532–539, 2013.

[14] G. Golub and C. Reinsch, "Singular value decomposition and least squares solutions," *Numerische Mathematik*, vol. 14, no. 5, pp. 403–420, 1970.

[15] F. Timm and E. Barth, "Accurate eye centre localisation by means of gradients," *Proc. 6th Int. Conf. Computer Vision Theory and Applications*, pp. 125–130, 2011.

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

[17] I. Bekerman, P. Gottlieb, and M. Vaiman, "Variations in eyeball diameters of the healthy adults," *Journal of Ophthalmology*, vol. 2014, p. 5 pages, 2014.

[18] Y.-M. Cheung and Q. Peng, "Eye gaze tracking with a web camera in a desktop environment," *IEEE Trans. Human-Machine Systems*, vol. 45, no. 4, pp. 419–430, 2015.

[19] H. Cai, X. Zhou, H. Yu, and H. Liu, "Gaze estimation driven solution for interacting children with ASD," *Proc. 26th 2015 International Symposium on Micro-Nano Mechatronics and Human Science*, pp. 1–6, 2015.

[20] X. Zhou, Y. Li, B. He, and T. Bai, "GM-PHD-based multi-target visual tracking using entropy distribution and game theory," *IEEE Trans. Industrial Informatics*, vol. 10, no. 2, pp. 1064–1076, 2014.