2016 23rd International Conference on Pattern Recognition (ICPR)
Cancún Center, Cancún, México, December 4-8, 2016

Real Time Eye Gaze Tracking with Kinect

Kang Wang
Rensselaer Polytechnic Institute
Troy, New York 12180
Email: wangk10@rpi.edu

Qiang Ji
Rensselaer Polytechnic Institute
Troy, New York 12180
Email: qji@ecse.rpi.edu

***Abstract*—Traditional gaze tracking systems rely on explicit infrared lights and high resolution cameras to achieve high performance and robustness. These systems, however, require complex setup and thus are restricted in lab research and hard to apply in practice. In this paper, we propose to perform gaze tracking with a consumer level depth sensor (Kinect). Leveraging on Kinect's capability to obtain 3D coordinates, we propose an efficient model-based gaze tracking system. We first build a unified 3D eye model to relate gaze directions and eye features (pupil center, eyeball center, cornea center) through subject-dependent eye parameters. A personal calibration framework is further proposed to estimate the subject-dependent eye parameters. Finally we can perform real time gaze tracking given the 3D coordinates of eye features from Kinect and the subject-dependent eye parameters from personal calibration procedure. Experimental results with 6 subjects prove the effectiveness of the proposed 3D eye model and the personal calibration framework. Furthermore, the gaze tracking system is able to work in real time (20 fps) and with low resolution eye images.**

I. Introduction

Gaze tracking is to predict where human looks in real time. Eye gaze can reflect human attention or interest in the world. Therefore gaze tracking/estimation techniques have been widely studied and applied in various fields. In advertising market, eye gaze can be utilized to study customer's interest so that more interesting advertisements can be developed. In Human Computer Interaction filed, eye gaze can serve as a replacement of traditional input like mouse and keyboard or as additional input to better interact with the computer. In game play field, more and more games start to integrate gaze input from eye tracker to enhance the gaming experience [1]. Furthermore, eye tracking data can also help research and analysis in marketing and understanding human's cognitive process [2], etc.

Gaze estimation/tracking has been studied for decades and various methods have been proposed. These methods can be divided into two main categories: model-based methods and appearance-based methods. Model-based methods [3], [4], [5], [6] rely on a geometric eye model representing the structure and function of human vision system. By analyzing human eye/head anatomy, key components in human vision system (cornea, pupil, fovea, etc) can be abstracted to specific features in the established eye model. Gaze estimation can be considered as mimicking the human vision system through the eye model to compute the exact gaze direction as human brain does. Eye model can approximate the real eyeball structure accurately, thus model-based methods are known for their high accuracy. Furthermore, the eye model is a 3D model in the camera coordinates system, thus model-based method can allow free head movement during gaze tracking. Constructing the 3D eye model requires the knowledge about relative position for different components in human vision system. These knowledge is typically abstracted to subject-dependent eye parameters which can be estimated through a personal calibration procedure. Differently, appearance-based methods [7], [8] and [9] rely on the eye appearance. The underlying assumption is that similar eye appearances correspond to similar gaze positions. Therefore a mapping function between eye appearances and gaze positions can be established. Gaze estimation is then simplified to learn the mapping functions. Unlike model-based method, appearance-based methods do not require special illumination and any knowledge about human vision systems. Besides, appearance-based methods only require simple eye detection techniques while model-based methods require accurate detection of eye features (pupil, glints, etc). However, appearance-based method may require large amount of data to learn the mapping function. Another major difference is that model-based methods consider the gaze estimation problem in 3D space while the appearance-based methods only work in 2D space. This makes the appearance-based methods sensitive to head movement since different head poses can result in same eye appearances. Therefore, compared to model-based methods, appearance-based methods typically suffer from head movement issues and the accuracy is rather low. We suggest readers refer to [10] for more detailed discussion on different eye tracking methods.

In this work, we focus on 3D model based gaze estimation method. Traditional approaches rely on explicit infrared illumination to produce glints on cornea surface. Glints position can be utilized to estimate the 3D cornea/eyeball center, which is essential to estimate gaze direction. Another benefit of infrared illumination is the bright/dark pupil effect, which make pupil detection much easier and more accurate. However, the system setup is rather complex, and typically multiple lights are required to enable enough operating ranges. Therefore, we propose to build a real time eye gaze tracking system with Microsoft Kinect. Notice Kinect also use infrared illumination, but it is designed for estimating depth information, not for gaze related applications. And

978-1-5090-4846-5/16/$31.00 ©2016 IEEE

a single Kinect sensor also enables portable gaze tracking. With Kinect, we first construct a 3D eye model similar to [5]. The gaze tracking system starts with estimating the head rotation and translation given the color and depth frame from Kinect. Eyeball center in camera coordinates system can then be estimated given rotation and translation and other subject-dependent parameters. The third step is to estimate the pupil position in camera coordinates system. Finally gaze direction can be computed given eyeball center, pupil center and related eye parameters.

II. Related Work

First of all, there are several commercially available remote eye trackers for ordinary customers. For example, Tobii eyeX [11], The Eye tribe [12], etc. Despite the good performance and affordable prices, these eye trackers still lies heavily on infrared lights, and are limited in indoor environments. Though Kinect V1 still requires infrared lights to sensing depth, our algorithm does not rely directly on infrared lights. Actually, we can replace with the newly Kinect V2 sensor, where a time of flight sensor is used to sensing depth.

Plenty of work has been proposed to remove explicit usage of IR lights. The first category is methods with web cameras only. Lu *et al* [7] proposed an appearance-based gaze estimation method with adaptive linear regression. The mapping functions can be accurately estimated via sparsely collected training samples. The sparsity nature of the proposed method enables much fewer training samples. However, the head pose issue is not well solved. Sugano *et al* [13] proposed to alleviate the head pose issue by adding head pose information. Specifically, they build a stereo vision system with multiple cameras. Then 3D facial landmark positions can be recovered and utilized to define head position. However, the proposed method requires large amount of data to cover enough head pose spaces and the system setup is rather complex. Hirotake [14] proposed a head-eye model combined with eye appearance to estimate the 3D gaze direction. Follow the idea similar to structure from motion, they can estimate the face model and head pose given a sequence of images. However, their method is sensitive to head pose, and the accuracy is rather low (6 degree) because of the ignorance of personal parameter.

The second category is appearance-based methods with depth sensor. Among them, Funes Mora *et al* [15] proposed a head pose invariant gaze estimation method with Kinect. An appearance generative process is built to obtain head-pose rectified eye images. Gaze can therefore be estimated with the rectified eye appearance. Li *et al* [16] proposed a real time gaze tracking system with a HD web camera and a Kinect. Gaze motion is approximated by local pupil motion and global face motion. However, such approximation is not correct since real gaze motion is the coupling of pupil motion and face motion. Therefore their system is limited in subject's head positions. Jafari *et al* [17] proposed to estimate gaze with Kinect and a PTZ camera. They first estimate the eye-gaze direction based on the relative displacement of the iris in terms of reference point. The final gaze direction is the correction of eye-gaze direction by considering the head pose and orientation from Kinect. However, the relative displacement cannot accurately model gaze directions and thus the accuracy is rather low.

The third category is model-based methods with depth sensor. Li *et al* [18] proposed a model-based gaze estimation method with Kinect. The basic idea is to estimate gaze related eye features (eyeball center, pupil center) and estimate gaze given these eye features and subject-dependent parameters. They proposed a personal-calibration procedure which requires subjects to gaze at a 3D target instead of pre-defined points on display surface. However, the proposed calibration procedure only estimates the eyeball center in head coordinates system and sets other subject-dependent parameters to human average. Therefore the estimated parameters may not apply to different subjects and results in poor accuracy of the proposed gaze tracking system. Xiong *et al* [19] proposed another model-based gaze estimation method with Kinect. Head pose information is estimated from facial landmark tracking results from Kinect, from which eyeball center position can be recovered. Together with pupil position and subject-dependent parameters, gaze direction can be effectively computed. However, their 3D eye model ignores cornea center and use eyeball center instead to model the difference between optical and visual axis. Such model is simple but may not give good gaze estimation results. Sun *et al* [20] propose a similar model-based gaze estimation method. Pupil and eyeball center position are estimated from pupil detection algorithm and pose estimation algorithm. Despite the high accuracy ($<$2 degree) achieved by the system, they did not model the difference between optical and visual axis properly. Therefore, the gaze tracking system might not apply to subjects with large optical-visual angle differences. In addition, they sacrifice the speed (12 FPS of their system) to obtain high resolution color images (1280×960), which make their approach limited in applications with larger FPS. Besides, the proposed methods are only evaluated at a distance of approximately $550mm$, the optical-visual angle difference might have a much larger effect with larger operating distances.

In summary, existing model-based gaze estimation methods make additional assumptions on the eye model to simplify the computation and reduce the number of personal parameters. However, we propose to mimic the human vision system as much as possible by using a complex eye model with more personal parameters. Despite the complexity of the eye model, the calibration process and the gaze estimation process remain simple and intuitive. Besides, we also build the gaze tracking system with low resolution images 640×480 compared to 1280×960 in [20] to enable real-time gaze tracking (20 FPS).

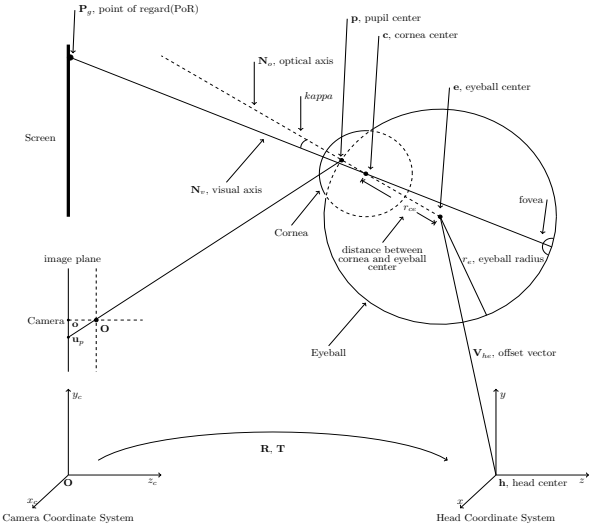

Fig. 1. 3D eye model.

III. 3D Eye Model

Model-based gaze estimation methods first construct a 3D eye model to mimic human vision system and how eye gaze is generated. From eyeball anatomy, we can construct the eye model as illustrated in Figure 1. Eyeball system can be approximated as two spheres intersecting with each other: the eyeball sphere and the cornea sphere. Two spheres can rotate around the eyeball center together to look at different directions. Optical axis $\mathbf{N}_o$ is defined as the line connecting eyeball center $\mathbf{e}$, cornea center $\mathbf{c}$ and pupil center $\mathbf{p}$. However, the real gaze direction is determined by the line connecting fovea and cornea center $\mathbf{c}$, since fovea is a small depression in the retina of the eye where visual acuity is highest. Visual axis $\mathbf{N}_v$ is therefore defined to model the real gaze direction. Since eyeball and cornea rotate around eyeball center together as shown in Figure 2 , the angle between optical and visual axis is a fixed angle called kappa. Kappa is typically represented as a two dimensional vector $[\alpha, \beta]$. Eyeball radius r_e and the distance between cornea center and eyeball center r_{ce} are also assumed to be fixed for the same subject. Offset vector $\mathbf{V}_{he}$ represents the eyeball center position in head coordinates system. Since human head is a rigid object and eyeball is rigidly attached within head, thus $\mathbf{V}_{he}$ is also a fixed vector for the same subject. In summary, we use $\theta = [\alpha, \beta, r_e, r_{ce}, \mathbf{V}_{he}]$ to represent all the subject-dependent eye parameters. These parameters can be effectively estimated from the proposed personal-calibration framework.

IV. Gaze Estimation with 3D Eye Model

In this work, we plan to use the depth sensor (Kinect) to perform gaze estimation. Depth information from depth sensor is used in two folds:

- Obtain the 3D coordinates of pupil center $\mathbf{p}$ in camera coordinates system.

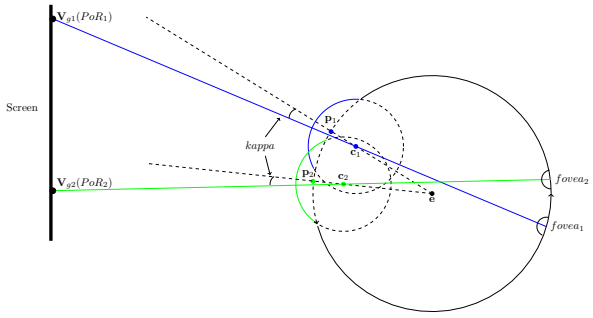

Fig. 2. Eyeball rotation and gaze direction.

- Perform head pose estimation to obtain the rotation $\mathbf{R}$ and translation $\mathbf{T}$ of the head relative to camera coordinates system.

Given the information $\{\mathbf{p}, \mathbf{R}, \mathbf{T}\}$ from depth sensor and the subject-dependent parameters θ from personal-calibration, we can perform gaze estimation with the proposed 3D eye model.

From Figure. 1, the same point in camera coordinates system $\mathbf{z}^c$ and head coordinates system $\mathbf{z}^h$ are related by the rotation $\mathbf{R}$ and translation $\mathbf{T}$:

$$\mathbf{z}^c = \mathbf{R}\mathbf{z}^h + \mathbf{T} \tag{1}$$

Therefore eyeball center in camera coordinates system can be computed as:

$$\mathbf{e} = \mathbf{R}\mathbf{V}_{he} + \mathbf{T} \tag{2}$$

Given eyeball center $\mathbf{e}$ and pupil center $\mathbf{p}$, optical axis $\mathbf{N}_o$ can be computed as:

$$\mathbf{N}_o = (\mathbf{p} - \mathbf{e})/||\mathbf{p} - \mathbf{e}|| \tag{3}$$

Cornea center $\mathbf{c}$ can then be computed as:

$$\mathbf{c} = \mathbf{e} + r_{ce}\mathbf{N}_o \tag{4}$$

The unit length vector $\mathbf{N}_o$ is typically expressed as two angles ϕ and γ:

$$\mathbf{N}_o = \begin{pmatrix} \cos(\phi)\sin(\gamma) \\ \sin(\phi) \\ -\cos(\phi)\cos(\gamma) \end{pmatrix} \tag{5}$$

By adding α and β to optical axis, we can obtain the visual axis $\mathbf{N}_v$:

$$\mathbf{N}_v = \begin{pmatrix} \cos(\phi+\alpha)\sin(\gamma+\beta) \\ \sin(\phi+\alpha) \\ -\cos(\phi+\alpha)\cos(\gamma+\beta) \end{pmatrix} \tag{6}$$

Gaze direction is then computed as:

$$\text{Gaze direction} = \mathbf{c} + \lambda\mathbf{N}_v \tag{7}$$

Point of Regard can be computed by intersecting the gaze direction with the display surface. The display surface equation

can be obtained by a one-time display camera calibration. For the purpose of simplicity, we denote PoR estimation as:

$$\text{PoR} = f(\mathbf{p}, \mathbf{R}, \mathbf{T}; \theta) \tag{8}$$

Overall, to perform gaze estimation, we first need robustly detected eye/head features $\mathbf{p}$, $\mathbf{R}$ and $\mathbf{T}$ from Kinect frame by frame. Given these detected features, a one-time personal calibration is performed to estimate the subject-dependent eye parameters. Finally, we can perform real time gaze tracking.

A. Detect Eye/Head Features and Estimate Head Pose

Kinect provides us with synchronized color stream and depth stream. Besides, Kinect SDK is able to perform facial landmark tracking, from which we can obtain the rough regions of left and right eyes. We treat pupil center as a special facial landmark, and propose to detect the 2D pupil center on the rough eye regions with supervised descent method as described in [21]. The corresponding depth value of the pupil is extracted from the depth frame, from which we can compute the 3D pupil center $\mathbf{p}$. Kinect SDK also provides us with estimated head pose $\mathbf{R}$ and $\mathbf{T}$. Note that $\mathbf{T}$ represents the head position in camera coordinates system, and $\mathbf{R}$ is computed based on a built-in defined head coordinates system from Kinect. Finally, observation pair $\{\mathbf{p}, \mathbf{R}, \mathbf{T}\}$ can be retrieved efficiently frame by frame.

B. Personal Calibration

During personal calibration procedure, subject is required to look at K pre-defined points $\mathbf{g}_i, i = 1, ..., K$. The corresponding eye/head features $\{\mathbf{p}_i, \mathbf{R}_i, \mathbf{T}_i\}, i = 1, ..., K$ are collected, from which we can compute the PoR using Eqn. 8:

$$\text{PoR}_i = f(\mathbf{p}_i, \mathbf{R}_i, \mathbf{T}_i; \theta) \tag{9}$$

The subject-dependent eye parameters θ can be estimated by minimizing the gaze prediction error:

$$\begin{aligned} \theta^* &= \arg\min_{\theta} \sum_{i=1}^{K} ||\text{PoR}_i - \mathbf{g}_i||^2 \\ &= \arg\min_{\theta} \sum_{i=1}^{K} ||f(\mathbf{p}_i, \mathbf{R}_i, \mathbf{T}_i; \theta) - \mathbf{g}_i||^2 \\ &\text{subject to } \ \theta_l < \theta < \theta_h \end{aligned} \tag{10}$$

Eye parameters θ represent the physical structure of human eye/head, therefore their values are limited in a reasonable range (θ_l, θ_h). The optimization problem in Eqn. 10 can be solved iteratively using interior-point algorithm. $[\alpha, \beta, r_e, r_{ce}]$ can be initialized to human average values. $\mathbf{V}_{he}$ can be initialized given $\mathbf{T}$ and initial estimation of eyeball center $\mathbf{e}$.

V. Experimental Results

A. Implementation Details

1) System Setup: A computer with Inter Core i7-4770 3.4 GHz CPU and 16.0 GB memory is used in the experiments. The monitor is 20.5 inch and the resolution is set to 1920×1080. Kinect is placed under the monitor. The resolution of color and depth stream are both set to 640×480. Display-camera calibration is achieved with the method proposed in [22], where only a thread is utilized to estimate the four screen corners' coordinates in camera coordinates system.

TABLE I
Subject-dependent personal parameters.

Subjects	$\theta = [\alpha, \beta, r_e, r_{ce}, \mathbf{V}_{he}]$						
1	4.8	1.2	16.6	5.1	[−34.6	41.6	45.0]
2	−4.8	3.3	15.5	5.1	[−30.3	40.3	58.6]
3	−3.7	4.7	17.3	7.6	[−32.3	48.6	43.2]
4	4.2	−3.5	16.8	5.4	[−29.4	44.7	59.3]
5	4.9	2.1	16.3	6.3	[−29.5	49.8	57.5]
6	−3.2	−4.9	15.9	5.2	[−30.6	47.9	47.1]

2) Camera and Stereo Calibration of Kinect: In order to decrease Kinect's build-in error, we perform camera calibration on both the color and depth cameras to estimate their intrinsic parameters. Furthermore, a stereo calibration procedure is performed on color and depth cameras to better align color stream and depth stream. Stereo calibration is essential since the physical distance between color and depth cameras causes the mis-alignment between synchronized color and depth streams.

3) Noise Reduction and Outlier Removal: The detected observation pair $\{\mathbf{p}, \mathbf{R}, \mathbf{T}\}$ are usually contaminated by noise and outliers. $\mathbf{R}$ and $\mathbf{T}$ are provided by Kinect, thus a simple smoothing operation is implemented. Incorrect $\mathbf{p}$ may result from poor 2D pupil detection results or missing depth values. We implement the method proposed in [23] to fill the missing values. Due to low resolution color image, 2D pupil detection is a challenge, we expect more advanced techniques developed to improve the 2D feature detection accuracy. Besides these global operation to reduce noise and remove outliers, we also implement the RANSAC method during calibration. Since we know groundtruth gaze points during calibration, RANSAC enables us to remove outliers and find inliers to better estimate subject-dependent eye parameters.

4) Fusion Results from Left and Right Eyes: Contrary to most IR lights based gaze tracking system, where camera is focused on human eyes to improve feature detections. Web camera and Kinect based gaze tracking system can capture the upper body of subjects. Therefore Kinect based system is more likely to capture image of two eyes than IR lights based system. In the experiments, we perform the same personal calibration and gaze estimation procedure for left and right eyes, the final PoR is the average of the PoR from left and right eyes. However, if the PoR from one of the eye falls out of the display region, we consider this as outliers and simply use the PoR from another eye. This also applies to the case where results from two eyes are both outliers.

B. Experiments with Real Subjects

We test the proposed gaze tracking system with Kinect for 6 subjects. None of the subjects wear glasses since glasses block eyes and cause the depth estimation error of pupil. Each subject is firstly asked to perform a 5-points personal

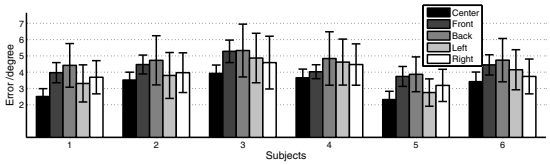

Fig. 3. Comparison of Different Operating Regions. The line segment on the bar represents the standard deviation.

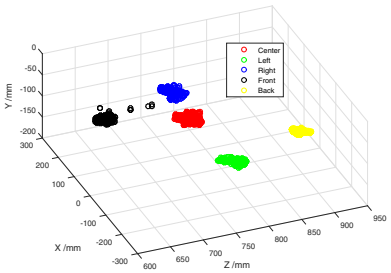

Fig. 4. Visualization of Different Regions in Camera Coordinates System. Kinect lies approximately at [200, -100, 0] mm, and the optical axis is along positive z direction.

calibration. Then they are asked to look at 15 random points, which is used for evaluation. Each point is displayed for approximately 4 seconds where 80 frames of data can be collected. We use the PoR prediction error to evaluate the performance of the gaze tracking system.

1) Personal Calibration Evaluation: For each subject, we solve the optimization problem in Eqn. 10 to estimate the subject-dependent parameters. Table I shows the estimated eye parameters for the 6 subjects. The first two numbers are the kappa angles α and β in degree. Then third and fourth number are eyeball radius and the distance between eyeball and cornea center. The final parameter is the offset between head and eyeball center $\mathbf{V}_{he}$. We can see the kappa angles and other parameters are different from each other, which illustrate the importance of personal calibration. This difference also causes poor performance of methods ignoring these personal parameters. Moreover, personal calibration not only estimates the subject-dependent parameters, but also compensate the built-in consistent error of the gaze tracking system. For example, the incorrect camera calibration, display-camera calibration results or the facial landmark detection bias can be compensated through the personal calibration procedure. Besides, as for efficiency, the 5-points calibration procedure takes averagely 20 seconds to collect the calibration data and 1 second to solve the subject-dependent parameters. This means after approximately 21 seconds of personal calibration procedure, subjects are able to perform free gaze tracking with the proposed Kinect based gaze tracking system.

2) Gaze Estimation Evaluation with Different Operating Regions: To better evaluate the proposed gaze estimation method with Kinect, we test the estimated eye parameters on different operating regions. In particular, we consider 5 rough regions (Center, Front, Back, Left and Right). Within each region, subjects are allowed to move or rotate their head freely as long as head positions still belong to the specific region. During the experiments, subjects perform one-time personal calibration in the "Center" region, then they are asked to perform the testing tasks 5 times in the 5 different regions. Figure 3 shows the gaze estimation error for 6 subjects in the 5 regions. To better understand the rough positions of the 5 regions, we visualize the 3D coordinates of one rigid facial landmark (nose tip) from Subject 1 as shown in Figure 4. The rigid facial landmark can represent the head translations ($\mathbf{T}$). Note that the rest 5 subjects operates at similar regions like subject 1.

From Eqn. 8, we can see that the gaze estimation error comes from two factors. One is the incorrect features $\{\mathbf{p}, \mathbf{R}, \mathbf{T}\}$ resulted from poor feature detections. Another is the incorrect eye parameters θ. In Figure 3, the best result comes from "Center" (3.2 degree) region. This is because the personal calibration is performed in the same region and the estimated eye parameters can match perfectly to the testing data. The worst results come from "Front" (4.3 degree) and "Back" (4.6 degree) regions. In these two regions, the operating distance differs a lot compared to "Center" region, results in the size change of face and eye in image coordinates. Therefore both feature detection and the mis-matched eye parameters contribute to the increased gaze estimation error. The results from "Left" (3.8 degree) and "Right" regions (3.9 degree) are worse than the "Center" results but better than "Front" and "Back" results. Since the operating distance of "Left" and "Right" regions are similar to "Center" regions, thus the eye parameters estimated in "Center" region are able to adapt to data from these two regions. Despite the increased gaze estimation error in regions outside the calibration region, we can see the overall gaze estimation error can still achieve 4.0 degree, which is suitable for may applications.

3) Gaze Estimation Evaluation with Manually Labeled Pupil Positions: The novelty of this paper mainly lies in the new gaze estimation framework with Kinect. However, the accuracy of the system relies on accurate feature detections like the pupil detections. We have evaluated the pupil detection algorithm on the popular BioID dataset, the percentage for 0.05 normalized error (approximately within pupil region) is 89.2. This explains why we can get good gaze estimation accuracy. But in order to better evaluate the proposed gaze estimation framework, we also manually label the 2D pupil positions on color image to remove the effect of poor pupil detections. Figure 5 shows the gaze estimation error using manually labeled pupil and automatically detected pupil for the 6 subjects in "Left" region. With manually labeled pupil positions, gaze estimation error reduces significantly from 3.9 degree to 3.0 degree. The reduced variance also proves the improved robustness with manually labeled pupil positions. We believe the performance of the proposed gaze tracking

system will further increase with better feature detection techniques.

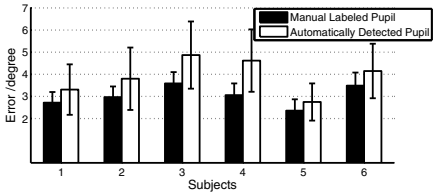

Fig. 5. Comparison of Manual Labeled and Automatically Detected Pupil Positions.

4) Comparison with State-of-the-Art Kinect-based gaze estimation methods: Generally speaking, gaze estimation methods under infrared lights can achieve relative good gaze estimation accuracy ($<$2 degree), since pupil positions can be accurately detected in infrared spectrum. However, methods with infrared lights suffer from complex system setup and limited working environments. Therefore we mainly focus on comparing methods using Kinect. Table II shows the comparison results. We can find the proposed method outperforms methods proposed from [19], [18] and [15]. The method proposed by Sun *et al* [20] achieves the best accuracy among methods using Kinect. However, their method suffers from several limitations. First of all, they set the resolution of color stream to be 1280×960, which benefits the pupil detection procedure and results in better gaze estimation accuracy. However, the good accuracy is at the cost of sacrificing speed (12 fps) compared to 30 fps with resolution 640×480. Besides, the operating distance is relatively short (550 mm) compared to 800 mm (Figure 4) in our case. Overall, the proposed methods achieve comparable gaze estimation accuracy with low resolution images and higher speed (20 fps).

TABLE II
COMPARISON WITH STATE-OF-THE-ART KINECT-BASED METHODS

Method	Error /degree
proposed	4.0
[19]	4.4
[18]	$<$10
[15]	5
[20]	$<$2

VI. CONCLUSION

In this paper, we propose a simple low-cost gaze tracking system with Kinect. We adapt the 3D eye model in IR-based approaches into this work to model the human vision system as accurate as possible. With the 3D eye model, we propose a personal calibration framework to estimate the 7-dimensional personal parameters, which can be used to estimate gaze directions or PoRs on the screen. Experimental results with 6 subjects under different head poses prove the effectiveness of the proposed method. Besides, by using manually labeled pupil positions, we can obtain a lower bound of the gaze estimation error. We can only expect the improvement of gaze estimation accuracy with new eye feature detection techniques. Furthermore, benefited from using low resolution color stream (640×480, 30 fps) instead of high resolution color stream (1280×960, 12 fps), we are able to achieve real time gaze tracking with 20 fps.

ACKNOWLEDGMENT

The work described in this paper is supported in part by an award from the National Science Foundation under the grant number IIS 1539012.

REFERENCES

[1] Steelseries, "https://steelseries.com/gaming-controllers/sentry-gaming-eye-tracker."
[2] M. Mason, B. Hood, and M. C, "Look into my eyes : Gaze direction and person memory," *Memory*, 2004.
[3] D. Beymer and M. Flickner, "Eye gaze tracking using an active stereo head," *IEEE Conference in Computer Vision and Pattern Recognition*, 2003.
[4] S. W. Shih and J. Liu, "A novel approach to 3d gaze tracking using stereo cameras," *IEEE Transactions on Systems, Man and Cybernetics, PartB*, 2004.
[5] E. D. Guestrin and M. Eizenman, "General theory of remote gaze estimation using the pupil center and corneal reflections," *IEEE Transcations on Biomedical Engineering*, 2006.
[6] J. Chen, Y. Tong, W. Gary, and Q. Ji, "A robust 3d eye gaze tracking system using noise reduction," *Proceedings of the 2008 symposium on Eye tracking research and applications*, 2008.
[7] F. Lu, Y. Sugana, T. Okabe, and Y. Sato, "Inferring human gaze from appearance via adaptive linear regression," *In Proc. International Conference on Computer Vision*, 2011.
[8] K. H. Tan, D. Kriegman, and N. Ahuja, "Appearance-based eye gaze estimation," *In Proc. 6th IEEE Workshop on Applications of Computer Vision*, 2002.
[9] O. Williams, A. Blake, and R. Cipolla, "Sparse and semi-supervised visual mapping with the s3gp," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2006.
[10] D. Hansen and Q. Ji, "In the eye of the beholder: A survey of models for eyes and gaze," *TPAMI*, 2010.
[11] Tobii, "http://www.tobii.com/xperience/."
[12] EyeTribe, "https://theeyetribe.com/."
[13] Y. Sugano, Y. Matsushita, and Y. Sato, "Learning-by-synthesis for appearance-based 3d gaze estimation," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.
[14] H. Yamazoe, A. Utsumi, T. Yonezawa, and S. Abe, "Remote and head-motion-free gaze tracking for real environments with automated head-eye model calibrations," in *Computer Vision and Pattern Recognition Workshops, 2008. CVPRW '08. IEEE Computer Society Conference on*, June 2008, pp. 1–6.
[15] K. A. Funes Mora and J. Odobez, "Geometric generative gaze estimation (g3e) for remote rgb-d cameras," *Computer Vision and Pattern Recognition*, 2014.
[16] Y. Li, D. Monaghan, and N. O Connor, "Real time gaze estimation using a kinect and a hd webcam," *The 20th Anniversary International Conference on MultiMedia Modeling*, 2014.
[17] R. Jafari and D. Ziou, "Gaze estimation using kinect/ptz camera," *Robotic and Sensors Environments (ROSE), 2012 IEEE International Symposium on*, 2012.
[18] J. Li and S. Li, "Eye-model-based gaze estimation by rgb-d camera," *Computer Vision and Pattern Recognition Workshops*, 2014.
[19] X. Xiong, Q. Cai, Z. Liu, and Z. Zhang, "Eye gaze tracking using an rgbd camera: A comparison with a rgb solution," *UBICOMP*, 2014.
[20] L. Sun, M. Song, Z. Liu, and M. Sun, "Real-time gaze estimation with online calibration," *IEEE Multimedia and Expo*, 2014.
[21] X. Xiong and F. D. la Torre, "Supervised descent method and its application to face alignment," *CVPR*, 2013.
[22] S. Li, K. Ngan, and L. Sheng, "Screen-camera calibration using a thread," *ICIP*, 2014.
[23] M. Camplani and L. Salgado, "Efficient spatio-temporal hole filling strategy for kinect depth maps," *Proceedings of the SPIE*, 2012.