Gaze determination via images of irises

Jian-Gang Wang and Eric Sung

Division of Control and Instrumentation, School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798
eericsung@ntu.edu.sg

Abstract

In this paper, we present a new approach for accurately measuring the eye gaze of faces from images of irises. Eye iris contours are modeled as two planar circles and we estimate the ellipses of their projections onto a retinal plane. The unique solution of the orientations was obtained based on a geometric constraint, namely that the normal directions of the two iris contours should be consistently parallel to each other, irrespective of eyeball rotations and head movements. The robustness of this approach was verified by experiments on synthetic and real image data.
In order to obtain a high-resolution iris image, a zoom-in camera is used. A general approach that combines head pose determination with gaze estimation is proposed. A second camera is used for the determination of the head pose. The camera, which tracks the irises using the information from the estimated pose of human head, provides sufficient resolution to measure accurately the rotation of eyeball.

1 Introduction

There are two components to gaze orientation: the pose of human head and the orientation of the eye within their sockets. We have investigated these two aspects and will concentrate on the second part in this paper. However, 3D location of the human eyeball can also be determined besides the gaze orientation in our eye gaze determination algorithm. A method to determine the gaze of a person by using the person's irises' contours is demonstrated. The prior knowledge of the eye model has been utilized in our method, namely that left and right iris boundaries are reasonably circular and that the normals of the two iris planes are parallel irrespective of eyeball rotations and head movement. This "normal constraint" means we assume the person is focused at infinity. We found that this assumption is applicable when the observed target is at a given distance of more than half meter away. It is hence applicable for our human-machine interaction applications. A gaze camera is mounted on the top of the monitor, the distance between the operator and the camera is about one meter. But in fact, this constraint can be slightly relaxed. The two iris planes need not be exactly parallel. If the person is viewing an object not too far away, the two eyes will verge at an angle. If the iris planes are close to being parallel, this condition is sufficient for disambiguation of the two possible solutions. Once disambiguated, the actual gaze direction of the two irises can be computed. We then take the average of these two gaze directions to define the gaze of the person. Differing from the existing eye models, the iris contours are modeled as two circles respectively and their projections are so fitted as ellipses rather than as circles. In most existing approaches, the iris contours on the image plane are assumed to be circles, so the circular geometry is utilized and iris outer

boundaries (limbus) are detected using circle edge operator. The gaze, defined as the vector starting from center of the eyeball and ending at center of iris contour, should be consistent for two eyes irrespective of eyeball rotations and head movements. And the gaze can be determined using its circle-ellipse correspondent. Furthermore, if we know the radius of the iris, then we can calculate the relative center of the iris. Now, the iris is partially occluded by the upper and lower eyelids so it will be difficult to fit ellipse with high precision. A zoom-in camera is used for iris tracking in order to improve the resolution. A zoom-in camera, with 45 *mm* lens, is used for iris tracking in order to improve the resolution. A subject size of 12 $cm \times 9$ *cm* fill up the whole image when the distance between the subject and the camera is one meter. The problem of having possible out-of-field views can be settled by guiding the gaze camera using the head pose information obtained from the pose camera. The robustness of our algorithm was verified with the experimental results on synthetic and real image data.

Facial features such as the corners of the eye and mouth, the tip of the nose, ratios of distances between feature points can be utilized to determine the pose of human head [11] and gaze [5]. More compact features such as eye contours has been utilized to improve the robustness. For instance, affine deformations of the left and right eye contours are used to estimate the head displacement in [12]. A technique was presented to infer the computer screen location currently observed by the user from the measurement of image of the eye pupil displacements [2]. In [7, 13], eyes are tracked for human-computer interaction by embedding an infra-red camera, provided with a transmitter and receiver, which evaluates the current pupil position by exploiting the differences in infrared reflection between the pupil and the surrounding iris and sclera. However, the system's calibration is sensitive to movement of the subject's head, so the subject must either remain perfectly still, or wear cumbersome headgear to maintain a constant separation between the sensor and eye. A. Gee et al [5] did not consider the eyeball rotation and estimate gaze direction by rotating the facial normal by about 10^0 around the eye-line. K. Talmi et al [3] introduced a dynamic model which distinguishes between the changes of gaze vectors due to head movement and due to the varied gaze direction and compensates the head movement.

From the observed perspective projection of a circle having known radius, it is possible to infer analytically the plane on which the circle lies as well as where the center of the circle lies [6]. In this paper, we develop it further for human gaze determination in conjunction with two iris contours. The unique solution of gaze can be determined based on the "normal constraint".

2 Positioning

We determine the human gaze by using the curved features, namely the contours of the iris. They are assumed to be circles in the real world. The perspective of a circle on image plane is an ellipse. The equation of the ellipse can be obtained from least square fitting of quadratic curve using the detected iris edges. With two irises, it will be shown that the two respective equations lead to a unique solution of the eye gaze. A monocular camera has been used to capture these two irises (circles).

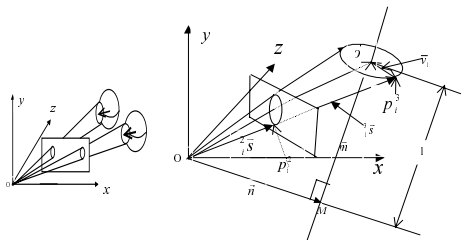
2.1 Two-circles algorithm

Given the image of a circle in 3D space and the corresponding ellipse in the image, its pose relative to the camera frame can be analytically solved. Researchers have

extensively exploited methods for recovering pose from circles, e.g. [14, 4, 6]. In this paper, the unique solution can be obtained from two ambiguous solutions by using the constraint, namely that two iris contours should have the same normal direction (Fig. 1(a)). We will discuss the constraint in following section (on eye model) in detail.

The spatial relation between a circle and its perspective on the image plane is shown in Fig. 1(b). Here we assume (x_i, y_i) is the image coordinates of point p_i^2, (X_i, Y_i, Z_i) corresponds to the 3D coordinates of point p_i^3 under the camera coordinate system. The camera coordinate system is *O-XYZ*, the circle center is O' and the radius is *r*. p_i' represents any point that lies on the circle, p_i is the projection of point p_i' on the image plane.

In order to infer the equation of ellipse from perspective geometry, we define some variables as follows



(a) (b)

Fig. 1 (a) Projection of two irises (b) Projection of a circle and the rotation axis of the circle

The circle can be considered as being generated by a point rotating about rotation axis $\vec{m}$ while preserving radius *r*.

So the equation of the circle will be:

$$[{}_i^3\vec{s} - (\vec{n} + l\vec{m})]^T [{}_i^3\vec{s} - (\vec{n} + l\vec{m})] = r^2 \tag{1}$$

Using normalized camera model, we have

$$[x \quad y \quad z]^T = {}_i^2\vec{s} = \frac{1}{Z}{}_i^3\vec{s} = \frac{1}{Z}[X \quad Y \quad Z]^T \tag{2}$$

So we obtain the expression of the projection of the circle on the image plane [10, 8], i.e. ellipse, it is:

$$({}_i^2\vec{s}^T)(l^2 I - l(\vec{n}\vec{m}^T + \vec{m}\vec{n}^T) + c_1\vec{m}\vec{m}^T)({}_i^2\vec{s}) = 0 \tag{3}$$

where $c_1 = \vec{n}^T\vec{n} - l^2 - r^2$ and I represents the unit matrix.

Define

$$\pi_1 = (l^2 I - l(\vec{n}\vec{m}^T + \vec{m}\vec{n}^T) + c_1\vec{m}\vec{m}^T) \tag{4}$$

If the eigenvalues of π_1 are λ_1, λ_2 and λ_3, where $\lambda_1>0$, $\lambda_2>0$, $\lambda_1<=\lambda_2$, $\lambda_3<0$, then

$$\vec{n} = \cos\theta \cdot \vec{n}_2 - \sin\theta \cdot \vec{n}_3 \tag{5}$$

$$\vec{m} = \sin\theta \cdot \vec{n}_2 + \cos\theta \cdot \vec{n}_3 \tag{6}$$

where $\vec{n}_1$ is an eigenvector corresponding to λ_1. The other two eigenvectors are $\vec{n}_2$ and $\vec{n}_3$ associated with the eigenvalues λ_2 and λ_3 respectively; θ is angle between $\vec{n}_2$ and $\vec{n}$. This formula is in accordance with the conclusion given in [1].

Note that the signs of the eigenvectors $\vec{n}_1$, $\vec{n}_2$ and $\vec{n}_3$ are arbitrary, so four solutions of $\vec{m}$ exist in equation (6) if $\lambda_1 \neq \lambda_2$. For the object to be in front of the camera, the *Z*-axis direction of $\vec{m}$ is always positive, so $\vec{m}$ and $\vec{n}$ have two different groups of solution. Fortunately, the normal lines of the two iris contours should be consistently parallel to

each other irrespective of eyeball rotations and head movements, so we can remove the ambiguous solutions and find the unique solution of rotating axis $\bar{m}$. The centers of the two iris contours i.e. $\bar{O}\bar{O}'$ vector can be determined then.

2.2 Degenerate cases

In our human-machine applications, the "two-circles" algorithm will be degenerate when the following condition is satisfied: the two iris contours are symmetrical about the *Y-Z* plane of the camera, where *Y*- and *Z*- axis is vertical and optical axis of the camera respectively.

We obtain two normal solutions for each circle from its projection respectively. The two solutions of the first circle are the same as the two solutions of the second circle. So we cannot obtain the unique solution based on our normal coincidence constraint in this case. Actually, this condition corresponds to the case that user is facing front, whereby the two irises are symmetrical about the optical axis. Thus it is impossible to distinguish from the ellipses whether the person is looking upwards or downwards. It is ill-posed for this case. We can see this in Fig. 4. Fortunately, this degenerate case can be prevented in our application by simply putting the camera slightly skewed to the face.

3 Eye model and experimental results

We have tested our gaze determination method using a Pentium450 PC, 128MB RAM. The algorithm runs on Image-Pro Plus image processing software with Matrox Meteor-II imaging board. The focal length of the camera can be adjusted from 12.5 to 75 *mm*. The distance between the camera and the operator is about half a meter.

3.1 Simulations on exact data

In order to understand the accuracy of the proposed algorithm, a model of the human eye is used in our simulations.

3.1.1 Eye model

A simple eye model is defined (Fig. 2). The eyeball is assumed to be a sphere with a radius *R*. The iris is located at the front of eyeball and modeled as a circular ring of radius *r* (see Fig. 2(a)). The distance from the center of the eyeball to the iris plane is *d*. The relation among *R*, *r* and *d* is (Fig. 2(c)):

$$R^2 = r^2 + d^2 \tag{7}$$

The optical axis of the eye is the line passing through the center of the eyeball and the center of the iris and it is defined as the ***gaze direction***. By changing the gaze direction, the eyeball rotates around its center (Fig 2(d)).

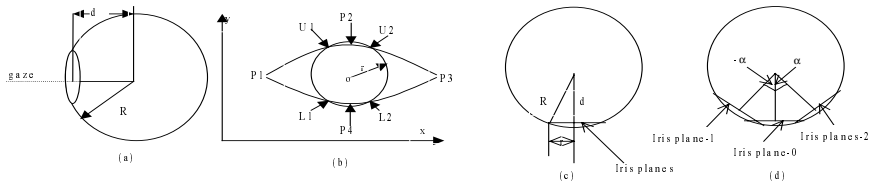


Fig 2 The eye model (a) eyeball, iris and gaze (b) coordinate system, top and lower eyelids (c) radius of eyeball, iris, distance between center of the eyeball and iris (d) rotation of iris about center of eyeball

The upper and lower eyelids are modeled as two parabolas (Fig. 2 (b)). The equation of the eyelids can be obtained by using the coordinates of four points: $P_1(x_1, y_1)$, $P_2(x_2, y_2)$, $P_3(x_3, y_3)$ and $P_4(x_4, y_4)$. In general, upper and lower eyelids in real face images occlude parts of the iris contours and only the unoccluded iris edges can be used to fit the iris contour on the image plane. Consistently, the synthetic iris contour edges that lie between the upper and lower eyelids are located and used to fit the elliptical contour in our simulations. For instance, curves U_1L_1 and U_2L_2 shown in Fig. 2(b) are the fitting edges we want. These fitting edges can be located by using the equations of iris and eyelids.

3.1.2 Experimental results on synthetic data

The intrinsic parameters of the camera and 3D centers of the two irises are set initially. The coordinate system of the camera is defined as above, i.e., the image plane is defined as the *X*-*Y* plane and the *Z*-axis is along the optical axis of the camera and pointing toward the frontal object. The initial plane is where two irises are positioned parallel to the *X*-*Y* plane of the camera. The projective image can be obtained using the perspective transform matrix of the camera. The two projective ellipses can be estimated by least-squares. The 3D location (six degrees of freedom, three for translation and three for rotation) can be calculated using "two-circles" algorithm. The errors between the calculated results and corresponding original synthetic data can be obtained. These sets of errors are obtained under different poses. An example is given as follows.

The size of the image is 640×480. The intrinsic parameters of the camera are described as follows.

$$u_0=320, \quad v_0=240, \quad f_x=2500, \quad f_y=2500 \tag{8}$$

where (u_0, v_0) are coordinates of the image-center point (principle point), f_x and f_y are the scale factors of the camera along the x- and y-axis respectively. The setting of f_x and f_y here imply that there is a constraint in our application. The camera (zoom-in) requires a larger focal length in order for the eyes to appear big enough on the image. 40-45 *mm* focal length is used in our real image experiments that keep a distance about 60-80 *cm* between human face and camera. The 3D coordinate (represented in the coordinate system of camera) of the centers of the two circles (named as circle 1 and circle-2) are:

$$(x_{1c}, y_{1c}, z_{1c})=(-3.25, 0, 60)\ (cm)\ , \qquad (x_{2c}, y_{2c}, z_{2c}) = (3.25, 0\ 60)\ (cm) \tag{9}$$

i.e. the initial contours of two irises (circles) lie parallel to the image plane. The radii of the two irises are set as:

$$r_1=0.65\ cm, \qquad r_2=0.65\ cm \tag{10}$$

The 3D position of any point lie in the circle-1 can be obtained using following parametric equations:

$$x = x_{1c}+r_1\cos(\theta), \quad y = y_{1c}+r_1\sin(\theta), \quad z = z_{1c} \tag{11}$$

where $0 \le \theta \le 360$. There is a similar formula for circle-2.

The ratio of the radius of a person's eyeball and the radius of his/her iris is assumed to be a generic constant. It is assumed to be 2 here. So the radius of the eyeball is: R=1.3 *cm*. The distance from eyeball center to the iris plane can be calculated using equation (7), we arrive at:

$$d = 1.13\ cm \tag{12}$$

The distance between two extreme corners of the parabola is assumed: $P_1P_3 = 3.5\ cm$. The top and bottom points of parabola are symmetry about initial iris center, the distance: $P_2O = P_4O = 0.6\ cm$.

We consider the possible poses of eye irises as follows. Let the head look toward the camera and the eyeball are rotated around center of the eyeball about *Y*-axis (vertical)

and *X*-axis (horizontal) of camera coordinate system, Consequently two iris contour will rotate follow the eyeball. We project the rotated eyeball and two irises onto the image plane and re-estimate the poses and compute the error with the actual poses. The errors here are measured using the angle between the actual and estimated gaze directions. The iris contours, being two ellipses, are fitted and their 3D locations are recalculated using "two-circles" algorithm. The points that lie within the upper and lower parabolas are found and used to fit the two irises elliptical contour respectively.

Rotating the eyeball about vertical axis from -40^0 to 40^0 step 1^0 (azimuth) and about horizontal axis from -10^0 to 10^0 step 1^0 (elevation) forms a set of synthetic images.

The image when the rotation angle around *Y*-axis -30^0 and *X*-axis -15^0 is shown in Fig. 3 (a). The results of normal vectors (gaze) and centers of circle-1 and circle-2 are listed in Table 1.

	Solution-1	Solution-2
Circle-1		
Gaze	(-0.353, -0.211, 0.912)	(0.234, 0.198, 0.952)
Center	(-3.737,-0.392, 59.769)	(-3.7406,-0.395, 59.460)
Circle-2		
Gaze	(-0.356, -0.248, 0.901)	(0.435, 0.236, 0.869)
Center	(2.688,-0.390,59.460)	(2.683, -0.393, 59.461)

Table 1 The results of the two circles' normal vectors

The solution-1 of circle-1 (gaze and center) and the solution-1 of circle-2 (gaze and center) are accepted based on the normal constraint. The actual gaze of the two irises are equal exactly and can be obtained from synthetic data equals:

(-0.342, -0.243, 0.908)

So the error of the estimated gaze equals 0.893^0.

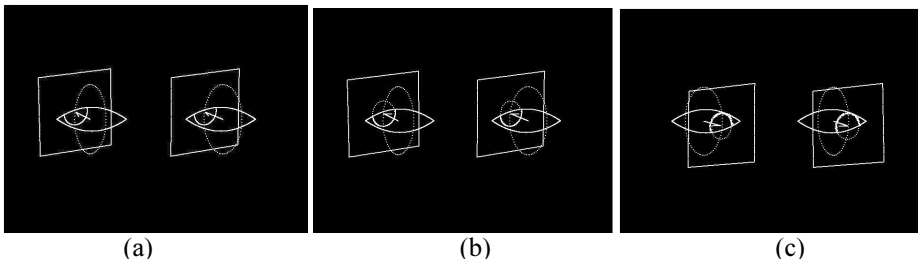
The centers of two circles are calculated and compared with the actual values, listed in Table 2.

	Calculated (cm)	Real (cm)	Error (cm)
Circle-1	(-3.737,-0.392,59.769)	(-3.787,-0.382, 9.855)	0.100
Circle-2	(2.688,-0.390,59.460)	(2.713,-0.382, 59.855)	0.396

Table 2 The errors of the two circles' center

The results are re-projected to the original image, see Fig 3(b).

The errors between calculated results and actual synthetic values of different poses are shown as Fig. 4(a).



(a) (b) (c)

Fig. 3 Results of the synthetic eye images (a) Rotating iris -30^0 about *Y*-axis and -15^0 about *X*-axis, the actual gaze vectors and ellipse fitting points are shown. (b) Overlaying the results onto the original image shown in (a). The results include the least-squares fitted iris contour (dotted ellipse) and the estimated gaze. The gaze direction is represented as an arrow that starting from the eyeball center and ending at the iris center. (c) Overlaying the result onto the original image where the rotation angle of the iris is 30^0 on *Y*-axis and 10^0 on *X*-axis and the positions of the fitting points are perturbed by Gaussian noise. The magnitude, *R_noise,* of the noise is one pixel.

3.2 Robustness to geometrical disturbances

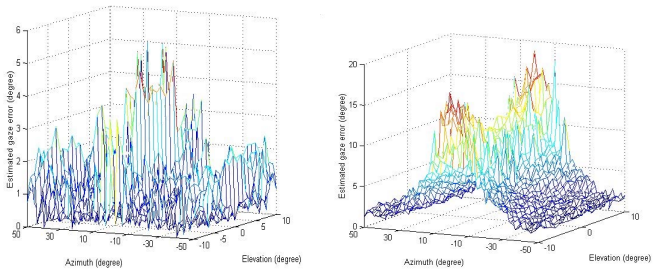
Using the same virtual camera and two circles, we test for the errors caused by geometrical disturbances. The imaged feature locations were corrupted by zero-mean Gaussian noise with a standard deviation of one pixel.

Consider we produce a *jth* point on circle as section 3.1. Then the image coordinate (x_j, y_j) of *jth* point will be disturbed as:

$$x\_j = x\_j + R\_noise \times rand\_x() \quad (13)$$

$$y\_j = y\_j + R\_noise \times rand\_y() \quad (14)$$

where *rand_x*() and *rand_y*() are two orthogonal random Gaussian noise generators (zero mean and unity variance). *R_noise* represents the magnitude of the noise (pixels).



(a) (b)

Fig. 4 Error of the gaze for different poses (a) no noise (b) perturbed by Standard Gaussian noise (zero mean, standard deviation equals 1)

We calculate the errors of the angle between estimated gaze and actual gaze. The errors of the centers of two circles have also been calculated respectively. We do this kind of simulation 100 times for every gaze and the mean error of 100 results is recorded to be the error of the gaze. One example of the results of geometrical disturbances applied in the simulation is shown in Fig. 3(c).

The experiments show that the algorithm we developed is robust enough. When we disturb 2D position of fitting points with a standard Gaussian noise (zero mean, standard variance equals 1), the errors of the gaze of different poses are less than 5 degree (Fig. 4 (b)), the error of the centers of two circles are both less than 1.5 *cm*.

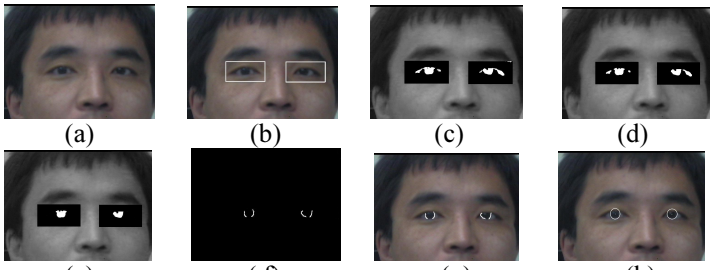
3.3 Experiments on real scene

The contour of an iris is modeled as a circle so the projection of the iris is modeled as an ellipse whose center locates the pupil. It is observed that some eye regions have relatively high contrast, such as boundary points between eye white and eyeball. This can be employed to guide the iris detection process. An effective iris detection method is proposed in this paper.

The ellipse contour of the irises can be fitted from the iris edge results by using the method proposed by Bookstein [9]. It is stable to detect ellipse even under the condition when the iris moves to one extreme side of the eye.

For the settings of f_x and f_y in the section on simulations, it imply that there are constraints for our application, i.e. the camera must use larger focal length in order for the eyes to appear bigger on the image. COSMICAR TV ZOOM lens 12.5-75 *mm* is used, 40-45 *mm* focal length is set in our real image experiments that keeps the eyes big enough from a distance of about 60-90 *cm* between the human face and the monitor. One example of iris contour detection and gaze determination result is shown in Fig. 5.

There are 62 and 60 edge points for fitting left and right ellipses respectively. The two solutions of the normal vector and the center of two ellipses are calculated and listed in Table 3.



(a) Original image. (b) The eye-regions located by the pose determination method. (c) Segmentation results of the eye-regions. (d) Separating the irises using morphological open (2×2 structuring element) operation. (e) The largest segmented regions within the eye-region are deemed to be irises. (f) The longest vertical edges (3×3 vertical operator) are deemed to be irises edges. (g) Overlaying the s onto the original image. (h) Overlaying the least-squares fitted ellipse onto the original image.

Fig. 5 iris contour detection and gaze determination

	Solution-1	Solution-2
Circle-1		
Gaze: NR_1	(-0.042, -0.328, 0.944)	(0.0402, -0.365, 0.887)
Center:CR_1	(-2.435, 1.161, 65.803)	(-2.435, 1.111, 65.803)
Circle-2		
Gaze:NL_2	(-0.079, -0.329, 0.941)	(0.106, 0.269, 0.957)
Center:CL_2	(4.853, 1.210, 66.967)	(4.053,1.206, 66.969)

Table 3 Results of normal vectors and center points of two irises

Fig. 6 Some examples of eye gaze determination results (overlaying the iris edges, fitted ellipses and gaze direction onto the original images, gaze direction are represented by an arrow that starting from center of the eyeball and ending at the center of the iris contour)

The angle between *NR_1* and *NL_1*, *NR_1* and *NL_2*, *NR_2* and *NL_1*, *NR_2* and *NL_2* are 2.08, 35.84, 17.93 and 41.01 respectively. From the normal constraint, solution 1 of

right circle and the solution 1 of left circle should be consistent, i.e. the unique of solution can be obtained.
The normal should be the average of the two gaze directions:

NR_1: (-0.042, -0.328, 0.944) and NL_1: (-0.079, -0.329, 0.941)

The center of the right and left ellipse should be:

(-2.435, 1.161, 65.803) and (4.853, 1.210, 66.967)

Hence, the distance between the centers of the two circles equals to 7.38 *cm.* The actual distance between the centers of the two circles is manually measured to be about 7 *cm.* We can see that the relative error of the distance between the centers of two circles is less than *6%* within the distance range of 65 *cm.*

Some gaze determination results are shown in Fig 6, where we have assumed the radius of the irises are 0.65 *cm* each, and the radius of eyeball is 1.3 *cm*. The gaze is represented by an arrow starting from center of eyeball and ending at the center of iris contour. This is calculated using the center of iris, the radius of eyeball and the gaze result. In order to obtain high-resolution iris images, we use a camera with 45 *mm* lens. The drawback of this approach is the problem caused by the narrow field of view causing the eye irises to be easily lost from image. A new approach that we use here is to use two cameras instead: one for pose determination (pose camera) and the other for gaze determination using "two-circles" method (gaze camera). Our method of pose determination has been presented elsewhere and will not be dealt with in this paper. The gaze camera, which is mounted on a pan-tilt device, can track the viewer's two eyes using the 3D information about the two eye positions that obtained by the pose camera. The left picture of Fig. 7 is the result of pose determination from an image using the pose camera (lens: 12.5 *mm*); the right picture of Fig. 7 is the corresponding gaze determination result. Because only the extrinsic parameters of the gaze camera need to be adjusted, it is easy to link the pose and gaze results.

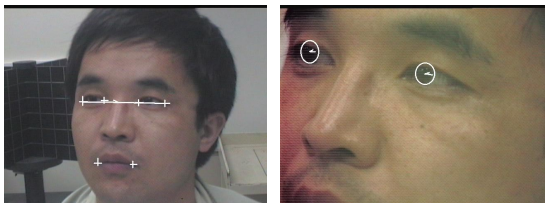


Fig. 7 Pose (left) and the corresponding gaze (right) determination results

4 Conclusion

In this paper, we present an original approach that obtains gaze based on iris information. It effectively combines iris detection and quadratic-based pose determination method. We obtain the unique solution based on the normal constraint (that the normals of two iris planes are invariant for eyeball rotations and head pose). Degenerate cases under the normal consistency assumption occurred when gaze orientation nearly parallel to the view axis of the camera. This degeneracy has been prevented in our application (human-machine interaction) by putting the camera on the top of the monitor and slightly skewed to the face. Simulations on a virtual camera model have shown that the algorithm is robust under geometric disturbances. When we disturb the 2D position of iris edge points using a standard Gaussian noise (zero mean and standard deviation is one pixel), the error of the rotation angle is less than 5^0, the

error of centers of two circles is less than 1.5 *cm* within the range 60 *cm*. We are working toward to improve the precision further by using image of one eye rather than two irises.

In order to tolerate inaccuracies due to low resolution of the iris images, a camera zoom lens is used. A general approach that uses two cameras is developed in this paper: one camera for pose determination of human head and one for gaze determination. The gaze camera can track irises by utilizing 3D information of human head obtained from the pose camera.

References

[1] Kenichi Kanatani, Geometric Computation for Machine Vision, Chapter 8: Analysis of Conics, Clarendon Press, 1993.

[2] Colombo C., Bimbo A. D., Interacting through eyes, *Robotics and Autonomous Systems* Vol. 19, 1997, pp.359-368.

[3] Kay Talmi, Jin Liu, Eye and gaze tracking for visually controlled interactive stereoscopic displays, *Signal Processing: Image communication*, Vol. 14, 1999, pp.799-810.

[4] Shiu Y.C. and Ahmad S., 3D location of circular and spherical features by monocular model based vision, *IEEE International Conference on system, Man and cybernetics*, pp.576-581 (1989).

[5] Gee A. and Cipolla R., Determining the gaze of faces in images, *Image and Vision Computing,* Vol. 12, No. 10, December 1994, pp.639-647.

[6] Haralick R. M., Shapiro L. G., Computer and Robot Vision, Chapter 13: Perspective projection geometry, Addison-Wesley Publishing Company.

[7] Hutchinson T.E., White K. P., JR, Martin W. N., Reichert K.C. and Frey L. A., Human-computer interaction using eye-gaze input*, IEEE Transaction on System, Man and Cybernetics*, Vol. 19, No. 6, November/December 1989, pp. 1527-1533.

[8] J.G.Wang and Eric Sung, Measuring pose of human head for human-machine interaction*, Technical Report*, School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore, April, 1999.

[9] F.L.Bookstein, Fitting conic sections to scattered data, *Computer Graphics and Image Processing*, 9, 1979, pp.56-71.

[10] H. S. Sawhney, J. Oliensis and A.R. Hanson, Description and reconstruction from image trajectories of rotational motion, *Proceedings of the third International Conference on Computer Vision*, pp.494-498 (1990).

[11] Stiefelhagen R., Yang J., A model-based gaze tracking system*, International Journal of Artificial Intelligence Tools*, Vol. 6, No.2, 1997, pp.193-209.

[12] Carlo Colombo, Alberto Del Bimbo, Real-time head tracking from the deformation of eye contours using a piecewise affine camera, *Pattern Recognition Letters*, Vol. 20, 1999, pp.721-730.

[13] S. Pastoor, J. Liu and S. Renault, An experimental multimedia system allowing 3-D visualization and eye-controlled interaction without user-worn devices, *IEEE Trans. on Multimedia*, Vol. 1, No. 1, March 1999, pp.41-52.

[14] Forsyth D., Mundy J. L., Zisserman A. and C. Coelho, and A. Heller and C. Rothwell, Invariant descriptors for 3-D object recognition and pose, *IEEE Transaction on PAMI*, Vol. 13, No. 10, October 1991, pp.971-991.