Remote Gaze Estimation with a Single Camera Based on Facial-Feature Tracking without Special Calibration Actions

Hirotake Yamazoe* Akira Utsumi* Tomoko Yonezawa* Shinji Abe*
ATR Intelligent Robotics and Communication Laboratories

Abstract

We propose a real-time gaze estimation method based on facial-feature tracking using a single video camera that does not require any special user action for calibration. Many gaze estimation methods have been already proposed; however, most conventional gaze tracking algorithms can only be applied to experimental environments due to their complex calibration procedures and lacking of usability. In this paper, we propose a gaze estimation method that can apply to daily-life situations. Gaze directions are determined as 3D vectors connecting both the eyeball and the iris centers. Since the eyeball center and radius cannot be directly observed from images, the geometrical relationship between the eyeball centers and the facial features and eyeball radius (face/eye model) are calculated in advance. Then, the 2D positions of the eyeball centers can be determined by tracking the facial features. While conventional methods require instructing users to perform such special actions as looking at several reference points in the calibration process, the proposed method does not require such special calibration action of users and can be realized by combining 3D eye-model-based gaze estimation and circle-based algorithms for eye-model calibration. Experimental results show that the gaze estimation accuracy of the proposed method is 5° horizontally and 7° vertically. With our proposed method, various application such as gaze-communication robots, gaze-based interactive signboards, etc. that require gaze information in daily-life situations are possible.

CR Categories: I.4.8 [Image Processing and Computer Vision]: Scene Analysis—Tracking; H.1.2 [Models and Principles]: User/Machine Systems—Human Information Processing

Keywords: remote gaze tracking, non-intrusive, daily-life situations

1 Introduction

Since gaze information is useful in such applications as human-computer interaction and user intention detection, various gaze estimation algorithms have been proposed.

As gaze estimation algorithms, corneal reflection-based methods are popular [Ohno et al. 2003; Morimoto and Mimica 2005]. They employ infrared (IR) illumination mounted near the camera to illuminate the eye region and to estimate gaze direction by detecting the pupil positions and the reflected illumination images on the cornea (Purkinje images). Since Purkinje images are very small, high-resolution images of the eye region are required. In addition, for easy detection of the Purkinje images and the pupils, the eye regions must get enough bright illumination. Therefore, these methods have a limitation: they require a very short distance (less than 1[m]) between user and camera (IR illumination).

Recently, gaze estimation methods only based on normal video camera(s) have been proposed that can be divided into two types: appearance-based and model-based.

In appearance-based methods, gaze directions can be estimated from eye-area images using pattern recognition algorithms such as the neural network or the nearest neighbor method [Baluja and Pomerleau 1994; Shiele and Waibel 1995; Ono et al. 2006; Morency et al. 2006]. These methods require a large amount of learning data and are not robust against changes in users and user head poses. In addition, accurate eye-region extraction is difficult, and errors in eye-region extraction also cause gaze estimation errors. Ono et al. proposed a gaze estimation method based on N-mode SVD [Ono et al. 2006] that considered eye-region extraction errors; however, different users and user head pose changes were not considered.

Model-based methods can also be divided into two types. One is the so-called "circle algorithm" [Wang and Sung 2001; Wu et al. 2004] method that employs a property in which irises are observed as ellipses to estimate gaze direction by ellipse shape. Ellipses are fitted to the observed iris regions to estimate gaze direction from fitting the ellipse parameters. The circle algorithm doesn't need any calibration process. However, for accurate fitting of ellipses, high-resolution images are required. In addition, estimation accuracy worsens when the angle between the gaze and camera directions is small (users are looking in the camera direction), since in that case the irises are observed as nearly circles.

Another model-based approach is founded on a 3D eye-model [Matsumoto and Zelinsky 2000; Miyake et al. 2002; Ishikawa et al. 2004], where gaze directions are estimated as a vector from the eyeball center to the iris centers. Since the eyeball center cannot be directly observed in the images, a calibration process is required to estimate the geometrical relationship between the facial features and the eyeball centers as well as the eye model parameters. To calibrate these parameters, users need to look at several reference points, significantly limiting the usability of such systems.

Most gaze tracking systems described above require user actions to calibrate their gaze estimation models, such as looking at some reference points. Gaze information can be useful in daily-life situations or outdoor scenes; however, the conventional gaze tracking systems methods cannot apply to such situations due to their calibration procedures and lacking of usability.

In this paper, we propose a single-camera-based gaze estimation algorithm that does not require the attachment of any devices or markers for gaze estimation or such special user action as looking at reference points in the calibration. In the eye-model calibration process, we employ an intermediate approach of the above two types of model-based algorithms. Then gaze direction can be determined based on a 3D eye-model.

Takegami et al. proposed a similar method that does not require special user actions for calibration [Takegami et al. 2002]. However, their method assumes that users wear a gaze estimation camera on

*e-mail: {yamazoe,utsumi,yone,sabe}@atr.jp

ETRA 2008, Savannah, Georgia, March 26–28, 2008.
© 2008 ACM 978-1-59593-982-1/08/0003 $5.00

their heads and does not consider cases where the relationship between camera and eyes is changed.

In our proposed method, eye model parameters and the geometrical relationship between eyeball centers and facial features are estimated in advance. Gaze direction can be determined by tracking facial features and locating iris centers. Since our method does not require special user actions for calibration, users do not need awareness of the calibration process.

2 Overview of Proposed Method

2.1 Gaze Estimation Model

In the proposed method, we assume that gaze direction can be determined as a 3D vector connecting the eyeball and the iris centers (Figure 1). Let $\boldsymbol{x}_c = [x_c, y_c]$ be the 2D positions of the eyeball centers, $\boldsymbol{x}_{iris} = [x_{iris}, y_{iris}]$ be the 2D positions of the iris centers, and r be the radius of the eyeballs in the image. The gaze direction (angles between the camera's optical axis and gaze direction) can be estimated as the following equations:

$$\begin{aligned} \alpha &= \sin^{-1}\left(\frac{x_{iris} - x_c}{r}\right) \\ \beta &= \sin^{-1}\left(\frac{y_{iris} - y_c}{r}\right), \end{aligned} \tag{1}$$

where α and β are the x and y axial components of the gaze direction, respectively.

The iris centers can be observed in the image sequences; however, other eye-model parameters, such as the positions of eyeball centers and eyeball radii cannot be determined from images.

This paper proposes a method to estimate these eye-model parameters without requiring special user actions.

2.2 Process flow of proposed method

Figure 2 shows the process flow of the proposed method that consists of three processes: facial-feature detection/tracking, face/eye model estimation, and gaze estimation.

We detect the facial features in the facial-feature detection/tracking process. These features are used in both the face/eye model estimation and gaze estimation processes. First, we detect the face position in the image using a face detection method based on the Six-Segmented Rectangular (SSR) filter [Kawato et al. 2005]. Rough positions of the facial features can be determined from the face-detection results. Then, we extract the facial features by searching

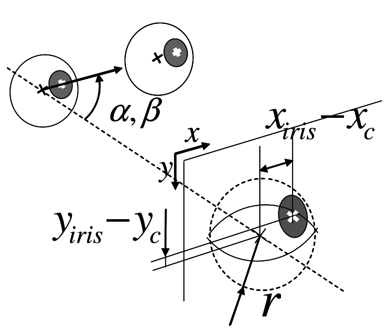

Figure 1: *Gaze estimation model*

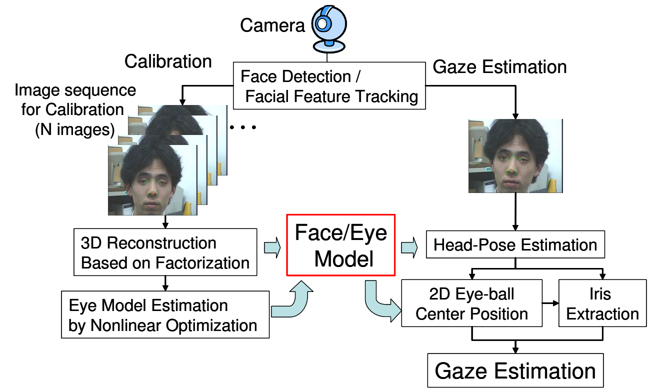

Figure 2: *Process flow of proposed method*

around the rough positions of each feature with the Lucas-Kanade Tracker [Lucas and Kanade 1981] to obtain the facial-feature positions in the image.

In the face/eye model estimation process, first we capture the N images for face/eye model estimation. Here, our method does not require special user action. Then, we track the facial-feature points in image sequences. Using the extracted feature points, the 3D position of each feature point (face model) can be calculated based on the factorization method [Poelman and Kanade 1992]. Eye model parameters (3D positions of the eyeball centers, the eyeball, and the iris radii) are estimated using the same image sequences by using nonlinear optimization.

In the gaze estimation process, we extract and track facial features from newly observed images, and the 2D positions of the eyeball centers are calculated using a 3D face model and the 2D facial-feature positions. Finally, iris centers are obtained by fitting an ellipse into the observed image, and gaze direction can be determined by Equation (1).

In the next section, we explain the details of the proposed method.

3 Gaze Estimation Algorithms based on Facial-Feature Tracking

3.1 Facial-feature Detection and Tracking

In this section, we explain facial-feature extraction and tracking. Facial-feature detection/tracking results are used in both the face/eye model estimation and gaze estimation processes.

For facial-feature detection, first, we detect the 2D positions of the face and locate both eyes in the images using a face detection method based on a Six-Segmented Rectangular (SSR) filter and a Support Vector Machine (SVM) [Kawato et al. 2005]. Next, rough positions of facial features such as eyes, node and mouth are determined based on the face detection results. Then the Lucas-Kanade's feature points [Lucas and Kanade 1981] are detected by searching around the rough facial-feature positions (Figure 3).

For facial-feature tracking, we also employ Lucas-Kanade's tracking method.

3.2 Face/Eye Model Estimation

Next, we describe face and eye model estimation. The processes described in this section are preprocessing for gaze estimation that

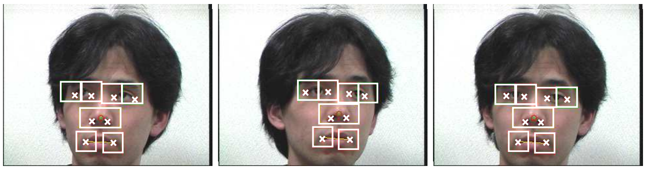

Figure 3: *Examples of facial-feature extraction (white boxes denote searching area of each facial feature, ×s denote extracted facial features.)*

is performed once before gaze estimation.

3.2.1 Face Model Estimation

First, we capture N images $I_i (i = 1, ..., N)$ while users arbitrarily change their head poses and gaze direction. As described above, users do not need to look at specific reference points or be aware of the calibration process.

Next, we extract facial-feature points from the N images with above mentioned Lucas-Kanade method. We assume M facial-feature points $p_j (j = 1, ..., M)$ are obtained. Let $\boldsymbol{x}_j^{(i)}$ be the 2D positions of feature points, $\bar{\boldsymbol{x}}^{(i)}$ be the average position of M features at image I_i $\left(\bar{\boldsymbol{x}}^{(i)} = \frac{1}{M}\sum_{j=1}^{M} \boldsymbol{x}_j^{(i)}\right)$, and $\boldsymbol{W}$ be

$$\boldsymbol{W} = \begin{bmatrix} \boldsymbol{x}_1^{(1)} - \bar{\boldsymbol{x}}^{(1)} & \cdots & \boldsymbol{x}_M^{(1)} - \bar{\boldsymbol{x}}^{(1)} \\ \vdots & & \vdots \\ \boldsymbol{x}_1^{(N)} - \bar{\boldsymbol{x}}^{(N)} & \cdots & \boldsymbol{x}_M^{(N)} - \bar{\boldsymbol{x}}^{(N)} \end{bmatrix}. \quad (2)$$

Then $\boldsymbol{W}$ can be decomposed as follows using the factorization method [Poelman and Kanade 1992]:

$$\begin{aligned} \boldsymbol{W} &= \boldsymbol{M}\boldsymbol{S} \\ &= \begin{bmatrix} \boldsymbol{m}_1 \\ \vdots \\ \boldsymbol{m}_N \end{bmatrix} \begin{bmatrix} \boldsymbol{s}_1 & \cdots & \boldsymbol{s}_M \end{bmatrix}, \end{aligned} \quad (3)$$

where $\boldsymbol{M}$ includes user pose change information, and $\boldsymbol{m}_i$ are 2×3. $\boldsymbol{S}$ denotes 3D positions of the facial features (face model), and $\boldsymbol{s}_j$ are 3×1. Then the relationship between the 2D feature positions $\boldsymbol{x}_j^{(i)}$ and the face model $\boldsymbol{S}$ can be expressed as

$$\boldsymbol{x}_j^{(i)} = \boldsymbol{m}_i \boldsymbol{s}_j + \bar{\boldsymbol{x}}^{(i)}. \quad (4)$$

Since $\boldsymbol{m}_i$ can be regarded as weak perspective projection matrices, $\boldsymbol{R}_i$ as the face poses in the image I_i can be obtained as follows by using QR decomposition [Quan 1996].

$$\boldsymbol{m}_i = \boldsymbol{A} \begin{bmatrix} \boldsymbol{r}_{i,1} \\ \boldsymbol{r}_{i,2} \end{bmatrix} \quad (5)$$

$$\boldsymbol{R}_i = \begin{bmatrix} \boldsymbol{r}_{i,1} \\ \boldsymbol{r}_{i,2} \\ \boldsymbol{r}_{i,1} \times \boldsymbol{r}_{i,2} \end{bmatrix} \quad (6)$$

where $\boldsymbol{r}_{i,1}$ and $\boldsymbol{r}_{i,2}$ are 1×3 vectors and $\boldsymbol{A}$ is 2×2 matrix.

$\boldsymbol{t}_i$, face position in I_i can be estimated as:

$$\boldsymbol{t}_i = \boldsymbol{A}^{-1} \bar{\boldsymbol{x}}_i. \quad (7)$$

Then, face model $\boldsymbol{S}$, the 2D facial-feature positions, $\boldsymbol{x}_i^{(j)}$, and face transitions and poses $\boldsymbol{t}_i, \boldsymbol{R}_i$ in image $I_i (i = 1, ..., N)$ can be estimated. They are used in the following process.

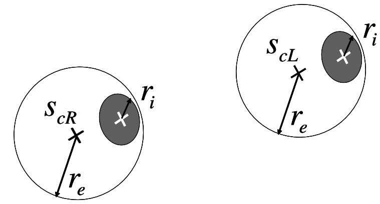

Figure 4: *Eye model*

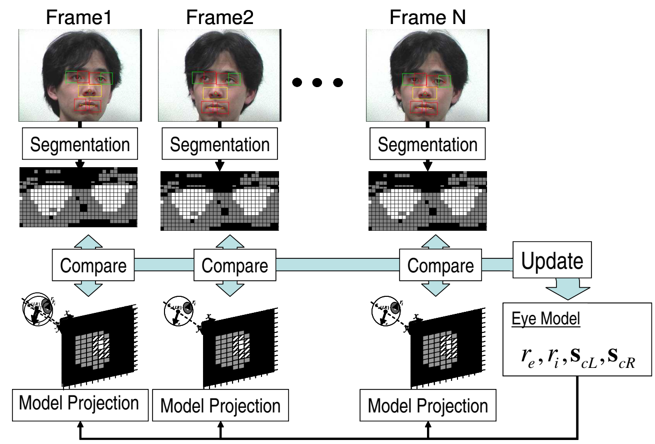

Figure 5: *Process flow of the eye model estimation*

3.2.2 Eye Model Estimation

For eye model estimation process, we employ identical image sequences $I_i (i = 1, ..., N)$ used in the face model estimation. Eye model parameters are estimated by using the 2D facial features $\boldsymbol{x}_i^{(j)}$, face positions and poses $\boldsymbol{t}_i, \boldsymbol{R}_i$.

We assume the eyeball is the sphere of radius r_e, the iris is the circle plane of radius r_i that moves along the eyeball surface (Figure 4). Let $\boldsymbol{s}_{cL}$ and $\boldsymbol{s}_{cR}$ be the 3D positions of the eyeball in the face model coordinates, the eye model parameters that we estimate in this section can be expressed as $\boldsymbol{\xi} = [r_e, r_i, \boldsymbol{s}_{cL}, \boldsymbol{s}_{cR}]$.

Let eye model parameters $\boldsymbol{\xi}$ and gaze directions in each frame $\boldsymbol{\theta} = [(\alpha_1, \beta_1), ..., (\alpha_N, \beta_N)]$ be known, the eye model can be projected onto image plane. When the true values of the $\boldsymbol{\xi}$ and $\boldsymbol{\theta}$ are given, the projected eye-model image should correspond to the actual observed image I_i. Therefore, optimal eye-model parameters

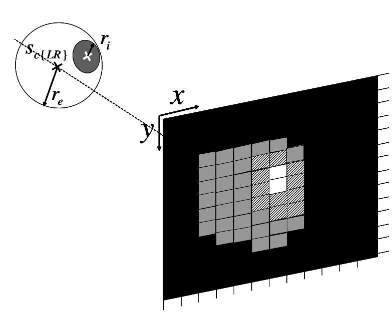

Figure 6: *Eye model projection (grey: sclera, white: iris, shaded: boundary region)*

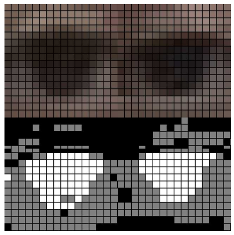

Figure 7: *Segmentation results of eye region (black: skin region, gray: sclera, white: iris)*

can be determined by searching eye-model parameters that satisfy the image constraints with nonlinear optimization (Figure 5).

First, we perform the following process for the actual N images. Based on the detected facial-feature positions and pixel colors and brightness, we segment the eye-region images into three regions: iris, sclera, and skin. Here u_x denote the segmentation results at pixel $\boldsymbol{x}$.

$$u_x^{(i)} = \begin{cases} 0 & \text{(skin region)} \\ 1 & \text{(iris region)} \\ -1 & \text{(sclera region)} \end{cases} \tag{8}$$

Figure 7 shows an example of segmentation results. We also locate the 2D positions of the iris center $\boldsymbol{x}_{iris\{LR\}}^{(i)}$ by fitting an ellipse to the segmented iris region.

Next, we determine the initial values of the eye model parameters $\boldsymbol{\xi}$. For the eyeball centers, the rough positions of the eyeball centers are estimated based on RANSAC using images in which users are looking in the camera direction. The initial values of the eyeball and iris radii are determined based on the anatomical model, and the initial gaze directions in each image are determined based on the initial eyeball centers and initial iris positions.

Using these values as the initial values for the nonlinear optimization, the optimal eye model parameters $\hat{\boldsymbol{\xi}}$ can be estimated by maximizing the next equation:

$$\left[\hat{\boldsymbol{\xi}}, \hat{\theta}_i\right] = \arg\max_{\xi,\theta} \sum_i \sum_x u_x^{(i)} \cdot v_x^{(i)} y \tag{9}$$

Here $v_x^{(i)}$ denote which region is projected into pixel $\boldsymbol{x}$ in the projected eye-model image,

$$v_x^{(i)} = B_x - W_x. \tag{10}$$

where B_x and W_x are areas of the iris and sclera regions projected in pixel $\boldsymbol{x}$, respectively. $v_x^{(i)}$ becomes

$$\begin{cases} v_x^{(i)} = -1 & \text{in iris region} \\ v_x^{(i)} = 1 & \text{in sclera region} \\ -1 < v_x^{(i)} < 1 & \text{in boundary region} \end{cases} \tag{11}$$

The optimal eye model parameters $\hat{\boldsymbol{\xi}}$ can be determined by maximizing Eq. (9).

3.3 Gaze Estimation

Gaze directions in the newly observed images are determined as follows.

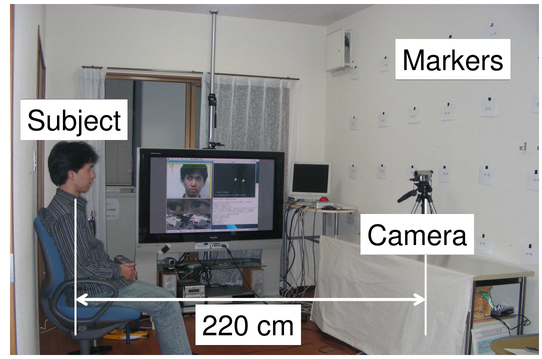

Figure 8: *Experimental environment*

First, we track the facial features and locate the iris centers from image I_t taken at time t. Let $\boldsymbol{x}_1^{(t)}, ..., \boldsymbol{x}_M^{(t)}$ be the obtained facial-feature positions. Let $\boldsymbol{x}_{irisL}^{(t)}$ and $\boldsymbol{x}_{irisR}^{(t)}$ be the 2D iris centers. Equation (4) can be transformed as:

$$\boldsymbol{x}_j^{(t)} = \boldsymbol{P}_t \begin{bmatrix} \boldsymbol{X}_j \\ 1 \end{bmatrix}. \tag{12}$$

Here $\boldsymbol{P}_t$ can be calculated when more than four facial features are observed. ($\boldsymbol{P}_t$ is 2×4 matrix). Then, the 2D positions of the eyeball centers in I_t, $\boldsymbol{x}_{cL}^{(t)}$, and $\boldsymbol{x}_{cR}^{(t)}$ can be calculated using the 3D positions of eyeball centers $\boldsymbol{s}_{c\{LR\}}$ and $\boldsymbol{P}^{(t)}$ as:

$$\begin{aligned} \boldsymbol{x}_{cL}^{(t)} &= \boldsymbol{P}_t \boldsymbol{s}_{cL} \\ \boldsymbol{x}_{cR}^{(t)} &= \boldsymbol{P}_t \boldsymbol{s}_{cR}. \end{aligned} \tag{13}$$

Gaze direction (angles between the camera's optical axis and gaze direction) can be estimated as the following equations:

$$\begin{aligned} \alpha_t &= \sin^{-1}\left(\frac{x_{iris}^{(t)} - x_c^{(t)}}{r^{(t)}}\right) \\ \beta_t &= \sin^{-1}\left(\frac{y_{iris}^{(t)} - y_c^{(t)}}{r^{(t)}}\right), \end{aligned} \tag{14}$$

where α_t and β_t are x and y axial components of the gaze direction and eyeball radius $r^{(t)}$ can be calculated as $r^{(t)} = a_1 \cdot r_e$.

4 Experiments

To confirm the effectiveness of the proposed method, we performed the following experiments whose environment is shown in Figure 8. The camera (Panasonic NV-GS200K, captured image size is 320×240) was set in front of the subjects at 220 [cm]. The size of the eye regions in the images is about 30×15 [pixel]. The 28 visual markers are arranged, as shown in Figure 9, for evaluating gaze estimation accuracy. Here, the 3D head positions of the users were estimated in advance. The distance between the subjects and the markers was 240 [cm].

We evaluated estimation accuracy using five subjects: three men, subjects 1-3; one woman, subject 4, without glasses; and one man with glasses, subject 5. In the experiment, first, we captured 50 images for face/model estimation and estimated the face/eye model of each subject. The processing time for model estimation was about 3 [sec]. Then the subjects sequentially looked at each marker for

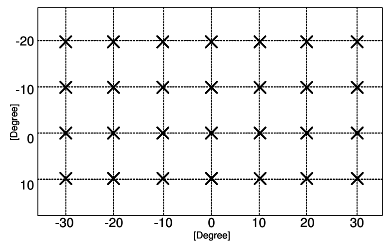

Figure 9: *Marker arrangement*

Table 1: *Gaze estimation errors (average)*

	Horizontal [°]	Vertical [°]
Subject 1	3.7	8.7
Subject 2	4.6	6.1
Subject 3	8.7	9.1
Subject 4	3.8	7.8
Subject 5	5.6	6.6
Average	5.3	7.7

five seconds. Gaze directions were estimated by our method using 30 frames. Processing speed for gaze estimation was about 10 [frame/sec] (PC: Intel Core2 Duo 2.66GHz). Table 1 shows the gaze estimation accuracy of each subject. Figures 12-14 show examples of the gaze estimation results of subjects 1 (lines denotes estimated gaze direction).

As seen, the average of the estimation errors is about 5° horizontally and about 7° vertically (5° corresponds to about 20 [cm] on the marker plane). Gaze direction can be properly estimated when the subject's head poses are changed. The proposed method can detect gaze direction even when users are wearing glasses according to the results of the subject 5 (with glasses).

Figures 10 and 11 show the averages of the estimation errors at each marker (averages of the 150 (30×5) frames). Gaze estimation errors increase when the vertical angles between the gaze direction and the camera's optical axis are increased because reducing the accuracy of the iris center estimation due to the areas of iris region occluded by the eyelid become large.

5 Applications based on Remote Gaze-tracking

Our remote gaze-tracking method enables to detect the user's gaze without any attachment and any calibration. These features are applicable for estimating user attention and targets of interest in daily-life by using user's gaze information and offering proper information/contents for each user[Utsumi et al. 2004]. Moreover, the features enable to provide a gaze-reactive robot for evoking communication for people who have lost the desire to communicate due to dementia or trauma [Yonezawa et al. 2007b; Yonezawa et al. 2007a]. In "Gazecoppet" system (Figure 15), a stuffed-toy robot reacts gaze-communicative interaction expressing its *gazing behaviors* as if looking at some object. Corresponding to the gaze-tracking results, the robot reacts in the following two ways: i) joint attention, which means "looking at the same object" expressed by facing the direction when the user looks around the robot; and ii) eye-contact reaction by voices and gestures when the user gazes at the robot[Yonezawa et al. 2007a]. The system decides which reaction the robot should manifest and synthesizes the motion and/or voice of the robot corresponding to the reaction type. The free and ambient gaze-tracking method is effective when the interaction design needs naturalness without any attachments. Thus the robot reacts to the user gaze as if communication exists between the human and the robot; the gaze-communicative robot is capable of natural and tentative communication by the ambient gaze-tracking like our proposed method.

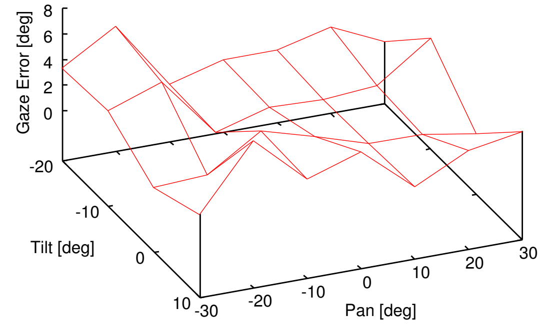

Figure 10: *Estimation error at each marker (horizontal)*

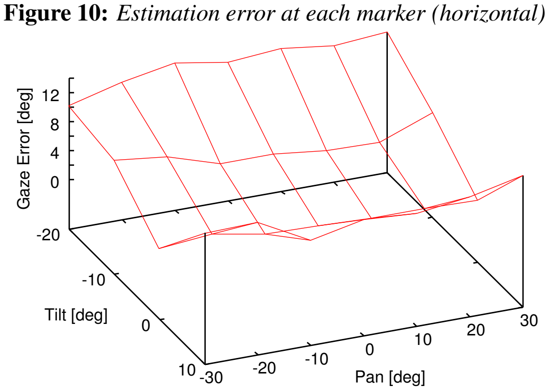

Figure 11: *Estimation error at each marker (vertical)*

6 Conclusion

In this paper, we proposed a remote gaze estimation method based on facial-feature tracking using a single video camera. The proposed method requires no special user action for eye/head model estimation, such as looking at reference points. This simple calibration algorithm can be realized by combining 3D eyemodel-based and circle-based methods (geometrical-based and texture-based eyemodels). Then gaze directions are determined as 3D vectors connecting the eyeball and the iris centers. Experimental results show that gaze estimation errors are 5° horizontally and 7° vertically. These results suggest that our method can estimate gaze direction with satisfactory accuracy to estimate user gazing targets in indoor scenes. Since our method is only based on a single normal camera, we can easily expand the gaze detection area with a pan-tilt camera. In addition, our method can be applied to situations in which users are farther from the camera if enough high resolution images can be obtained. Therefore, our method can be applied to such various applications as gaze-based interactive signboards, communication evocation using gaze motion, etc.

Future works include improving gaze estimation accuracy by employing a more accurate facial-feature detection/tracking method. We will also investigate sequential methods for face/eye model estimation and a new gaze estimation model for more accurate gaze estimation.

Figure 12: *Gaze estimation results ($\alpha = -20^\circ, \beta = -20^\circ$)*

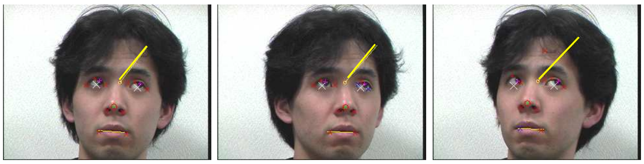

Figure 13: *Gaze estimation results ($\alpha = 20^\circ, \beta = -20^\circ$)*

Figure 14: *Gaze estimation results ($\alpha = 0^\circ, \beta = -10^\circ$)*

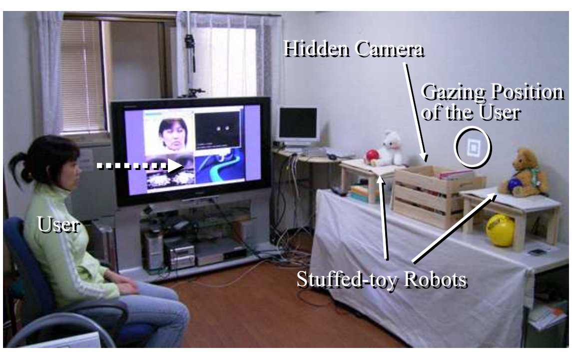

Figure 15: *"GazeCoppet": Gaze-communication in Ambient Space*

Acknowledgements

This research was supported in part by the National Institute of Information and Communications Technology (NICT).

References

BALUJA, S., AND POMERLEAU, D. 1994. Non-intrusive gaze tracking using artificial neural networks. Tech. Rep. CMU-CS-94-102, CMU.

ISHIKAWA, T., BAKER, S., MATTHEWS, I., AND KANADE, T. 2004. Passive driver gaze tracking with active appearance models. In *Proc. 11th World Congress on Intelligent Transportation Systems*.

KAWATO, S., TETSUTANI, N., AND HOSAKA, K. 2005. Scale-adaptive face detection and tracking in real time with ssr filters and support vector machine. *IEICE Trans. on Info. and Sys. E88-D*, 12, 2857–2863.

LUCAS, B., AND KANADE, T. 1981. An iterative image registration technique with an application to stereo vision. In *Proc.Int'l Joint Conf. Aretificial Intelligence*, 674–679.

MATSUMOTO, Y., AND ZELINSKY, A. 2000. An algorithm for real-time stereoo vision implementation of head pose and gaze direction measurement. In *Proc. Int. Conf. Automatic Face and Gesture Recognition*, 499–504.

MIYAKE, T., HARUTA, S., AND HORIHATA, S. 2002. Image based eye-gaze estimation irrespective of head direction. In *Proc. IEEE Int. Symp. Industrial Electronics*, vol. 1, 332–336.

MORENCY, L. P., CHRRISTOUDIAS, C., AND DARRELL, T. 2006. Recognizing gaze aversion gestures in embodied conversational discourse. In *Proc. ICMI'06*, 287–294.

MORIMOTO, C. H., AND MIMICA, M. R. M. 2005. Eye gaze tracking techniques for interactive applications. *Computer Vision and Image Understanding 98*, 1, 4–24.

OHNO, T., MUKAWA, N., AND KAWATO, S. 2003. Just blink your eyes: A head-free gaze tracking system. In *Proc. CHI2003*, 950–951.

ONO, Y., OKABE, T., AND SATO, Y. 2006. Gaze estimation from low resolution images. In *Proc. IEEE Pacigic-Rim Symp. on Image and Video Technology(PSIVT'06)*, 178–188.

POELMAN, C., AND KANADE, T. 1992. A paraperspective factorization method for shape and motion recovery. Tech. Rep. 92-208, CMU-CS.

QUAN, L. 1996. Self-calibration of an affine camera from multiple views. *Int'l Journal of Computer Vision 19*, 93–105.

SHIELE, B., AND WAIBEL, A. 1995. Gaze tracking based on face-color. In *Proc. Int'l Workshop on Automatic Face and Gesture Recognition*, 344–349.

TAKEGAMI, T., GOTOH, T., AND OHYAMA, G. 2002. An algorithm for an eye tracking system with self-calibration. *Systems and Computers in Japan 33*, 10, 10–20.

UTSUMI, A., KAWATO, S., SUSAMI, K., KUWAHARA, N., AND KUWABARA, K. 2004. Face-orientation detection and monitoring for networked interaction therapy. In *Proc. of SCIS & ISIS 2004*.

WANG, J., AND SUNG, E. 2001. Gaze determination via images of irises. *Image and Vision Computing 19*, 12, 891–911.

WU, H., CHEN, Q., AND WADA, T. 2004. Conic-based algorithm for visual line estimation from image. In *Proc. Automatic Face and Gesture Recognition*, 260–265.

YONEZAWA, T., YAMAZOE, H., UTSUMI, A., AND ABE, S. 2007. Gaze-communicative behavior of stuffed-toy robot with joint attention and eye contact based on ambient gaze-tracking. 140–145.

YONEZAWA, T., YAMAZOE, H., UTSUMI, A., AND ABE, S. 2007. Gazecoppet: Hierarchical gaze-communication in ambient space.