FreeGaze: A gaze tracking system for everyday gaze interaction

Conference Paper · January 2002

DOI: 10.1145/507072.507098 · Source: DBLP

3 authors, including:

Naoki Mukawa
Tokyo Denki University
107 PUBLICATIONS **1,574** CITATIONS
SEE PROFILE

Atsushi Yoshikawa
Tokyo Institute of Technology
81 PUBLICATIONS **516** CITATIONS
SEE PROFILE

FreeGaze: A Gaze Tracking System for Everyday Gaze Interaction

Takehiko Ohno* Naoki Mukawa Atsushi Yoshikawa†

NTT Communication Science Laboratories, NTT Corporation

Abstract

In this paper we introduce a novel gaze tracking system called *FreeGaze*, which is designed for the use of everyday gaze interaction. Among various possible applications of gaze tracking system, Human-Computer Interaction (HCI) is one of the most promising fields. However, existing systems require complicated and burdensome calibration and are not robust to the measurement variations. To solve these problems, we introduce a geometric eyeball model and sophisticated image processing. Unlike existing systems, our system needs only two points for each individual calibration. When the personalization finishes, our system needs no more calibration before each measurement session. Evaluation tests show that the system is accurate and applicable to everyday use for the applications.

CR Categories: H.5.2 [Information Interfaces and Presentation]: User Interfrace——Interaction Styles, Gaze Tracking System,The Eyeball Model,Personal Calibration,Gaze Interaction

Keywords: Gaze tracing system, FreeGaze, the eyeball model, gaze interaction

1 Introduction

We humans acquire much information from our eyes. Gaze reflects our attention, intention and desire. Thus, detection of the gaze direction makes possible to extract such information that is valuable in Human-Computer Interaction (HCI). Computers integrated with gaze tracking function must potentially provide an intuitive and effective interactive system.

Many interactive systems integrated with a gaze tracking system have been proposed. One features task selection by eye. Without using a mouse or keyboard, one can select an icon or a menu item by looking at it for a while [Hansen et al. 1995; Jacob 1990; Jacob et al. 1993]. Selection does not necessarily require long dwell time. For example, separating the selection area from the menu enables very fast selection by eye [Ohno 1998]. Integration of a gaze tracking system with other input devices also accelerates selection [Zhai et al. 1999]. Another way of using the eye is to extract the user's attention. Starker et al. proposed a method to detect the user's attention from the user's eye movement pattern while the user is looking at objects on the screen [Starker and Bolt 1990]. In their system, a narration about the object of interest is given.

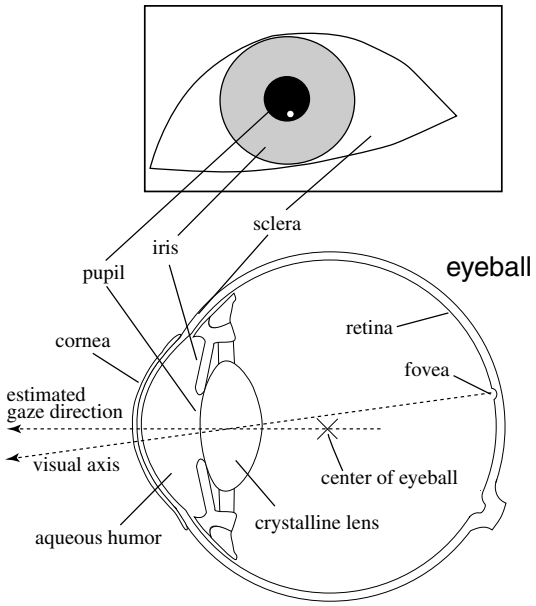


Figure 1: The structure of the eye.

However, the development of an interactive system requires the accurate, useful, comfortable and reliable real-time gaze tracking system. Put in another way, if it is difficult to track gaze direction, gaze interaction is just pie in the sky.

One of the most important issues in gaze tracking systems is simplifying the personal calibration procedures. Personal calibration is a system configuration process carried out at the beginning of use. In general, personal calibration requires the user to look at 5 to 20 markers on the screen. It is a time consuming and a burdensome task for users. Gazing at the marker unwaveringly is not easy. It is not required for other input devices like the keyboard and mouse.

Most of existing systems need some kind of personal calibration at the beginning of the measurement. The reasons for this are:

- Individual differences of eyeball size. Among adult persons, there is about 10 % individual difference in the radius of eyeball.

- Difficulty in measuring the position of the fovea. The fovea is the highest resolution area on retina, and the human visual axis can be defined as a vector from fovea to the center of the crystalline lens (Fig. 1). On the other hand, existing gaze tracking techniques measure the estimated gaze direction from the center of cornea curvature and the center of the pupil, instead of the real visual axis. The estimated direction is then corrected by the personal calibration.

*Communication Scene Analysis Research Group, Media Information Science Research Laboratory, 3-1 Morinosato Wakamiya, Atsugi, Kanagawa, 243-0198, Japan. takehiko@brl.ntt.co.jp

†Current: NTT DATA Research and Development Headquarters

Proceedings of the symposium on ETRA 2002: eye tracking research & applications symposium, pp.125-132, 2002.

Figure 2: The gaze tracking system *FreeGaze*.

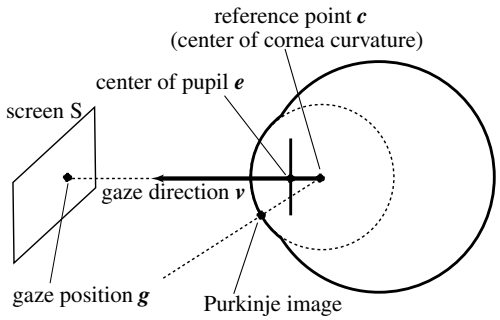


Figure 3: Gaze direction derived from the pupil and the Purkinje image.

In this paper, we propose a gaze tracking system called *FreeGaze*. Unlike existing systems, the novel system drastically reduces the burden of personal calibration. The system requires the user just look at the two marks on the display in personal calibration procedure. Once the personal calibration is completed, no further calibration is required for the person and the user can use the system as soon as s/he sits in front of the computer.

To accomplish this advantage, we introduce a eyeball model. In general, the derived gaze direction is not only different from the visual axis, but also different from the estimated gaze direction described in Fig. 1. Therefore, personal calibration becomes complicated and may cause estimation error. The eyeball model compensates factors necessary for calculating the correct gaze direction, so that, the minimal personal calibration is realized.

Moreover, FreeGaze has the robust and sub-pixel order image detection of the pupil and the Purkinje image. Thus, even if the lighting environment in the room changes, it is possible to detect the pupil and the Purkinje image without any parameter adjustment in image processing.

In the rest of the paper, related works of gaze detection are introduced first. Then, our gaze detection algorithm is described in detail, and the implemented system is presented. Next, the results of an evaluation are presented and we discuss the benefits and remaining problem of our system. We also introduce an example of application of the system, human computer interaction by gaze. Finally, future work is mentioned.

2 Related Works

Several types of gaze tracking systems have been developed. The most popular type uses infrared light and camera to detect the eye. Especially, the use of the pupil and the Purkinje image is an accurate gaze tracking method and it is used widely including many commercial products. Research efforts are focused on accurate, robust and easiness of calibration.

To improve the robustness, Morimoto et al.'s system arranges light-emitting diode (LED) arrays at two different places, around the camera lens and far from the lens, to detect the bright and the dark pupil images [Morimoto and Flickner 2000]. From the difference between two images, it is possible to detect the pupil robustly. The idea of combining bright and dark pupil images comes from Ebisawa's research [Ebisawa 1998].

Shih et al. proposed a calibration-free gaze tracking technique [Shih et al. 2000] in which multiple cameras and multiple point light sources are used to estimate gaze direction. Correction of the refraction on the surface of cornea is also done. They performed computer simulations to confirm their method. In their algorithm, the difference between the visual axis and the estimated gaze direction is not considered; therefore, some measurement error still remains.

The limbus, the boundary of the sclera and the iris, is also used to detect gaze direction. While sclera is white and iris is darker, the limbus can be easily detected by the camera. Matsumoto et al. used a real-time stereo vision technique to measure head position and gaze direction [Matsumoto and Zelinsky 2000]. Their system uses the limbus to detect gaze direction. The eye size in the captured image size is small; therefore, the accuracy is not high. However, gaze tracking is robust; for example, system can be used while driving a car.

3 Detection of Gaze Position

The developed gaze tracking system, FreeGaze, detects gaze position by the two processes. First, the pupil and the Purkinje image are detected from the captured image. Then, the gaze position is computed from these two images by using the eyeball model. Finally, due to the personal calibration, accurate gaze direction is estimated.

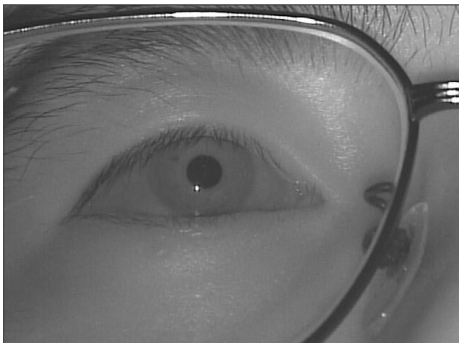
Fig. 3 shows the relation between the images and gaze direction. Hereafter, we use $\boldsymbol{c}$ as the position vector of the reference point. We define $\boldsymbol{c}$ as the position vector of the reference point at the center of cornea curvature, $\boldsymbol{v}$ as the vector of the gaze direction in 3-D coordinate system, and $\boldsymbol{g}$ as the gaze position on the screen. The reference point $\boldsymbol{c}$ is derived from the Purkinje image position. Gaze direction $\boldsymbol{v}$ is determined by the center of the pupil and the center of cornea curvature. Therefore, given the center of pupil and the position of Purkinje image, it is possible to calculate the gaze direction $\boldsymbol{v}$ and the reference point $\boldsymbol{c}$, and consequently gaze position $\boldsymbol{g}$ on the screen as well.

3.1 Detection of Pupil and Purkinje Image

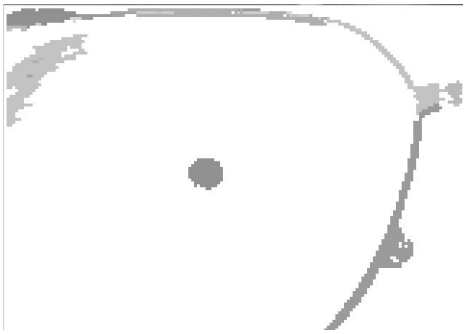
In order to determine the position of the pupil and the Purkinje image, it is necessary to detect these two areas in the grabbed image.

Step 1. Eye Image Grabbing

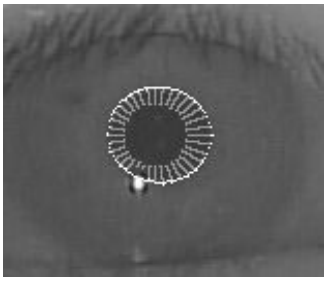
First, an eye image taken by an infrared camera is captured by a frame grabber with a resolution of 640×480 [Fig.4 (a)]. To reduce noise and processing cost, a quarter size image (resolution is 160×120) is generated from the original image. This image is used in the following Step 2 and Step 3.



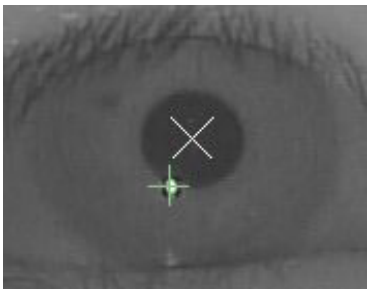
(a). Grabbed eye image.



(b).Area segmentation.



(c). Edge detection. Searching from inside of pupil to the circumference of pupil is done.



(d). Result of pupil and Purkinje image detection.

Figure 4: Detection of the pupil and the Purkinje image.

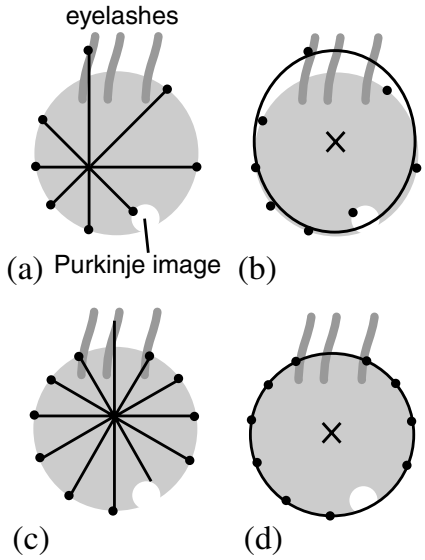


Figure 5: Double ellipse fitting, which consists of four steps: (a) rough edge detection, (b) temporal fitting of ellipse, (c) detailed edge detection, and (d) final fitting of ellipse.

Step 2. Image Segmentation
The eye image is segmented into small regions [Fig.4 (2)]. Each region is generated by connecting the pixels having similar pixel values.

Step 3. Pupil Image Detection
For each region, its boundary shape is calculated to detect pupil candidates. The neighboring pixel values are used to distinguish between pupil and other circular objects like spectacles' frame and nostrils.

Steps 2 and 3 are repeated from the dark to the bright areas until the pupil is detected. If a pupil is not detected after the steps, the system judges that there is no pupil in the image.

Step 4. Pupil Edge Refinement
If candidate pupil areas are detected, the edge refinement process starts for the original resolution image. Sometimes eyelashes, lower eyelid, and also Purkinje image overlap on the pupil area. These noises are eliminated by fitting ellipses on the edges. The fitting process is repeated twice, which we call it *double ellipse fitting*.

Step 4.1. First, the edge of the pupil is detected roughly [Fig. 5 (a)]. The threshold of the edge detection is calculated from the average brightness of the candidate area. Pixels are scanned radially from the center of candidate pupil, and those brighter than the threshold are added to the edge set.

Step 4.2. Ellipse fitting is then done for the edge set, and the provisional center of the pupil is calculated [Fig.5 (b)].

Step 4.3. Next, the fine edge detection is done using the provisional center. The edge is estimated by verifying if each pixel in the edge set is on the ellipse edge or not, and pixels far from the edge are disregarded. This process eliminates the overlapped noises.

Step 4.4. Again, the ellipse fitting is performed. The derived ellipse is described as

$$x^2 + a_1xy + a_2y^2 + a_3x + a_4y + a_5 = 0, \qquad (1)$$

where $a_1, ..., a_5$ are ellipse parameters determined by the fitting. The center of the ellipse (e_{x_0}, e_{y_0}) is also derived from Eq. (1):

$$(e_{x_0}, e_{y_0}) = (\frac{2a_3a_4 - a_2a_5}{{a_2}^2 - 4a_3}, \frac{2a_5 - a_2a_4}{{a_2}^2 - 4a_3}). \qquad (2)$$

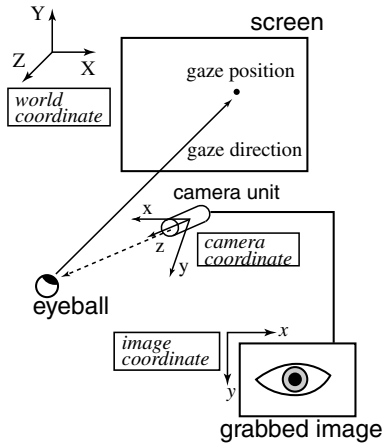


Figure 6: Three-coordinate system in gaze position computation

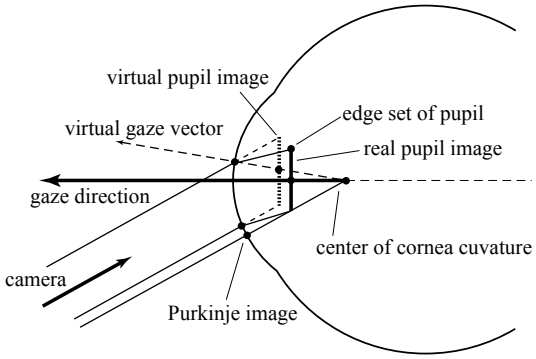


Figure 7: The eyeball model

Step 5. Purkinje Image Detection
The Purkinje image is detected by searching near the pupil. When the Purkinje image is detected, its centroid is calculated. ∎

The accuracy and the resolution of estimated gaze direction highly depends on the pupil and the Purkinje image detection. The double ellipse fitting method contributes to improve the accuracy of pupil detection. For detecting the pupil and the Purkinje image, no specific parameters of threshold are used (it appears only in the edge refinement). Therefore, even if the radiance of infrared light and the ambient light conditions change, the center of the pupil and the centroid of the Purkinje image can be detected accurately.

3.2 The Eyeball Model

From the detected pupil and the Purkinje image, gaze direction in the 3-D coordinate system is calculated based on an eyeball model. Figure 7 overviews the eyeball model.

Most conventional systems use the position of observed pupil image directly to compute the gaze direction. However, due to the refraction on the surface of cornea, its position of observed pupil is not in the right place. To calculate the accurate gaze direction, it is necessary to compensate the pupil position for obtaining the right position. We introduce the eyeball model for this purpose.

In our eyeball model, the right-handed coordinate system is used. Fig. 6 illustrates the three-coordinate systems; world coordinate, camera coordinate, and image coordinate.

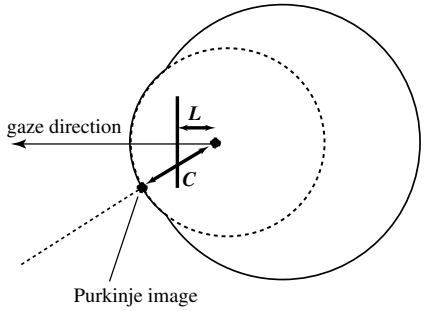


Figure 8: Personal parameters in the eyeball model

The camera was calibrated by Tsai's camera calibration algorithm [Tsai 1987]. The camera's intrinsic parameters (focal length, radial lens distortion, the principal point, i.e. intersection of the camera coordinate's Z axis and the camera's charge coupled device (CCD) image plane) and extrinsic parameters (translational and rotation parameters for the transform between the world and the camera coordinate) were determined by the camera calibration. By using Tsai's camera calibration algorithm, it is possible to transform from the image coordinate (x_0, y_0) to the world coordinate (x_w, y_w, z_w) by giving z_w, the Z coordinate in the world coordinate system,

$$\begin{pmatrix} x_w \\ y_w \\ z_w \end{pmatrix} = f_{iw}(\begin{pmatrix} x_0 \\ y_0 \end{pmatrix}, z_w) \ , \tag{3}$$

where f_{iw} is a function derived from the intrinsic parameters. Transformation from the world coordinate system (x_w, y_w, z_w) to the camera coordinate system (x, y, z) is given by

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = R_c \begin{pmatrix} x_w \\ y_w \\ z_w \end{pmatrix} + T_c, \tag{4}$$

where R_c is the 3×3 matrix, and T_c is the 3×1 translational vector derived from the extrinsic parameters.

The eyeball model has two personal parameters (Fig. 8):

C the radius of cornea curvature, (5)

L the distance between the pupil and the center of cornea curvature. (6)

In general, C and L varies depending on the user; however, those parameters are difficult to be derived from the grabbed eye image. If we use the same values C and L for all users, gaze direction error may remains. This will be discussed later.

3.2.1 Derivation of Center of Cornea Curvature

First, the center of cornea curvature in the camera coordinate is determined.

Given the position vector $\boldsymbol{u_0} = (u_{x_0}, u_{y_0})^T$ of the Purkinje image in the image coordinate, the position vector $\boldsymbol{u} = (u_x, u_y, u_z)^T$ in the camera coordinate is transformed by Eq. (3) as

$$\boldsymbol{u} = f_{iw}(\boldsymbol{u_0},\ u_{z_w}) \ , \tag{7}$$

where u_{z_w} is the Z coordinate of the Purkinje image in the world coordinate. In our system, u_{z_w} is derived from the focus value of the camera (depth from focus). Because the Purkinje image appears

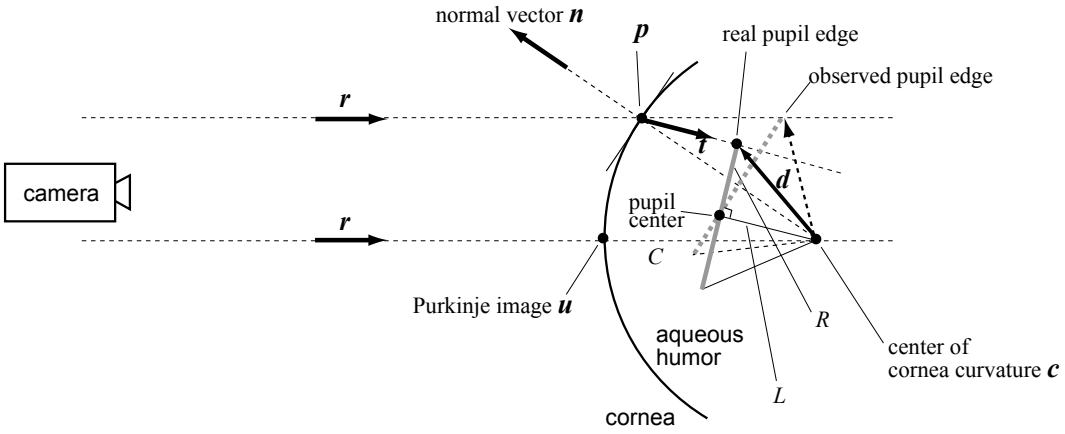


Figure 10: Detection of real pupil center from virtual pupil image.

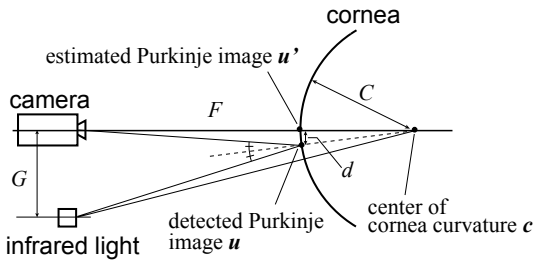


Figure 9: Correction of Purkinje image.

on the spherical surface, the focus value can be determined where the image size is the smallest.

In the implemented system, infrared light is not on the camera axis, so it is necessary to derive the position of the Purkinje image on the camera axis (Fig. 9). The distance between the Purkinje image and the image on camera axis, d is derived as

$$d \approx \frac{C \cdot G}{2\ F} \tag{8}$$

where

C	the radius of cornea curvature given by Eq. (5),
F	the distance between camera and surface of cornea,
G	the distance between camera and infrared light.

(9)

In the implemented system, the camera and infrared light are located on the Y-Z plane. Therefore, the estimated Purkinje image $\boldsymbol{u}'$ is

$$\boldsymbol{u}' = \boldsymbol{u} - \begin{pmatrix} 0 \\ d \\ 0 \end{pmatrix}. \tag{10}$$

From now, we use $\boldsymbol{u}$ instead of $\boldsymbol{u}'$ to describe the corrected Purkinje image.

The Purkinje image is a reflection of infrared light on the cornea surface. Therefore, the position vector $\boldsymbol{c}$ of the center of cornea curvature in the camera coordinate system is given by

$$\boldsymbol{c} = \boldsymbol{u} + C\ \frac{\boldsymbol{u}}{\|\boldsymbol{u}\|}, \tag{11}$$

where $\|\boldsymbol{u}\|$ is a norm of $\boldsymbol{u}$.

3.2.2 Compensation of Refraction at the Surface of Cornea

Here, we describe a method to calculate the real pupil position from the observed pupil image based on the eyeball model. To compensate the refraction at the surface of cornea and to obtain the center of real pupil, we use an edge set of the observed pupil. Consider there are n_0 edge points

$$p_{1_0},\ p_{2_0},\ \ldots,\ p_{n_0}$$

that are sampled from the ellipse given by Eq. (1). As we showed before, the ellipse is the approximation of the pupil image in the image coordinate. There are three steps to detect the center of pupil:

Step 1. For each edge point p_{i_0}, the projected point $\boldsymbol{p}$ on the cornea's surface is calculated (see Fig. 10). This is done analytically by using the camera's direction vector $\boldsymbol{r}$, the position vector of the Purkinje image $\boldsymbol{u}$, the radius of cornea curvature C, and the distance between $\boldsymbol{u}_0$ and p_{i_0} in image coordinate.

Step 2. The ray from the camera is bended at the position $\boldsymbol{p}$ on the surface of the cornea. If the refracted vector at $\boldsymbol{p}$ is derived, it is possible to calculate the real pupil edge. Here, we introduce $\boldsymbol{t}$, the unit vector of the refracted ray at $\boldsymbol{p}$, which is described as

$$\boldsymbol{t} = \frac{n_1}{n_2}\left\{\boldsymbol{r} - \left((\boldsymbol{r},\boldsymbol{n})\ + \sqrt{\left(\frac{n_2}{n_1}\right)^2 - 1 + (\boldsymbol{r},\boldsymbol{n})^2}\ \right)\ \boldsymbol{n}\right\}, \tag{12}$$

where $\boldsymbol{n}$ is the unit normal vector of the curvature at $\boldsymbol{p}$, $\boldsymbol{r}$ is a unit vector from the camera to $\boldsymbol{p}$, n_1 is the refractive index of air, n_2 is the refractive index of aqueous humor, and $(\boldsymbol{r},\boldsymbol{n})$ denotes the inner product of $\boldsymbol{r}$ and $\boldsymbol{n}$.

The real edge point exists on the vector $\boldsymbol{t}$. Next, we must find out the position of the real edge point. Let $\boldsymbol{d}$ be a vector from the center of cornea curvature to the real pupil edge. The length of $\boldsymbol{d}$ can be derived from L and R

$$\|\boldsymbol{d}\| = \sqrt{L^{\,2} + R^{\,2}} \tag{13}$$

where R is the radius of the pupil, which is estimated from the grabbed pupil image. Then, $\boldsymbol{d}$, $\boldsymbol{t}$, $\boldsymbol{d}$ and $\boldsymbol{p}$ have the following relation:

$$\boldsymbol{p} + m\ \boldsymbol{t} = \boldsymbol{c} + \boldsymbol{d}, \tag{14}$$

$$(\boldsymbol{p} - \boldsymbol{c}) + m\ \boldsymbol{t} = \boldsymbol{d}, \tag{15}$$

where m represents the distance between cornea surface and pupil edge point. From this,

$$m = -(\boldsymbol{p}-\boldsymbol{c},\boldsymbol{t}) + \sqrt{(\boldsymbol{p}-\boldsymbol{c},\boldsymbol{t})^2 - (\|\boldsymbol{p}-\boldsymbol{c}\|^2 - \|\boldsymbol{d}\|^2)}, \quad (16)$$

and by putting m into Eq. (15), the real pupil edge $\boldsymbol{d}$ is calculated.

Step 3. By repeating the step 1 and 2 for all edge points p_{i_0} $(i = 1, ..., n_0)$ to compute the real edges $\boldsymbol{d}_i$ $(i = 1, ..., n_0)$. Finally, by ellipse fitting is done for the set of $\boldsymbol{d}_i$ and center of the ellipse $\boldsymbol{e}$ is obtained. This is the estimation of real pupil center in the camera coordinate. ∎

3.2.3 Calculation of Gaze Direction and Gaze Position

The eyeball model enables to derive the center of cornea curvature $\boldsymbol{c}$ and the center of the pupil $\boldsymbol{e}$ in the camera coordinate system. By transforming $\boldsymbol{c}$ and $\boldsymbol{e}$ to the world coordinate system,

$$\boldsymbol{c_w} = {R_c}^{-1}(\boldsymbol{c} - T_c), \quad (17)$$

and

$$\boldsymbol{e_w} = {R_c}^{-1}(\boldsymbol{e} - T_c). \quad (18)$$

Then the gaze vector in the world coordinate $\boldsymbol{v_w}$ can be expressed as

$$\boldsymbol{v_w} = \boldsymbol{e_w} - \boldsymbol{c_w}. \quad (19)$$

Finally, the gaze position $\boldsymbol{g_w}$ is obtained as the intersection of the gaze vector $\boldsymbol{v_w}$ and the screen S.

3.3 Personal Calibration

Most existing gaze tracking systems which are available in the marketplace need bothersome task for personal calibration. They need be calibrated before every measurement session even for frequent users. In addition the burden of calibration is large since users must watch many calibration points, e.g. 5-20 points. Existing systems mostly lack robustness to variation of measurement conditions.

Conventional personal calibration uses second- or higher-order approximation is used in general (e.g. [Morimoto and Flickner 2000]). However, second- or higher-order approximation sometime affects robustness. For example, if the user does not gaze at the point on the screen accurately, error sometimes grows in the calibrated data. In addition, gazing at the point on the screen many times often causes burden and fatigue.

In our personal calibration method, the gaze vector is calibrated by its angle and scale. First, the gaze vector $\boldsymbol{v_w}$ is described as the homogeneous vector in polar coordinate,

$$\boldsymbol{v_w} \longrightarrow \boldsymbol{v_{w\,\theta}} = \begin{pmatrix} l \\ \theta \\ \phi \\ 1 \end{pmatrix}. \quad (20)$$

Next, calibration matrix W is defined as a 4×4 matrix in the homogeneous space,

$$W = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & w_1 & 0 & w_2 \\ 0 & 0 & w_3 & w_4 \\ 0 & 0 & 0 & 1 \end{pmatrix}. \quad (21)$$

To determine the calibration matrix W, the user gazes at least two points on the screen S. It is also possible to use more than two points in the calibration. In this case, the variance of the gaze points may become small, so the accuracy of the gaze direction may increase. Once W is calculated, it is possible to compute the calibrated vector ${\boldsymbol{v_w}'}_\theta$

$${\boldsymbol{v_w}'}_\theta = W\ \boldsymbol{v_{w\,\theta}}, \quad (22)$$

and it is described in rectangular coordinate system as

$${\boldsymbol{v_w}'}_\theta \longrightarrow \boldsymbol{v_w}'. \quad (23)$$

Calibrated gaze point is computed by using $\boldsymbol{v_w}$' instead of $\boldsymbol{v_w}$.

4 Implementation

The gaze tracking system FreeGaze that detects the gaze direction in real time is implemented. Its body is shown in Fig.2. It has a CCD camera that is sensitive to near infrared light. Underneath the camera, there is an infrared LED array. FreeGaze is controlled by a Windows 2000 PC Pentium III 866 MHz. The PC controls the brightness of the LEDs, the gain of the CCD, and the focal distance of the camera. The eye image is captured by a frame grabber (Matrox Meteor II) with a resolution of 640×480 pixels. The sampling rate is 30 fps (frame per second).

In the current version, the camera direction is fixed and the user's head position is restricted to about 4-cm-square area when the distance between the display and the user is 60 cm.

We use the personal parameters as $C = 7.7mm$ and $L = 4.5mm$ which are taken from literatures. The refractive index of air n_1 is 1.000 and that of aqueous humor n_2 is 1.336.

5 Evaluation

To evaluate the accuracy of FreeGaze, an experiment was conducted. In the experiment, head position was either fixed or free, and the number of calibration markers was 2, 4 or 20.

Subjects: Participants were nine researchers and university students. Eight tested in the fixed condition and four in the free condition. Three of four who tested in the free condition also participated the fixed condition.

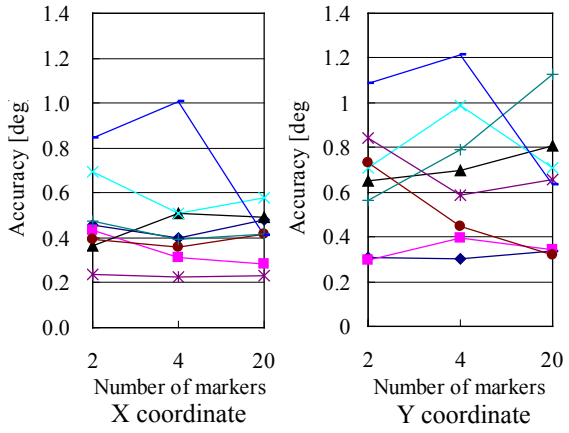
Procedure: Personal calibration was done first and then the accuracy of gaze position was evaluated. In the personal calibration procedure, a marker was displayed on the screen at one time. When a participant pressed a key, the color of the marker changed for one second, and gaze position was recorded. After personal calibration, nine markers were displayed in different places, and the participant was asked to press a key while gazing at each.

Result: The results are shown in Fig.11. These two graphs shows the average accuracy in terms of X coordinate and Y coordinate. The accuracy of the Y coordinate is lower than the X coordinate. The accuracy of X coordinate is generally between 0.23 [deg] to 0.7 [deg] in view angle, and the accuracy of Y coordinate is between 0.3 [deg] to 0.9 [deg]. In the worst case, it was larger than 1.0 [deg]. In such case, personal calibration should be done again.

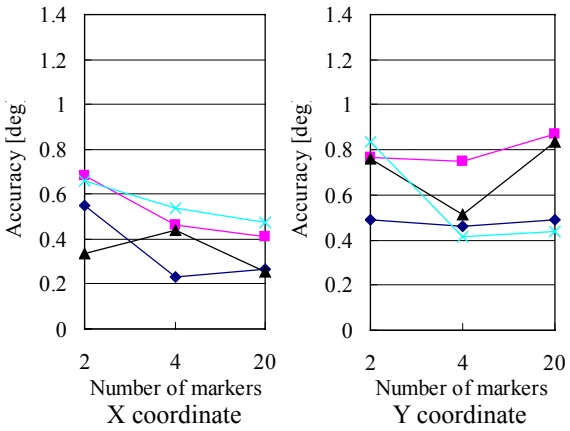
Before the experiments, we expected that increase of the number of calibration points would improve the accuracy. However, the results showed that it did not effect the accuracy, suggesting that personal calibration with two markers is enough for accurate gaze detection.

6 Discussion

From the analysis of the experimental data, we found that the user sometimes moved their eyes between gazing period and confirmation period in the calibration session. It may be cause degradation of accuracy. Detection method is needed whether the gaze position is really fixed during the calibration. When the personal calibration



(a). Head fixed condition.



(b). Head free condition.

Figure 11: The result of user evaluation.

is done well, the accuracy is between 0.3 [deg] and 0.6 [deg]; if the user notices that it does not achieve the expected performance, the user should try it again.

During the experiment, we found that FreeGaze sometimes could not detect the Purkinje image of subjects who wore soft contact lenses. This is because system detects two Purkinje images, one in front of the lens, and another in front of the cornea. If these two images overlap each other, it becomes impossible to detect the correct centroid of the cornea's Purkinje image. This problem will be avoided by properly arranging the camera and infrared light.

6.1 The Variance of Personal Parameter

As we explained earlier, there are two personal parameters, L and C, in the eyeball model. These values were regarded as constant in this work. Although the variations are compensated in the personal calibration, variance of their values may still cause the detection error, so a computer simulation was done to measure the effect of variations. The pupil and the Purkinje position were taken from the real eye image, where an user was looking at the center of the screen and the distance between the screen and the face was 65 cm. L and C were changed between 10 % smaller and 10 % larger. Fig.12 shows the result of the computer simulation. The change of L affected the Y coordinate of the gaze position. On the other hand,

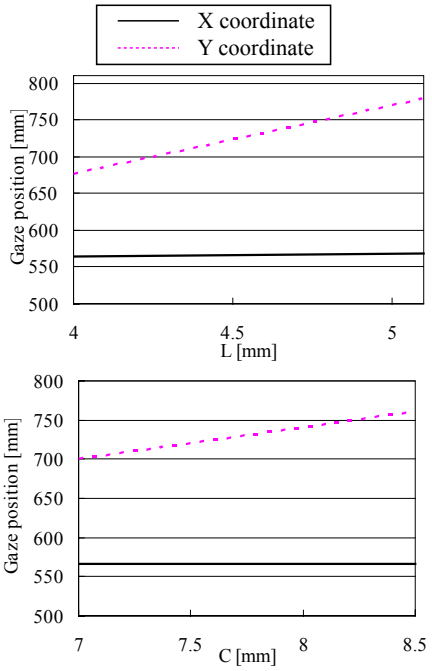


Figure 12: The variance of personal parameter L and C.

the effect of C was not large. If L of each user can be determined, the accuracy will be improved. Development of estimation method of L remains for future work.

6.2 Gazer: Gaze Interaction with Gaze Agent

To demonstrate FreeGaze, a gaze interaction system with a computer agent was developed (Fig. 13) [Mukawa et al. 2001]. The agent, named *Gazer*, is a fortune-teller, and he asks questions to the user. The user can select the answer by looking at "Yes" or "No" on the screen. During the question and answer, Gazer also recognizes the user's eye movement pattern through FreeGaze and responds to the user by moving his eyes. For example, when the user looks into Gazer's eyes, Gazer looks downward shyly.

At the beginning of the demonstration, personal calibration is necessary. To avoid this boring procedure, we developed a personal calibration technique with no active selection such as key pressing. In the procedure, a small Gazer appears in the corner of the screen [Fig.14 (a)], and he jiggles about to attract the user's attention. After the user looks at Gazer, he moves to the opposite corner [Fig.14 (b)], and just when the user looks at him there, the personal calibration is completed. This easy-to-use calibration is possible because FreeGaze can estimate the user's gaze position approximately before personal calibration and it is also possible to calibrate using only two calibration points on the screen.

This system was demonstrated at an exhibition in our laboratory, and more than one hundred people enjoyed interactions with Gazer. Sometimes the accuracy of the personal calibration was insufficient because the system detected gaze position while the user was not looking at the small Gazer. However, in the demonstration, the user soon realized the accuracy was bad and executed the personal calibration again.

Application of FreeGaze is not limited to the fortune-teller task; many application areas are possible, including an online document browser, digital Kiosk, and information desk. It was demonstrated

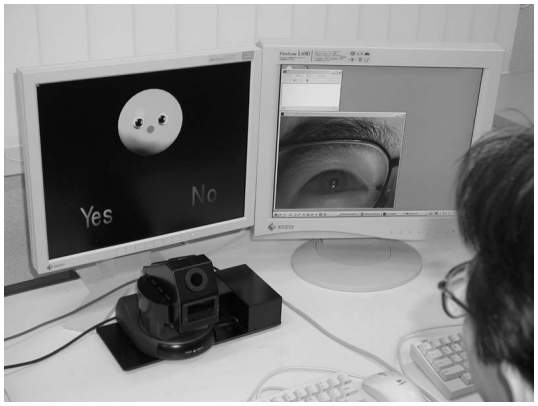


Figure 13: Gazer, a fortune-teller who analyzes the personality of the user. The user can select the answer by looking at “Yes” or “No” on the screen.

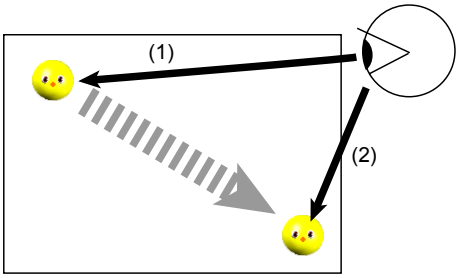


Figure 14: Personal calibration with a small Gazer. First, he is in the left corner, and after the user looks at him there, he moves to the right corner, and when the user looks at him again, the calibration finishes.

that it is possible for anyone to operate the system with just their eyes, no other devices.

7 Conclusion and Future Work

A real-time gaze tracking system was proposed for the everyday use of gaze interaction. It uses a robust pupil detection technique and an accurate gaze computation method based on an eyeball model. The system achieves the personal calibration easily. The user is needed to look at only two markers on the display for the calibration. And once the personal calibration is finished, no further calibration is necessary.

In future work, the restriction of the user’s head position should be reduced. For that purpose we are planning to combine this system with face tracking system. We also have a plan to apply FreeGaze system to various Human-Computer Interaction applications.

Acknowledgment

We thank Dr. Kenichiro Ishii, the Director of NTT Communication Science Laboratories, Dr. Hiroshi Murase, the Executive Manager of Media Information Laboratory and Dr. Norihiro Hagita, the director of ATR Media Information Science Laboratories for their support for our project. We are also grateful to the member of Communication Scene Analysis Research Group for their invaluable comments on this research.

References

EBISAWA, Y. 1998. Improved Video-Based Eye-Gaze Detection Method, *IEEE Transaction on Instrumentation and Measurement 47*, 4, 948–955.

HANSEN, J. P., ANDERSON, A. W., AND ROED, P. 1995. Eye-Gaze Control of Multimedia Systems. In *Symbosis of Human and Artifact* (1995), Y. Anzai, K. Ogawa, and H. Mori, Eds., vol. 20A, Elsevier Science, 37–42.

JACOB, R. J. K. 1990. What You Look At is What You Get: Eye Movement-Based Interaction techniques. In *Proceedings of CHI’90*, ACM Press, 11–18.

JACOB, R. J. K., LEGGETT, J. J., MYERS, B. A., AND PAUSCH, R. 1993. Interaction Styles and Input/Output Devices, *Behaviour and Information Technology 12*, 69–79.

MATSUMOTO, Y. AND ZELINSKY, A. 2000. An Algorithm for Real-time Stereo Vision Implementation of Head Pose and Gaze Direction Measurement. In *Proceedings of Fourth IEEE International Conference on Automatic Face and Gesture Recognition*, 499–504.

MORIMOTO, C., AND FLICKNER, M. 2000. Real-Time Multiple Face Detection Using Active Illumination. In *Fourth IEEE International Conference on Automatic Face and Gesture Recognition*, 8–13.

MUKAWA, N., FUKAYAMA, A., OHNO, T., SAWAKI, M. AND HAGITA, N. 2001. Gaze Communication between Human and Anthroporphic Agent -Its Concept and Examples. In *Proceedings of The 10th IEEE International Workshop on Robot-Human Interactive Communication (ROMAN2001)* , 366-370.

OHNO, T. 1998. Features of Eye Gaze Interface for Selection Tasks. In *Third Asia Pasific Computer Human Interaction - APCHI’98* , IEEE Computer Society, 176–181.

SHIH, S.W., WU, Y.T., AND LIU, J. 2000. A Calibration-Free Gaze Tracking Technique. In *Proceedings of the International Conference on Pattern Recognition (ICPR’00)*, 201–204.

STARKER, I., AND BOLT, R. A. 1990. A Gaze-Responsive Self-Disclosing Display. In *Proceedings of CHI’90*, ACM Press, 3–9.

TSAI, R. Y. 1987. A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-The-Shelf TV Cameras and Lenses, *IEEE Journal of Robotics and Automation RA-3*, 4, 323–344.

ZHAI, S., MORIMOTO, C., AND IHDE, S. 1999. Manual and Gaze Input Cascaded (MAGIC) Pointing. In *Proceedings of CHI’99*, 246–253.