Eye-Model-Based Gaze Estimation by RGB-D Camera

LI Jianfeng
Graduate School of Engineering
Tottori University
Tottori, Japan
E-mail: popqlee@sina.com

LI Shigang
Graduate School of Engineering
Tottori University
Tottori, Japan
E-mail: li@ele.tottori-u.ac.jp

Abstract— **This paper proposes a method of eye-model-based gaze estimation by RGB-D camera, Kinect sensor. Different from other methods, our method sets up a model to calibrate the eyeball center by gazing at a target in 3D space, not predefined. And then by detecting the pupil center, we can estimate the gaze direction. To achieve this algorithm, we first build a head model relying on Kinect sensor, then obtaining the 3D information of pupil center. As we need to know the eyeball center position in head model, we do a calibration by designing a target to gaze. Because the ray from eyeball center to target and the ray from eyeball center to pupil center should meet a relationship, we can have an equation to solve the real eyeball center position. After calibration, we can have a gaze estimation automatically at any time. Our method allows free head motion and it only needs a simple device, finally it also can run automatically in real-time. Experiments show that our method performs well and still has a room for improvement.**

Keywords-gaze estimation; kinect sensor; eyeball calibration; free head motion; pupil center detection

I. INTRODUCTION

In recent years, how to understand human intentions has become a hot issue in various applications. Gaze estimation, as an important part of human behavior, also be focused gradually as it can be used in the areas such as driver behavior analysis, security monitoring, behavior investigation and human-computer interfaces. In order to be applied widely and conveniently, current research is working on estimating the gaze via the fewest devices while achieving the best results. However, most of current gaze estimation methods request additional devices and models. On the other hand, both texture and depth information can be acquired from a RGB-D camera. By integrating both information, some tasks can be carried out much easily.

The RGB-D camera used widely is the Kinect, which is developed by Microsoft Co. for game. Via Kinect sensor, we can acquire the pose and 3D position of human heads easily. Moreover, as a commercial product, the accuracy and efficiency can be guaranteed. Last, as it has opened its source code, it became a convenient platform for developing.

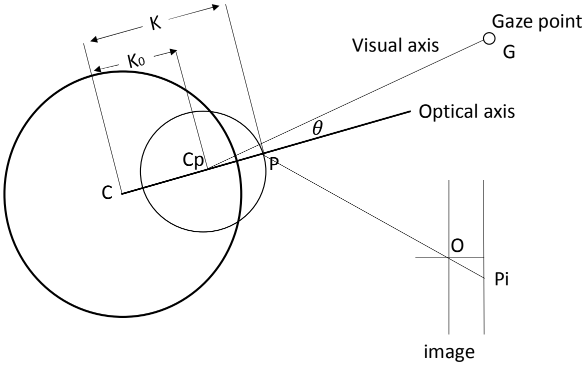


Figure 1: 3D eye model

In this paper, we exploit the Kinect sensor for gaze estimation of humans. Since the head pose relative to the Kinect can be acquired easily, what we have to do is to determine the line of sight at the coordinate system of human head.

Fig. 1 shows the simple eye-model of humans [1]. The eyeball is composed of two spheres with different sizes. The anterior smaller sphere is the cornea, and the pupil is inside the cornea. The optical axis is defined as the 3D line connecting the eyeball center C and the pupil center P. Since the corneal center C_0 is at the optical axis, and the distances, K and K_0, among the eyeball center C, the corneal center C_0 and the pupil center P and the angle θ between the optical axis and the visual axis are constant for each person, the visual axis can be determined from the optical axis. Therefore, the gaze estimation results in the determination of the eyeball center C and the pupil center P.

In this paper we propose a novel method to estimate gaze direction based on Kinect sensor. We can easily set up a head coordinate system by Kinect, then we detect the pupil center from the image. In the next stage, 3D eye model is designed to calibrate the eyeball center in head coordinate system. The calibration requests human to gaze at a target while the 3D position can also be achieved by Kinect. At last, we can compute the gaze direction through calibrated eyeball center and pupil center.

In this way, our method only rely on a Kinect sensor, no additional devices are needed. We obtain the gaze estimation that can allow free head motion. Moreover, after calibration, our method can calculate automatically at any time.

In the rest of the paper, Section 2 describes related works. Section 3 introduces the principle of our method. Section 4 shows the experimental results of our method. Section 5 concludes the work.

II. RELATED WORKS

As gaze estimation can be used in the Human Computer Interaction (HCI), there has been a plenty of methods proposed [1]. In common, the current gaze estimation methods can be divided into two groups: feature-based and appearance-based methods. Appearance-based methods do not explicitly extract features, but rather learn a direct mapping from the high dimensional eye images to the low dimensional space of gaze coordinates [2, 3, 4]. However, the extracted 2D eye image features is variable due to head pose. To handle this problem, Noris et al. [8] used specialized head mounted hardware to track the gaze in unconstrained environments. Zhu et al. [9] constructed a highly non-linear generalized gaze mapping function that accounts for head movement by using support vector regression. These methods are all needing an additional device. Since Kinect can provide an effective solution, it has been exploited to estimate the gaze by modelling the head and eye [12, 13]. Mora et al. [14] exploited the depth sensor to perform an accurate tracking of a 3D mesh model and robustly estimate a person head pose via Kinect, and they also used the Kinect sensor to collect ground truth.

Unlike Appearance-based method, Feature-based methods extract some features such as corneal infrared reflections, pupil center, and iris contour [5, 6, 7]. These features are used to set up a 3D eye model, then estimate the gaze direction. Because its high accuracy under free head movement, it became more popular. But at the same time, this kind of method also has a disadvantage that special cameras or lights are always required because of extracting the eye features [10, 11], one or multiple infrared lights are used to illuminate the eye region and to build the corneal refection on the corneal surface, while one or multiple cameras are used to capture the image of the eye. Instead of using the infrared lights and the corneal reflections, Chen et al. [15] proposed a 3D eye gaze estimation and tracking algorithm based on facial feature tracking using a single camera. But they labelled the pupil center manually.

In this paper, we proposed a 3D gaze estimation method that can allow free head movement and calculate automatically. The Kinect sensor is exploited to supply with the head pose, at the same time to collect 3D position of a target like in [14]. But different from [14], our method is based on a 3D eye model which is also unlike [15]. Thanks to the detecting pupil center algorithm [16], we can calibrate the eyeball center accurately by setting up a model.

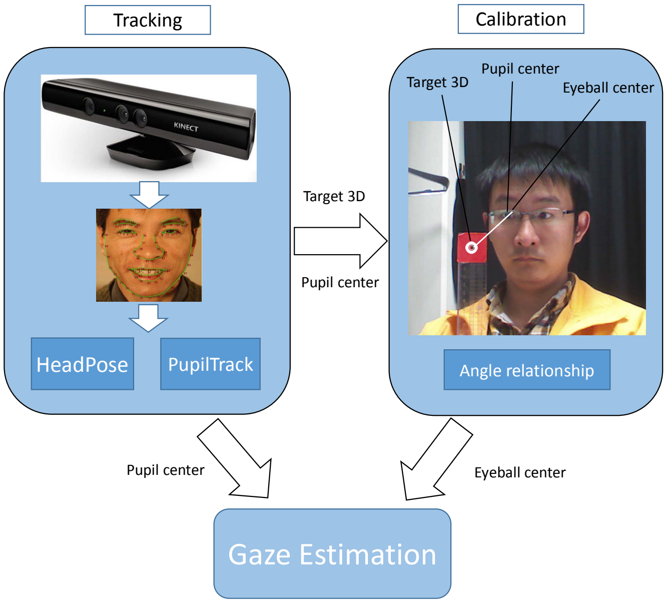


Figure 2: Main steps of method

The proposed method has an advantage that does not need any additional special cameras and infrared lights, and can calculate gaze direction automatically in real-time. The final experiment shows that our method performs better than [14] with the same evaluation.

III. PROPOSED METHOD

The main steps of our method are shown in Fig. 2. Based on the Microsoft Kinect sensor, we can estimate head pose and build a head coordinate system (section A). Then using the pupil center tracking algorithm [16] to get the 2D position of pupil center, which would be transformed to 3D later, from image (section B). Since we need to know the eyeball center in the head coordinate, we do a calibration in section C. This calibration refers to the angle relationship, so we designed a target, which 3D information is known by Kinect. In section D, as we know the pupil center and eyeball center, gaze direction can be estimated. The following sections describe in detail for each step.

A. Face pose estimation

We decided to use the face tracking algorithm supplied by Kinect because of its reliable and convenience. Microsoft Kinect is a commercial product that widely used in various aspects, it can estimate the head pose accurately and robustly, moreover, additional devices are not needed.

When tracking the head by Kinect sensor, it supplies us with translation matrix and head pose which is captured by three angles: pitch, roll, and yaw. Based on angles, we can calculate the rotation matrix as below.

$$R = \begin{bmatrix} cos\alpha cos\gamma - sin\alpha sin\beta sin\gamma & -cos\beta sin\alpha & cos\alpha sin\gamma + cos\gamma sin\alpha sin\beta \\ cos\gamma sin\alpha + cos\alpha sin\beta sin\gamma & cos\alpha cos\beta & sin\alpha sin\gamma - cos\alpha cos\gamma sin\beta \\ -cos\beta sin\gamma & sin\beta & cos\beta cos\gamma \end{bmatrix} \quad (1)$$

Which $\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}$ represent roll, yaw and pitch. As we know the translation matrix $\boldsymbol{T}$ and rotation matrix $\boldsymbol{R}$, we can build a head coordinate system. The origin of system is inside of the head.

B. *Pupil center estimation*

The algorithm proposed by Febian Timm and Erhardt Barth in [16] gives us a solution to track the pupil center from image. This approach locates the pupil center accurately and robustly by using image gradients.

As the pupil center is given $\boldsymbol{p} = (\boldsymbol{u_p}, \boldsymbol{v_p})$, its 3D position $\boldsymbol{P} = (\boldsymbol{x_p}, \boldsymbol{y_p}, \boldsymbol{z_p})$ can be estimated by assuming the distance between pupil center $\boldsymbol{P}$ and eyeball center $\boldsymbol{C}$ is a constant $\boldsymbol{K}$. Because we can take the eyeball as a standard ball, and if we know the internal parameters of camera, a ray in 3D space which through the 2D point of image can be formulated. Related equations listed as below:

$$\begin{cases} \frac{x_p}{u_p - u_0} = \frac{y_p}{v_p - v_0} = \frac{z_p}{f} \\ \|P - C\| = K \end{cases} \quad (2)$$

Where $(\boldsymbol{u_0}, \boldsymbol{v_0})$ is the center of image, $\boldsymbol{f}$ is the focus length. Eyeball center $\boldsymbol{C}$ can be estimated by calibration.

C. *Eyeball center calibration*

To calibrate the eyeball center, we designed a target to gaze, as shown in Fig.1. When people are focusing on the target $\boldsymbol{G}$, the vector $\overrightarrow{\boldsymbol{C_p G}}$ and $\overrightarrow{\boldsymbol{C_p P}}$ would have a angle $\boldsymbol{\theta}$, since the gaze point is defined as the intersection of visual axis, and the angle between the optical axis and the visual axis is a constant value [17].

First assuming the eyeball center in head coordinate system as $\boldsymbol{C_0}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$. To transform $\boldsymbol{C_0}$ to Kinect coordinate $\boldsymbol{C}$,

$$C = R * C_0 + T \quad (3)$$

As (2) described, pupil center $\boldsymbol{P}$ can be expressed by unknown $\boldsymbol{C}$,

$$P = f(C) \quad (4)$$

Because $\boldsymbol{K}$ and $\boldsymbol{K_0}$ are also constants [17]. $\boldsymbol{C_p}$ can be estimated as,

$$C_p = C + \frac{K_0}{K}(P - C) \quad (5)$$

Finally, according to the relationship between two vectors, we can have the equation below,

$$\frac{\overrightarrow{C_p G} \cdot \overrightarrow{C_p P}}{\| \overrightarrow{C_p G} \| \| \overrightarrow{C_p P} \|} = \cos\theta \quad (6)$$

To solve the nonlinear equation, we use the Levenberg-Marquardt Method. To let the calibration more accurate, RANSAC is used before calculation.

D. *Gaze estimation*

After calibration, the eyeball center can be transformed to Kinect coordinate at any time, and the pupil center $\boldsymbol{P}$ is also known, then $\boldsymbol{C_p}$ can be calculated by (5). Thus, the direction of estimated gaze $\mathbf{g}$ can be estimated and expressed as horizontal angle and vertical angle $(\boldsymbol{\delta}, \boldsymbol{\varphi})$. At last, the visual axis can be obtained by adding the constant angle values.

IV. EXPERIMENTS

To verify the effects of our method, we have conducted a series of experiments. In our experiments, the size of RGB image from Kinect is 1280*960, and size of depth image is 640*480. As Kinect cannot have the depth data if the distance is below 50cm, we controlled the distance between Kinect and human to about 70cm.

To evaluate the result, we also design a target which is totally red and a black point in center of the target. So the Kinect sensor can catch it easily and get the 3D position from depth sensor. When doing the experiment, we employed the same evaluation in [14], we let the people gaze at the target which is shown in Fig.3. So we can have a gaze ground-truth data. But the difference is that we calculated the eyeball center, not predefined in the topology. Another difference is that to estimate the gaze, we set up a 3D eye model to calculate vectors, not the same as estimating from the eye-in-head images in [14].

First calibrating the eyeball center, we use the method which is mentioned in chapter 3. To get a more accurate result, we collect the data by gazing the target from different directions. In total, 20 sets of data. After RANSAC, we use Levenber-Marquardt Method to solve the equation. The constants inside are fixed as average human eye value: *K*=13.1mm, *K0*=5.3mm, horizontal angle between visual and optic axes of the eye is 5 degrees, vertical angle is 1.5 degrees [17]. And after calibration, we obtained the eyeball center in head coordinate system: left eyeball $C_l = (-0.0315, 0.0405, 0.1032)^T(m)$, right eye ball $C_r = (0.0311, 0.0398, 0.1079)^T(m)$.

Next, the target moves in front of observer while the observer is gazing at the target, the distance between the observer and target is about 20cm. We tested our method in each direction including looking upward, downward, leftward and rightward. And the comparison diagram between ground truth and estimation for both eyes are shown in Fig. 4. Besides, the final average error of gaze estimation for both eyes are illustrated in Table 1. As we can see from these graphs, ground truth δ represents the horizontal angle of target while φ represents the vertical angle. Take Fig.4 (a) as an example, it shows a result of looking upward, the target is moving horizontally above the eyes, so the vertical angle

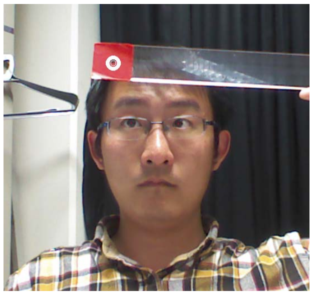
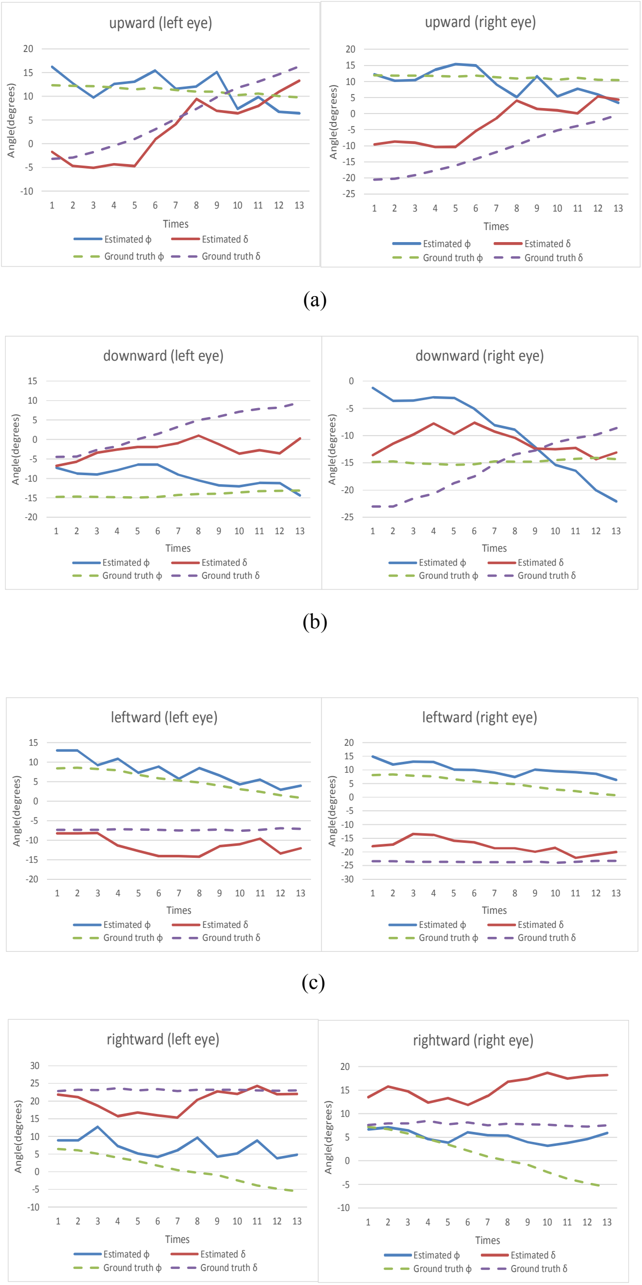


Figure 3: Gazing the target



(a)

(b)

(c)

(d)

Figure 4: Comparison of estimated gaze and ground truth. a) The target is moving horizontally above eyes. b) The target is moving horizontally below eyes. c) The target is moving in the left side of eyes. d) The target is moving in the right side of eyes.

of ground truth is nearly not changing while the horizontal angle of ground truth is changing regularly. As we can see, the estimation of our method is close to the ground truth. Comparing with the method in [14], we also allow free head movements, the final result shows that our method is better than theirs. As the table I. shows, whatever the direction is, the average error is definitely below 10 degrees, sometimes even below 5 degrees.

But at the same time the experiments show that it is not very stable, that is because the performance of our method is depending on the detecting pupil center algorithm to some extent. This pupil tracking algorithm requests the pupil shown on the image as complete as possible. If it can detect more stable and accurate, we believe our method can have a better performance.

TABLE I. AVERAGE ERROR OF GAZE ESTIMATION

Direction	**Error of left eye (degrees)**		**Error of right eye(degrees)**	
	Average of horizontal angle	***Average of vertical angle***	***Average of horizontal angle***	***Average of vertical angle***
upward	3.1976	2.2012	8.4482	3.1565
downward	5.2264	4.6642	6.6024	7.9100
leftward	4.2144	2.5384	5.4345	5.2521
rightward	3.4191	6.1998	7.7528	4.1983

V. CONCLUSION

In this paper, we proposed a gaze estimation method that can calculate automatically after calibration. First we built a head coordinate system based on Kinect sensor. Then we set up an eye model and design a calibration method. When people are gazing at a target, the vector from eyeball center to target and the vector from eyeball center to pupil center should meet a relationship. According to this principle, we can estimate the eyeball center in head coordinate system. After calibration, the gaze estimation can be realized easily.

Our experiments show that our method can have a good performance even in every direction, moreover, our method is simple and reliable, and it can also run automatically in real-time. But we also notice that our method could be more accurate if the pupil center can be detected better. Our future work would consist in finding a more stable and accurate algorithm on detecting pupil center.

REFERENCES

[1] D. W. Hansen and Q. Ji. In the eye of the beholder: a survey of models for eyes and gaze. *IEEE Trans. on Pat. Analysis and Machine Intelligence*, 32(3):478-500, Mar. 2010.

[2] K. Tan, D. J. Kriegman, and N. Ahuja. Appearance-based eye gaze estimation. In *Proceedings of the 6th IEEE Workshop on Applications of Computer Vision*, pp. 2667-2674, 2010.

[3] F. Lu, T. Okabe, Y. Sugano, and Y. Sato. A head pose-free approach for appearance-based gaze estimation. *British Machine Vision Conference*, pp. 1-11, 2011.

[4] F. Lu, Y. Sugano, T. Okabe, and Y. Sato. Inferring Human Gaze from Appearance via Adaptive Linear Regression. In *ICCV: International Conference on Computer Vision*, Barcelona, Spain, 2011.

[5] R. Valenti, T. Gevers. Accurate eye center location and tracking using isophote curvature. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2008)*, pp. 1-8, 2008.

[6] J. G. Wang, E. Sung, and R. Venkateswarlu. Eye gaze estimation from a single image of one eye. In *Proceedings of the 9th IEEE*

International Conference on Copmputer Vision (ICCV 2003), pp. 136-143, 2003.

[7] E. D. Guestrin and M. Eizenman. Remote point-of-gaze estimation requiring a single-point calibration for applications with infants. In *Proceedings of the 2008 symposium on eye tracking research & applications*, pp. 267-274, 2008.

[8] B. Noris, J. Keller, and A. Billard. A wearable gaze tracking system for children in unconstrained environments. *Computer Vision and Image Understanding*, pp. 1-27, 2010.

[9] Z. Zhu, Q. Ji, and K. P. Bennett. Nonlinear eye gaze mapping function estimation via support vector regression. *International Conference on Pattern Recognition*, pp. 1132-1135, 2006.

[10] T. Nagamatsu, J. Kamahara, and N. Tanaka. 3D gaze tracking with easy calibration using stereo cameras for robot and human communication. In *Proceedings of the 17th IEEE International Symposium on Robot and Human Interactive Communication*, pp. 59-64, 2008.

[11] Z. Zhu, Q. Ji. Novel eye gaze tracking techniques under natural head movement. *IEEE Transaction on Biomedical Engineering*, 54(12):2246-2260, 2007.

[12] R. Jafari, D. Ziou. Gaze estimation using Kinect/PTZ camera. In *Proceedings of 2012 IEEE International Symposium on Robotic and Sensors Environments*, pp. 16-18, 2012.

[13] Y. Li, D. S. Monaghan, and N. E. Connor. Real-time gaze estimation using a Kinect and a HD webcam. *MultiMedia Modeling*. 8325:506-517, 2014.

[14] K. A. F Mora, J. Odobez. Gaze estimation from multimodal Kinect data. *Computer Vision and Pattern Recognition Workshops*, pp. 25-30, 2012.

[15] J. Chen, Q. Ji. 3D gaze estimation with a single camera without IR illumination. In *Proceedings of the 19th International Conference on Pattern Recognition*, pp. 1-4, 2008.

[16] F. Timm, E. Barth. Accurate eye center localization by means of gradients. In *Proceedings of the International Conference on Computer Theory and Applications*, volume 1, pp. 125-130, 2011.

[17] E. D. Guestrin, M. Eizenman. General theory of remote gaze estimation using the pupil center and corneal reflections. *IEEE Transactions on biomedical engineering*, 53(6): 1124-1133, 2006.