3D Gaze Estimation with a Single Camera without IR Illumination

Conference Paper · January 2009

DOI: 10.1109/ICPR.2008.4761343 · Source: IEEE Xplore

2 authors:

Jixu Chen
General Electric

Qiang Ji
Tongji Medical University

3D Gaze Estimation with a Single Camera without IR Illumination

Jixu Chen Qiang Ji
Rensselaer Polytechnic Institute, Troy, NY, 12180-3590, USA
chenj4@rpi.edu jiq@rpi.edu

Abstract

This paper proposes a 3D eye gaze estimation and tracking algorithm based on facial feature tracking using a single camera. Instead of using the infrared(IR) lights and the corneal reflections (glint), this algorithm estimates the 3D visual axis using the tracked facial feature points. For this, we first introduce an extended 3D eye model which includes both the eyeball and the eye-corners. Based on this eye model, we derive the equations to solve for the 3D eyeball center, the 3D pupil center and the 3D visual axis, from which we can solve for the point of gaze after a one-time personal calibration. The experimental results show the accuracy of this algorithm is less than 3^o. Compared with the existing IR-based eye tracking methods, the proposed method is simple to setup and can work both indoor and outdoor.

1 Introduction

Gaze tracking is the procedure of determining the point-of-gaze in the space, or the visual axis of the eye. Since it is very useful in many applications, such as natural human computer interaction and human state detection, many algorithms have been proposed.

Currently, most gaze tracking algorithms ([7],[12],[3]) and commercial gaze tracking products ([1], [2]) are based on Pupil Center Corneal Reflection(PCCR) technique. One or multiple Infrared (IR) lights are used to illuminate the eye region and to build the corneal refection (glint) on the corneal surface. At the same time, one or multiple cameras are used to capture the image of the eye. By detecting the pupil position and the glints in the image, the gaze direction can be estimated based on the relative position between the pupil and the glints. However, the gaze tracking tracking systems based on IR illumination have many limitations. First, the IR illumination can be affected by the sunshine in outdoor scenario. Second, the relative position between the IR lights and the camera need to be calibrated carefully. Third, because the pupil and the glint are very small, usually a high-resolution camera is needed. So, most current gaze tracking system can only work in indoor scenario.

Recently, different eye gaze-tracking algorithms without IR lights have also been proposed. Wang et. al[10] and Kohlbecher [5] propose to perform eye gaze tracking based on estimating the shape of the detected iris or pupil through an ellipse fitting, and using the estimated pupil or iris ellipses to infer the eye gaze. The accurate estimation of the shapes of iris or pupil is often difficult since they are often occluded by eyelids or face pose. In addition, it requires a high resolution camera in order to accurately capture the pupil or iris. To overcome these problems, facial features based eye tracking methods have been proposed recently. This kind of methods can work properly without IR lights and the facial features can be more easily detected than either pupil or iris. In [4] and [11], the locations of the iris and the eye-corners in the image are tracked from a single camera. Then these 2D eye features in image can be used to estimate the horizontal and vertical gaze angles (θ_x, θ_y) in 3D space. Matsumoto et. al [6] proposed to use stereo camera to estimate the 3D eye position and 3D visual axis. Our algorithm is similar to Matsumoto's method, the difference is that we use only one camera to estimate the 3D eye features. Furthermore, all the above methods ignore the difference between the eyeball center and the corneal center, and the difference between optical axis and visual axis. Based on the anatomical structure of the eyeball, we propose an extended 3D eye model which includes eye corners. By solving the equations of this 3D eye model, we can estimate the 3D visual axis. Our method can achieve accurate gaze estimation under free head movement.

The following sections are organized as follows: In section 2, we proposed the 3D eye model and gaze estimation algorithms. Then, the calibration algorithm is discussed in section 3. Finally, experiment result is shown to verify our method.

978-1-4244-2175-6/08/$25.00 ©2008 IEEE

2 Gaze estimation algorithm

The key of our algorithm is to compute the 3D position of the eyeball center **C** based on the middle point **M** of two eye corners ($\mathbf{E}_1$, $\mathbf{E}_2$) (Fig. 1). The 3D model is based on the anatomical structure of the eye [3, 8]. As shown in Figure 1, the eyeball is made up of the segments of two spheres with different sizes. The anterior smaller segment is the cornea. The cornea is transparent, and the pupil is inside the cornea. The optical axis is defined as the 3D line connecting the corneal center $\mathbf{C}_0$ and the pupil center **P**. Since the gaze point is defined as the intersection of visual axis rather than the optical axis with the scene. The relationship between these two axis has to be modeled. The angle between the optical axis and the visual axis is named as $kappa$, which is a constant value for each person.

When the person gazes at different directions, the corneal center $\mathbf{C}_0$ and the pupil center **P** will rotate around the eyeball center **C**, and $\mathbf{C}_0$,**P**, **C** are all on the optical axis. Since **C** is inside the face, we have to estimate its position from the facial point on the face surface: $\mathbf{E}_1$ and $\mathbf{E}_2$ are the two eye corners and **M** is their middle point. The offset vector **V** between **M** and **C** is related to the face pose. Based on the eye model, we can estimate the gaze step by step as follows.

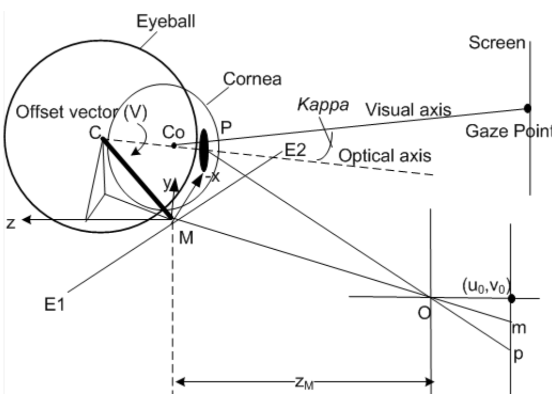


Figure 1. 3D eye model

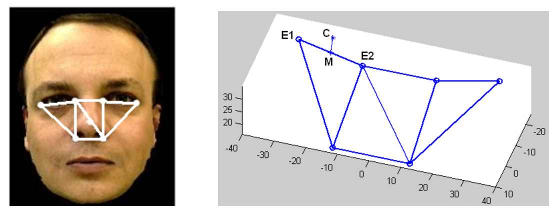
2.1 Gaze estimation algorithm

2.1.1 Step 1. Facial Feature Tracking and Face Pose Estimation

First, we employ the facial feature tracking algorithm in [9] to track the facial points and estimate the face pose vector $\alpha = (\sigma_{pan}, \phi_{tilt}, \kappa_{swing}, s)$, where $(\sigma_{pan}, \phi_{tilt}, \kappa_{swing})$ are the three face pose angles and s is the scale factor. Same as [9] , we use a generic 3D face model, which is composed of six rigid face points (Figure 2(a)), to estimate the face pose. Actually, the 3 face pose angles can define a 3×3 rotation matrix R to rotate the 3D face point from the face-model coordinate to the camera coordinate. Assuming weak perspective projection, the projection from 3D point in face-model coordinate to the the 2D image point is defined as

$$\begin{pmatrix} u_i \\ v_i \end{pmatrix} = sR_{1,2} \begin{pmatrix} x_i \\ y_i \\ z_i \end{pmatrix} + \begin{pmatrix} u_0^f \\ v_0^f \end{pmatrix} \qquad (1)$$

$R_{1,2}$ is a 2×3 matrix which is composed of the first two rows of the rotation matrix R. $(u_0^f, v_0^f)^T$ is the projection of the face-model origin. (Here, the face origin is defined as the nose tip and the z axis is pointing out the face.)



(a) A frontal face image and the selected rigid points and triangles.

(b) 3D face model with middle point and eyeball center.

Figure 2. 3D generic face model.

2.1.2 Step 2. Estimate the 3D point M

To estimate the 3D eyeball center **C**, we extend the traditional face model by adding two points(**C** and **M**) in the 3D face model, as shown in Fig.2(b). So, although the camera cannot capture the eyeball center **C** directly, we can first estimated the 3D point **M** from the tracked facial feature points, and then estimate the **C** position from that of **M**. Our implicit assumption here is that the relative spatial relationships between **M** and **C** is fixed independent of gaze direction and head movement. Such relationships may, however, vary slightly from person to person.

As shown in Fig. 1, from the tracked 2D eye corners points $E1$ and $E2$, the eye corner middle point in the image is estimated $\mathbf{m} = (u_m, v_m)^T$. If we can estimate the distance z_M from 3D point **M** to the camera, the 3D coordiantes of **M** can be recovered.

Same as [4], the 3D distance from the camera can be approximated using the fact that the distance is inversely proportional to the scale factor (s) of the face: $z_M = \frac{\lambda}{s}$. Here, the inverse-proportional factor λ can be recovered automatically in the user-dependent calibration procedure. (Section 3).

After the distance z_M is estimated, **M** can be recovered as follows:

$$\mathbf{M} = \frac{z_M}{f} \begin{pmatrix} u_m - u_0 \\ v_m - v_0 \\ f \end{pmatrix} \qquad (2)$$

where f is the camera focus length, $(u_0, v_0)^T$ is the projection of the camera origin, as shown in Fig. 1. Assuming we use a calibrated camera, f and $(u_0, v_0)^T$ are, therefore, known.

2.1.3 Step 3. Compute 3D position of C

We compute the eyeball center $\mathbf{C}$ based on the middle point $\mathbf{M}$ and the offset vector $\mathbf{V}$ in Fig. 1. And the offset vector $\mathbf{V}$ in the camera frame is related to the face pose.

Since the middle point $\mathbf{M}$ and the eyeball center point $\mathbf{C}$ are fixed relative to the 3D face model (Figure 2(b). Their position in this face-model coordinate are $\mathbf{M}^f = (x_M^f, y_M^f, z_M^f)^T$ and $\mathbf{C}^f = (x_C^f, y_C^f, z_C^f)^T$. Let the rotation matrix and the translation between the face coordinate and the camera coordinate be R and $\mathbf{T}$, so the offset vector between $\mathbf{M}$ and $\mathbf{C}$ in camera coordinate is:

$$\begin{aligned} &\mathbf{C} - \mathbf{M} \\ &= (R\mathbf{C}^f + \mathbf{T}) - (R\mathbf{M}^f + \mathbf{T}) \\ &= R(\mathbf{C}^f - \mathbf{M}^f) \\ &= R\mathbf{V}^f \end{aligned} \tag{3}$$

where $\mathbf{V}^f = \mathbf{C}^f - \mathbf{M}^f$ is a constant offset vector in the face model, independent of gaze direction and head position. During tracking, given the face pose R, and the $\mathbf{M}$ position in camera coordinate, then $\mathbf{C}$ in camera coordinate can be written as:

$$\mathbf{C} = \mathbf{M} + R\mathbf{V}^f \tag{4}$$

2.1.4 Step 4. Compute P

Given $\mathbf{C}$ and the pupil image $\mathbf{p} = (u_p, v_p)^T$, the 3D pupil position $\mathbf{P} = (x_P, y_P, z_P)^T$ can be estimated from its image corrdinates and using the assumption that the distance between $\mathbf{C}$ and $\mathbf{P}$ is a constant K. Specifically,**P** can be solved using the following equations.

$$\left\{ \begin{array}{c} \begin{pmatrix} u_p \\ v_p \end{pmatrix} = \frac{f}{z_P} \begin{pmatrix} x_P \\ y_P \end{pmatrix} + \begin{pmatrix} u_0 \\ v_0 \end{pmatrix} \\ \|\mathbf{P} - \mathbf{C}\| = K \end{array} \right. \tag{5}$$

2.1.5 Step 5. Compute Gaze

Since the distance (K_0) between the corneal center and the pupil center is also a constant, given the $\mathbf{P}$ and $\mathbf{C}$, the corneal center $\mathbf{C}_0$ can be estimated as:

$$\mathbf{C}_0 = \mathbf{C} + \frac{K_0}{K}(\mathbf{P} - \mathbf{C}) \tag{6}$$

Then, same as the method in [3], the visual axis can be obtained by adding a person-specific angle to the optical axis as follows:

$$\mathbf{V}_{vis} = f(\alpha, \beta; \mathbf{P} - \mathbf{C}_0) \tag{7}$$

Here $f(\alpha, \beta; \mathbf{V})$ is a function to add the horizontal angle (α) and vertical angle (β) to a vector $\mathbf{V}$.

3 Eye parameter calibration

In our gaze estimation algorithm, we use many person-specific parameters, including the offset vector $\mathbf{V}^f$, the distance between pupil and eyeball center K, the distance between cornea center and eyeball center K_0, the horizontal angle α and the versicle angle β between the optical axis and visual axis. In this section, we propose the following method to calibrate there parameters.

In our calibration procedure, the subject's head is fixed, and he/she gazes at 9 fixed points on the screen sequentially.

Figure 3 shows the procedure of calibration. When the subject is gazing at the calibration point $\mathbf{G}_i$ on the screen, the 3D pupil position is $\mathbf{P}_i$, and the 3D middle point is $\mathbf{M}_i$, their projection on 2D image are $\mathbf{p}_i$ and $\mathbf{m}_i$.

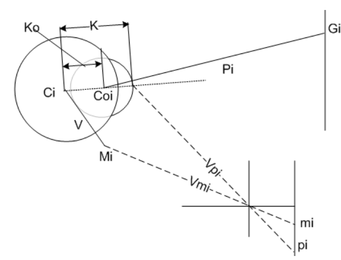


Figure 3. Eye parameter calibration.

From the 2D image projection, we can obtain the two lines going through $\mathbf{M}_i$ and $\mathbf{P}_i$, the direction (unit vector) of these two lines are $\mathbf{V}_{mi}$ and $\mathbf{V}_{pi}$.

During calibration we can assume the face pose R is fixed and known, so the offset vector in camera frame is fixed $\mathbf{V} = R\mathbf{V}^f$. Then, we can have the following equations:

$$\left\{ \begin{array}{lr} \|(K_{mi}\mathbf{V}_{mi} + \mathbf{V}) - K_{pi}\mathbf{V}_{pi}\| = K & (a) \\ \mathbf{G}_i = K_{gi} \cdot f(\alpha, \beta; [K_{pi}\mathbf{V}_{pi} - (K_{mi}\mathbf{V}_{mi} + \mathbf{V})]) + \mathbf{C}_{0i} & (b) \\ \mathbf{C}_{0i} = (K_{mi}\mathbf{V}_{mi} + \mathbf{V}) + \frac{K_0}{K}[K_{pi}\mathbf{V}_{pi} - (K_{mi}\mathbf{V}_{mi} + \mathbf{V})] & (c) \end{array} \right. \tag{8}$$

In our experiment, the K and $K0$ are fixed as average human eye value: $K = 13.1$mm, $K_0 = 5.3$mm. So, For N calibration point we have $7N$ equations. The unknowns are $\mathbf{V}$, α,β,$\mathbf{C}_{0i}$, K_{pi}, K_{mi} and K_{gi}. There are totally $6N + 5$ unknowns. So theoretically, $N = 9$ calibration points are enough to estimate the parameters. In practice we use optimization method to minimize the error between the estimated $\mathbf{G}_i$ and the ground truth gaze.

Note that, during calibration we can estimate the 3D middle point position ($M_i = K_{mi}\mathbf{V}_{mi}$). So, given the face scale s_i during calibration, the inverse-proportional factor λ can be estimated as the average of nine estimations $\lambda_i = s_i \cdot z_{Mi}$.

4 Experiment result

In this preliminary experiment, because the pupil is not clear in the camera, we manually label the iris center as the pupil center. The facial feature tracking result to estimate face pose and the eye features to estimate gaze are shown in Figure 4.

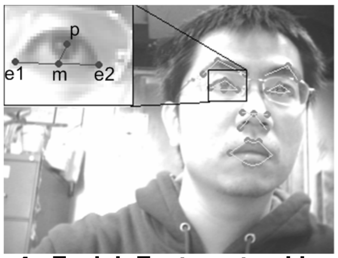


Figure 4. Facial Feature tracking result and the points to estimate gaze

During the calibration, the subject is calibrated in a fixed position (500mm from the camera), and the calibrated eye parameters are: Offset vector $V^f = [0.14, 0.09, -16.9]^T(mm)$, $\alpha = 0.026^o$, $\beta = -0.29^o$.

Then we first test the gaze estimation accuracy without head movement. The subject still keep his head in the calibration position and gaze at 9 points on the screen sequentially, we use the calibrated parameter to estimate the gaze. The estimated gazes (scan pattern) are shown in Figure 5. The accuracy can achieve: X accuracy =17.7mm (1.8288^o),Y accuracy =19.3mm (2.0^o).

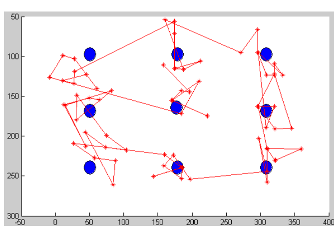


Figure 5. Gaze estimation result. The large dark (blue) circles are the 9 points showed on the screen. The stars and the lines denote the estimated gaze points and the saccade movement.

Finally, we test our algorithm under free head movement. The head moves to a new position and we still use the same set of eye parameters to estimate the gaze. The estimate gaze accuracy can achieve X accuracy = 22.42mm (2.18^o),Y accuracy =26.17mm (2.53^o).

For the experiment, we can see that our algorithm can give reasonable gaze estimation result ($< 3^o$), and the accuracy doesn't decrease much with free head movement. To maintain and improve eye tracking accuracy, we need improve our facial feature tracking accuracy.

5 Conclusion

In this paper, an gaze estimate algorithm based on the facial feature tracking is proposed. We set up the eye model and the equations based on the anatomical structure of the eye. By solving the equations, the 3D visual axis can be estimated, then the gaze point on the screen can be obtained by intersecting this visual axis with the screen. The preliminary experiment shows that this method can achieve the accuracy under 3 degree under free head movement.

References

[1] Lc technologies, inc. *http://www.eyegaze.com*, 2005.

[2] *http://www.a-s-l.com*, 2006.

[3] E. D. Guestrin and M. Eizenman. General theory of remote gaze estimation using the pupil center and corneal reflections. *IEEE Transactions on Biomedical Engineering*, 53:1124–1133, 2006.

[4] T. Ishikawa, S. Baker, I. Matthews, and T. Kanade. Passive driver gaze tracking with active appearance models. *Proceedings of the 11th World Congress on Intelligent Transportation Systems*, 2004.

[5] S. Kohlbecher, S. Bardinst, K. Bartl, E. Schneider, T. Poitschke, and M. Ablassmeier. Calibration-free eye tracking by reconstruction of the pupil ellipse in 3d space. *Symposium on Eye Tracking Research and Applications(ETRA)*, 2008.

[6] Y. Matsumoto and A. Zelinsky. An algorithm for real-time stereo vision implementation of head pose and gaze direction measurement. *Proceedings of the IEEE International Conference on Automatic Face and Gesture Recognition*, 2000.

[7] C. H. Morimoto and M. R. Mimica. Eye gaze tracking techniques for interactive applications. *Computer Vision and Image Understanding, Special Issue on Eye Detection and Tracking*, 98:4–24, 2005.

[8] C. W. Oyster. *The Human Eye: Structure and Function*. Sinauer Associate, Inc., 1999.

[9] Y. Tong, Y. Wang, Z. Zhu, and Q. Ji. Robust facial feature tracking under varying face pose and facial expression. *Pattern Recognition*, 2007.

[10] J.-G. Wang and E. Sung. Study on eye gaze estimation. *IEEE Transactions on Systems, Man and Cybernetics, PartB*, 32(3):332–350, 2002.

[11] H. Yamazoe, A. Utsumi, T. Yonezawa, and S. Abe. Remote gaze estimation with a single camera based on facial-feature tracking without special calibration actions. *Proceedings of the 2008 symposium on Eye tracking research and applications*, 2008.

[12] Z. Zhu and Q. Ji. Novel eye gaze tracking techniques under natural head movement. *to appear in IEEE Transactions on Biomedical Engineering*, 2007.