

Optical Engineering

SPIEDigitalLibrary.org/oe

New method for face gaze detection in smart television

Won Oh Lee
Yeong Gon Kim
Kwang Yong Shin
Dat Tien Nguyen
Ki Wan Kim
Kang Ryoung Park
Cheon In Oh



New method for face gaze detection in smart television

Won Oh Lee,^a Yeong Gon Kim,^a Kwang Yong Shin,^a Dat Tien Nguyen,^a Ki Wan Kim,^a Kang Ryoung Park,^{a,*} and Cheon In Oh^b

^aDongguk University, Division of Electronics and Electrical Engineering, 26 Pil-dong 3-ga, Jung-gu, Seoul 100-715, Republic of Korea

^bElectronics and Telecommunications Research Institute, 218 Gajeong-ro, Yuseong-gu, Daejeon 305-700, Republic of Korea

Abstract. Recently, gaze detection-based interfaces have been regarded as the most natural user interface for use with smart televisions (TVs). Past research conducted on gaze detection primarily used near-infrared (NIR) cameras with NIR illuminators. However, these devices are difficult to use with smart TVs; therefore, there is an increasing need for gaze-detection technology that utilizes conventional (visible light) web cameras. Consequently, we propose a new gaze-detection method using a conventional (visible light) web camera. The proposed approach is innovative in the following three ways. First, using user-dependent facial information obtained in an initial calibration stage, an accurate head pose is calculated. Second, using theoretical and generalized models of changes in facial feature positions, horizontal and vertical head poses are calculated. Third, accurate gaze positions on a smart TV can be obtained based on the user-dependent calibration information and the calculated head poses by using a low-cost conventional web camera without an additional device for measuring the distance from the camera to the user. Experimental results indicate that the gaze-detection accuracy of our method on a 60-in. smart TV is 90.5%. © The Authors. Published by SPIE under a Creative Commons Attribution 3.0 Unported License. Distribution or reproduction of this work in whole or in part requires full attribution of the original publication, including its DOI. [DOI: [10.1117/1.OE.53.5.053104](https://doi.org/10.1117/1.OE.53.5.053104)]

Keywords: gaze detection; smart television; web camera; head pose; facial feature position.

Paper 140033 received Jan. 8, 2014; revised manuscript received Mar. 31, 2014; accepted for publication Apr. 2, 2014; published online May 5, 2014.

1 Introduction

In recent times, with the proliferation of digital television (TV), internet protocol (IP) TV, and smart TV, all of which provide a variety of multimedia services as well as conventional TV channels, the broadcast environment has changed significantly. A TV is no longer a passive device for receiving television broadcast signals; it has evolved into a smart device. A TV now offers multimedia services, such as video on demand, social network services, and teleconferencing, in addition to multichannel television broadcasting. The popularity of smart TV has recently increased as a result of the various functionalities it provides; however, this technology is in need of a more convenient interface design. Gesture recognition has been researched intensively using the Kinect device from the Microsoft Corporation in this regard.¹⁻³ Other research projects have proposed methods based on eye gaze-detection technology using near-infrared (NIR) light cameras and illuminators.⁴⁻⁷ However, the devices based on additional NIR cameras and illuminators are difficult to use with smart TVs owing to their large size and high cost, and using conventional (visible light) web cameras can resolve these drawbacks for gaze-detection technology. Previously, Nguyen et al.⁸ used a conventional (visible light) web camera for gaze detection with a smart TV. They measured horizontal head pose using face boundary and facial feature positions, and estimated vertical head pose using facial features and shoulder positions. In particular, the horizontal head pose was measured using the distance from the left face boundary to the left eye and from the right eye to the right face

boundary. In addition, the vertical head pose was measured using the distance between the center of the face and the shoulder, measuring the center of the face region from the average position of the eyes and the center of both nostrils. However, it is difficult to accurately detect shoulder positions because of individual variation in shoulder shapes and accurate detection of the shoulder line requires much processing time. Corcoran et al.⁹ and Zhang et al.¹⁰ proposed gaze-tracking methods that utilize visible light cameras. However, in these methods, the freedom of the user's head movement is limited and the Z distance between the user's face and the gaze-tracking system is short, which is an obstacle to such system applications to TVs. Mardanbegi and Hansen¹¹ proposed a gaze-detection method for 55-in. TVs; however, they used a wearable gaze-tracking device shaped as a pair of glasses, which may be inconvenient for users.

In order to solve these problems, we propose a new gaze-detection method. Using the user-dependent facial information obtained in an initial calibration stage, head pose can be calculated accurately. Further, the horizontal and vertical head poses can be calculated with theoretical and generalized models of the changes in facial feature position. In addition, accurate gaze positions on the TV screen can be obtained based on the user-dependent calibration information and calculated head poses by using a single low-cost conventional web camera without an additional device for measuring the Z distance.

The remainder of this paper is organized as follows. An overview of our proposed gaze-tracking system and the methods used is presented in Sec. 2. Experimental results are presented in Sec. 3. Finally, we discuss our findings and conclude the paper in Sec. 4.

*Address all correspondence to: Kang Ryoung Park, E-mail: parkgr@dgu.edu

2 Proposed System

2.1 Overview of the Proposed Method

Figure 1 shows the environment in which our proposed gaze-tracking system for smart TVs is used. A conventional web camera equipped with a zoom lens ($\times 1.75$) is used. The image resolution captured by the camera is 1600×1200 pixels. The Z distance between the user and the TV screen is ~ 2 m, and the size of the TV screen is 60 in.

Figure 2 gives an overview of our proposed method: Fig. 2(a) shows the user-calibration procedure, while Fig. 2(b) shows the gaze-detection procedure.

In the user-calibration procedure, RGB color images are first captured by a camera equipped with a zoom lens (step 1 in Fig. 2). In step 2, the user turns his head to look at five points on the TV screen—center, left-top, right-top, right-bottom, and left-bottom—which helps in data acquisition of facial features in step 3. In the gaze-detection procedure, RGB color images are captured by the camera (step 4). In step 5, the face regions of the input images are detected using adaptive boosting (AdaBoost) face detection,¹² and the detected face regions are tracked continuously using the adaptive mean shift (CamShift) method in step 6.¹³ In steps 7 and 8, the eye and nostril regions are detected with adaptive template matching (ATM) and sub-block-based template matching, respectively. Based on the detected eyes and nostril regions, the head pose is estimated and the gaze’s position on the TV screen is determined in step 9.

2.2 Initial Calibration

In order to estimate a user’s head poses, an initial calibration must be performed. In the initial stage, the user is positioned centered in front of the smart TV at a Z distance (between user and camera) of 2 m. The user then looks at five points on the TV: center, left-top, right-top, right-bottom, and left-bottom, by rotating his face as shown in Fig. 3.

The face and eye regions of the input image are detected using the AdaBoost method.^{8,12} The nostril area is located using sub-block-based template matching. Figure 4 shows five examples of face images acquired in the calibration stage. To remove the background area from the detected face region, the face region is redefined based on the detected

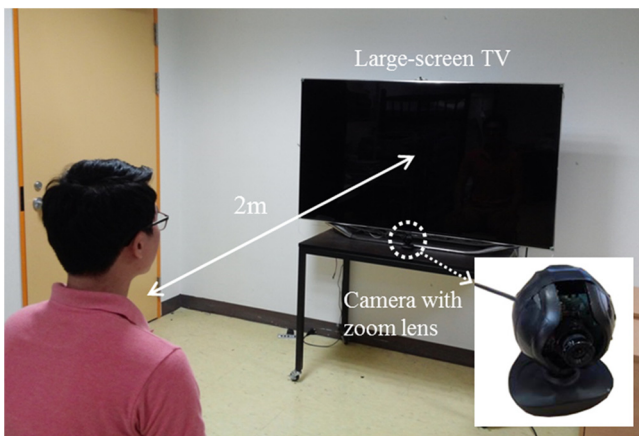


Fig. 1 Environment in which our proposed gaze tracking system for smart TVs is utilized.

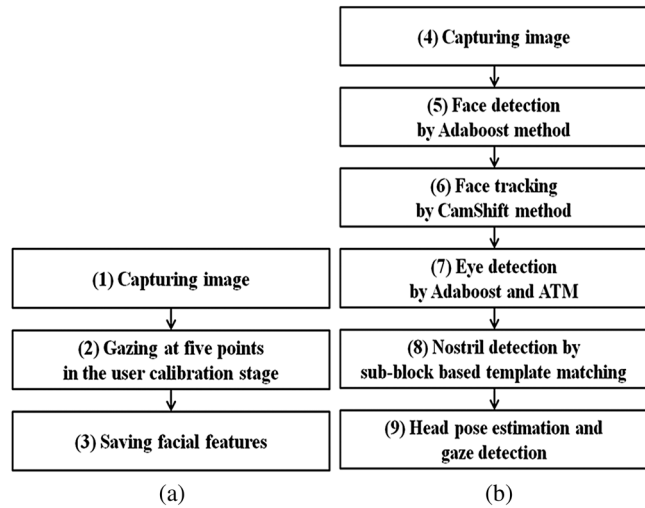


Fig. 2 Procedures used in our proposed method: (a) user calibration procedure and (b) gaze detection procedure.

eye positions, as shown in Fig. 4. In addition, the distance between the eyes (l of Fig. 4) of the frontal face is obtained. In addition, the distances between the eyes and the nose (m_1 and m_2 of Fig. 4) are obtained from the two images when a user is looking at the left-bottom and right-bottom positions. These distances are used as reference values for calculating the head pose in the gaze-detection step in Fig. 2(b).

2.3 Face and Facial Feature Detection in the Gaze-Detection Stage

In the gaze-detection stage of Fig. 2(b), information about the succession of frames can be used; the face region detected with AdaBoost is tracked using the CamShift method.¹³ This is because although the AdaBoost method shows high accuracy of face detection, it requires a long processing time. Face tracking using the CamShift method has the advantages of processing speed and being less affected by the variations of head pose. The CamShift algorithm has been widely used for object detection and tracking.¹³ This algorithm is based upon the MeanShift method,¹⁴ and both the MeanShift and CamShift methods

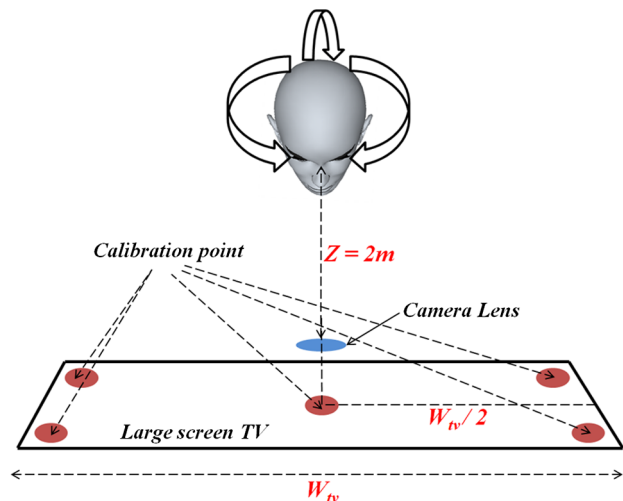


Fig. 3 Schematic of the calibration process.

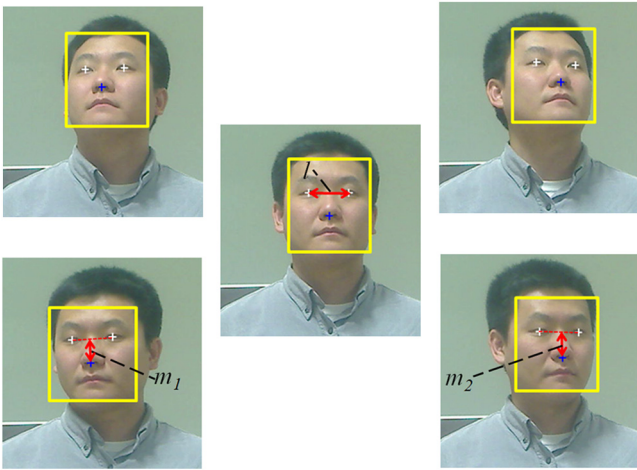


Fig. 4 Examples of the face and facial feature information obtained in the calibration stage.

usually use color histograms of the target to be tracked. Considering the probability distributions of the target that change in time, the CamShift algorithm tracks the object.¹⁵ In our method, we use the CamShift method based on color images (the hue histogram of HSV color space). Because the CamShift method tracks the target based on the correlation of the histogram, it is less affected by the head pose than pixel-based matching, and it has a fast processing speed.

The eyes are detected and tracked using AdaBoost and ATM, respectively, within the predetermined area of the detected face box. Only in the first frame, the eyes are detected by the AdaBoost method, and they are tracked by ATM in successive frames. That is, in the second frame, the left and right eyes are detected with the two templates that were determined by the AdaBoost method in the first frame. If the eyes are successfully detected in the second frame, the templates are updated with the newly detected eye regions, and these are used for template matching in the third frame. Owing to the template-update scheme, this method is called adaptive template matching. If they are not detected in the second frame, the eyes are located by the AdaBoost method within the predetermined area, based on the detected face region in the second frame. This procedure is iterated in successive images. For template matching, the correlation value between the template and the corresponding region to be matched is calculated by moving the region with an overlap of 1 pixel in the horizontal and vertical directions, respectively. The position that maximizes the correlation value is determined to be the final matching area.

Further, the nostril region is detected using a nostril-detection mask, and it is tracked using ATM. That is, in the first frame, the nostril region is detected using the nostril-detection mask, and it is located using ATM from the second frame on. If it is not located in the second frame, the nostril area is detected by the nostril detection mask within the predetermined area based on the detected face region in the second frame. This procedure is iterated in successive images.

The nostril-detection mask is designed based on the shape of the nostrils. In general, nostrils exhibit similar shape characteristics, i.e., the intensity differences between the nostril and its neighbors (skin region) are significant. Using this property, nostril-detection masks were defined as shown in Fig. 5, and sub-block-based template matching was performed.⁸ Figures 5(a) and 5(c) are frontal face and nostril-detection masks, respectively. Figures 5(b) and 5(d) show the rotated face and nostril-detection masks, respectively. The numbers of sub-blocks in Figs. 5(c) and 5(d) are 5×3 and 3×3 , respectively. When the face is rotated as shown in Fig. 5(b), it is often the case that the two nostrils overlap, and we use the mask with only one black area, as shown in Fig. 5(d). Because it is difficult to know whether the face is oriented frontally or rotated, matching with both masks, in Figs. 5(c) and 5(d), is performed. If the matching value using the mask in Fig. 5(c) is less than a predetermined threshold, further matching using the mask in Fig. 5(d) is performed.

The nostril region varies according to the Z distance between the camera and the user. Therefore, we change the size of the mask based on the Z distance estimated from the width of the detected face box. Accordingly, if the estimated Z distance is large, we use a small mask, and vice versa.

For sub-block-based template matching, we compute the average intensity of each sub-block in Figs. 5(c) and 5(d). Within the searching area of the nostril region, sub-block-based matching is iterated by moving the masks of Figs. 5(c) and 5(d) by an overlap of 2 pixels in the horizontal and vertical directions. If the average intensity of the black sub-blocks in Figs. 5(c) and 5(d) is lower than other sub-blocks, this location is assigned to the nostril candidate. Among the candidate areas, the region that has a maximum value of intensity differences between the black sub-block and its neighboring sub-blocks is determined to be the nostril region. To reduce computational complexity, the integral image technique is used to calculate the average intensity.¹² The detected nostril region is tracked using ATM in successive frames.

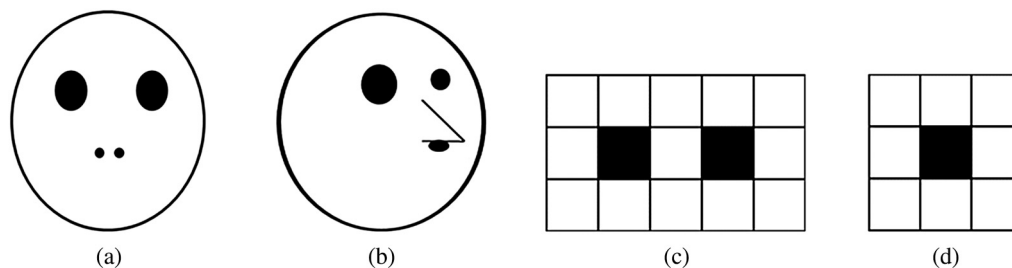


Fig. 5 Nostril detection masks: (a) frontal face, (b) rotated face, (c) nostril detection mask designed for frontal face, and (d) nostril detection mask designed for rotated face.

2.4 Head Pose Estimation

We estimate the head pose based on the detected eye and nostril positions. In order to estimate rotation in the direction of the X axis (horizontal rotation), we utilize the change in distance between the two eyes. Specifically, we calculate the rotation angle in the direction of the X axis based on the distance between the eyes in the rotated-face image and in the front-facing image in the calibration stage (gazing at the center position of the TV screen in Fig. 4).

Figure 6(a) is a frontally directed face in the calibration stage, and Fig. 6(b) is a face rotated in the direction of the X axis in the gaze-detection stage. The angle θ_x' in Fig. 6(b) is the face rotation angle in the direction of the X axis; it can be calculated as follows.

Initially, the distance l_x' in Fig. 6(a), the distance between the two eyes of the frontally directed face in the image plane (1 of the center image of Fig. 4), follows the following relationship:

$$l_x : l_x' = z_1 : f, \tag{1}$$

$$l_x = \frac{z_1}{f} l_x', \tag{2}$$

where l_x is the distance between the two eyes and z_1 is the distance between the camera and the user in the calibration stage. f is the camera focal length, which is measured during the initial camera calibration. Distance l_x'' in Fig. 6(b) is calculated using Eq. (2) as follows:

$$l_x'' = l_x \cos \alpha_x = \frac{z_1}{f} l_x' \cos \alpha_x. \tag{3}$$

Distance l_x''' in Fig. 6(b), the distance between the two eyes of the rotated face in the image plane, can be represented, using Eq. (3), as follows:

$$l_x'' : l_x''' = z_2 : f, \tag{4}$$

$$l_x''' = \frac{f}{z_2} l_x'' = \frac{z_1}{z_2} l_x' \cos \alpha_x, \tag{5}$$

where z_2 is the distance between the camera and the user.

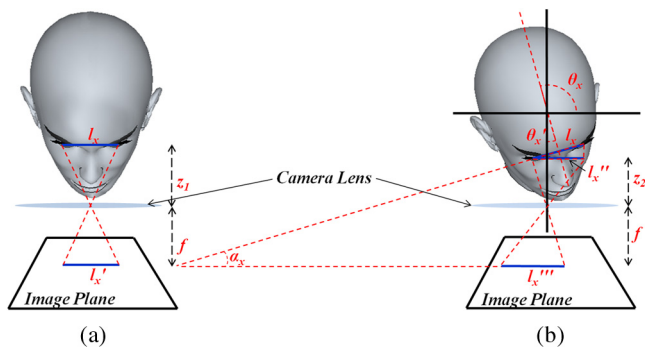


Fig. 6 Face rotation model in the direction of the X axis: (a) front-facing and (b) rotated.

Angle α_x in Fig. 6(b) is calculated from Eq. (5) as follows:

$$\alpha_x = \cos^{-1} \frac{l_x''' z_2}{l_x' z_1}. \tag{6}$$

In Eq. (6), l_x' and l_x''' are the distances between both detected eyes in the capture image during the calibration stage and the current input image, respectively. z_1 and z_2 are measured from the widths of face box detected in the image captured in the calibration stage and the current input image, respectively.

In Eq. (6), α_x is identical to θ_x' , as follows. The line with gradient $\tan \theta_x$ is orthogonal to the line with gradient $\tan \alpha_x$. Thus, θ_x is calculated as in Eq. (7).

$$\theta_x = \tan^{-1} \frac{-1}{\tan \alpha_x}. \tag{7}$$

Consequently, the angle θ_x' in Fig. 6(b), the face rotation angle in the direction of X axis, can be calculated using Eq. (8).

$$\theta_x' = \theta_x - \frac{\pi}{2} = \tan^{-1} \frac{-1}{\tan \alpha_x} - \frac{\pi}{2}. \tag{8}$$

With further manipulation of Eq. (8), we can obtain Eqs. (9) and (10).

$$\tan \left(\theta_x' + \frac{\pi}{2} \right) = \frac{-1}{\tan \alpha_x}, \tag{9}$$

$$-\cot \theta_x' = -\cot \alpha_x. \tag{10}$$

Thus, we confirm that α_x is identical to θ_x' and that the angle θ_x' in Fig. 6(b), the face rotation angle in the direction of X axis, is calculated using Eq. (6).

In Fig. 6(b), it is difficult to determine whether the face is rotated in the clockwise or counterclockwise direction with Eq. (6) alone. Therefore, we measured the X -distance (horizontal distance) between the left eye and the nostril, and between the right eye and the nostril, as d_1 and d_2 , respectively. If d_1 is larger than d_2 as shown in Fig. 6(b), the face has been rotated in the counterclockwise direction. If d_1 is smaller than d_2 , the face is determined as having been rotated in the clockwise direction.

If the face is rotated in the direction of the Y axis (vertical direction), we can calculate the rotation angle in the direction of the Y axis based on the changing distance between the eyes and the nostril, as illustrated in Fig. 7. Figure 7(a) is a face (when a user is looking at the left-lower or right-lower position) in the calibration stage, and Fig. 7(b) is the rotated face in the direction of the Y axis in the gaze-detection stage. As shown in Fig. 1, the camera is positioned below the TV screen and a user gazes at the position on the TV screen. So, the image plane of the camera is positioned below the gaze vector extending from the user, as shown in Fig. 7(b). Angle θ_y' in Fig. 7(b), the face rotation angle, can be calculated as follows.

Distance l_y' in Fig. 7(a), the distance between the eyes and the nostril, is measured from the face images when a user is looking at the left-lower and right-lower positions

of Fig. 4 during the calibration stage (m_1 and m_2 of the left-lower and right-lower images of Fig. 4). The reasoning behind why we used face images of a user looking at the left-lower and right-lower position, instead of an image of the user gazing at the center position [as with the measurement of l_x' of Fig. 6(a)] goes as follows. Since the camera is positioned below the TV, when a user gazes at the (left or right) lower position, the resolution of distance l_y' in Fig. 7(a) is maximized. Because the vertical head pose is measured based on the distance l_y' , it is necessary to obtain the largest distance l_y' for the best accuracy in estimating the vertical head pose.

Distance l_y' follows the following relationship:

$$l_y : l_y' = z_1 : f, \tag{11}$$

$$l_y = \frac{z_1}{f} l_y', \tag{12}$$

where l_y is the distance between the eyes and the nostril. Distance l_y'' in Fig. 7(b) is calculated with Eq. (12) as follows:

$$l_y'' = l_y \cos \alpha_y = \frac{z_1}{f} l_y' \cos \alpha_y. \tag{13}$$

Distance l_y''' in Fig. 7(b), the distance between the eyes and the nostril of the rotated face in the image plane, can be represented using Eq. (13) as follows:

$$l_y'' : l_y''' = z_2 : f, \tag{14}$$

$$l_y''' = \frac{f}{z_2} l_y'' = \frac{z_1}{z_2} l_y' \cos \alpha_y. \tag{15}$$

Angle α_y in Fig. 7(b) is calculated from Eq. (15) as follows:

$$\alpha_y = \cos^{-1} \frac{l_y''' z_2}{l_y' z_1}. \tag{16}$$

In Eq. (16), l_y' and l_y''' are measured from the distance between the detected eyes and nostrils in the image captured in the calibration stage and the current input image,

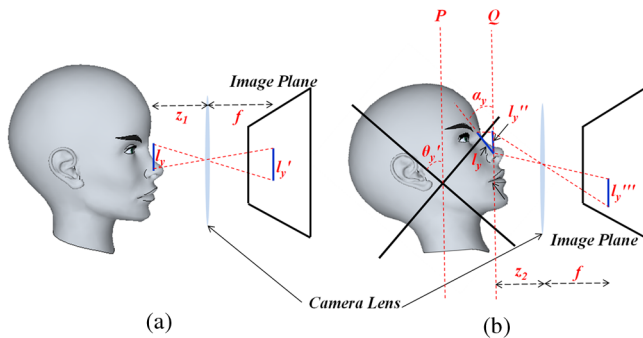


Fig. 7 Face rotation model in the direction of the Y axis: (a) face when a user is looking at the left-lower or right-lower position of Fig. 4 and (b) rotated face.

respectively. z_1 and z_2 are measured from the width of face box detected in the image captured in the calibration stage and the current input image, respectively. The width (w_1) of a user's detected face box is obtained in the image of initial calibration stage. If the width (w_2) of the face box in a current image is compared to w_1 , the change in the Z distance can be estimated based on the camera-perspective model. Since the initial calibration is done at a Z distance (z_1) of 2 m, the Z distance [z_2 of Eq. (18)] in the current frame is measured as $z_1 \times w_1/w_2$ from Eqs. (17) and (18).

$$w_1 : f = W : z_1, \tag{17}$$

$$w_2 : f = W : z_2, \tag{18}$$

where W is the actual width of the face.

Since line P is parallel with line Q , the angle α_y is equal to the angle θ_y' in Fig. 7(b), the face rotation angle, and the face rotation angle can be calculated using Eq. (16).

In order to estimate the gaze's position on TV screen, the five rotation angles acquired when a user looks at the five positions in the calibration stage are used. By using these angles, the user-dependent thresholds that define the candidate areas for gaze position are calculated. Based on thresholds and the head pose of θ_x' and θ_y' in Figs. 6(b) and 7(b), respectively, we obtain the final gaze position on the TV screen.

3 Experimental Results

We tested our proposed gaze tracking method using a desktop computer with an Intel Core™ I7 3.5 GHz CPU and 8 GB of RAM. Our proposed algorithm was implemented using Microsoft Foundation Class-based C++ programming and the DirectX 9.0 software development kit.

To measure the gaze-detection performance of our proposed method, we utilized a database comprising head poses for five people.⁸ In our experiment, each user was asked to look at nine different positions on a 60-in. TV screen (as shown in Fig. 8) by rotating his/her head up and down and left to right (left-upper, left-middle, left-lower, center-upper, center-middle, center-lower, right-upper, right-middle, and right-lower positions).⁸ The Z distance between the camera and each user's face was ~2 m. Our database had 1350 head pose images (five persons \times nine gazing positions)

1	4	7
2	5	8
3	6	9

Fig. 8 Nine gaze positions (target regions) on the TV screen in our experiment.

×30 images/gazing position).⁸ In addition, an additional five images per person were obtained for user calibration when the user gazed at the five positions 1, 3, 5, 7, and 9 indicated in Fig. 8.

We used the strictly correct estimation rate (SCER)^{8,16} to measure the accuracy of our proposed gaze-detection method. The SCER is the ratio of the number of strictly correctly determined frames to the number of total image frames. For example, if a user gazes at region 2 in an image frame and our system correctly determines that he/she is gazing at region 2, this image is determined to be the correct frame. If 1100 images among 1350 are determined to be correct frames, the consequent SCER is ~81.5% [(1100/1350) × 100].⁸ Thus, a higher SCER value signifies a better estimation performance.

Table 1 shows the SCER results obtained with the collected database for both the method at hand and the scheme we proposed previously.⁸ As shown in Table 1, the average SCER value of all nine target positions is ~90.5%, which means that our method correctly detected the gaze position of >90% of the images. In addition, the average SCER of the proposed method is improved over our previous effort.⁸ The proposed method outperforms our previous method because of the shoulder detection error, which came from measuring the vertical head pose using facial features and shoulder positions.

Table 2 shows the confusion matrix for gaze detection results from the proposed method. For example, 7 in the (2, 3) cell (reference gaze positions in Fig. 8, calculated gaze positions) represents that seven times the system predicted the user looked at position 3 although users actually gazed at position 2 of Fig. 8. The diagonal in Table 2 shows the number of correct detections by the proposed gaze-detection method. The matrix confirms that the correct detection rates for all the gaze positions are similar.

Table 3 shows the accuracies of the proposed gaze-detection method for horizontal (column) and vertical (row)

Table 1 Strictly correct estimation rate (SCER) result for each gaze region (target region) for our proposed method and previous one (Ref. 8).

Target region	SCER (%)	
	Previous method ⁸	Proposed method
1	89.3	99.3
2	60.7	84.7
3	70	94
4	100	90.7
5	88.7	92
6	76	87.3
7	70.7	96
8	76.7	83.3
9	100	87.3
Average	81.3	90.5

Table 2 The confusion matrix of gaze detection results by the proposed method.

	Calculated gaze positions									
	1	2	3	4	5	6	7	8	9	
Reference gaze positions of Fig. 8	1	149	1	0	0	0	0	0	0	0
	2	5	127	7	0	11	0	0	0	0
	3	0	0	141	2	1	6	0	0	0
	4	10	0	0	136	3	0	1	0	0
	5	0	0	4	3	138	5	0	0	0
	6	0	0	0	0	19	131	0	0	0
	7	0	0	0	2	1	0	144	3	0
	8	0	0	0	6	16	0	1	125	2
	9	0	0	1	0	0	10	0	8	131

rotations. In the first column, 95.6 represents the rate of the system calculating the gaze position to be 1, 2, or 3 when the user actually gazes at position 1, 2, or 3 in Table 2. Table 3 confirms that the horizontal and vertical accuracies are similar.

For the next experiment, we measured the influence of the accuracy of facial feature detection/tracking. With the 1350 head pose images (five persons × nine gazing positions × 30 images/gazing position) used in Table 1, we obtained noisy data by including the Gaussian random noise in the detected positions of both eyes and nostrils according to sigma values of 0.5, 1.0, and 1.5. As shown in Table 4, the accuracy of gaze detection is not much affected by the Gaussian random noise. Here, the case of sigma value 0 indicates no Gaussian random noise included in the detected positions of facial features.

In addition, we measured the influence of the detection accuracy on each facial feature. In Table 5, with each of the three sigma values for Gaussian random noise added to the detected position of the left eye, we measured the accuracy of gaze detection according to the Gaussian random noise added to the detected position of the right eye.

By a similar method, we measured the accuracy of gaze detection with the detected positions of the left eye and nostrils, and those of the right eye and nostrils, as shown in Tables 6 and 7, respectively. As shown in Tables 5, 6, and 7, the influence of the detection accuracy for each facial feature on the accuracy of gaze detection is similar. In

Table 3 Horizontal and vertical accuracies of the proposed method (%).

Accuracy	95.6	96.7	92
98.2	1	4	7
92.7	2	5	8
93.3	3	6	9

Table 4 SCER results for each gaze region (target region) according to the sigma value of Gaussian random noise added to the detected positions of facial features (%).

Sigma value	Target region									Average
	1	2	3	4	5	6	7	8	9	
0	99.3	84.7	94	90.7	92	87.3	96	83.3	87.3	90.5
0.5	99.3	83.3	93.3	90.7	92	87.3	95.3	83.3	86	90.1
1	99.3	82	91.3	90	92	86	94.7	83.3	86	89.4
1.5	98.7	81.3	91.3	88.7	90.7	85.3	92	81.3	85.3	88.3

addition, we confirm that the accuracy of gaze detection is not much affected by the Gaussian random noise added to the detected positions of facial features.

Figure 9 shows the successful and unsuccessful results for gaze detection by the proposed method. Figure 9(a) includes images that gave good results. The left, center, and right images of Fig. 9(a) are the cases when the user is gazing at regions 1, 7, and 2 of Fig. 8, respectively. Figures 9(b) and 9(c) are images that gave bad results. The user is actually gazing at region 3 in Fig. 9(b). However, since the rotation (in the lower direction) of his head is small [the head rotation seems to be similar to that of the right image of Fig. 9(a)], his gaze is incorrectly determined as being directed at region 2. The user’s eye rotation explains the small head rotation.

The left and right images of Fig. 9(c) are cases in which the user is gazing at regions 8 and 5 of Fig. 8, respectively. However, due to the user’s eyes blinking, the left eye positions are incorrectly detected, which causes the incorrect detection of gaze position. The gazes for the left and right images of Fig. 9(c) were incorrectly determined as being directed at regions 9 and 6, respectively.

For the next test, we measured the accuracy of gaze detection in a case where the training images for the calibration stage and the current testing image are from different users. With the 1350 head pose images (five persons × nine gazing positions × 30 images/gazing position) used to compile Table 1, we randomly selected the cases where the training and testing images are from different persons and measured the accuracy (SCER) of gaze detection. The accuracy was 59.4%, which is much less than that for user-dependent training data in Table 1. The reasons are as follows.

Table 5 Average SCER of nine target regions according to the sigma values of Gaussian random noise added to the detected positions of left (S_LE) and right eyes (S_RE), respectively (%).

S_LE	S_RE			
	0	0.5	1	1.5
0	90.5	90.5	90.4	90.4
0.5	90.5	90.2	89.6	89.4
1	90.4	89.7	89.6	89.3
1.5	90.4	89.3	89.2	88.9

As shown in Figs. 6 and 7, l_x' and l_y' are obtained from the distance between two facial features in the images obtained in the initial user-dependent calibration stage. l_x''' and l_y''' are obtained from the current captured image. In addition, z_1 and z_2 are measured by the widths of face boxes detected in the image of the initial user-dependent calibration stage and the current input image, respectively.

If the training images in the calibration stage and the current testing image are of different people, l_x of Fig. 6(a) is different from that of Fig. 6(b). In addition, l_y of Fig. 7(a) is different from that of Fig. 7(b). These do not satisfy our assumption that l_x and l_y are equal in the initial calibration stage [Figs 6(a) and 7(a)] and current input image [Figs. 6(b) and 7(b)]. In addition, the actual width of the face in the calibration stage [Figs. 6(a) and 7(a)] is different from that in the current input image [Figs. 6(b) and 7(b)], which does not match our assumptions, either. As shown in Figs. 6 and 7, l_x' and l_x''' are calculated from l_x , and l_y' and l_y''' are obtained from l_y . In addition, z_1 and z_2 are measured based on the actual width of the face.

So, we inevitably obtain inaccurate l_x'''/l_x' , l_y'''/l_y' , and z_2/z_1 [Eqs. (6) and (16)], which degrades the accuracy of the estimation of α_x (θ_x') of Eq. (6) [Eq. (10)], and α_y (θ_y') of Eq. (16). Consequently, the gaze-detection accuracy is much reduced.

Next, we compared the processing time of the proposed method with that of our previous effort.⁸ Experimental results showed that the processing time of the method at hand was 50.67 ms/frame, which is much faster than that of the previous method (528.53 ms/frame).⁸

In the next test, we performed the experiments with two open datasets, the CAS-PEAL-R1 database^{17,18} and the FEI

Table 6 Average SCER of nine target regions according to the sigma values of Gaussian random noise added to the detected positions of left eye (S_LE) and nostril (S_N), respectively (%).

S_N	S_LE			
	0	0.5	1	1.5
0	90.5	90.5	90.4	90.4
0.5	90.4	90.3	89.5	89.3
1	90.4	89.7	89.5	89.2
1.5	90.3	89.2	89.2	88.8

Table 7 Average SCER of nine target regions according to the sigma values of Gaussian random noise added to the detected positions of right eye (S_RE) and nostril (S_N), respectively (%).

S_N	S_RE			
	0	0.5	1	1.5
0	90.5	90.5	90.4	90.4
0.5	90.4	90.2	89.6	89.3
1	90.4	89.7	89.5	89.2
1.5	90.3	89.2	89.1	88.7

face database.¹⁹ Although many open face databases exist, such as the AR database and Pal database, few include a variety of poses for each face. The CAS-PEAL-R1 database includes 30,900 images of 1040 Mongolian subjects (595 males and 445 females). Among them, 21,840 images (21 poses \times 1040 individuals) with pose variations were acquired according to different camera positions (*C1* to *C7*) as shown in Fig. 10.^{17,18} The image resolution is 360×480 pixels.

We use only nine images that were captured by *C3* to *C5* of Fig. 10 for each face in the CAS-PEAL-R1 database for our experiments because we divide the gazing regions into a 3×3 grid as shown in Fig. 8. We assume that the upper, middle, and lower images from *C5* are obtained when a user gazes at the 1, 2, and 3 positions of Fig. 8. *C4* corresponds with the 4, 5, and 6 positions similarly, as does *C3* with the 7, 8, and 9 positions of Fig. 8.

The remained images from *C1*, *C2*, *C6*, and *C7* in Fig. 10 are not used for our experiments because severe rotation of head occurs, which does not happen when a user looks at a TV normally. In addition, one of the eyes or nostrils is occluded due to the severe rotation of the head. Consequently, a total of 9360 images (nine poses \times 1040 individuals) were used for our experiments. We used nine images for each face in the database, and five images (upper and lower of *C5*, middle of *C4*, and upper and lower of *C3*) among nine were used for user calibration, as in Fig. 4. So, we define five images as the training set and the remaining four as the test set. Thus, the accuracies of our gaze-detection method were measured with the training and test sets as shown in Table 8.

As shown in Table 8, the average SCERs for the training and test sets are similar, and the average SCER of all of the sets is $\sim 86.15\%$. Figure 11 shows examples of the detection results of facial features using the CAS-PEAL-R1 database.

The FEI open face database consists of 2800 images of 200 subjects (100 males and 100 females). Among them, 2200 images (11 poses \times 200 individuals) include pose variations. Ten images were obtained by profile rotation (up to ~ 180 deg) with each person in an upright frontal position, and one additional frontal image was acquired.¹⁹ The image resolution is 640×480 pixels. All participants were between 19 and 40 years and were Brazilian. Figure 12 shows examples from the FEI face database.

The FEI face database does not include images with rotations in the direction of the *Y* axis (upper, middle, and lower directions). Therefore, we define new gazing positions of a 1 (row) by 7 (column) grid on the screen (instead of using the positions of a 3×3 grid in Fig. 8), and the seven images of Figs. 12(a) and 12(b) are used for experiments, while

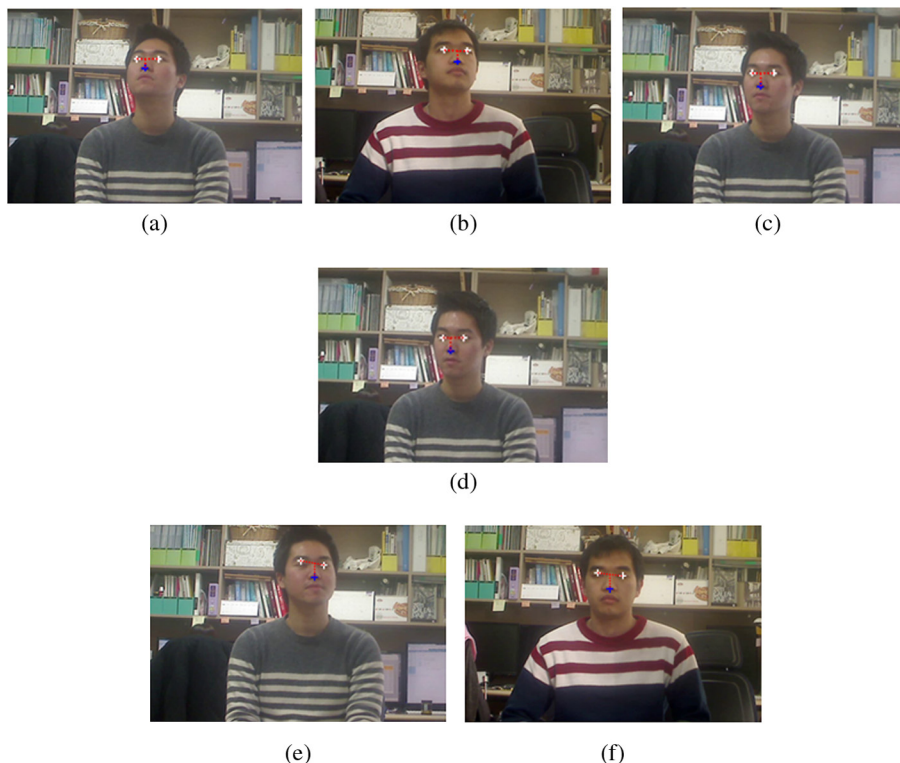


Fig. 9 Resulting images from our method: (a) good result cases, (b) bad result cases (when user's head rotation is small, and (c) bad result cases (when user eye blinking occurs).

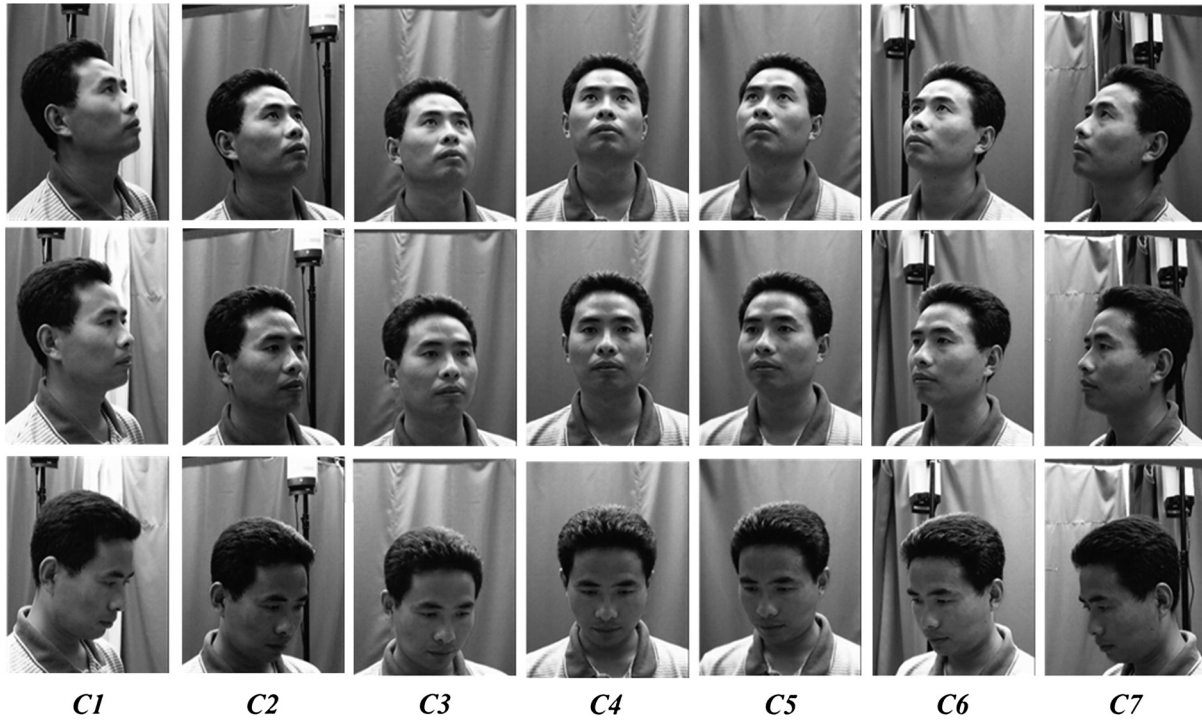


Fig. 10 Twenty-one images of one subject with pose variations in the CAS-PEAL-R1 database.

Table 8 SCER results with CAS-PEAL-R1 face database.

Reference gaze positions	Training sets					Test sets			
	1	3	5	7	9	2	4	6	8
SCER (%)	94.63	81.19	85.12	95.59	79.85	81.77	83.97	90.88	82.34
Average (%)	87.28					84.74			
Average of total sets (%)	86.15								

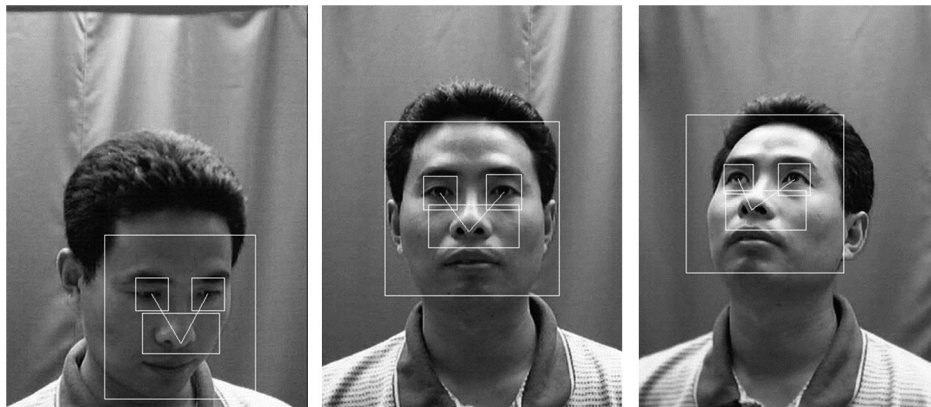


Fig. 11 Examples of detected facial features from the CAS-PEAL-R1 database.

assuming that they are obtained when the user gazes at these seven positions (1 to 7). Among these seven images, we used three in Fig. 12(a) for user calibration (training), which are assumed to be obtained when the user gazes at the 2, 4, and 6 gazing positions, respectively. The other four images in

Fig. 12(b) were used for testing, which are assumed to be obtained when the user gazes at the 1, 3, 5, and 7 gazing positions, respectively. The remaining four images of Fig. 12(c) were not used for experiment because severe head rotation occurs, as discussed above. Therefore, 1400



Fig. 12 Examples of 11 images of one subject with pose variations in the FEI face database. (a) Images used for user calibration (training). (b) Images used for testing. (c) Images that were not used in our experiment.

Table 9 SCER result with FEI face database.

Reference gaze positions	Training sets			Test sets			
	2	4	6	1	3	5	7
SCER (%)	92	100	98.5	76.5	88	84.5	78
Average (%)	96.83			81.75			
Average of total sets (%)				88.21			

images (seven poses \times 200 individuals) were used for our experiments. Table 9 shows the SCER results from the FEI face database.

As shown in Table 9, the average SCER of all the sets is $\sim 88.21\%$. This lower SCER of testing data compared to that of training data can be explained by it often being the case

that the head rotations in images 1 and 2 of Figs. 12(a) and 12(b) are similar in the database. In addition, the head rotations in images 6 and 7 of Figs. 12(a) and 12(b) tend to be similar. Figure 13 shows examples of detection results for facial features from the FEI database.

In our experiments, we used three databases (the database we collected, CAS-PEAL-R1, and FEI databases). The rough size (width \times height) of faces in the database we collected is $\sim 185 \times 185$ pixels. Those in the CAS-PEAL-R1 and FEI databases are $\sim 180 \times 180$ pixels and 200×220 pixels, respectively.

Our research aims at developing gaze detection for use in a smart TV based upon a conventional, low-cost web camera without needing a high-power zoom lens and additional devices for pan and tilt functionalities. So, eye gaze cannot be detected due to the low image resolution in the eye region. Figure 14 shows examples of nine images, which are obtained when a user gazes at the nine positions in Fig. 8

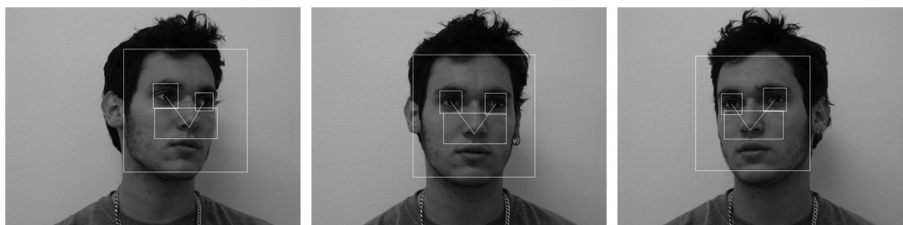


Fig. 13 Examples of detection results of facial features from the FEI database.



Fig. 14 Examples of nine images of one subject who gazes at nine gazing positions of Fig. 8 only by eye movement (without head rotation) and the corresponding detected coordinates of the left and right eyes. Gazing at the (a) 1, 4, and 7 positions, (b) 2, 5, and 8 positions, and (c) 3, 6, and 9 positions, respectively.

using only eye movement. As shown in Fig. 14, the X- and Y-disparities of eye positions according to each gazing position are very small and the detected eye positions using an AdaBoost eye detector are inaccurate due to the low image resolution of the eye region. So, the detected eye positions are impractical for use in gaze detection. Consequently, we propose that gaze detection in smart TVs be based on natural user head movements and not eye-gaze detection.

4 Conclusion

In this paper, we proposed a new gaze-detection method that uses a conventional (visible light) web camera. By using facial information obtained in the initial calibration stage, accurate head poses can be calculated. Horizontally and vertically rotated head poses were calculated based upon a geometrical analysis of the changes in facial feature position. The results of experiments conducted indicate that the gaze-detection accuracy for our method using a 60-in. smart TV is 90.5%. When the user's eye is falsely detected due to blinking eye, the accuracy of the proposed method decreases. In addition, when the user's head rotation is

small due to eye movement accompanying the gaze, the accuracy of the proposed method is also reduced.

In our research, we aim at developing a gaze-detection system using only a conventional visible-light web camera without additional special devices, through which we can reduce the cost and size of our system, and easily adopt our system for applications in smart TV. Thus, a conventional device for measuring the Z distance, such as a Microsoft Kinect, is not considered in our research.

In future work, we plan to combine the facial feature positions and shoulder positions in order to enhance gaze-detection accuracy. Further, we plan to incorporate facial texture information into the calculation of the gaze position.

Acknowledgments

This research was supported by the Korea Communications Commission, Korea, under the title of Development of Beyond Smart TV Technology (11921-03001). The research in this paper uses the CAS-PEAL-R1 face database collected under the sponsor of the Chinese National Hi-Tech Program and ISVISION Tech. Co. Ltd.

References

1. Z. Ren et al., "Robust hand gesture recognition with Kinect sensor," in *Proc. of the 19th ACM Int. Conf. on Multimedia*, pp. 759–760, ACM, New York, NY (2011).
2. K. K. Biswas and S. K. Basu, "Gesture recognition using Microsoft Kinect®," in *Proc. of the 5th Int. Conf. on Automation, Robotics and Applications*, pp. 100–103, IEEE Xplore Digital Library, USA (2011).
3. J. L. Raheja, A. Chaudhary, and K. Singal, "Tracking of fingertips and centres of palm using KINECT," in *Proc. of the 3rd Int. Conf. on Computational Intelligence, Modelling & Simulation*, pp. 248–252, IEEE Xplore Digital Library, USA (2011).
4. D.-C. Cho et al., "Long range eye gaze tracking system for a large screen," *IEEE Trans. Consum. Electron.* **58**(4), 1119–1128 (2012).
5. H. Lee et al., "Multi-modal user interaction method based on gaze tracking and gesture recognition," *Signal Process.: Image Commun.* **28**(2), 114–126 (2013).
6. C. Hennessey and J. Fiset, "Long range eye tracking: bringing eye tracking into the living room," in *Proc. of the Symp. on Eye Tracking Research and Applications*, pp. 249–252, ACM, New York, NY (2012).
7. W. O. Lee et al., "Auto-focusing method for remote gaze tracking camera," *Opt. Eng.* **51**(6), 063204 (2012).
8. D. T. Nguyen et al., "Gaze detection based on head pose estimation in smart TV," in *Proc. of Int. Conf. on ICT Convergence*, pp. 283–288, IEEE Xplore Digital Library, USA (2013).
9. P. M. Corcoran et al., "Real-time eye gaze tracking for gaming design and consumer electronics systems," *IEEE Trans. Consum. Electron.* **58**(2), 347–355 (2012).
10. Y. Zhang, A. Bulling, and H. Gellersen, "Sideways: a gaze interface for spontaneous interaction with situated display," in *Proc. of the SIGCHI Int. Conf. on Human Factors in Computing Systems*, pp. 851–860, ACM, New York, NY (2013).
11. D. Mardanbegi and D. W. Hansen, "Mobile gaze-based screen interaction in 3D environments," in *Proc. of the Conf. on Novel Gaze-Controlled Applications*, ACM, New York, NY (2011).
12. P. Viola and M. J. Jones, "Robust real-time face detection," *Int. J. Comput. Vis.* **57**(2), 137–154 (2004).
13. M. Boyle, "The effects of capture conditions on the CAMSHIFT face tracker," Technical Report, Department of Computer Science, Report No. 2001-691-14, University of Calgary (2001).
14. D. Comaniciu, V. Ramesh, and P. Meer, "Real-time tracking of non-rigid objects using mean shift," in *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 142–149, IEEE Xplore Digital Library, USA (2000).
15. G. R. Bradski, "Real time face and object tracking as a component of a perceptual user interface," in *Proc. of IEEE Workshop on Applications of Computer Vision*, pp. 214–219, IEEE Xplore Digital Library, USA (1998).
16. S. J. Lee et al., "Real-time gaze estimator based on driver's head orientation for forward collision warning system," *IEEE Trans. Intell. Transp. Syst.* **12**(1), 254–267 (2011).
17. W. Gao et al., "The CAS-PEAL large-scale Chinese face database and baseline evaluations," *IEEE Trans. on Syst. Man, Cybern.* **38**(1), 149–161 (2008).
18. "CAS-PEAL-R1 face database," <http://www.jdl.ac.cn/peal/JDL-PEAL-Release.htm> (3 March 2014).
19. C. E. Thomaz, "FEI face database," 14, March, 2012, <http://fei.edu.br/~cet/face-database.html> (3 March 2014).

Won Oh Lee received his BS degree in electronics engineering from Dongguk University, Seoul, Republic of Korea, in 2009. He is currently pursuing a combined course of MS and PhD degrees in electronics and electrical engineering at Dongguk University. His research interests include biometrics and pattern recognition.

Yeong Gon Kim received his BS and MS degrees in computer engineering and electronics and electrical engineering from Dongguk University, Seoul, Republic of Korea, in 2011 and 2013, respectively. He is currently pursuing his PhD degree in electronics and electrical engineering at Dongguk University. His research interests include biometrics and pattern recognition.

Kwang Yong Shin received his BS in electronics engineering from Dongguk University, Republic of Korea, in 2008. He also received a combined MS and PhD degrees in electronics and electrical engineering at Dongguk University in 2014. He is a researcher at the Korea Research Institute of Standards and Science. His research interests include biometrics and image processing.

Dat Tien Nguyen received his BS degree in electronics and telecommunication technology from Hanoi University of Technology, Hanoi, Vietnam, in 2009. He is currently pursuing a combined course of MS and PhD degrees in electronics and electrical engineering at Dongguk University. His research interests include biometrics and image processing.

Ki Wan Kim received his BS degree in computer science from Sangmyung University, Seoul, Republic of Korea, in 2012. He is currently pursuing his MS degree in electronics and electrical engineering at Dongguk University. His research interests include biometrics and image processing.

Kang Ryoung Park received his BS and MS degrees in electronic engineering from Yonsei University, Seoul, Republic of Korea, in 1994 and 1996, respectively. He received his PhD degree in electrical and computer engineering from Yonsei University in 2000. He has been a professor in the Division of Electronics and Electrical Engineering at Dongguk University since March 2013. His research interests include image processing and biometrics.

Cheon In Oh received his BS degree in electronic engineering from the Sungkyunkwan University, Suwon, Republic of Korea, in 2005 and his MS degree from the University of Science and Technology, Daejeon, Republic of Korea, in 2007. He is now a senior researcher at the ETRI, Daejeon, South Korea. His research interests lie in the areas of broadcasting system, with particular emphasis on audience recognition, and advertising services/system.