

Calibration of Iris Radius in Single Camera and Single Light Gaze Tracking System

Lin-Na Liu, Xiao-Cui Zhang, Jing-Yao Hu, Yong-Yue Xing, Jian-Nan Chi

Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing, 100083, China

E-mail: liulinna_1166@sina.com, zsc814064@sina.com

jingyao_hu@sina.com, xingyongyue@outlook.com, sy_jnchi@126.com

Abstract—In a single camera single source gaze tracking system, if we can accurately estimate the 3-D coordinates of the iris center and the center of corneal surface curvature, we can reconstruct 3-D gaze direction. A novel method is proposed in this paper, different from other gaze tracking systems, the user calibration in this paper is the calibration of the iris radius, corneal curvature radius and kappa angle. Through the simulation experiments, the feasibility of the algorithm proposed in this paper is analyzed, and the error of the algorithm is relatively small.

Keywords—gaze tracking system; calibration of iris; single camera single source; cornea

I. INTRODUCTION

A 3D gaze direction vector can't be obtained by single camera and single light approach in theory. It was proved in the literature [1] that in order to obtain the 3-D gaze direction, the gaze tracking system contains at least two cameras. But there are still documents discuss the method of estimating the 3-D gaze direction that bases on single camera system [2,3,4]. The current single camera single source system has no yet achieved the 3-D gaze estimation, methods rely more on priori eye parameters. Because these parameters are individual differences, using constant geometric models of the line of sight to calculate the direction of the optical axis and gaze direction, will inevitably cause large errors. In this paper, the iris is regarded as a circular object, and the normal direction and center coordinates of iris are estimated to reconstruct the optical axis of eye.

II. THE METHOD OF 3-D GAZE ESTIMATION BASED ON THE SINGLE CAMERA AND SINGLE LIGHT SYSTEM

The goal of gaze tracking is to determine the 3-D visual line LoS. But it is difficult to estimate the visual line LoS directly. At present, the general approach is to estimate the LoG firstly, and then converted into LoS. As figure 1 shows, the optical axis of the eye contains four principal points of the eyeball, and their positional vectors are denoted by E—the eyeball center, P—the pupil center, C—the cornea center, and I—the iris center. In the above four points, point I, P and C are used to reconstruct the optical axis. Besides, optical axis is always perpendicular to the pupil plane and the iris margin plane. Therefore, if the pupil and iris are regarded as circular objects in space, the optical axis can be

regarded as the normal line of circular objects.

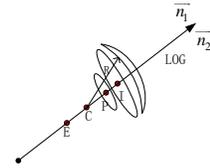


Figure 1 Spatial geometry of the optical axis.

In summary, If the space coordinates of any two points among I, P, C are determined, or identify the normal direction of iris and any one space coordinate of I, P, C, or identify the normal direction of pupil and any one space coordinate of I, P, C, the optical axis can be reconstructed. The normal direction of the pupil plane and the iris plane are denoted by \vec{n}_1 and \vec{n}_2 respectively. In fact, \vec{n}_1 and \vec{n}_2 are the same, both of them represent the direction of optical axis. The space coordinates of I, P and C are denoted by $(x_I, y_I, z_I), (x_P, y_P, z_P), (x_C, y_C, z_C)$ respectively. The above conclusions are shown in figure 2.

$$\left. \begin{array}{l} \vec{n}_1 + I(x_I, y_I, z_I) \\ \vec{n}_1 + P(x_P, y_P, z_P) \\ \vec{n}_1 + C(x_C, y_C, z_C) \\ \vec{n}_2 + I(x_I, y_I, z_I) \\ \vec{n}_2 + P(x_P, y_P, z_P) \\ \vec{n}_2 + C(x_C, y_C, z_C) \\ I(x_I, y_I, z_I) + C(x_C, y_C, z_C) \\ I(x_I, y_I, z_I) + P(x_P, y_P, z_P) \\ C(x_C, y_C, z_C) + P(x_P, y_P, z_P) \end{array} \right\} \Rightarrow LoG$$

Figure2. Reconstruction of line of sight

III. CALIBRATION OF IRIS RADIUS

When the iris radius is known, the 3-D coordinates and the normal direction of the iris center in the camera coordinate system can be determined based on the single camera image. Different from the pupil, the iris radius of the human eye is constant. Therefore, if the iris radius can be calibrated in the user calibration process, the 3-D coordinates of iris center can be calculated and the optical axis of the eye can be reconstructed with the normal

direction of the iris center.

The 3-D coordinates of iris center, denoted by I_1 , I_2 , and the unit normal vector of the iris edge plane, denoted by D_1 , D_2 , can be obtained respectively by using iris image, and can be expressed by iris radius as follows:

$$I_1 = R * \begin{bmatrix} i_{11} \\ i_{12} \\ i_{13} \end{bmatrix}, I_2 = R * \begin{bmatrix} i_{21} \\ i_{22} \\ i_{23} \end{bmatrix} \quad (1)$$

$$D_1 = \begin{bmatrix} d_{11} \\ d_{12} \\ d_{13} \end{bmatrix}, D_2 = \begin{bmatrix} d_{21} \\ d_{22} \\ d_{23} \end{bmatrix} \quad (2)$$

in formula, in addition to the radius of the iris, the remaining amount is known. Therefore, the approximate formula of optical axis is expressed as follows:

$$\frac{x - r * i_{11}}{d_{11}} = \frac{y - r * i_{12}}{d_{12}} = \frac{z - r * i_{13}}{d_{13}} \quad (3)$$

$$\frac{x - r * i_{21}}{d_{21}} = \frac{y - r * i_{22}}{d_{22}} = \frac{z - r * i_{23}}{d_{23}} \quad (4)$$

Fig. 3. shows the spatial relationship between the iris center I_1 , I_2 , the direction vector D_1 , D_2 , and the optical axis. Because the program assumes that the symmetrical axis of the iris circle is the optical axis, the axis determined by I_1 , I_2 , D_1 , D_2 must pass the center of the outer surface of the cornea.

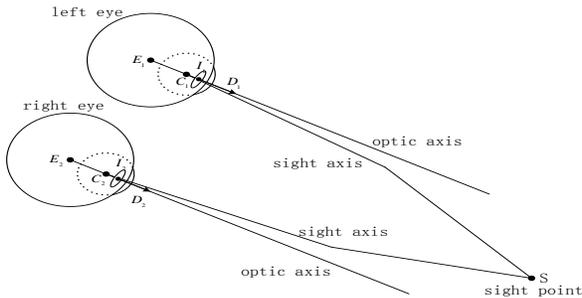


Figure 3. the relationship between the characteristics of the eye.

In addition to the above relation, according to the literature [6], the cornea center C is also contained in the plane determined by light source, the camera projection center and the glint position in the image. As shown in Fig. 4.

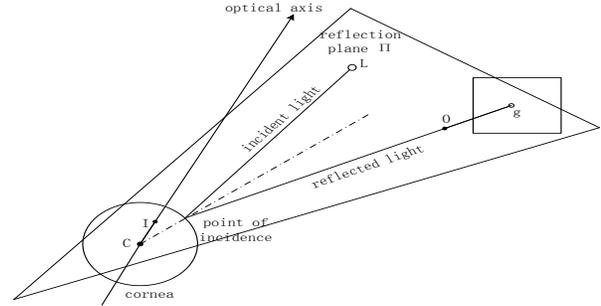


Figure 4. The cornea center is contained in a plane together with the glint, light source, projection center of the camera.

In Fig. 4, light source L produces a reflection on the corneal surface. This reflection is reproduced in the images as an area of bright connected pixels, and the glint position in the image is denoted by g . By the law of reflection, the incident and reflected light, and the normal vector are coplanar. It is deduced that the cornea center C is contained in the same plane, since it is contained in the normal line contained by the plane. In addition, the light source L , the camera projection center O , and the image of Purkinje spot g , will also be located in the same plane. The special position of the light source L in the camera coordinate system, denoted by $L(x_l, y_l, z_l)$, can be calibrated by the system calibration. The special position of the image of Purkinje spot g in the camera coordinate system, denoted by $g(x_g, y_g, z_g)$, can be obtained by image processing and imaging model. Therefore, based on the special coordinates of light source L , image of Purkinje spot g and the camera projection center O , a plane containing the cornea center C can be obtained. The plane is calculated as follows:

$$\Pi = L \times g \quad (5)$$

substituting $L(x_l, y_l, z_l)$ and $g(x_g, y_g, z_g)$ into Eq. (5) and rearranging, we have the equation of the plane:

$$(y_l z_g - z_l y_g)x + (z_l x_g - x_l z_g)y + (x_l y_g - y_l x_g)z = 0 \quad (6)$$

therefore, the special position of the cornea center C can be determined by the plane Π and the optical axis of the human eye. Taking the left eye as an example, this paper introduces the process of obtaining the spatial position of cornea center C_1 . Passing through the left and right eyes the reflected light form different images, and the image formed by the reflection of the light source through the left eye is denoted by $g_1(x_{g1}, y_{g1}, z_{g1})$. The plane determined by light source L and Purkinje spot g is calculated as follows:

$$(y_l z_{g1} - z_l y_{g1})x + (z_l x_{g1} - x_l z_{g1})y + (x_l y_{g1} - y_l x_{g1})z = 0 \quad (7)$$

The cornea center C can be formulated by solving Eq. (3) and Eq. (7):

$$C_1 = r * \begin{bmatrix} \frac{(z_l x_{g1} - x_l z_{g1})(d_{12} i_{12} - d_{11} i_{12}) + (x_l y_{g1} - y_l x_{g1})(d_{13} i_{11} - d_{11} i_{13})}{(y_l z_{g1} - z_l y_{g1})d_{11} + (z_l x_{g1} - x_l z_{g1})d_{12} + (x_l y_{g1} - y_l x_{g1})d_{13}} \\ \frac{(y_l z_{g1} - z_l y_{g1})(d_{11} i_{12} - d_{12} i_{11}) + (x_l y_{g1} - y_l x_{g1})(d_{13} i_{12} - d_{12} i_{13})}{(y_l z_{g1} - z_l y_{g1})d_{11} + (z_l x_{g1} - x_l z_{g1})d_{12} + (x_l y_{g1} - y_l x_{g1})d_{13}} \\ \frac{(y_l z_{g1} - z_l y_{g1})(d_{11} i_{13} - d_{13} i_{11}) + (z_l x_{g1} - x_l z_{g1})(d_{12} i_{13} - d_{13} i_{12})}{(y_l z_{g1} - z_l y_{g1})d_{11} + (z_l x_{g1} - x_l z_{g1})d_{12} + (x_l y_{g1} - y_l x_{g1})d_{13}} \end{bmatrix} \quad (8)$$

in the Eq. (8), only the iris radius is unknown. Based on the assumption that the iris radius of the left and right eyes are the same, the cornea center C_1 , C_2 can be expressed as follows:

$$C_1 = r * \begin{bmatrix} c_{11} \\ c_{12} \\ c_{13} \end{bmatrix}, C_2 = r * \begin{bmatrix} c_{21} \\ c_{22} \\ c_{23} \end{bmatrix} \quad (9)$$

where $c_{11}, c_{12}, c_{13}, c_{21}, c_{22}, c_{23}$ is constant.

The intersection of sight line is the focus of attention, which can be used to obtain the spatial position of object of attention. Binocular strategy has received a lot of attention from researchers recently [7] and this technology has been applied to various areas, such as driver's helmet and virtual reality headset. In this work, it is assumed that the kappa angles of left and right eyes are equal. The unit vector of the optical axis D_1 D_2 has been obtained and the unit vector of binocular visual axis can be expressed as follows:

$$V_1 = \frac{S - C_1}{\|S - C_1\|}, V_2 = \frac{S - C_2}{\|S - C_2\|} \quad (10)$$

where S is the 3-D coordinates of the fixation point, and a many times a dollar equation in terms of iris radius can be obtained as follows:

$$\frac{D_1 \cdot (S - C_1)}{\|S - C_1\|} = \frac{D_2 \cdot (S - C_2)}{\|S - C_2\|} \quad (11)$$

Taking the left eye as an example, the iris radius R is calculated by solving Eq. (1), (3), (7), (11).

IV. SIMULATION EXPERIMENTS

In this paper, the algorithm is verified by simulation with Rhino5.0 and Matlab R2015a. The simulation procedure is as follows:

- 1) Establish a three-dimensional simulation model in the Rhino5.0 software, as shown in Fig. 5.

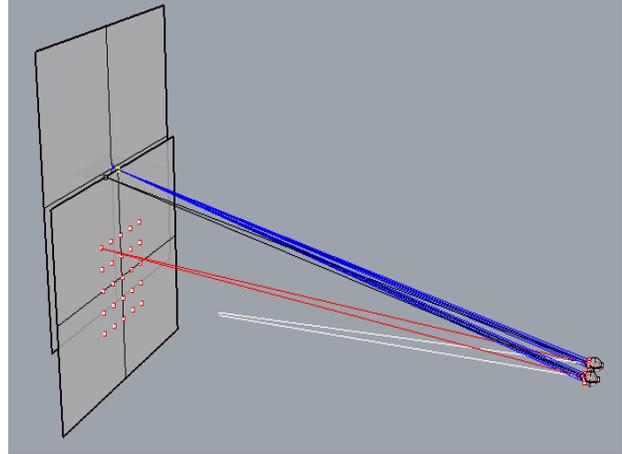


Figure 5. The three dimensional simulation model (the red line simulate the line of sight, the black simulate the incident and reflected light in the cornea, the blue line simulate iris edge.)

- 2) Extract line of sight model parameters from three-dimensional simulation diagram: These parameters include: (a) eye parameters: the distance between the eyes is 53mm, the eyeball radius is 12mm, the corneal curvature radius is 7.8mm, the distance between the cornea center and the eyeball center is 5.84mm, the iris radius is 6.57mm, the distance between the pupil center and cornea center is 4.2mm, (b) camera parameters: the camera lens focus is 16mm, (c) description of coordinate system: the origin of the camera coordinate system is aligned with the optical center of the camera, x-axis and y-axis are in the plane of screen, and the positive direction of z-axis is defined by the outward direction perpendicular to the screen, (d) the coordinates of source light L is (0,50,0), (e) 9 calibration points on the screen: S1(-170,50,0), S2(-170,0,0), S3(-170,-50,0), S4(-200,50,0), S5(-200,0,0), S6(-200,-50,0), S7(-230,50,0), S8(-230,0,0), S9(-230,-50,0), (f) 25 test points on the screen: S1(-120,80,0), S2(-120,40,0), S3(-120,0,0), S4(-120,-40,0), S5(-120,-80,0), S6(-160,80,0), S7(-160,40,0), S8(-160,0,0), S9(-160,-40,0), S10(-160,-80,0), S11(-200,80,0), S12(-200,40,0), S13(-200,0,0), S14(-200,40,0), S15(-200,-80,0), S16(-240,80,0), S17(-240,40,0), S18(-240,0,0), S19(-240,40,0), S20(-240,-80,0), S21(-280,80,0), S22(-280,40,0), S23(-280,0,0), S24(-280,40,0), S25(-280,-80,0)

- 3) Use the Matlab to make up simulation programs: Input the experimental data obtained from Rhino5.0 software to the Matlab programs and run the programs to get the simulation results. The simulation data are as

follows:

- (a) The simulation results of user calibration:
- (b) The simulation results of line of sight estimation

TABLE I. TABLE RESULTS FOR CORNEA RADIUS CALIBRATION

Calibration point	Right eye			Left eye		
	gama angle/degree	Pupil center/mm	Purkinje image/mm	gama angle/degree	Pupil center/mm	Purkinje image/mm
S1	6.32	(-4.38937,-0.550461)	(-4.38937,-0.550461)	2.92	(-4.39967,0.59975)	(-4.35959,0.593656)
S2	3.03	(-4.38956,-0.562157)	(-4.38937,-0.550461)	3.11	(-4.39966,0.587961)	(-4.35959,0.593656)
S3	2.96	(-4.38956,-0.57391)	(-4.38937,-0.550461)	6.42	(-4.39947,0.576232)	(-4.35959,0.593656)
S4	5.91	(-4.39643,-0.550452)	(-4.38937,-0.550461)	1.81	(-4.40678,0.599759)	(-4.35959,0.593656)
Cornea radius/mm	7.7693			7.8733		
Mean cornea radius/mm						
S1	7.8213					

TABLE II. TABLE RESULTS FOR KAPPA ANGLE AND IRIS CALIBRATION

Calibration point	Right eye		Left eye	
	Kappa angle/degrees	iris radius/mm	Kappa angle/degrees	iris radius/mm
S1	5.0358	6.5906	5.0422	6.5911
S3	4.9822	6.565	5.0325	6.5903
S7	5.0461	6.5903	5.0492	6.5904
S9	5.05	6.5905	5.0456	6.5902
Mean value	5.0285	6.5841	5.0424	6.5905
Error	0.0285	0.0141	0.0424	0.0205

TABLE III. TABLE RESULTS FOR LINE OF SIGHT ESTIMATION

Binocular center position/mm	Observation point	Right eye		Left eye		Gaze point error/m
		Kappa angle/degrees	kappa angle error/degrees	Kappa angle/degrees	kappa angle error/degrees	
(-200,0,300)	S1	4.8382	0.1618	4.8283	0.1717	1.5137
	S13	5.0248	0.0248	5.0386	0.0386	0.2816
(-200,-300,500)	S5	4.9501	0.0499	4.9513	0.0487	2.6392
	S17	5.0266	0.0266	5.0416	0.0416	3.7217
(-200,0,1000)	S1	4.9919	0.0081	5.0046	0.0046	0.4603
	S25	5.0262	0.0262	5.0399	0.0399	0.2661

Simulation results show that the error of cornea radius is $7.8213-7.8=0.0213\text{mm}$, kappa angle errors are 0.0424 and 0.0285 for the right and left eyes, respectively. These errors are in a small range, which shows that the calibration results are accurate in the simulation environment. In summary, the algorithm this paper proposed is feasible.

V. CONCLUSION

Through the simulation experiments, the feasibility of the algorithm proposed in this paper is analyzed, and the error of the algorithm is relatively small. According to analysis, the following conclusions can be drawn: the results of line of sight estimation is mainly influenced by the accuracy of the iris edge detection, and the impact of deviation of the light source and Purkinje spot noise is very

small. Therefore, the iris segmentation is very important in the gaze tracking system.

ACKNOWLEDGMENT

This work is supported by the Fundamental Research Funds for the Central Universities under Grant FkF-TP-15-059A3.

REFERENCES

- [1] Sheng-Wen Shih, Jin Liu. A Novel Approach to 3D Gaze Tracking Using Stereo Cameras[J]. IEEE TRANSACTIONS ON SYSTEMS MAN AND CYBERNETICS-PART B: CYBERNETICS.2004, 34(1):234-256.
- [2] H Morimoto, A. Amir, M. Flickner. Detecting eye position and gaze from a single camera and 2 light sources[C]. In Proc. ICPR. Washington, DC, USA: IEEE Press, 2002:314-317.

- [3] T Ohno, N. Mukawa, A Yoshikawa. Freegaze: A gaze tracking system for everyday gaze interaction[J]. In Processings of Symposium. ETRA. New Orleans, USA, ACM Press: 2002. 125–132.
- [4] C. Hennessey, B. Nouredin, and P. Lawrence. A Single Camera Eye-Gaze Tracking System with Free Head Motion[C]. Processings of Symposium. Eye Tracking Research and Applications, pp. 87-94, 2006.
- [5] Shiu Y C, Ahmad S. 3D location of circular and spherical features by monocular model-based vision[C]. Conference Proceedings of IEEE International Conference on Systems, Man and Cybernetics, 1989: 576-581.
- [6] Jacob R J K. The use of eye movements in human computer interaction techniques: what you look at is what you get[J]. ACM Transactions on Information Systems, 1991, 9: 152-169
- [7] J. Wang, E. Sung. Study on eye gaze estimation[J], IEEE Transactions on systems, man, and cybernetics - PART B 32 (3) (2002) 332–350