A Free-head, Simple Calibration, Gaze Tracking System That Enables Gaze-Based Interaction

Takehiko Ohno*
NTT Communication Science Laboratories,
NTT Corporation

Naoki Mukawa†
Tokyo Denki University

Abstract

Human eye gaze is a strong candidate to create a new application area based on human-computer interaction. To implement a really practical gaze-based interaction system, gaze detection must be realized without placing any restriction on the user's behavior or comfort. This paper describes a gaze tracking system that offers free-head, simple personal calibration. It does not require the user wear anything on her head, and she can move her head freely. Personal calibration takes only a very short time; the user is asked to look at two markers on the screen. An experiment shows that the accuracy of the implemented system is about 1.0 degrees (view angle).

CR Categories: H.5.2 [Information Interfaces and Presentation]: User Interfrace——Interaction Styles, Gaze Tracking System, Head-Free Gaze Tracking, Stereo Camera System.

Keywords: gaze tracking system, free-head gaze detection, geometric eyeball model, personal calibration, human-computer interaction, corneal reflection method

1 Introduction

The recent improvements in computing power and image processing technology have made human eye gaze a strong candidate to support human-computer interaction(HCI). It conveys rich information that enhances the interaction between people and computers. For example, we acquire perceptual information whenever we choose items on the computer screen. Indeed, when we select an icon by operating a mouse, we look at the exact position of the icon on the screen before pressing the mouse button for selection. As a result, detection of eye gaze reduces object selection time [Zhai et al. 1999]. Another example appears in the conversation between people. While talking to another, eye gaze plays an important role in regulating the conversation. This suggests that it is possible to use human gaze direction to regulate the interaction between people and multiple computers [Shell et al. 2003; Vertegaal 2003]. Gaze is also used for computer operation like mouse and keyboard. Especially, it is used to select objects including menu, icon, and screen-based keyboard [Jacob 1990; Jacob et al. 1993; Hansen et al. 1995; Ohno 1998; Majaranta and Räihä 2002].

To base HCI on gaze direction, we need a really practical gaze tracking system that satisfies the following three requirements.

- Minimal set up difficulty. Should be as simple to use as the keyboard and the mouse. Gaze detection must be accurate with rapid personal calibration.
- No device on the user's head. Even light head devices disturb the user and are a bother to put on and take off.
- No restriction on head movements. Many gaze tracking systems prevent the user from moving his/her head freely because a fixed camera is used to detect the user's eye. This makes gaze detection uncomfortable.

The ultimate gaze tracking system will use a distant camera to follow the user's head movements, and starts gaze detection when the user approaches the system. Such system will strongly improve the penetration of gaze-mediated HCI.

In this paper, we propose a free-head and simple calibration gaze tracking system. An overview of the system is given in Fig. 1. It consists of *the eye positioning unit* and *the gaze detection unit*; the eye positioning unit detects the user's eye position with a stereo-camera and controls the direction of the gaze detection unit accordingly. The gaze detection unit, which consists of a near-infrared camera and a near-infrared light-emitting diode (LED) array, detects the user's gaze direction (the gaze direction vector) and its anchoring point (the gaze positioning vector).

In a previous work, we developed a simple-calibration gaze tracking method that reduced the complexity of personal calibration [Ohno et al. 2002]. It enables highly accurate gaze detection with just two-point personal calibration. The implemented system placed an eye detection camera in front of the user, so no head-mounted devices are required. The camera in this system was fixed, so the user's head position was restricted to a 4cm square area. This prevented long term use. To overcome this problem, we extend the gaze tracking method by adding a stereo-camera based eye positioning system. The enhanced system satisfies the three requirements, and so drastically improves the usability of gaze detection.

This paper first discusses related works. It then describes the proposed gaze tracking method and also shows an implementation of the free-head gaze tracking system. Next, we present an experiment conducted to evaluate the system's accuracy and initial results. Next we discuss the possibility of improving the accuracy of gaze detection. Finally, conclusion and future works are mentioned.

2 Related Works

A lot of research has been conducted to develop a free-head gaze tracking system. Works in this area fall into two categories according to the eye position determination approach: image based or not.

We start by reviewing image-based eye positioning methods. Talmi et al. proposed a gaze tracking system that used a stereo

*3-1 Morinosato Wakamiya, Atsugi, Kanagawa, 243-0198, JAPAN. email:takehiko@brl.ntt.co.jp http://www.brl.ntt.co.jp/people/takehiko/
†e-mail: mukawa@sie.dendai.ac.jp

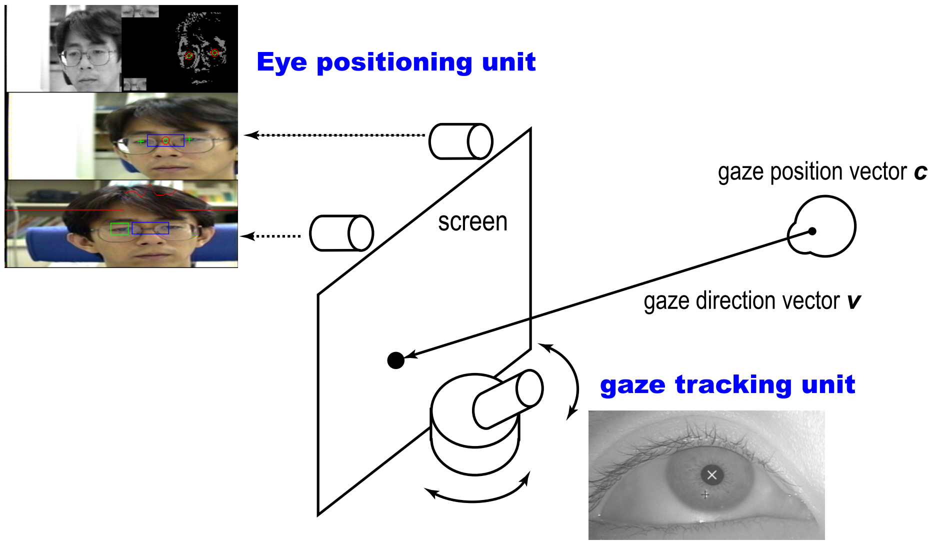


Figure 1: Overview of the stereo-camera based gaze tracking system. It consists of the eye positioning unit, which detects the user's eye position in three-dimensions, and the gaze tracking unit, which detects the user's gaze direction from the eye image taken by a near-infrared camera. The gaze tracking unit is placed on a pan-tilt stand so it can change its direction to detect the user's eye.

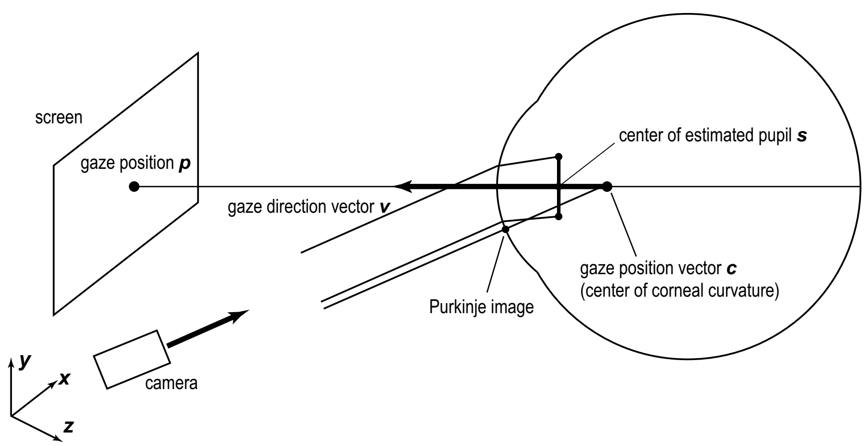


Figure 2: A geometric eyeball model that defines the user's gaze position by the gaze direction vector and the gaze position vector. Gaze direction vector is derived from the center of the corneal curvature and the center of the pupil. Gaze position vector is defined as the center of the corneal curvature, which is derived from the Purkinje image. The model compensates the refraction on the surface of cornea to estimate the exact position of the center of the pupil.

camera for eye positioning and a gaze detection camera. [Talmi and Liu 1999]. Principal Component Analysis was used to detect the user's eye position from the stereo camera output. Corneal reflection is used for gaze detection. The camera setting is similar to our system, however, no actual gaze data was presented in the paper, and nothing was mentioned about personal calibration.

Yoshikawa et. al developed a free-head gaze tracking system that detects the user's eye position with stereo camera, while another camera catches the user's gaze by moving reflection mirrors. Corneal reflection is used to detect gaze direction. To use the system, the user must wear a lightweight triangular frame over the eyes, which has three infrared reflective markers. The stereo camera detects the user's eye position from the markers. This method is robust in detecting eye position, however, the user has to wear the triangular frame, which sometimes slips out of the correct place and causes gaze measurement error. The mechanism of the system was described briefly in [Ohno 1998]. Park et al. also proposed a reflection marker based head-direction detection method [Park and Lim 2001]. In their method, three infrared reflective markers are attached to the user's eyeglasses. In their method, head direction rather than gaze direction is used to interact with the computer. The problem of these methods are that while they allow the user move her head freely, the use of the makers hinders the intermittent use of the computer.

Beymer et. al propoosed a head-free gaze tracking system which consists of wide angle stereo system for eye positioning, and narrow angle stereo system for gaze detection [Beymer and Flickner 2003]. In their system, pan and tilt directions of narrow angle camera are controlled using rotationg mirrors with galvo motors. They confirmed that the accuracy was 0.6 degress where the subject looked at nine points on the screen from two different head positions for personal calibration.

Shih et al. proposed a calibration-free gaze tracking method that employs multiple cameras and multiple spotlights [Shih et al. 2000]. Their method compensates the refraction on the surface of the cornea. They conducted a computer simulation to evaluate the method, but no implementation was reported. They also fail to compensate the difference between the visual axis and the estimated gaze direction. Therefore, some measurement error remains if there is no calibration. Morimoto et. al also proposed a calibration-free gaze tracking method [Morimoto et al. 2002]. This method also uses at least two spotlight sources, however, it requires only one camera. Their computer simulation resulted that the average error of 2.5 degrees was maintained even if the heas position was changed.

Iris contour can also be used to detect gaze direction. Matsumoto et al. proposed a gaze tracking method that uses a stereo camera to detect iris contour [Matsumoto and Zelinsky 2000]. This method allows the user to move her head freely within the visual field of the stereo camera. The method works well even in daylight conditions. Because it is robust, there are several application areas like gaze detection while driving a car. Wang et al. presented an approach to measure the eye gaze via images of the two irises [Wang and Sung 2001]. Their method uses a pose camera and a gaze camera, which is build on a pan-tilt unit, to obtain a higher resolution image of the iris. An evaluation test showed that the method is not so accurate. The problem of these approaches is that the accuracy of gaze detection is not high, which limits the application area.

The other type of approach is to detect the user's head position. For example, a magnetic position sensor system is used to determine the user's head position and direction. The problem with such systems is that the user has to wear the positioning sensor, which is wired to the measurement system. This is not free-head gaze detection. Another problem with magnetic position sensors is that they often exhibit serious measurement error, which causes gaze position error. Therefore, a really practical gaze tracking system for HCI use should be based on some form of image processing.

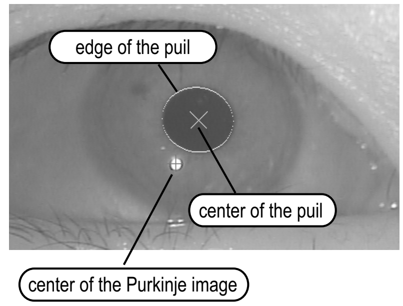


Figure 3: Pupil and Purkinje image captured by the near-infrared camera.

3 Simple-Calibration Gaze Tracking Method

This section briefly describes a gaze tracking method that enables simple personal calibration. This method is a refinement of our previous work [Ohno et al. 2002].

The proposed system defines eye gaze using the gaze direction vector $\boldsymbol{v}$ and position vector $\boldsymbol{c}$ (Fig. 2). Here, a right hand coordinate system is used. The origin of the world coordinate system is set near the gaze detection camera.

Gaze position vector $\boldsymbol{c}$ is equal to the center of the corneal curvature. $\boldsymbol{v}$ is defined as a vector that passes through the center of the cornea curvature and the center of the pupil, which is described as

$$\boldsymbol{v} = \boldsymbol{s} - \boldsymbol{c} \ . \tag{1}$$

where $\boldsymbol{s}$ is the center of the pupil.

To determine the user's gaze, we need to calculate the center of the pupil $\boldsymbol{s}$ and the center of the corneal curvature $\boldsymbol{c}$. These two points are derived from the pupil and the infrared light reflected from the front of the cornea (called the Purkinje image). Fig. 3 shows the pupil and the Purkinje image observed by a infrared-sensitive camera.

The basic strategy of the method is to estimate the position of $\boldsymbol{s}$ and $\boldsymbol{c}$ with high accuracy. Because the eyeball is small (its diameter is about 25 mm), even slight measurement error results large error in predicting gaze position. For example, the distance between $\boldsymbol{s}$ and $\boldsymbol{c}$ is about 4.5 mm, so 0.1 mm of pupil position error yields a view angle error of 1.2 degrees.

3.1 Gaze Detection Procedure

The gaze detection procedure consists of four steps.[1]

1. Pupil and Purkinje image detection from the eye image captured by the infrared-sensitive camera. To derive their position with sub-pixel resolution, the center of the pupil is determined by fitting an ellipse to the contour of the pupil image, and the center of the Purkinje image is defined as the center of gravity of the observed Purkinje image.

2. Calculate the center of the corneal curvature, $\boldsymbol{c}$, in the world coordinate system from the Purkinje image. To derive the center of the corneal curvature, it is necessary to calculate Purkinje image position in the world coordinate system. We use

[1] This procedure is fully described in [Ohno et al. 2002].

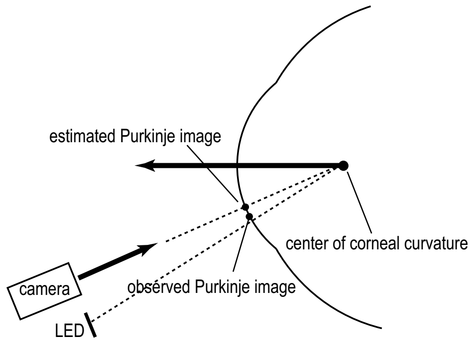


Figure 4: Estimation of correct Purkinje image position from the observed Purkinje image. Displacement of LED from camera axis causes difficulty in estimating the center of corneal curvature from the Purkinje image. Therefore, we estimate the Purkinje image position assuming that the Purkinje image is on the camera axis.

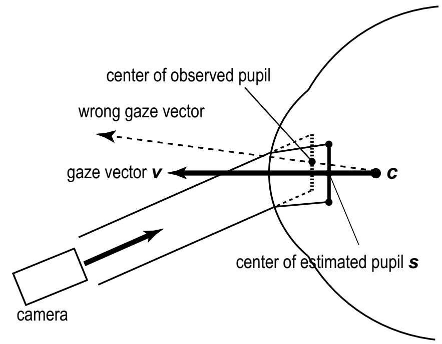


Figure 5: Estimation of real pupil position from the observed pupil image. Observed pupil position differst from the real pupil position due to refraction at the surface of the cornea. Therefore, to calculate the correct gaze direction, it is necessary to estimate the real pupil position from the observed pupil image.

the focal number of the camera to determine the Purkinje image position (depth-from-focus method). In our implemented system, an LED array is placed off the camera axis. Therefore, the center of the corneal curvature does not exist on the extension from the center of the camera to the Purkinje image. Therefore, we estimate the Purkinje image as if the LED array was on the camera axis (Fig. 4).

3. Calculate the center of the pupil, $\boldsymbol{s}$, from the pupil image. Due to the refraction present on the surface of the cornea, the observed pupil position differs from the real pupil position. To compensate this difference, we estimate the real pupil position from the observed pupil image (Fig. 5) [2].

4. Calculate the gaze direction and the gaze position on the screen. Steps 2 and 3 yield $\boldsymbol{s}$ and $\boldsymbol{c}$. Therefore, we get the gaze direction vector $\boldsymbol{v}$ defined by Eq.1. Projecting the gaze direction vector to the screen yields gaze position on the screen, p.

3.2 Personal Calibration

In most cases, the estimated gaze position contains measurement error which should be offset by personal calibration. For personal calibration, we use linear approximation of gaze position. This is possible because non-calibrated gaze position includes little or no non-linear factors. That is, when the user looks at markers arranged in lattice, measured gaze positions also form a lattice. For example, as shown in Fig. 6, when there are nine calibration markers that are arranged in 3 by 3 lattice on the screen. Non-calibrated gaze positions also form a lattice. Therefore, only two calibration markers are required for personal calibration.

In contrast, existing gaze tracking methods use second or higher order approximation (e.g. [Morimoto and Flickner 2000]). Therefore, five to twenty personal calibration markers are required.

Here, we detail personal calibration. Our calibration method compensates only the gaze direction vector, $\boldsymbol{v}$, because the major cause of measurement error exists only in the gaze direction. Even

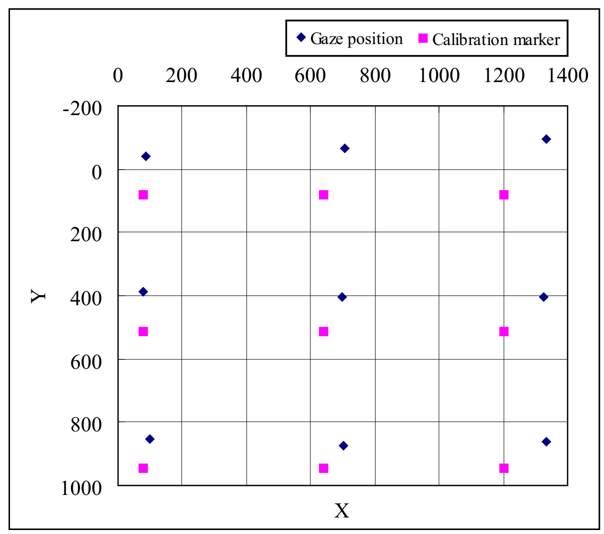


Figure 6: Non-calibrated gaze data. An user looked at nine calibration markers on the screen. The figure shows calibration markers and the user's gaze positions.

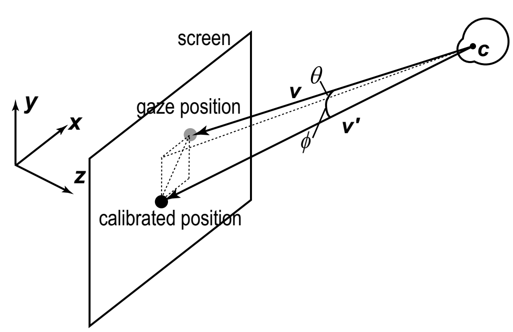


Figure 7: Personal calibration of the gaze vector. In the calibration, only the gaze direction vector is calibrated. the gaze position vector is not modified.

[2] In the previous work, radius of the observed pupil image is used to estimate the real pupil position. However, it is larger than the real pupil when we observe the pupil from the oblique direction. The current version compensates the direction when we use the radius of the observed pupil image.

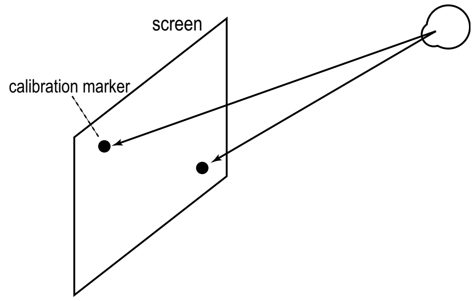


Figure 8: Simple personal calibration using two calibration markers on the screen.

if the gaze position vector contains some error, it appears as the same measurement error on the screen. On the other hand, if the gaze direction vector has some error, it is amplified on the screen. When the distance between the pupil and the screen is 600 mm, it is about 130 times long as the length from the center of the corneal curvature to the center of the pupil. Therefore, 0.1mm of gaze direction error becomes 13mm on the screen.

In the personal calibration method, the gaze direction vector is calibrated by its angle and scale. First, the gaze direction vector is described as the homogeneous vector in a polar coordinate system,

$$\boldsymbol{v} \longrightarrow \boldsymbol{v}_\theta = \begin{bmatrix} l \\ \theta \\ \phi \\ 1 \end{bmatrix} . \tag{2}$$

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

$$W = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & w_1 & 0 & w_2 \\ 0 & 0 & w_3 & w_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} . \tag{3}$$

In personal calibration, calibration matrix W is determined. The calibration matrix M has four unknown parameters $w_1, w_2, \cdots, w_4$, so at least two calibration points on the screen are required to determine those parameters. As shown in Fig. 8, the calibration matrix can be determined by looking at two calibration markers arranged at corners of the screen.

Once W is calculated, the calibrated gaze direction vector $\boldsymbol{v}'_\theta$ is described as

$$\boldsymbol{v}'_\theta = W \cdot \boldsymbol{v_\theta} \tag{4}$$

$$= \begin{bmatrix} l \\ w_1\theta + w_2 \\ w_3\phi + w_4 \\ 1 \end{bmatrix} . \tag{5}$$

Calibrated gaze vector $\boldsymbol{v'}$ in rectangular coordinate system is derived from $\boldsymbol{v'_\theta}$ as

$$\boldsymbol{v'_\theta} \longrightarrow \boldsymbol{v'} . \tag{6}$$

Calibrated gaze position on the screen can be computed by using $\boldsymbol{v'}$ instead of $\boldsymbol{v}$. However, if we use $\boldsymbol{v'}$, the effect of compensation becomes large when θ and ϕ are large. This means that in the surrounding area of the screen, gaze position is over compensated. To avoid the problem, we use the following method to derive the gaze position on the screen. First, the compensation angle

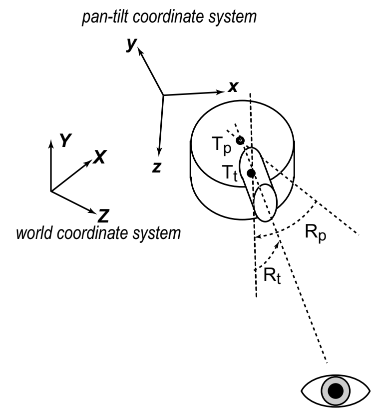


Figure 9: Pan and tilt rotation of gaze tracking unit to catch the user's eye.

$\Delta\boldsymbol{v_\theta} = [0 \;\; \Delta\theta \;\; \Delta\phi \;\; 0]^T$ is calculated.

$$\Delta\boldsymbol{v_\theta} = W \cdot \boldsymbol{v_\theta} - \boldsymbol{v_\theta} \tag{7}$$

$$= \begin{bmatrix} 0 \\ w_1\theta + w_2 - \theta \\ w_3\phi + w_4 - \phi \\ 0 \end{bmatrix} . \tag{8}$$

Next, compensated gaze position $\boldsymbol{p'}$ on the screen is derived as

$$\boldsymbol{p'} = \boldsymbol{p} + \begin{bmatrix} m \cdot sin\Delta\theta \cdot cos\Delta\phi \\ m \cdot sin\Delta\phi \end{bmatrix} , \tag{9}$$

where m is the distance between the gaze position vector and the non-calibrated gaze point on the screen, which is described as

$$m = \| \, \boldsymbol{p} - \boldsymbol{c} \, \| , \tag{10}$$

where $\|\boldsymbol{p} - \boldsymbol{c}\|$ is a norm of $\boldsymbol{p} - \boldsymbol{c}$.

4 Free-head Gaze Detection Method

4.1 Gaze Detection Procedure

The gaze detection method described in the previous section is designed for fixed-head systems. Now, we extend the method for free-head systems. The principal idea is unchanged from the head-fixed method; to determine user's gaze position, we calculate the gaze position vector, $\boldsymbol{c}$, and gaze direction vector, $\boldsymbol{v}$. We use a pan-tilt coordinate system and a world coordinate system to calculate the two vectors (Fig. 9). The world coordinate system described in Sec. 3 is described as pan-tilt coordinate system because it changes when the direction of camera changes.

At first, eye position $\boldsymbol{e} = [e_x \; e_y \; e_z]^T$ in the world coordinate system is determined by the eye positioning unit, which consists of two cameras. To calculate the eye position with the stereo camera, eye position is determined for the each camera [3]. Based on the eye position derived from the stereo image, the eye position in the world coordinate system is calculated. Actually, the center of the iris is detected as the user's eye.

[3]We use the eye detection method proposed in [Kawato and Tetsutani 2002], which detects the user's eye blink to create a template image of a two-eye area. Once the template image is created, the present eye position is located based on the measured position of the middle of the two-eye area and the previously detected eye position.

Next, a camera rotation matrix is calculated using a homogeneous matrix. Because we use a pan-tilt stand to mount the gaze tracking unit, there are two rotation axes, pan rotation axis and tilt rotation axis. For each axis, we define the position vector $\boldsymbol{t}$ relevant to the center of rotation, and the rotation matrix R, which describes the rotation around the axis. We assume that the center of the pan rotation $\boldsymbol{t_p} = [t_{px} \ \ t_{py} \ \ t_{pz}]^T$ and the center of the tilt rotation $\boldsymbol{t_t} = [t_{tx} \ \ t_{ty} \ \ t_{tz}]^T$ are given in advance.

To determine the rotation matrix of pan and tilt, we first derive the pan angle a_p and tilt angle a_t. The pan angle a_p is derived as

$$a_p = tan^{-1}\left(\frac{e_x - t_{px}}{e_z - t_{pz}}\right) , \tag{11}$$

and the tilt angle a_t is derived as

$$a_t = tan^{-1}\left(\frac{e_y - t_{py}}{\sqrt{(e_x - t_{px})^2 + (e_z - t_{pz})^2}}\right) . \tag{12}$$

Thus the rotation matrix of pan angle R_p is obtained as

$$R_p = \begin{bmatrix} cos(a_p) & 0 & sin(a_p) & 0 \\ 0 & 1 & 0 & 0 \\ -sin(a_p) & 0 & cos(a_p) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} , \tag{13}$$

and the rotation matrix of tilt angle R_t is obtained as

$$R_t = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & cos(a_t) & -sin(a_t) & 0 \\ 0 & sin(a_t) & cos(a_t) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} . \tag{14}$$

From the position vector of pan and tilt rotation, the transform matrix of pan rotation T_p is derived as

$$T_p = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ t_{px} & t_{py} & t_{pz} & 1 \end{bmatrix} , \tag{15}$$

and the transform matrix of tilt rotation T_t is also derived as

$$T_t = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ t_{tx} & t_{ty} & t_{tz} & 1 \end{bmatrix} . \tag{16}$$

Finally, gaze position and direction are calculated. As described in the previous section, gaze position vector $\boldsymbol{c}$ and gaze direction vector $\boldsymbol{v}$, estimated by the gaze detection camera, are given in the pan-tilt coordinate system. Therefore, it is necessary to transform the pan-tilt coordinates into world coordinates. The gaze direction vector $\boldsymbol{v}$ is derived as

$$\boldsymbol{v} = T_p \cdot R_p \cdot T_p^{-1}(T_t \cdot R_t \cdot T_t^{-1})\boldsymbol{v_0} , \tag{17}$$

and the gaze position vector $\boldsymbol{c}$ in the world coordinate system are derived as

$$\boldsymbol{c} = T_p \cdot R_p \cdot T_p^{-1}(T_t \cdot R_t \cdot T_t^{-1})\boldsymbol{c_0} , \tag{18}$$

where $\boldsymbol{v_0}$ is equal to the calibrated gaze direction vector $\boldsymbol{v'}$ derived in Eq. (6), and $\boldsymbol{c_0}$ is equal to the center of the corneal curvature $\boldsymbol{c}$ described in Sec. 3.1.

Projecting the gaze direction vector to the screen, yields the gaze position on the screen, p.

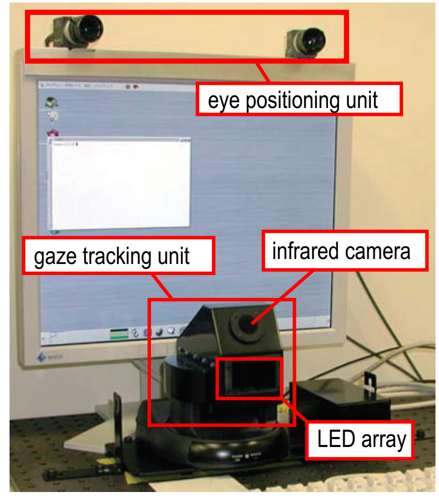


Figure 10: The overview of free-head gaze tracking system, which consists of eye positioning unit mounted on the display, and the gaze tracking unit placed under the display.

4.2 Personal Calibration

To calibrate the user's gaze position, we use the personal calibration method described in Sec. 3.2. The gaze vector $\boldsymbol{v}$ is corrected by the calibration matrix M. Even if the user's eye position changes, the same calibration matrix is used.

5 Implementation

We implemented a free-head gaze tracking system based on the gaze tracking method described in Sec. 4. Fig. 10 overviews the system. The eye positioning unit, controlled by a Linux PC with dual AMD Athlon 1.7GHz CPUs, has a stereo camera set which consists of two NTSC cameras. They are placed on a 18.1 inch LCD display EIZO L674, whose resolution is 1280×1024. Two images taken by those cameras are combined into one NTSC image, and captured by a Conexant Bt878 video capture card in RGB color mode 30 times per second. For image processing, only even lines of the captured image are used. Therefore, the capture size of each camera is 320×120 pixels.

The gaze tracking unit, controlled by a Linux PC with dual Intel Pentium III 1GHz CPUs, has a near-infrared sensitive NTSC camera placed on a pan-tilt stand. A near-infrared LED array is also placed under the camera. The eye image taken by the camera is captured by a Conexant Bt878 video capture card in gray image mode 30 times per second. The image size is 640×480 pixels. Pan-tilt stand, focus parameter of the camera, LED brightness, and camera gain are controlled by the PC through the RS-232C serial interface. These two PCs are connected by a TCP/IP network. The sampling rate of gaze detection is 30Hz.

Fig. 11 shows the flow used by the eye positioning unit and the gaze tracking unit to detect the user's gaze position. These two units run independently, and as we mentioned before, eye position data is sent to the gaze tracking unit if there is a request from the gaze tracking unit. The request is sent to the eye positioning unit if the gaze tracking unit can not detect the pupil and the Purkinje image from the captured eye image. If an answer is received from the eye tracking unit, the gaze detection unit controls the pan-tilt stand to track the user's eye. Focal number parameter of the gaze detection camera is also updated with the newest eye position data.

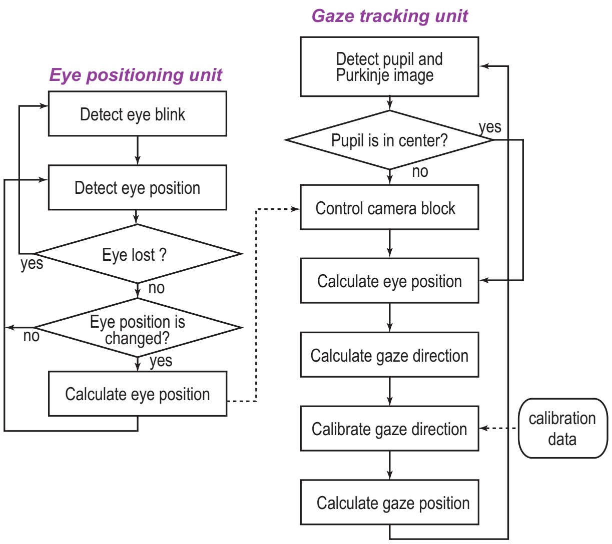


Figure 11: The flow of eye and gaze detection. Eye positioning unit and the gaze tracking unit run separately, and the eye position data is sent to the gaze tracking unit when the eye position is changed.

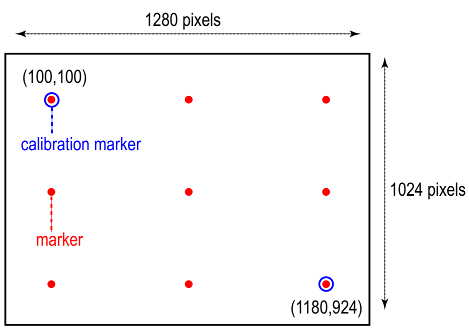


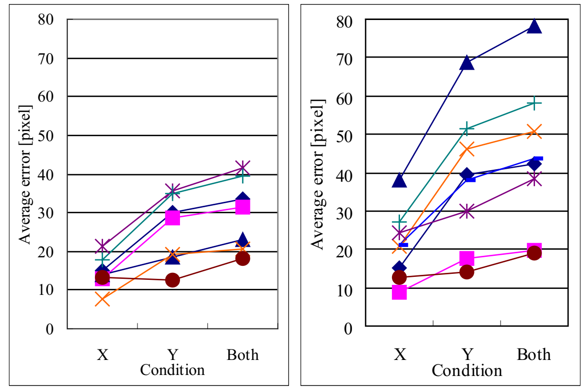
Figure 12: Two calibration markers and nine markers used in the experiment. During the experiment, only one marker appeared on the screen in each gaze detection trial.

The direction of the pan-tilt stand is also changed if the center of the pupil moves to the boundary area of the captured image for centering the pupil in the image. This function works well especially when the user's head moves slowly.

The recommended distance from the camera to the display is between 60cm and 90cm. When the distance is 60cm, the acceptable eye area is about 10cm $\times$ 10cm. When the distance is 90cm, the acceptable eye area is about 15cm $\times$ 15cm. This limitation comes from the eye detection method used in the eye positioning unit. That is, the method captures both eyes for eye detection. However, the measurable area can be expanded by changing the stereo cameras' focal lengths and the convergence angle.

6 Evaluation

The proposed method was tested in an experiment. In the experiment, The accuracy of gaze detection was measured just after calibration and after a short break. Because eye position was changed during the break time, we could compare the accuracy when the user's eye position was changed.



(a). Just after the personal calibration. (b). after the short break.

Figure 13: Results of the experiment. Two graphs show the average measurement error in x-coordinate, y-coordinate, and both coordinates for each subject. The y-coordinate of the graph shows the number of pixels on the screen.

Subjects Nine subjects participated the experiment. Seven subjects had naked eyes, and the rest wore eyeglasses.

Procedure We conducted the experiment in four steps.

1. The system was calibrated by the two-point personal calibration method. During the calibration, the subjects were asked to press a space key while they were looking at the calibration marker on the display. The pan-tilt stand was not moved during the personal calibration, and they were asked to fix their heads by themselves.

2. When calibration was finished, nine markers appeared at different positions one after the other, and for each maker, the subjects were asked to press a space key while looking at the marker (*condition 1*).

3. They took a break for a short period. During the break, they were asked to leave their seat.

4. They returned to the seat and then repeat step (2) again (*condition 2*).

The positions of the two calibration markers and the nine markers on the screen are shown in Fig. 12.

Result Fig. 13 shows the average measurement error of condition (1) and condition (2) in terms of x-coordinate, y-coordinate, and both coordinates for each subject. The distance between subjects and the screen was different in each session, so we can not derive the view angle accuracy from the data. If we assume that the distance between the screen and the user is 70cm, which is the typical distance in the implemented system, 40 pixels on the screen equals a view angle of 0.93 degrees.

In condition 1, the measurement error was between 18.3 pixels and 41.6 pixels, the average was 29.7 pixels. Assuming that the distance between the user and the screen was 70cm, this is equivalent to a view angle of 0.68 degrees.

In condition 2, the measurement error was increased; it was between 18.9 pixels and 78.4 pixels, the average was 43.8 pixels. Assuming that the distance between the user and the screen was 70 cm, this is equivalent to a view angle of 1.0 degrees. In the best case, ther error was 18.9 pixels or 5.3mm on the screen, which is equivalent to a view angle of 0.43 degrees when the distance between the user and the screen was 70cm.

In both cases, the x-coordinate accuracy was better than y-coordinate accuracy. This result matches that noted in our previous work [Ohno et al. 2002].

7 Discussion

We confirmed that the user could move his/her head freely during gaze detection. For some users, the accuracy is not changed even if they left the seat and then returned.

However, the experiment showed that for some users, the accuracy of gaze detection decreases when their eye position changed. Though the gaze tracking system works well in selecting larger objects on the screen, higher accuracy is desired. We are considering two ways of improving the accuracy.

- Increase the resolution of the eye positioning unit. Because the iris in the captured image is not clear, detected eye position is sometimes different from the center of the iris. This causes measurement error in eye position. Especially when the user moves their head front or back slightly, the eye position unit sometimes can not detect the movement, which creates gaze detection error.

- Increase the resolution of the gaze tracking unit. When the distance between the user and the screen is long, the captured eye size is too small for accurate gaze tracking. It is better to use a longer focal length camera to improve the accuracy.

Another problem we should mention is the robustness of eye positioning. Vision based eye detection is sometimes not as robust as magnetic position sensors or reflection markers on eyeglasses,. Our eye positioning system occasionally loses or wrongly detects eye position. The problem of misdetection is corrected when the gaze tracking unit sends a signal to the eye positioning unit to restart eye detection (in the current implementation, it waits for the next eye blink). Therefore, it works better than the standalone eye positioning system. Furthermore, we are planning to use a more robust eye detection algorithm that does not require eye blink for starting eye detection. We believe that the integration of the gaze tracking unit and the current eye detection algorithm are already robust enough to be practical.

8 Conclusion and Future Works

We proposed a free-head, simple calibration, gaze tracking system for gaze-based human-computer interaction. It consists of an eye positioning unit, which has a stereo camera set, and a gaze tracking unit, which has a near-infrared camera and a near-infrared LED array. An evaluation test confirmed that the accuracy of gaze detection is about a view angle of 1.0 degrees. We are now trying to improve the accuracy. Development of the gaze tracking system that captures the user's eye from further away to provide even freer gaze detection is a future task.

Acknowledgement

We are grateful to the member of Recognition Research Group for their invaluable comments and advices on this research.

References

BEYMER, D., AND FLICKNER, M. 2003. Eye gaze tracking using an active stereo head. In *Proceedings of Computer Vision and Pattern Recognition*, vol. 2, 451–458. eye tracking, head-free, active camera.

HANSEN, J. P., ANDERSON, A. W., AND ROED, P. 1995. Eye-gaze control of multimedia systems. In *Symbiosis of Human and Artifact: Future Computing and Design for Human-Computer Interaction, Proceedings of The 6th International Conference on Human Computer Interaction*, Elsevier Science, Y. Anzai, K. Ogawa, and H. Mori, Eds., vol. 20A, 37–42.

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*, 2, 69–79.

JACOB, R. J. K. 1990. What you look at is what you get: Eye movement-based interaction techniques. In *Conference Proceedings on Human Factors in Computing Systems (CHI '90)*, ACM Press, 11–18.

KAWATO, S., AND TETSUTANI, N. 2002. Detection and tracking of eyes for gaze-camera control. In *Proeedings of VI2002*, 348–353.

MAJARANTA, P., AND RÄIHÄ, K. 2002. Twenty years of eye typing: Systems and design issues. In *Proceedings of Eye Tracking Research & Application (ETRA2002)*, 15–22.

MATSUMOTO, Y., AND ZELINSKY, A. 2000. An algorithm for real-time stereo vision implementation of head pose and gaze direction measurement. In *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.

MORIMOTO, C. H., AMIR, A., AND FLICKNER, M. 2002. Free head motion eye gaze tracking without calibration. In *CHI '02 extended abstracts on Human factors in computer systems*, vol. 586–587.

OHNO, T., MUKAWA, N., AND YOSHIKAWA, A. 2002. Freegaze: A gaze tracking system for everyday gaze interaction. In *Proceedings of Eye Tracking Research & Application (ETRA2002)*, 125–132.

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.

PARK, K. S., AND LIM, C. J. 2001. A simple vision-based head tracking method for eye-controlled human/computer interface. *International Journal of Human-Computer Studies 54*, 319–332.

SHELL, J. S., SELKER, T., AND VERTEGAAL, R. 2003. Interacting with groups of computers. *Communication of the ACM 46*, 3, 40–46.

SHIH, S., WU, Y., AND LIU, J. 2000. A calibration-free gaze tracking technique. In *Proceedings of the International Conference on Pattern Recognition*, 201–204.

TALMI, K., AND LIU, J. 1999. Eye and gaze tracking for visually controlled interactive stereoscopic displays. *Signal Processing: Image Communication 14*, 799–810.

VERTEGAAL, R. 2003. Attentive user interfaces. *Communication of the ACM 46*, 3, 31–33.

WANG, J., AND SUNG, E. 2001. Gaze determination via images of irises. *Image and Vision Computing 19*, 891–911.

ZHAI, S., MORIMOTO, C., AND IHDE, S. 1999. Manual and gaze input cascaded (magic) pointing. In *Proceedings of Conference on Human Factors in Computing Systems (CHI'99)*, ACM Press, 246–253.