IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received January 20, 2017, accepted March 7, 2017, date of publication August 7, 2017, date of current version March 15, 2018.

Digital Object Identifier 10.1109/ACCESS.2017.2734168

Unconstrained and Calibration-Free Gaze Estimation in a Room-Scale Area Using a Monocular Camera

KIMIMASA TAMURA, RAN CHOI, AND YOSHIMITSU AOKI, (Member, IEEE)
Graduate School of Integrated Design Engineering, Keio University, Kanagawa 223-8522, Japan

Corresponding author: Kimimasa Tamura (kimimasa.t@gmail.com)

ABSTRACT Gaze estimation using monocular cameras has significant commercial applicability, and many studies have been undertaken on head pose-invariant and calibration-free gaze estimation. The head positions in existing data sets used in these studies are, however, limited to the vicinity of the camera, and methods trained on such data sets are not applicable when subjects are at greater distances from the camera. In this paper, we create a room-scale gaze data set with large variations in head poses to achieve robust gaze estimation across a broader range of widths and depths. The head positions are much farther from the camera, and the resolution of the eye image is lower than in conventional data sets. To address this issue, we propose a likelihood evaluation method based on edge gradients with dense particles for iris tracking, which achieves robust tracking at low-resolution eye images. Cross-validation experiments show that our proposed method is more accurate than conventional methods on all the individuals in our data set.

INDEX TERMS Gaze estimation, iris tracking, particle filter, regression.

I. INTRODUCTION

As the price of cameras has decreased and the performance of computers continues to increase, more consumer-grade devices have been equipped with monocular cameras. Such cameras are used for many applications, i.e., video calls using laptop PCs, next-generation controllers for video games, natural user interfaces, and digital signage, and are widely employed for image recognition and human–computer interaction (HCI). If a gaze estimation function could be integrated into such built-in monocular cameras without additional hardware, they would have many applications. Gaze estimation is a key factor in determining user intent and interest; thus, it can be used for next-generation UIs and marketing analysis [1]. In addition, gaze estimation technology is expected to be applied to automatic vehicles. The estimation of driver and pedestrian gaze is helpful in improving automatic driving, because eye contact is an important element in driving situations.

To realize such applications at lower cost and complexity, we propose a novel gaze estimation system that achieves sufficient accuracy in real time. Our aim is to meet the following conditions. The proposed system should (1) work with only a single monocular camera, (2) be operable with unknown users, and (3) remain consistent under varied conditions, e.g., low-resolution eye images, unconstrained user postures, different locations, and varying distances from the camera.

Various gaze estimation methods have been proposed. In recent years, learning with a large-scale dataset has enabled user-specific calibration-free point of gaze (PoG) estimation with a monocular camera. However, the head positions in conventional datasets are constrained within a limited area, which considers only the region near to the camera, so that those systems have not been evaluated with images taken from more varied distances. In addition, the requirement of high-resolution eye images also limits the practical applicability.

In this paper, we propose a gaze estimation method that comprises a model-based iris tracker that is robust to the scarcity of information brought about by the high distance from the camera. We further propose a new gaze dataset, called the ''Room-scale Gaze Dataset,'' (RSGD) to achieve PoG estimation in a room-scale space without user-specific calibration. The remainder of this paper is organized as follows. In section II, we briefly review related studies on gaze estimation. Section III describes the dataset proposed in this paper. Section IV explains the theory of iris tracking and PoG estimation, before section V summarizes the experiments and results.

II. RELATED WORK

Many gaze estimation methods have been proposed [2]. These can be broadly divided into methods using an infrared light camera and those using a visible light camera.

A. METHODS USING INFRARED CAMERAS

In methods using infrared cameras [3]–[5], the vector between the reflection point on the cornea (Purkinje image) and the center point of the pupil is mapped to the PoG after some calibration processes. This method has two advantages. First, a Purkinje image is a suitable reference point because the eyeball is approximately spherical; therefore, the Purkinje image does not move when the gaze angle changes. Second, the pupil is easily observed by infrared cameras, because the iris reflects infrared light well. The iris and pupil are difficult to distinguish in visible light cameras. Generally, infrared methods have high accuracy. However, they require a high-resolution eye image and special equipment, such as infrared lights and cameras. In addition, the subject must be close to the camera, i.e., within the reachable range of the infrared light.

B. METHODS USING VISIBLE LIGHT CAMERAS

Methods using visible light cameras work with inexpensive and readily available cameras, and can be categorized as appearance-based and model-based methods.

1) APPEARANCE-BASED METHODS

Appearance-based methods directly use an eye image as input, and then estimate the PoG through machine learning. For the learning process, methods using adaptive linear regression [6], support vector regression [3], Gaussian process regression [7], and convolutional neural networks (CNN) [8], [9] have been proposed.

Generally, appearance-based methods are more robust to low-resolution eye images than model-based methods [9]. In contrast, previous studies on appearance-based methods [10]–[13] were easily influenced by head pose and environmental light changes. Later, some appearance-based methods used the head pose information obtained from facial feature point tracking to achieve head pose invariance [14]–[16]. Lu *et al.* [17] compensated the gaze biases caused by head pose changes.

The requirement of large user-specific training datasets (in other words, calibration) is an issue in appearance-based methods. To reduce the burden of calibration, Sugano *et al.* [18] proposed an automatic online calibration technique under the assumption that the position selected by PC users with the mouse was the correct PoG. However, hundreds of individual samples were required to achieve sufficient accuracy, and so an unknown user's gaze cannot be estimated instantly. Zhang *et al.* [9] collected the MPIIGaze dataset, which contains a large number of images of laptop users looking at on-screen markers in daily life. They trained a CNN using the dataset, then achieved person- and head pose-independent gaze estimation in the wild. However, the computational cost of a CNN is very high and requires a discrete GPU for real-time tracking. In addition, the targets of previously proposed datasets [9], [19]–[22] considered PC or tablet users; therefore, the area of the head position was necessarily close to the camera, as explained in section II-B4.

2) MODEL-BASED METHODS

Model-based methods estimate gaze by fitting face and eye models to the input image. As these methods use human anatomical features such as faces and eyeballs, simple parameters can describe the gaze state without a large amount of person-specific training data.

Early model-based methods [23], [24] estimate gaze direction from the shape of the iris, that are called "circle algorithm." An ellipse is fitted to the observed iris, and then the gaze is estimated from the ellipse parameters. These methods only work with an eye image; however, they require relatively high-resolution eye images.

Later model-based approaches use 3D eyeball models [25], [26]. In these approaches, the gaze direction is defined as the vector from the eyeball center to the iris center. Kitagawa *et al.* [27] employed the eyeball model with eyelids, but this requires manual annotation of the eye corners. The authors in [28] used tracked facial feature points to estimate 3D gaze vectors; however, this requires accurately detected eye corners and one-time calibration.

The advantage of model-based methods is that they consider head pose changes more effectively than appearance-based methods, because model-based methods often utilize the head position and rotation information obtained from the face image [29].

In [30], a tracker for 3D head pose, lips, eyebrows, and irises was proposed. In [31], a method of combining the head pose and eye location information to obtain enhanced gaze estimation was proposed, but this requires calibration phases to look at known targets.

One of the issues of model-based methods is the necessity of a calibration process prior to tracking each individual user in order to determine user-specific parameters such as the accurate position of the eyeballs in the face. This limits many potential uses. Yamazoe *et al.* [32] proposed an automatic calibration method to reduce the burden on users. This approach optimizes the face and eyeball models by minimizing the projected errors between model output and images through hidden online calibration. It is remarkable that this method does not require any special calibration action. However, although short, the time required for the calibration limits the applicability of this technique. In addition, the robustness to head pose changes is not clear.

Cazzato *et al.* [33] proposed an instant calibration-free gaze estimation method by generalizing the eyeball center to 12 mm from the surface of the eye using an RGB-D sensor. Although suitable for some situations, the necessity of the depth sensor also limits the available scenes. The experiment was conducted at 70 cm from the RGB-D sensor, but the gaze estimation at farther distances was not verified.

TABLE 1. Existing gaze datasets.

	Head positions	Head rotations	Gaze targets	Illumination conditions	Participants	Images
McMurrough *et al.* [36]	1	1	16	1	20	videos
Smith *et al.* [20]	1	5	21	1	56	5 880
UT Multi-view [22]	1	8+Synthesised	160	1	50	64 000
EYEDIAP [21]	continuous (small)	continuous	continuous	1	16	videos
MPIIGaze [9]	continuous (small)	continuous	continuous	daily life	15	213 659
RSGD (ours)	continuous (large)	continuous	continuous	1	16	53 180

Baltrušaitis *et al.* [34] proposed OpenFace, an open source tool for facial behavior analysis. This toolkit tracks facial feature points and estimates gaze based on conditional local neural fields (CLNF, [35]). OpenFace achieved a state-of-the-art score using the MPIIGaze dataset, exceeding that of a CNN appearance-based method [9]. Therefore, we compare this approach with the proposed method in section V.

In general, model-based methods require high-resolution images of faces and eyes, because the alignment accuracy of facial and eye feature points is important [9]. Consequently, some low-resolution conditions, such as head positions distant from the camera, are challenging. Furthermore, the translational freedom of users relative to the camera (i.e., the available head position range) is also important for HCI applications, digital signage, and TV users. However, the above-mentioned gaze estimation methods limit the user head positions to being close to the camera (less than 1 m) or restrict the user to a preset location using a chair.

3) GAZE ESTIMATION FOR DISTANT PERSONS

As described above, gaze estimation for remote distances or large translational freedom has been challenging. The methods described in [37] and [38] use head and body pose information instead of iris tracking. The authors in [39] presented a two-camera system that detects the face from a fixed wide-angle camera to estimate a rough location for the eye region and another active pan-tilt-zoom camera to focus in on this area. This method achieved gaze estimation for distant users (approximately 4 m from the camera), however, pan-tilt-zoom cameras cannot track multiple users at the same time. In addition, users must look straight into the camera for calibration; whether such behavior occurs in real applications is unknown. In [40] and [41], an RGB-D sensor is employed for head pose invariant gaze estimation of distant users. The work of [40] proposed a geometric generative model to avoid the critical feature tracking of geometric approaches, which requires high-resolution images. Cazzato *et al.* [41] proposed gaze estimation based only on head pose information, under the assumption that the head pose can supply the gaze direction. They tested several distances (70 cm, 150 cm, and 250 cm from the camera), but did not clarify the available width range.

4) EXISTING DATASETS

Gaze datasets, which contain images of participants looking at known markers, are used for the training and evaluation of gaze estimation methods. Because existing gaze datasets were constructed for laptop or desktop PC users, the head positions of participants are close to the camera, and there is little variation in head translation. Table 1 summarizes the existing datasets.

McMurrough *et al.* [36] collected gaze data with fixed head positions. In [20], the heads of the participants are fixed on a chin mount and images are captured from five different camera positions. UT Multi-view [22] used eight cameras to capture images of 50 participants, enabling the reconstruction of 3D shapes of the eye regions. In these datasets, the heads of participants are fixed, so there no variety in the head positions.

EYEDIAP [21] used RGB and RGB-D (Kinect) sensors to capture participants gazing at markers on the display and floating objects under static and moving head motions. MPIIGaze [9] contains the gazes of laptop users obtained with high-resolution front-facing cameras under large illumination changes. Both EYEDIAP and MPIIGaze allow head position and rotation changes, but the variations in translation are small and close to the camera. The methods trained using these gaze datasets are only applicable in the vicinity of the camera.

III. DATASET

We propose RSGD (available online) to achieve unconstrained calibration-free gaze estimation across a wide area using a monocular camera. The targets of the dataset include TV and digital signage users. We collected 53180 images of 16 subjects (2 females and 14 males) looking at markers on the display. In RSGD, head translations relative to the camera are quite large. Three subjects wear glasses, and two change their facial expression (laughing). All subjects look at the markers with their natural head orientations. We train and evaluate the proposed method based on this dataset.

A. DATA COLLECTION PROCEDURE

We used a 1.2 m×0.8 m television (55” diagonal) and a Kinect v2 for data collection. The Kinect v2 was positioned centrally at the bottom of the TV. The Kinect v2 has an RGB camera (1920×1080 pixels) and a depth sensor. In this study, we only use color images from the Kinect v2 as input to the proposed method. The Kinect v2 was used because it has a cost-effective wide-angle color camera that is suitable for capturing persons across a wide area. Depth data are not used for training and prediction, but only to confirm the distribution of head positions and for some evaluations.

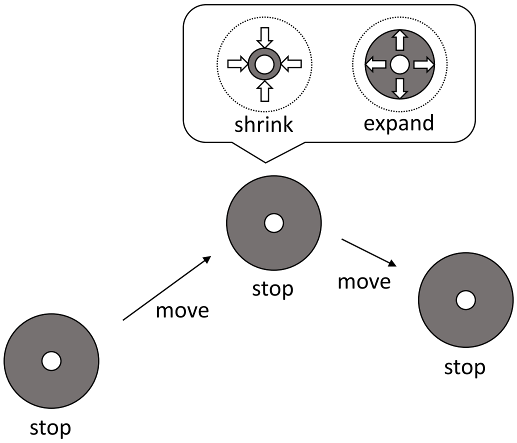


FIGURE 1. Example of the circular marker and its movement.

To collect the data, subjects were asked to sit at an arbitrary position in front of the TV and look at a circular marker (3 cm diameter) displayed on the screen. Fig. 1 shows the marker and its movement. This marker continuously moves for 1 s and then stops for 3 s. During the 3 s stationary period, the subjects are expected to look at the center of the marker. The marker shrinks and expands during the stationary period as an indication that the subjects should look at it. When the marker is stationary, three pictures are taken, but no pictures are taken while the marker is moving. The locations at which the marker stops follow a uniform distribution. After 40 stop-start cycles, the system temporarily stops capturing images and shows the message "Please change position" on the TV. The subject changes their sitting position freely, and pushes a button to resume the image capture process. This procedure is repeated for about one hour for each subject. Fig. 2 shows some images from the dataset.

B. DATASET DETAILS

In this subsection, we describe the features of RSGD.

1) WIDE VARIETY OF HEAD POSITIONS

Previous datasets were designed for laptop or desktop PC users, so the ranges of head positions were limited. As head position changes affect PoG accuracy, evaluations should consider a wide range of head positions. Fig. 3a shows the distribution of head positions in RSGD. The range of head positions in RSGD is -0.9–0.9 m in the X-axis and 0.5–2.5 m in the Z-axis. This corresponds to the available angle from the camera and covers the typical TV viewing area. In contrast,

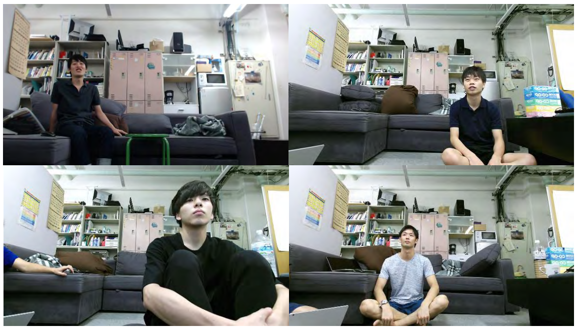


FIGURE 2. Example images from RSGD.

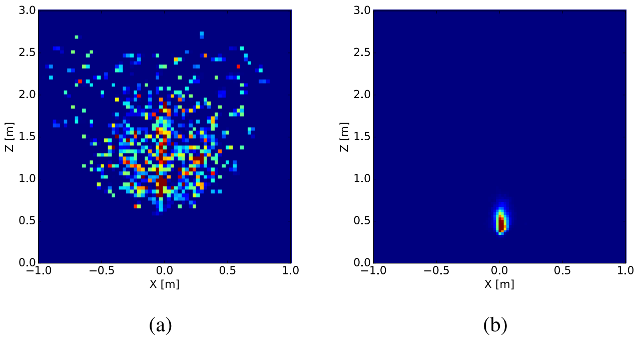


FIGURE 3. Distribution of head positions for (a) RSGD and (b) MPIIGaze.

Fig. 3b shows the distribution of head positions in MPIIGaze on the same scale as Fig. 3a. The range of head position in MPIIGaze is -0.1–0.1 m in the X-axis and 0.3–0.8 m in the Z-axis. From Fig. 3, it can be seen that RSGD has a wider head position range than MPIIGaze.

The head positions denote the translation vectors in the world coordinate system. The world coordinate system is defined as a right-handed system with the origin at the center of the camera.

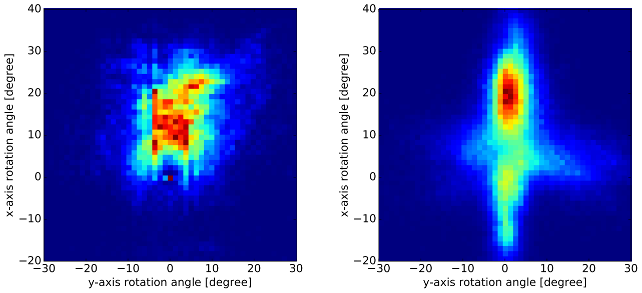
Fig. 3 shows the distribution of head rotations for RSGD and MPIIGaze. Both datasets have continuous head rotations.

2) LOW RESOLUTION OF EYE IMAGES

The resolution of eye images also affects the accuracy of gaze estimation. To evaluate the effectiveness and robustness of model-based methods in varied situations, testing with low-resolution eye images is important. Fig. 5 shows examples of eye images from RSGD and MPIIGaze. RSGD does not contain illumination variations as large as those in MPIIGaze, but has a greater variation of eye image resolutions. Here, we define "eye resolution" as the pixel distance between the inner and outer eye corners in the image, as shown in Fig. 6. Fig. 7 shows the distributions of eye resolutions in both datasets. RSGD contains more low-resolution eye images than MPIIGaze.

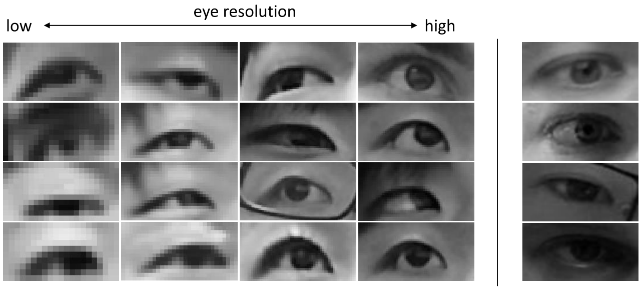
IV. PoG ESTIMATION

This section explains the process of estimating PoG from an input RGB image. The proposed method comprises head pose



(a) (b)

FIGURE 4. Distributions of head rotations for (a) RSGD and (b) MPIIGaze.



(a) (b)

FIGURE 5. Example eye images from (a) RSGD and (b) MPIIGaze.

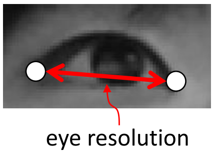


FIGURE 6. Definition of "Eye resolution" in this paper. The two white points represent the inner and outer eye corners.

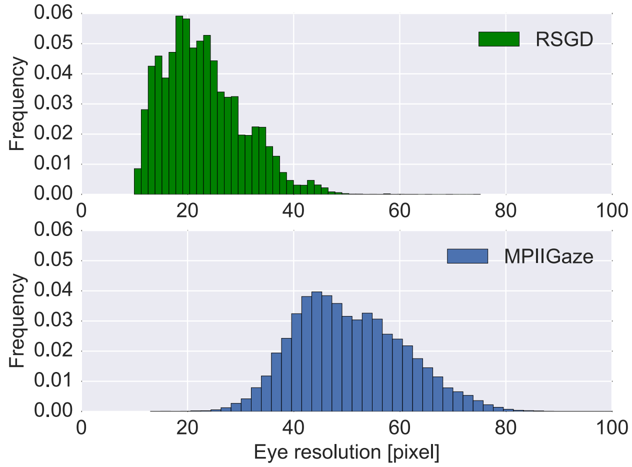


FIGURE 7. Two histograms to compare the distributions of eye resolution in RSGD and MPIIGaze.

estimation, iris tracking, and PoG regression. First, from the tracking result of facial feature points, we estimate the head position **t** and rotation **r**. In addition, the left and right eyeball centers $(\mathbf{e}_l, \mathbf{e}_r)$ are estimated from **t** and **r**. Second, based on $(\mathbf{e}_l, \mathbf{e}_r)$, iris tracking is performed in the eye region to estimate the gaze directions $(\mathbf{g}_l, \mathbf{g}_r)$. Finally, PoG **p** is estimated from **t**, **r**, $\mathbf{g}_l$, and $\mathbf{g}_r$ by a regressor, which is trained using RSGD. **t**, **r**, $\mathbf{e}_l$, $\mathbf{e}_r$, $\mathbf{g}_l$, $\mathbf{g}_r$ are three-dimensional vectors in the world coordinate system. The flow of the system is shown in Fig. 8.

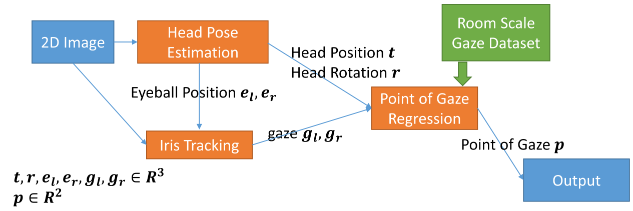

2D Image
Head Pose Estimation
Room Scale Gaze Dataset
Head Position $\boldsymbol{t}$
Head Rotation $\boldsymbol{r}$
Eyeball Position $\boldsymbol{e}_l, \boldsymbol{e}_r$
Point of Gaze Regression
Iris Tracking
gaze $\boldsymbol{g}_l, \boldsymbol{g}_r$
Point of Gaze $\boldsymbol{p}$
Output
$\boldsymbol{t}, \boldsymbol{r}, \boldsymbol{e}_l, \boldsymbol{e}_r, \boldsymbol{g}_l, \boldsymbol{g}_r \in \boldsymbol{R}^3$
$\boldsymbol{p} \in \boldsymbol{R}^2$

FIGURE 8. System flow of the proposed method.

A. HEAD POSE ESTIMATION

1) HEAD POSE IN WORLD COORDINATES

Because our aim is to achieve gaze estimation across a wide space using a monocular camera, we estimate the 3D head pose (**t** and **r**) from a 2D image. The head pose is calculated from the tracking result of facial feature points. The tracking of facial feature points is not the target of this study; therefore, we use Baltrušaitis *et al.*'s OpenFace [34].

The point distribution model (PDM) of OpenFace follows a model proposed in [42]. The PDM of OpenFace is noted in [35], expressed by eq.1:

$$\mathbf{x_i} = a\mathbf{R}_{2D}(\bar{\mathbf{x}}_i + \mathbf{V}_i\mathbf{q}) + (x, y)^T, \tag{1}$$

where $\mathbf{x_i} = (x_i, y_i)^T$ represents the ith projected facial feature point in the image, and $\bar{\mathbf{x}}_i = (\bar{x}_i, \bar{y}_i, \bar{z}_i)^T$ is the mean location of the ith feature point in a face model space. $\mathbf{V}_i$ is a $3 \times m$ principal component matrix and $\mathbf{q}$ is an m-dimensional coefficient vector that represents facial deformation. $\mathbf{R}_{2D}$ represents the first two rows of the 3×3 face rotation matrix **R**, and a is the face scale, which is the ratio of the face model space to the image space. $(x, y)^T$ is a mean face coordinate in the image.

The head position $\mathbf{t} = (X, Y, Z)$ is calculated from (a, x, y) using a perspective transform as follows:

$$X = (x - \mathrm{c_x})\frac{1}{\mathrm{f_x}}Z, \tag{2}$$

$$Y = (y - \mathrm{c_y})\frac{1}{\mathrm{f_y}}Z, \tag{3}$$

$$Z = \frac{\mathrm{f_x}}{a}, \tag{4}$$

where $(\mathrm{f_x}, \mathrm{f_y}, \mathrm{c_x}, \mathrm{c_y})$ represents the focal length and optical centers of the camera. The head rotation **r** is the Euler angle of the rotation represented by **R**.

2) EYEBALL CENTER IN IMAGE

In model-based gaze estimation, the gaze vector is defined as the line through the center of the eyeball and the iris. However, the center of the eyeball cannot be observed directly in an image. Thus, it must be estimated from the facial feature points and the head pose.

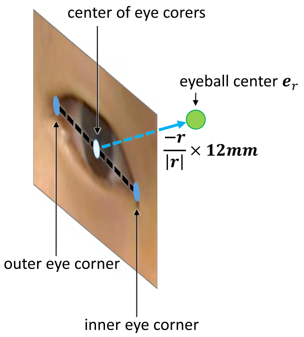


FIGURE 9. Estimation of eyeball center position.

As shown in Fig. 9, the eyeball position is set to be 12 mm from the center point of the inner and outer eye corners in the opposite direction to face rotation. The length of 12 mm is the average radius of an eyeball, as used in [33] and [40].

B. IRIS TRACKING

An overview of the iris tracking procedure is shown in Fig. 10. Iris tracking is performed using a cropped eye image.

1) EYE IMAGE CROPPING

The purpose of this process is to obtain a stable eye image from various head poses. The cropping process is as follows. The image center is defined as $(\hat{e}_{lx}, \hat{e}_{ly})$ and its cropping size is defined as $(\hat{h} \times \hat{w})$. Here, $(\hat{e}_{lx}, \hat{e}_{ly})$ denotes the 2D coordinates of the left eyeball center in the input image, which is projected from $\mathbf{e}_l$. We take $(\hat{h}, \hat{w})$ as $(k_r a, 2k_r a)$, where a is the face scale in eq. (1) and k_r is a scaling factor from the face scale a to the cropping width $\hat{w}$. Thus, $\hat{w}$ is double the distance between the inner and outer eye corners. We determined empirically that $k_r = 23.45$. After cropping, the eye image is resized to 200 × 100 pixels using bilinear interpolation. We call this resized eye image the cropped eye image. The cropped eye image has u and t axes, as shown in Fig. 11. The coordinates of the projected eyeball center in the cropped eye image are denoted as (e_{lu}, e_{lt}).

Iris tracking is composed of an initial template matching and particle filter refinement.

2) INITIAL TEMPLATE MATCHING

Initial template matching is performed to determine the rough iris position quickly. The template image is a black circle of diameter 50 pixels, which is determined empirically. The matching is performed using the normalized cross-correlation method, and the result is (u_{base}, t_{base}), which represents the coordinates of the detected iris center.

3) PARTICLE FILTER REFINEMENT

To obtain an accurate iris position, we use a particle filter. Tracking is based on the eye model shown in Fig. 11. The state vector is defined as $\mathbf{s} = (\phi, \theta, d)$. Here, ϕ and θ represent the yaw and pitch angle of eyeball rotation and d represents the radius of the iris.

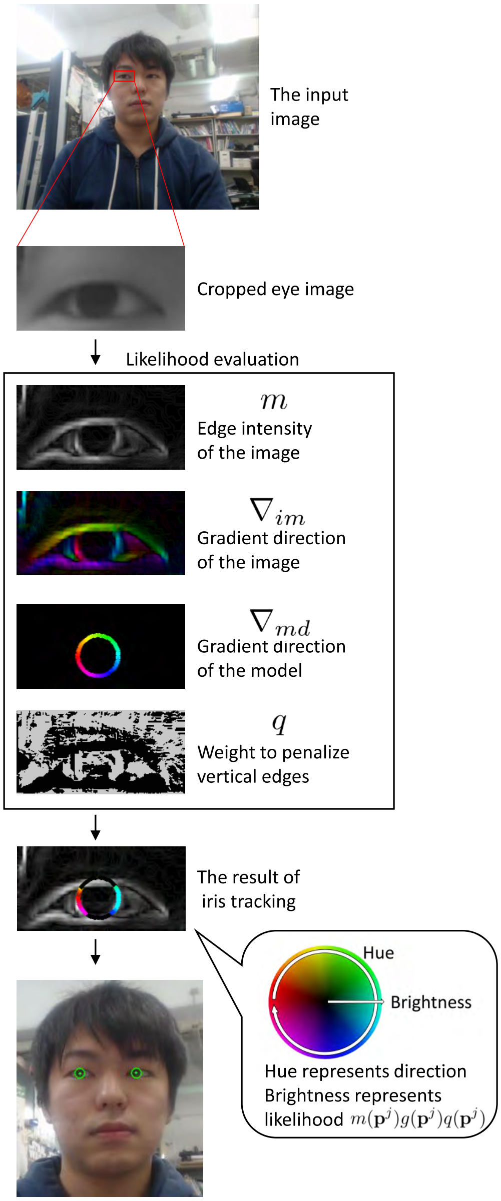

The input image
Cropped eye image
Likelihood evaluation
m
Edge intensity of the image
∇_{im}
Gradient direction of the image
∇_{md}
Gradient direction of the model
q
Weight to penalize vertical edges
The result of iris tracking
Hue
Brightness
Hue represents direction
Brightness represents likelihood $m(\mathbf{p}^j)g(\mathbf{p}^j)q(\mathbf{p}^j)$

FIGURE 10. Overview of the iris tracking process.

First, we generate random samples. For ϕ and θ, we add Gaussian noise within a range of ±5° of the rough eyeball rotation angle $(\theta_{base}, \phi_{base})$, which is calculated from (u_{base}, t_{base}) following:

$$\theta_{base} = \arcsin\left(\frac{-(t_{base} - c_{lt})}{R}\right), \tag{5}$$

$$\phi_{base} = \arcsin\left(\frac{-(u_{base} - c_{lu})}{R\cos(\theta)}\right). \tag{6}$$

As for d, we add Gaussian noise within a range of 25 ± 2 pixels. As a result, 200 state vectors $\mathbf{s}^i = (\theta^i, \phi^i, d^i)$ are generated. R represents the eyeball radius in the cropped eye image. We determined empirically that $R = 70$ pixels.

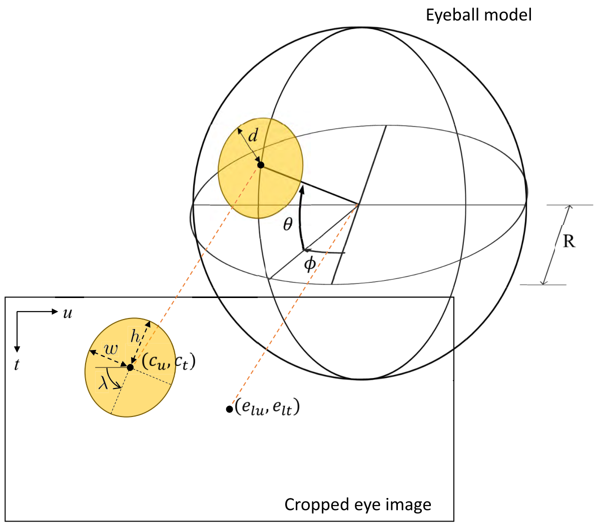


FIGURE 11. Eyeball model used in the proposed method. The relation between an eyeball state $\mathbf{s} = (\phi, \theta, d)$ and an ellipse shape in the cropped eye image is $\mathbf{o} = (c_u, c_t, h, w, \lambda)$.

Second, we spread the observation vector $\mathbf{o}^i$ as a particle based on the state vector $\mathbf{s}^i$. $\mathbf{o}^i = (c_u^i, c_t^i, h^i, w^i, \lambda^i)$ is the ith ellipse candidate in an eye image. Here, (c_u^i, c_t^i) is the ith center of the ellipse, (h^i, w^i) is the ith ellipse size, and λ^i is the ith rotation angle. The relation between $\mathbf{s}^i$ and $\mathbf{o}^i$ is given by:

$$c_u^i = -R\sin(\phi^i)\cos(\theta^i) + e_{lu}, \tag{7}$$

$$c_t^i = -R\sin(\theta^i) + e_{lt}, \tag{8}$$

$$h^i = d^i, \tag{9}$$

$$w^i = |d^i\cos(\phi^i)\cos(\theta^i)|, \tag{10}$$

$$\lambda^i = \arctan\left(\frac{\sin(\theta^i)}{\cos(\theta^i)\sin(\phi^i)}\right). \tag{11}$$

For each $\mathbf{o}^i$, a likelihood $L(\mathbf{o}^i)$ is calculated as:

$$L(\mathbf{o}^i) = \sum_{\mathbf{p}^j \in \mathbf{o}^i} m(\mathbf{p}^j)g(\mathbf{p}^j)q(\mathbf{p}^j), \tag{12}$$

$$m(\mathbf{p}^j) = \sqrt{d_u(\mathbf{p}^j)^2 + d_t(\mathbf{p}^j)^2}, \tag{13}$$

$$g(\mathbf{p}^j) = \frac{1}{(\nabla_{im}(\mathbf{p}^j) - \nabla_{md}(\mathbf{p}^j))^2 + 1}, \tag{14}$$

$$\nabla_{im}(\mathbf{p}^j) = \arctan\left(\frac{d_t(\mathbf{p}^j)}{d_u(\mathbf{p}^j)}\right), \tag{15}$$

$$\nabla_{md}(\mathbf{p}^j) = \arctan\left(\frac{t(\mathbf{p}^{j+1}) - t(\mathbf{p}^j)}{u(\mathbf{p}^{j+1}) - u(\mathbf{p}^j)}\right), \tag{16}$$

$$q(\mathbf{p}^j) = \begin{cases} 1 & \left(\begin{array}{l} |\nabla_{im}(\mathbf{p}^j)| \le 45 \text{ or} \\ |\nabla_{im}(\mathbf{p}^j) - 180| \le 45 \end{array}\right) \\ 0.01 & (otherwise), \end{cases} \tag{17}$$

where $\mathbf{p}^j$ represents the jth of 120 points equally located along the ellipse arc. $d_u(\mathbf{p}^j)$ and $d_t(\mathbf{p}^j)$ are the u-direction and t-direction differences of the pixel value at a point $\mathbf{p}^j$, respectively. $u(\mathbf{p}^j)$ and $t(\mathbf{p}^j)$ are the u and t coordinates of $\mathbf{p}^j$.

$m(\mathbf{p}^j)$ is the edge intensity at $\mathbf{p}^j$, and $g(\mathbf{p}^j)$ is the similarity of the gradient direction between the image and the model. $\nabla_{im}(\mathbf{p}^j)$ denotes the gradient direction of the image at a point $\mathbf{p}^j$, and $\nabla_{md}(\mathbf{p}^j)$ denotes the gradient direction of the model at a point $\mathbf{p}^j$. $q(\mathbf{p}^j)$ denotes a weight that penalizes edge directions $\mathbf{p}^j$ that are approximately vertical, because such edges are likely to be part of the eyelid, although we are attempting to find the iris independently from the eyelid. In other words, when the angle between the u-axis and the edge direction at $\mathbf{p}^j$ is less than or equal to 45°, $q(\mathbf{p}^j)$ takes a value of 1; otherwise, it takes a value of 0.01. These numbers were chosen empirically.

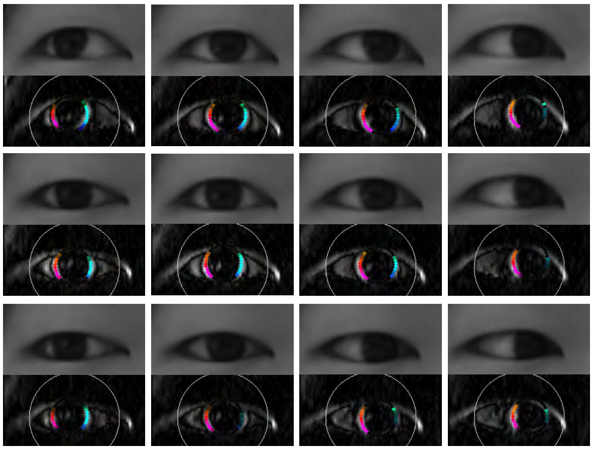


FIGURE 12. Example eye edges. Colored points represent sampled edges that have high likelihood. The color (hue) denotes the edge gradient direction and the brightness denotes the value of $m(\mathbf{p}^j)g(\mathbf{p}^j)q(\mathbf{p}^j)$.

In summary, the likelihood is high when many edges on the ellipse candidate $\mathbf{o}^i$ have strong intensity and similar directions to the slope of $\mathbf{o}^i$. Some sample eye images and edges are shown in Fig. 12. The large white circle represents the projected contour of the 3D eyeball model. The set of $\mathbf{p}^j$ is shown as points on the edges. The set of points looks like a curve because it is dense. The color (hue) of $\mathbf{p}^j$ represents the edge gradient direction and the brightness represents the value of $m(\mathbf{p}^j)g(\mathbf{p}^j)q(\mathbf{p}^j)$. As can be seen in the edge images in Fig. 12, only the edges of the iris are properly sampled.

Finally, the maximum likelihood state $\mathbf{s}^* = (\phi^*, \theta^*, d^*)$ is obtained by taking the weighted mean of all particles. The gaze direction $\mathbf{g}$ is obtained by:

$$\mathbf{g} = \begin{pmatrix} g_x \\ g_y \\ g_z \end{pmatrix} = \begin{pmatrix} 1\cos(\theta^*)\sin(\phi^*) \\ 1\sin(\theta^*) \\ -1\cos(\theta^*)\cos(\phi^*) \end{pmatrix}. \tag{18}$$

Each eye gaze direction $(\mathbf{g}_l, \mathbf{g}_r)$ is obtained as described above.

C. PoG REGRESSION

The head position $\mathbf{t}$ and the gaze directions $(\mathbf{g}_l, \mathbf{g}_r)$ were estimated as described previously; thus, the gaze ray was obtained by adding $\mathbf{t}$ and ($\mathbf{g}_l$, $\mathbf{g}_r$) geometrically. However, experiments showed that the PoG results calculated by the geometric method contained significant errors. The main reason for the errors is an inaccuracy in the estimated eyeball center positions $\mathbf{e}_l$ and $\mathbf{e}_r$. Unlike the iris, that can be observed directly in an image, the eyeball position must be estimated from the head pose and face shape. However, biased errors occur between the actual position of the eyeball center and the estimated position according to the head pose because of differences in face shapes among individuals. We discuss this in section V-A.

To compensate for this error, we propose a PoG regressor that uses the head pose ($\mathbf{t}$, $\mathbf{r}$) and the gaze directions ($\mathbf{g}_l$, $\mathbf{g}_r$) as explanatory variables to estimate PoG as a set of objective variables. We use gradient boosting regression trees (GBRT) for training, as this technique can handle mixed data types effectively. In addition, a decision tree method such as GBRT is suitable because it can learn different regression coefficients according to the head pose.

V. EXPERIMENTS

We performed several experiments to evaluate the effectiveness of the proposed method. All experiments were performed using RSGD. The error was defined as the distance on the TV screen [m] between the predicted PoG and the actual marker position. Table 2 summarizes the methods compared in the experiments.

TABLE 2. Methods compared in experiments.

Methods Name	Head Pose Est.	Iris Tracking	PoG Est.
CLNF+Geometric	OpenFace [34]	CLNF [34] [35]	Geometric
CLNF+Training	OpenFace	CLNF	GBRT
xSobel+Training	OpenFace	PF(x-Sobel) [43]	GBRT
Head	OpenFace	-	GBRT
Proposed	OpenFace	PF(edge-gradient)	GBRT

To assess generalizability, we evaluated the training methods in a leave-one-person-out test, which uses one subject for testing and the rest for training.

A. EFFECT OF TRAINING

In this subsection, we evaluate the effect of the PoG regressor proposed in this study.

The previous model-based gaze estimation method [34] outputs the head position $\mathbf{t}$ and the gaze directions $\mathbf{g_l}$ and $\mathbf{g_r}$. By combining these, the gaze vector $\mathbf{g}_{line}$ can be calculated geometrically as $\mathbf{g}_{line} = \mathbf{t} + l\frac{\mathbf{g_l}+\mathbf{g_r}}{2}$ (l is a parameter). If the position and size of the TV is known in the world coordinate system, the PoG on the TV can be obtained by calculating the intersection of the gaze vector and the TV plane. We refer to this as the [CLNF+Geometric] method.

The [CLNF+Training] method uses the head pose ($\mathbf{t}$, $\mathbf{r}$) and the gaze direction ($\mathbf{g}_l$, $\mathbf{g}_r$), similar to [CLNF+Geometric], to predict the PoG using the regression model proposed in section IV-C.

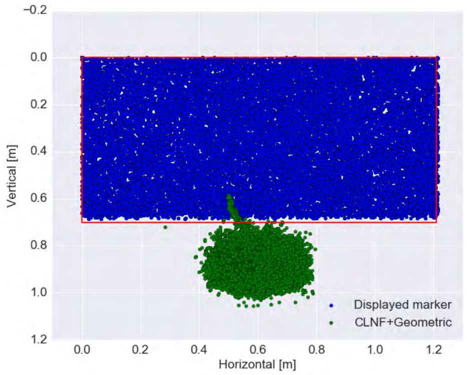
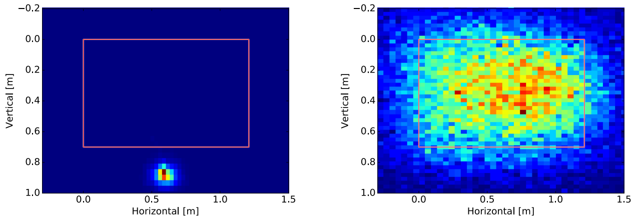


FIGURE 13. Distribution of markers and results of [CLNF+Geometric].



(a) (b)

FIGURE 14. Two histograms showing the distributions of predicted PoGs from (a) [CLNF+Geometric] and (b) [CLNF+Training].

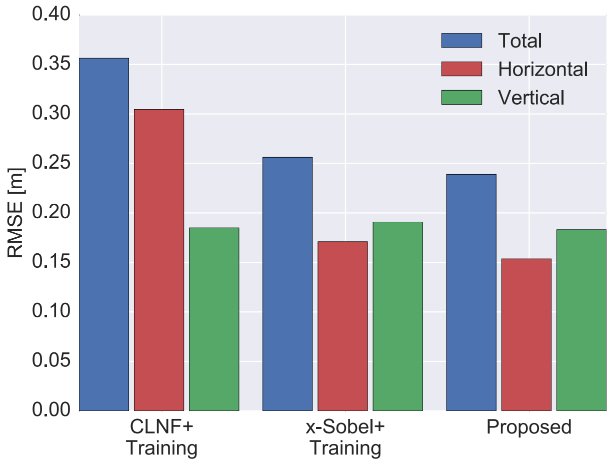


FIGURE 15. RMSEs in horizontal direction, vertical direction, and overall.

Fig. 13 shows the marker positions displayed on the TV as blue points, and shows the TV size as the red rectangle. The green points represent the PoG predicted by [CLNF+Geometric]. Note that the horizontal and vertical directions are the same in the real world. As can be seen in Fig. 13, the PoGs obtained by [CLNF+Geometric] were concentrated in the middle-bottom area of the TV. Fig. 14a shows a histogram of predicted PoGs, which indicates that the predicted PoGs lie in a very narrow range. The center point of this range is $(x, y) = (0.6 \text{ m}, 0.87 \text{ m})$, which corresponds to the installation point of the camera. This result means that the predicted gaze vectors are concentrated in the

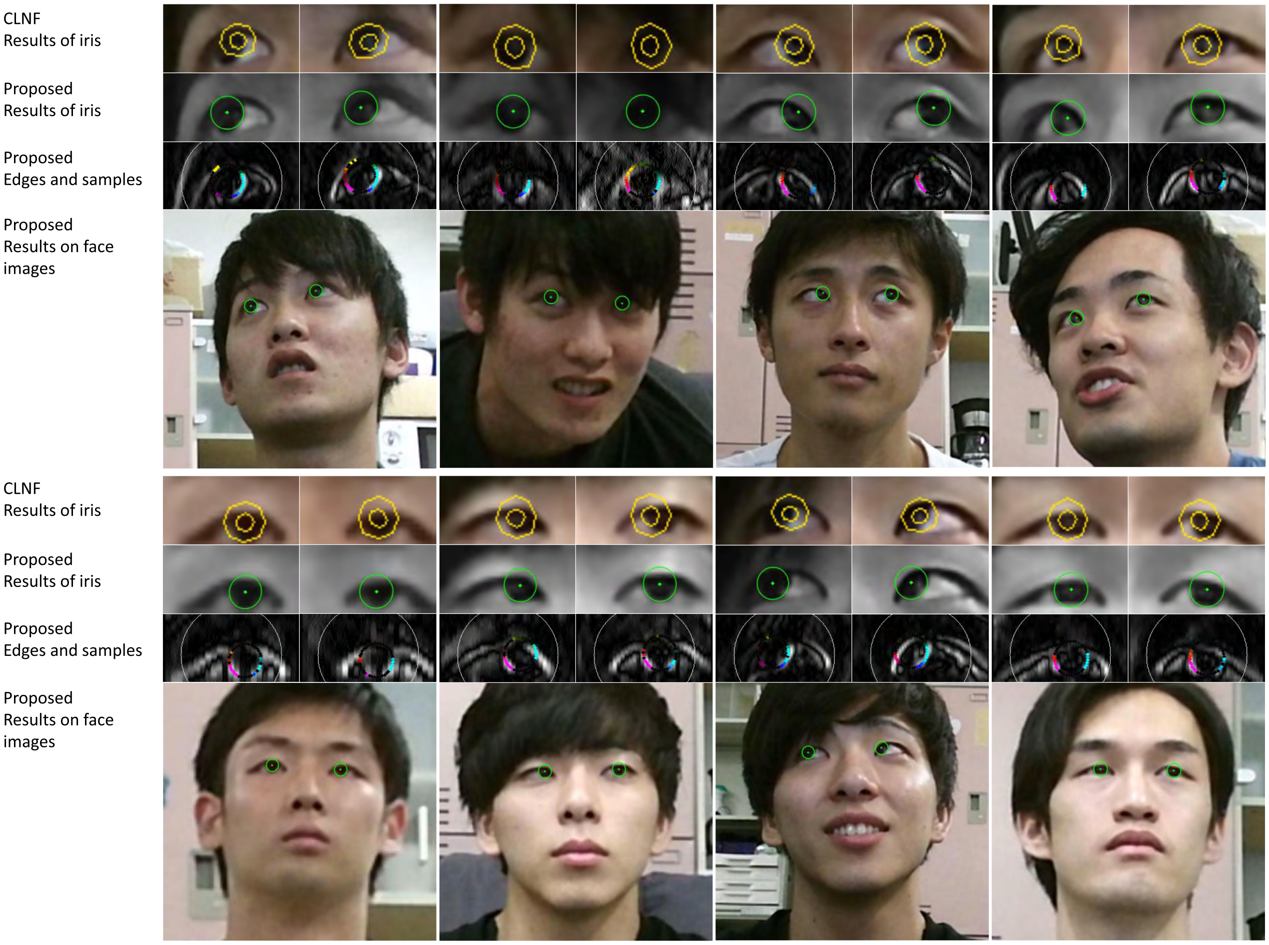


FIGURE 16. Iris tracking results given by CLNF and the proposed method using RSGD.

camera position. Therefore, we infer that the predicted eyeball center positions in the image are consistently shifted from their actual positions away from the center of the image. In other words, biased shifts occur according to the head poses.

Individual calibration is commonly used to optimize geometric methods [32]; however, it is difficult to obtain a precise result under a non-calibration condition. Employing an RGB-D sensor helps to obtain accurate eyeball center positions by generalizing the eyeball radius as 12 mm [33] and calculating the eyeball center position from the face surface. However, without depth information, accurate eyeball positions cannot be estimated.

The main aim of the proposed PoG regression using RSGD is to compensate for the error caused by the biased shift of the predicted eyeball center according to the head pose. Fig. 14b shows the histogram of the PoGs predicted in [CLNF+Training], whose range corresponds to the TV size. The root mean square error (RMSE) improved from $(x, y) = (0.38\text{ m}, 0.56\text{ m})$ in [CLNF+Geometric] to $(x, y) = (0.30\text{ m}, 0.18\text{ m})$ in [CLNF+Training].

Therefore, we believe that a decision tree-based ensemble training method like GBRT is effective for calibration-free gaze estimation across a wide range of head poses, because it optimizes different regression coefficients according to the head poses. However, [CLNF+Training] still exhibited large PoG prediction errors because of the inaccuracy of iris tracking. In the next subsection, we demonstrate the improvement given by the proposed iris tracking.

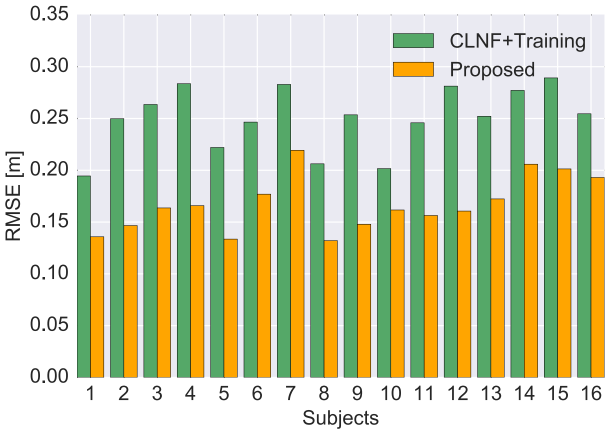


FIGURE 17. RMSE of each subject from [CLNF+Training] and [Proposed].

B. COMPARISON OF IRIS TRACKING

To evaluate the proposed iris tracking accuracy, we compared three iris tracking methods (CLNF [34], x-Sobel edge [43], and the proposed method) with head pose estimation using OpenFace and regression using GBRT. We refer to each method as [CLNF+Training], [x-Sobel+Training], and [Proposed], respectively. The CLNF method is used for comparison because it has achieved the state-of-the-art score in gaze estimation, as mentioned in section II. In the x-Sobel edge method [43], which is our previous technique, a particle filter is also used for iris tracking, but the likelihood is calculated using only the edge intensity from the Sobel filter output in the x-direction.

Fig. 15 shows the RMSE of each method in the horizontal direction, vertical direction, and overall. The total RMSEs of [CLNF+Training], [x-Sobel+Training], and [Proposed] are 0.356 m, 0.256 m, and 0.239 m, respectively. Thus, the proposed method demonstrates the highest accuracy.

In the horizontal direction, in particular, the accuracy was greatly improved. In many cases, the iris edges on the upper and lower sides are covered by the eyelid, i.e., only the edges on the left and right sides can be seen. The proposed method samples 120 points on the iris ellipse arc and estimates the gaze parameters from points that fit well with the model. Therefore, we can estimate the yaw angle very accurately from the limited left and right edges. In contrast, [CLNF+Training] only uses eight sample points as patch exports on the iris, so it is influenced by eyelid occlusion.

The CLNF-based method achieved a state-of-the-art score using the MPIIGaze dataset. However, with RSGD, it could not capture the features of the iris correctly because of noise, eyelid occlusions, and low-resolution eye images.

Fig. 16 shows example images of iris tracking results given by CLNF and the proposed method using RSGD. As can be seen in the first row, CLNF failed to track the irises in many frames because of the low-resolution of the eye images in RSGD. However, the second row shows that the proposed method tracked the irises properly, ignoring outlier edges (e.g., shadow and eyelids). It only extracted iris edges efficiently; thus, it was possible to track correctly under poor conditions.

Fig. 17 shows the RMSEs of [CLNF+Training] and [Proposed] for each subject. As can be seen, improvements were confirmed for all subjects.

C. ROBUSTNESS AGAINST DISTANCE

When the distance of the user from the camera becomes greater than around 1 m, model-based methods tend to fail because the eye image resolution is insufficient for tracking. To estimate the gaze in conditions where the eye image resolution is relatively poor, Cazzato *et al.* [41] proposed a method for estimating the gaze from the head pose information alone. Through a series of experiments, they demonstrated that the errors in their method were comparable to those in other state-of-the-art methods.

The proposed method also attempts to estimate the gaze of persons who are far from the camera (up to 2.5 m), and uses head pose information for the gaze estimation. Therefore, there is concern that the gaze is estimated simply from the head pose, and iris tracking does not have any meaning at this distance. To confirm whether iris tracking is effective at this distance, we compared the proposed method and a method using only head pose information, which is referred to as [Head]. In addition, we also compared [CLNF+Training].

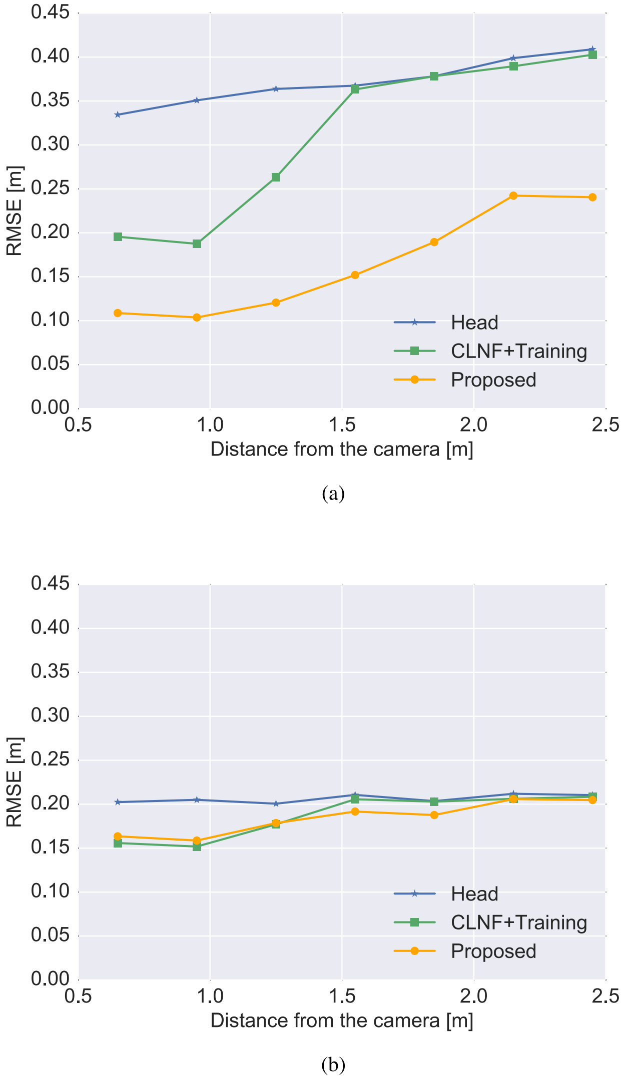


FIGURE 18. RMSEs of [Head], [CLNF+Training], and [Proposed] in the (a) horizontal and (b) vertical directions.

Fig. 18 shows the RMSE results at intervals of 0.3 m from 0.5–2.6 m from the camera. Fig. 18a shows the results in the horizontal direction, and Fig. 18b shows those in the vertical direction. The x-axis of the figures represents the distance from the camera, and the y-axis represents the RMSE.

With respect to the vertical direction, at distances greater than 1.5 m, the RMSE of each method is very similar. This indicates that the iris tracking information is not effective at this distance.

TABLE 3. Comparison with other representative methods.

Method	Head Positions [m]		Head Rotations	Category	Reported Error [°]	Subjects	Camera
Proposed	continuous (wide)	X: -0.9∼0.9 Y: -0.2∼0.5 Z: 0.5∼2.5	continuous	model	Hor.: 4.02±2.57 Ver.: 5.94±2.79 Total: 7.58±4.48	16	RGB
CLNF+Training	continuous (wide)	X: -0.9∼0.9 Y: -0.2∼0.5 Z: 0.5∼2.5	continuous	model	Hor.: 8.80±4.06 Ver.: 6.04±2.65 Total: 11.25±6.15	16	RGB
Yamazoe *et al.* [44]	1	X: 0 Y: 0 Z: 2.2	continuous	model	Hor.: 5.3 Ver.: 7.7	5	RGB
Cazzato *et al.* [41]	discrete (wide)	X: unknown Y: unknown Z: 0.7, 1.5, 2.5	continuous	model	Hor.: 4∼12 Ver.: 4.5∼8	6	RGB-D
Sugano *et al.* [18]	continuous (limited)	range of X: 0.22 range of Y: 0.05 range of Z: 0.20	continuous	appearance	Total: 4∼5	3	RGB
Lu *et al.* [14]	continuous (limited)	X: -0.1∼0.09 Y: 0∼0.07 Z: 0.54∼0.67	continuous	appearance	Total: 2∼3	7	RGB
Zhang *et al.* [9]	continuous (limited)	X: -0.1∼0.1 Y: -0.1∼0.1 Z: 0.3∼0.8	continuous	appearance	Total: 6.3	15	RGB

However, with respect to the horizontal direction, the proposed method produced lower errors not only in the vicinity of the camera but also at farther distances. It is noteworthy that the iris tracking in [CLNF+Training] failed above 1.5 m, and could not provide any further value. In contrast, the proposed iris tracking method remained effective up to 2.5 m.

Using RSGD, we could not verify the maximum distance at which the iris tracking is effective. In other words, we could not verify the intersection of the blue line [Head] and the orange line [Proposed]. However, Fig. 18a indicates the possibility that the proposed iris tracking is effective at distances greater than 2.5 m, which is a very interesting result.

D. COMPARISON WITH OTHER STATE-OF-THE-ART METHODS

In this subsection, we compare the proposed method with other representative state-of-the-art gaze estimation methods mentioned in section II. Table 3 summarizes the head pose-free methods, together with information on the variety of head positions, rotations, categories, reported errors, number of subjects, and number of cameras. As the other methods report the error in degrees, we converted our results from PoG error [m] to [degrees], and list the mean absolute error and standard deviation. The conversion used the cosine theorem in the world coordinate system. We calculated the angle between the predicted gaze vector and the vector from the head position to the displayed marker position on the TV. For this, we employed the head positions obtained from Kinect's depth measurements, which are more accurate than those of OpenFace.

The appearance-based methods [9], [14], [18] give more accurate results than the proposed method, but all verifications were performed in the vicinity of the camera (less than 1 m). In addition, the head position translations in the x-direction were very limited (less than 0.2 m). Therefore, these methods have only been verified in a very small area.

Cazzato *et al.* [41] verified the accuracy of gaze estimation at distances of 0.7 m, 1.5 m, and 2.5 m from the camera. They divided subjects into three groups and performed a comprehensive study [17]. Subjects in the first group were familiar with the system and had experience of using it. The second group was informed how the system works, but had no experience with it. The subjects in the third group were completely unaware of the system. According to this classification, our experiment involved five subjects in the second group and 11 subjects in the third group. Although we could not confirm a significant difference between the two groups using RSGD, we selected the results from the second and third groups in [41] for comparison under the same conditions (see Table 3). Although Cazzatos method requires an RGB-D sensor, the accuracy of the proposed method is similar in the vertical direction and higher in the horizontal direction. This result is consistent with the results obtained in section V-C.

Yamazoe *et al.* [44] estimated the gaze of subjects who were 2.2 m away using only a monocular camera. They used eye images of 30×15 pixels, similar to RSGD. However, the subjects were in a fixed chair in the experiment, meaning there was no variation in head positions. However, the proposed method allows subjects to sit in a wide range of arbitrary positions. In addition, the accuracy of the proposed method is higher than that reported in [44] in both the vertical and horizontal directions.

As described above, other representative gaze estimation methods restrict the user's position to the vicinity of the camera, to a preset location, or require additional hardware. In contrast, the proposed method is applicable to arbitrary head positions across a wide area using only a monocular camera. Furthermore, the accuracy is equal to or higher than that of conventional methods. Our method is a practical approach for gaze estimation applications in digital signage and smart TVs.

Regarding the processing speed, our system is implemented by a single thread of the C++ environment, and the frame rates are 20 fps at a resolution of 640×480 and 16 fps at a resolution of 1920×1080 using an Intel Core i7 3.4 GHz CPU. When implemented in a multi-thread environment, 30 fps can be achieved at both image resolutions. The processing speeds reported in [41] and [44] are 30 fps and 10 fps, respectively. Although their input data and machine specifications are different, out system works efficiently in real time.

VI. CONCLUSION

Although gaze estimation has been extensively studied, previous methods have only targeted users that are close to the camera. In this study, we achieved gaze estimation over a wide variety of positions with respect to the camera without calibration for each individual.

We provide two main contributions. The first is an iris tracking method that is robust to low eye image resolutions using dense particle sampling and improved likelihood evaluation. The second is RSGD, which contains a broader range of head positions than existing datasets. Using this dataset to train a regressor, we achieved gaze estimation in a room-scale environment.

We proved that the proposed method is more accurate than conventional methods using cross-validation, and showed that our approach is less affected by the eye image resolution and distance from the camera. The proposed method is applicable to arbitrary head positions in a wide space using only a monocular camera. Furthermore, even across a wide area, the accuracy is comparable to other state-of-the-art methods. Thus, we have developed a practical method for gaze estimation applications in digital signage and smart TVs.

In future work, we plan to verify the proposed method in various illumination environments, such as outdoors, and achieve gaze estimation at greater distances.

REFERENCES

[1] M. Wedel and R. Pieters, "A review of eye-tracking research in marketing," *Review of Marketing Research*, vol. 4. London, U.K.: Emerald Group Publishing Limited, 2008, pp. 123–147.

[2] D. W. Hansen and Q. Ji, "In the eye of the beholder: A survey of models for eyes and gaze," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 3, pp. 478–500, Mar. 2010. [Online]. Available: http://dx.doi.org/10.1109/TPAMI.2009.30

[3] Z. Zhu, Q. Ji, and K. P. Bennett, "Nonlinear eye gaze mapping function estimation via support vector regression," in *Proc. 18th Int. Conf. Pattern Recognit. (ICPR)*, vol. 1. Aug. 2006, pp. 1132–1135.

[4] T. Ohno and N. Mukawa, "A free-head, simple calibration, gaze tracking system that enables gaze-based interaction," in *Proc. Symp. Eye Tracking Res. Appl.*, 2004, pp. 115–122.

[5] A. Villanueva and R. Cabeza, "A novel gaze estimation system with one calibration point," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 38, no. 4, pp. 1123–1138, Aug. 2008.

[6] F. Lu, Y. Sugano, T. Okabe, and Y. Sato, "Adaptive Linear Regression for Appearance-Based Gaze Estimation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 10, pp. 2033–2046, Oct. 2014.

[7] B. Noris, K. Benmachiche, and A. Billard, "Calibration-free eye gaze direction detection with Gaussian processes," in *Proc. Int. Conf. Comput. Vis. Theory Appl.*, 2008, pp. 1–6.

[8] C. L. L. Jerry and M. Eizenman, "Convolutional neural networks for eye detection in remote gaze estimation systems," in *Proc. Int. MultiConf. Eng. Comput. Scientists*, vol. 1. 2008, pp. 1–6.

[9] X. Zhang, Y. Sugano, M. Fritz, and A. Bulling. (Apr. 2015). "Appearance-based gaze estimation in the wild." [Online]. Available: https://arxiv.org/abs/1504.02863

[10] D. Pomerleau and S. Baluja, "Non-intrusive gaze tracking using artificial neural networks," in *Proc. AAAI Fall Symp. Mach. Learn. Comput. Vis.*, Raleigh, NC, USA, 1993, pp. 153–156.

[11] K. Liang, Y. Chahir, M. Molina, C. Tijus, and F. Jouen, "Appearance-based gaze tracking with spectral clustering and semi-supervised gaussian process regression," in *Proc. Conf. Eye Tracking South Africa*, 2013, pp. 17–23.

[12] O. Williams, A. Blake, and R. Cipolla, "Sparse and semi-supervised visual mapping with the S^3GP," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 1. Jun. 2006, pp. 230–237.

[13] W. Sewell and O. Komogortsev, "Real-time eye gaze tracking with an unmodified commodity webcam employing a neural network," in *Proc. CHI Extended Abstracts Human Factors Comput. Syst.*, 2010, pp. 3739–3744.

[14] F. Lu, T. Okabe, Y. Sugano, and Y. Sato, "Learning gaze biases with head motion for head pose-free gaze estimation," *Image Vis. Comput.*, vol. 32, no. 3, pp. 169–179, 2014.

[15] K. A. F. Mora and J.-M. Odobez, "Gaze estimation from multimodal kinect data," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun. 2012, pp. 25–30.

[16] J. Choi, B. Ahn, J. Park, and I. S. Kweon, "Appearance-based gaze estimation using Kinect," in *Proc. 10th Int. Conf. Ubiquitous Robots Ambient Intell.*, 2013, pp. 260–261.

[17] F. Lu, T. Okabe, Y. Sugano, and Y. Sato, "A head pose-free approach for appearance-based gaze estimation," in *Proc. BMVC*, 2011, pp. 1–11.

[18] Y. Sugano, Y. Matsushita, Y. Sato, and H. Koike, "An incremental learning method for unconstrained gaze estimation," in *Proc. Eur. Conf. Comput. Vis.*, 2008, pp. 656–667.

[19] E. Wood and A. Bulling, "Eyetab: Model-based gaze estimation on unmodified tablet computers," in *Proc. Symp. Eye Tracking Res. Appl. (ETRA)*, New York, NY, USA, 2014, pp. 207–210. [Online]. Available: http://doi.acm.org/10.1145/2578153.2578185

[20] B. A. Smith, Q. Yin, S. K. Feiner, and S. K. Nayar, "Gaze locking: Passive eye contact detection for human-object interaction," in *Proc. 26th Annu. ACM Symp. User Interface Softw. Technol.*, 2013, pp. 271–280.

[21] K. A. Funes Mora, F. Monay, and J.-M. Odobez, "Eyediap: A database for the development and evaluation of gaze estimation algorithms from RGB and RGB-D cameras," in *Proc. Symp. Eye Tracking Res. Appl. (ETRA)*, New York, NY, USA, 2014, pp. 255–258. [Online]. Available: http://doi.acm.org/10.1145/2578153.2578190

[22] Y. Sugano, Y. Matsushita, and Y. Sato, "Learning-by-synthesis for appearance-based 3D gaze estimation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1821–1828.

[23] H. Wu, Q. Chen, and T. Wada, "Conic-based algorithm for visual line estimation from one image," in *Proc. 6th IEEE Int. Conf. Autom. Face Gesture Recognit. (FGR)*, Washington, DC, USA, May 2004, pp. 260–265. [Online]. Available: http://dl.acm.org/citation.cfm?id=1949767.1949816

[24] W. Zhang, T.-N. Zhang, and S.-J. Chang, "Eye gaze estimation from the elliptical features of one iris," *Opt. Eng.*, vol. 50, no. 4, p. 047003, 2011.

[25] Y. Matsumoto and A. Zelinsky, "An algorithm for real-time stereo vision implementation of head pose and gaze direction measurement," in *Proc. 4th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG)*, Washington, DC, USA, Mar. 2000, pp. 499–504. [Online]. Available: http://dl.acm.org/citation.cfm?id=795661.796234

[26] T. Ishikawa, S. Baker, I. Matthews, and T. Kanade, "Passive driver gaze tracking with active appearance models," in *Proc. 11th World Congr. Intell. Transp. Syst.*, 2004, paper CMU-RI-TR-04-08.

[27] Y. Kitagawa, H. Wu, T. Wada, and T. Kato, "On eye-model personalization for automatic visual line estimation," in *Proc. PRMU*, 2007, vol. 106. no. 469, pp. 55–60.

[28] J. Chen and Q. Ji, "3D gaze estimation with a single camera without IR illumination," in *Proc. 19th Int. Conf. Pattern Recognit.*, Dec. 2008, pp. 1–4.

[29] W.-Z. Zhang, Z.-C. Wang, J.-K. Xu, and X.-Y. Cong, "A method of gaze direction estimation considering head posture," *Int. J. Signal Process., Image Process. Pattern Recognit.*, vol. 6, no. 2, pp. 103–112, 2013.

[30] J. Orozco, O. Rudovic, J. Gonzàlez, and M. Pantic, "Hierarchical on-line appearance-based tracking for 3D head pose, eyebrows, lips, eyelids and irises," *Image Vis. Comput.*, vol. 31, no. 4, pp. 322–340, 2013.

[31] R. Valenti, N. Sebe, and T. Gevers, "Combining head pose and eye location information for gaze estimation," *IEEE Trans. Image Process.*, vol. 21, no. 2, pp. 802–815, Feb. 2012.

[32] H. Yamazoe, A. Utsumi, T. Yonezawa, and S. Ave, "Automatic calibration of 3D eye model for single-camera based gaze estimation," *Trans. IEICE*, vol. 94, pp. 998–1006, Jun. 2011.

[33] D. Cazzato, A. Evangelista, M. Leo, P. Carcagnì, and C. Distante, "A low-cost and calibration-free gaze estimator for soft biometrics: An explorative study," *Pattern Recognit. Lett.*, vol. 82, pp. 196–206, Oct. 2016.

[34] T. Baltrušaitis, P. Robinson, and L. P. Morency, "OpenFace: An open source facial behavior analysis toolkit," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2016, pp. 1–10.

[35] T. Baltrušaitis, P. Robinson, and L.-P. Morency, "Constrained local neural fields for robust facial landmark detection in the wild," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCVW)*, Sydney, NSW, Australia, Dec. 2013, pp. 354–361.

[36] C. D. McMurrough, V. Metsis, J. Rich, and F. Makedon, "An eye tracking dataset for point of gaze detection," in *Proc. Symp. Eye Tracking Res. Appl.*, 2012, pp. 305–308.

[37] N. Robertson, I. Reid, and J. Brady, "What are you looking at? Gaze estimation in medium-scale images," in *Proc. HAREM Workshop (BMVC)*, Oxford, U.K., vol. 9. 2005, pp. 1–11.

[38] S. O. Ba and J.-M. Odobez, "Recognizing visual focus of attention from head pose in natural meetings," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 39, no. 1, pp. 16–33, Feb. 2009.

[39] M. J. Reale, S. Canavan, L. Yin, K. Hu, and T. Hung, "A multi-gesture interaction system using a 3-D iris disk model for gaze estimation and an active appearance model for 3-D hand pointing," *IEEE Trans. Multimedia*, vol. 13, no. 3, pp. 474–486, Jun. 2011.

[40] K. A. F. Mora and J.-M. Odobez, "Geometric generative gaze estimation (G3E) for remote RGB-D cameras," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1773–1780.

[41] D. Cazzato, M. Leo, and C. Distante, "An investigation on the feasibility of uncalibrated and unconstrained gaze tracking for human assistive applications by using head pose estimation," *Sensors*, vol. 14, no. 5, pp. 8363–8379, 2014.

[42] J. M. Saragih, S. Lucey, and J. F. Cohn, "Deformable model fitting by regularized landmark mean-shift," *Int. J. Comput. Vis.*, vol. 91, no. 2, pp. 200–215, Jan. 2011.

[43] K. Tamura and Y. Aoki, "Eyelid and iris tracking method with novel eye models," in *Proc. IEEE/SICE Int. Symp. Syst. Integr. (SII)*, Dec. 2013, pp. 449–453.

[44] H. Yamazoe, A. Utsumi, T. Yonezawa, and S. Abe, "Remote gaze estimation with a single camera based on facial-feature tracking without special calibration actions," in *Proc. Symp. Eye Tracking Res. Appl.*, 2008, pp. 245–250.

KIMIMASA TAMURA received the M.S.Eng. degree from Keio University, Japan, in 2014. He is currently pursuing the Ph.D. degree at Keio University. His research interests include gaze estimation, tracking, human–computer interactions, and medical imaging.

RAN CHOI received the M.S.Eng. degree in information science and telecommunication from Hanshin University, South Korea, in 2014. She is currently pursuing the Ph.D. degree in engineering at Keio University. Her research interests include medical imaging and computer vision.

YOSHIMITSU AOKI received the Ph.D. degree in engineering from Waseda University in 2001. From 2002 to 2008, he was an Associate Professor with the Department of Information Engineering, Shibaura Institute of Technology. He is currently an Associate Professor with the Department of Electronics and Electrical Engineering, Keio University. He performs research in the areas of computer vision, pattern recognition, and media understanding.