Binocular gaze detection method using a fuzzy algorithm based on quality measurements

Chul Woo Cho
Hyeon Chang Lee
Su Yeong Gwon
Jong Man Lee
Dongwook Jung
Kang Ryoung Park
Hyun-Cheol Kim
Jihun Cha

Binocular gaze detection method using a fuzzy algorithm based on quality measurements

Chul Woo Cho,[a] Hyeon Chang Lee,[a] Su Yeong Gwon,[a] Jong Man Lee,[a] Dongwook Jung,[a] Kang Ryoung Park,[a,*] Hyun-Cheol Kim,[b] and Jihun Cha[b]
[a]Dongguk University, Division of Electronics and Electrical Engineering, 26 Pil-dong 3-ga, Jung-gu, Seoul 100-715, Republic of Korea
[b]Electronics and Telecommunications Research Institute, 218 Gajeong-ro, Yuseong-gu, Daejeon 305-700, Republic of Korea

Abstract. Due to the limitations of gaze detection based on one eye, binocular gaze detection using the gaze positions of both eyes has been researched. Most previous binocular gaze detection research calculated a gaze position as the simple average position of the detected gaze points of both eyes. To improve this approach, we propose a new binocular gaze detection method using a fuzzy algorithm with quality measurement of both eyes. The proposed method is used in the following three ways. First, in order to combine the gaze points of the left and right eyes, we measure four qualities on both eyes: distortion by an eyelid, distortion by the specular reflection (SR), the level of circularity of the pupil, and the distance between the pupil boundary and the SR center. Second, in order to obtain a more accurate pupil boundary, we compensate the distorted boundary of a pupil by an eyelid based on information from the lower half-circle of the pupil. Third, the final gaze position is calculated using a fuzzy algorithm based on four quality-measured scores. Experimental results show that the root-mean-square error of gaze estimation by the proposed method is approximately 0.67518 deg. © 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.53.5.053111]

1 Introduction

Gaze detection is a method of detecting where a user is looking. Previous gaze detection methods can be categorized into two types: monocular and binocular gaze detection methods. Most previous studies have focused on monocular gaze detection. For this category of detection, gaze detection methods based on the use of multiple cameras[1,2] and multiple illuminators with stereo cameras[3] have been proposed. In another approach, Murphy-Chutorian et al. proposed a gaze detection method for driver monitoring.[4] However, these methods are limited in that they require a complex calibration procedure for multiple cameras or result in low accuracy. In another approach to monocular gaze detection, methods in which a gaze position is estimated using a wearable head-mounted device with a camera[5] and near infrared (NIR) illuminators have been proposed.[6–9] Piccardi et al. used a single head-mounted camera for calculating gaze positions.[5] Ko et al. proposed a gaze detection method using four NIR illuminators attached to the four corners of a monitor and a helmet-type device with a single eye-capturing camera.[8] However, wearing additional equipment can be uncomfortable for users. In addition, monocular gaze detection has a limitation in that the gaze detection accuracy degrades when the pupil and the reference illumination points in a single-eye image are incorrectly detected. To overcome this problem, binocular gaze detection using the gaze positions of both eyes has been proposed.

A binocular gaze detection method using four monochrome cameras and three illuminators was proposed in Ref. 10. In this method, the gaze position is calculated after obtaining the optical axis of both eyes from two cameras. However, due to the number of cameras and illuminators required, the system is complicated, large in size, and expensive. In addition, the midpoint of the gaze of both eyes is determined as the final gaze position without considering the confidence level of each of the two gaze points.

Hennessey et al. calculated a three-dimensional gaze position using one camera and five illuminators.[11] By switching the illuminator on and off, the pupil center can be obtained. Through calibration, the angle between the visual and the optical axes is calculated, and the gaze point of both eyes is, thereby, obtained. The midpoint of the gaze of both eyes is determined as the final gaze position without considering the confidence level of the two gaze points.

Most recent researches in binocular gaze detection calculate the gaze position as the simple average of the detected gaze points of both eyes and do not consider the confidence level of the two gaze points. To solve this problem, we propose a new binocular gaze detection method using one camera and one NIR illuminator, without the need for wearable equipment. To combine the gaze points of both the left and right eyes, we measure the quality of each eye based on the distortions from the eyelid or specular reflection (SR), the level of circularity of the pupil, and the distance between the pupil boundary and the SR center. To obtain a more accurate pupil boundary, we compensate the distorted boundary of the pupil from the eyelid based on the information on the lower half-circle of the pupil. The final gaze position is calculated using a fuzzy algorithm based on the measurements of four quality scores. Table 1 shows a comparison between previous systems and the proposed method.

The remainder of this paper is organized as follows. The proposed method and the devices used are described in Sec. 2. The experimental results are presented in Sec. 3. Finally, some concluding remarks are given in Sec. 4.

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

Table 1 Comparison between previous systems and the proposed method.

| Category	Methods	Strength	Weakness	
Monocular gaze detection method	Using one camera to acquire the face region and calculating the gaze position by using the appearance information of face and eye[4]	Not requiring a user to wear any devices, this method can give convenience to user.	Due to the low resolution eye image, the gaze accuracy is low.	Gaze detection accuracy degrades when the pupil and the reference illumination points in one eye image are incorrectly detected.
	Using two cameras such as narrow view and wide view cameras[1,2] Using stereo cameras with multiple illuminators[3]		The system cost and complexity are high. The complicated calibrations are required.	
	Using one eye camera on HMD[6]	Not necessary to consider head movements.	HMD can give inconvenience or cybersickness to user	
	Using one head-mounted camera[5]	By using the camera on the wearable device, the accuracy of gaze tracking is high.	Since the user's eye and a frontal scene are captured in one image by the head-mounted camera, the eye image resolution is low, which degrades gaze detection accuracy.	
Binocular gaze detection method	Using four cameras and multiple illuminators[10] Using one camera and multiple illuminators[11]	Not requiring a user to wear any devices, this method can give convenience to user. In spite of the occlusion in one eye, gaze detection is possible by the other eye.	Gaze position is calculated as the simple average position of the detected gaze points of both eyes not considering the confidence level of two gaze points.	
	Using one camera and one NIR-LED illuminator considering the qualities of both eyes (Proposed method)	High accuracy of gaze estimation is obtained by using a fuzzy algorithm based on quality measurement of both eyes.	Gaze estimation time is increased by processing two eyes compared to that by monocular gaze detection.	

2 Proposed Gaze Detection Method

2.1 Overview of the Proposed Method

Figure 1 shows a flow chart of the proposed method. After capturing an image using a face-capturing camera, the eye region of the image is detected [steps (1) and (2) in Fig. 1]. A detailed explanation is provided in Sec. 2.3. The pupil center in the eye region is detected [step (3) in Fig. 1] (see Sec. 2.4). In addition, the SRs are localized [step (4) in Fig. 1], a detailed description of which is given in Sec. 2.5. To calculate the gaze point accurately, it is important to detect the exact centers of the pupil and the SR.[12,13] In Sec. 2.6, we describe the compensation method used to obtain a more accurate pupil boundary that is not distorted by an eyelid. The gaze point of each eye is then calculated using a geometric transform [step (5) in Fig. 1]. Then, the gaze positions of the two eyes are combined using a fuzzy algorithm based on quality measurements [step (6) in Fig. 1], a detailed description of which is provided in Sec. 2.7 through Sec. 2.9.

2.2 Proposed Gaze Detection System

Our gaze detection system includes two parts, an NIR illuminator and a single camera. As shown in Fig. 2, a conventional Web camera with a universal serial bus (USB) interface is used. The NIR cutting filter of the camera is

removed and a NIR passing filter is included in its place.[9,14–16] The NIR light-emitting diode (LED) illuminator is positioned below the camera. In this way, we can capture the facial image of the user without any adverse effects from changes in the visible environmental light conditions. In previous research, the edge between the pupil and the iris was
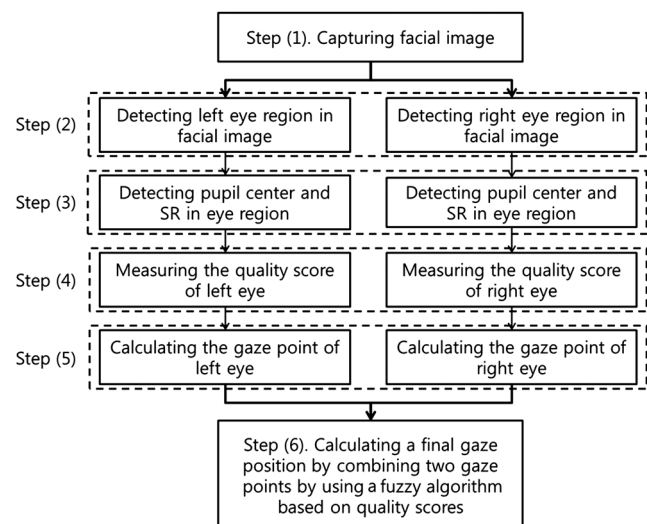
Fig. 1 Flow chart of the proposed method.

Fig. 2 Proposed gaze detection system.

Fig. 3 Example of images captured by our gaze detection system.

shown to be clearer with an NIR light of wavelength 850 nm or longer.[17] That is, the edge is hardly seen with the visible light in the special case of Asian people whose iris is dark. However, the camera sensor becomes less sensitive with an increase in wavelength.[18] Considering both the clearness of the edge and the camera sensitivity, we chose an NIR illuminator with a wavelength of 850 nm for our system. This wavelength avoids the glare from entering the user's eyes. To capture a magnified image of the user's face, a zoom lens is attached to the USB camera. The specifications of the camera and the NIR LED illuminator are as follows:

- *NIR LED illuminator*

  Wavelength: 850 nm
  Illuminative angle: $\pm 3 \times$ deg
  Number of NIR LEDs: 64

- *USB camera*

  Product name: Logitech Webcam C600[19]
  Spatial resolution: 1600×1200 pixels

2.3 *Eye Region Detection*

In this section, we describe the method of detecting the eye region in a facial image, which is step (2) in Fig. 1. Figure 3 shows examples of facial images captured by our gaze detection system. To reduce motion blur and increase the camera's depth of field (DOF), we used a zoom lens with a larger F# of 5.5 and reduced the camera exposure time. F# is usually defined by the ratio of focal length to the diameter of the camera lens (effective aperture).[20] Hence, the larger the focal length or the smaller the diameter of the camera lens is, the F# increases. The DOF is the Z-distance range in which a focused image can be obtained. In general, the DOF of a lens increases as the F# of the lens increases.

In addition, with a reduced camera exposure time, motion blur decreases. However, the captured image becomes darker with a lens with a larger F# and reduced camera exposure time, but a bright SR from the NIR illuminator can still be seen, as shown in Fig. 3. We, therefore, propose a method for detecting the eye region in a dark image based on the SR, as shown in Fig. 4. Our gaze detection system uses NIR light, and the eye region is detected using the SR near the pupil. By observing the captured image, the SR that occurs near the pupil of the user generally has a maximum gray value due to the high reflectance rate of the cornea surface. Therefore, when a window mask (3×3 pixels) is sequentially searched from the upper-left corner to the bottom-right corner, if all pixel values in the window mask are bigger than the fixed threshold value, it is determined that the window mask includes the SR. Three different results are possible: the user's eyes are not detected, only one eye is detected, and both eyes are detected.

Figure 4 shows an example of detecting an eye region based on the SR. Figure 4(a) shows an example in which the user's eyes are accurately detected. In this case, because the SR pixel values are brighter than the pixel values of the other skin regions, only the user's eyes are properly detected. However, in Fig. 4(b), there are three regions where all pixel values in the window mask are larger than the fixed threshold value. In this case, if one eye is closed, other skin regions can be incorrectly determined as an eye region. In addition, because the SR pixel value can be changed according to the Z-distance, using a fixed threshold value causes an eye detection error. We, therefore, propose the following method for detecting an eye region. In a captured facial image, when a user opens their eye, both the SR and the pupil can be seen. The SR is generally the brightest area, and the pupil is the darkest. Therefore, if the difference in value between the SR and the pupil is maximized within a certain region, the region is regarded as an eye candidate area. Figure 5 shows a flow chart of the proposed method used for eye region detection.

To detect an eye region, the input image is divided into $M \times N$ sub-blocks [step (2) of Figs. 5 and 6(b)]. The size of
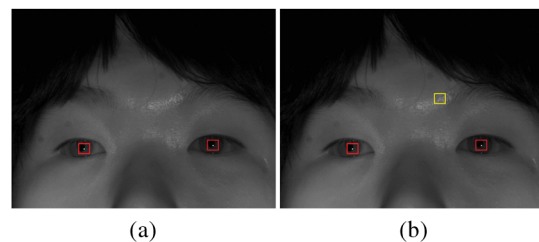

Fig. 4 Example of detecting the eye region based on the SR: (a) both eyes are detected, and (b) an erroneous detected region is found along with the detection of both eyes.
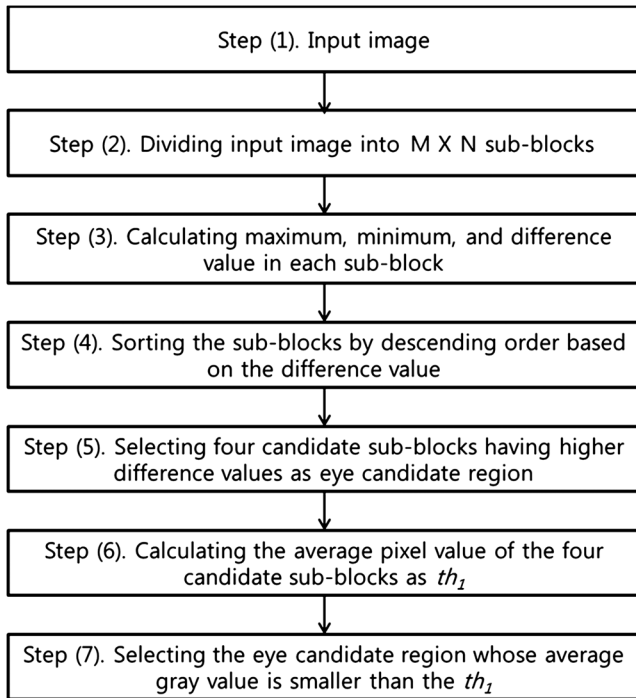
Step (1). Input image

↓

Step (2). Dividing input image into M X N sub-blocks

↓

Step (3). Calculating maximum, minimum, and difference value in each sub-block

↓

Step (4). Sorting the sub-blocks by descending order based on the difference value

↓

Step (5). Selecting four candidate sub-blocks having higher difference values as eye candidate region

↓

Step (6). Calculating the average pixel value of the four candidate sub-blocks as th_1

↓

Step (7). Selecting the eye candidate region whose average gray value is smaller than the th_1

Fig. 5 Flow chart of the proposed method for eye region detection.

(a)   (b)

(c)   (d)

Fig. 6 Image results of eye region detection based on the SR: (a) the original image, (b) the input image divided into $M \times N$ sub-blocks, (c) the resulting image after selecting four candidate regions, and (d) the final eye detection image.

Fig. 7 Image results of eye region detection.

each sub-block is 64×60 pixels, and total 500 sub-blocks (25×20) are defined as shown in Fig. 6(b). In each sub-block, the maximum and minimum pixel values are measured. Then, the difference between the maximum and minimum values is also calculated [step (3) of Fig. 5]. Based on the difference value, the sub-blocks are sorted in descending order [step (4) of Fig. 5]. For example, assuming that there are five sub-blocks that have the difference values of 200, 220, 205, 202, and 230, respectively, the sub-blocks are sorted as follows: the fifth sub-block (230), the second block (220), the third block (205), the fourth block (202), and the first block (200). The four candidate sub-blocks with the first, second, third, and fourth highest difference values are selected as eye candidate regions [step (5) of Figs. 5 and 6(c)]. With this example, the fifth sub-block (230), the second block (220), the third block (205), and the fourth-block (202) are selected. The average value of all candidate regions is then calculated as the threshold (th_1) [step (6) of Fig. 5]. The two regions, whose average gray level is smaller than th_1, are determined to be an eye region [step (7) of Figs. 5 and 6(d)].

Finally, once the two eye regions are detected, the position of the maximum gray value is determined to be the SR position. An example of images of eye region detection are shown in Fig. 7. In successive images, the accuracy of the eye region detection based on the SR is improved by tracking the SR within the specified range based on the average position of the Y coordinate of the eye position in the previous image.

2.4 Pupil Center Detection

To calculate a detected gaze, it is necessary to accurately detect the center position of the pupil and the SR. In this section, we describe the method for detecting the pupil center in an eye region, which corresponds to step (3) in Fig. 1.
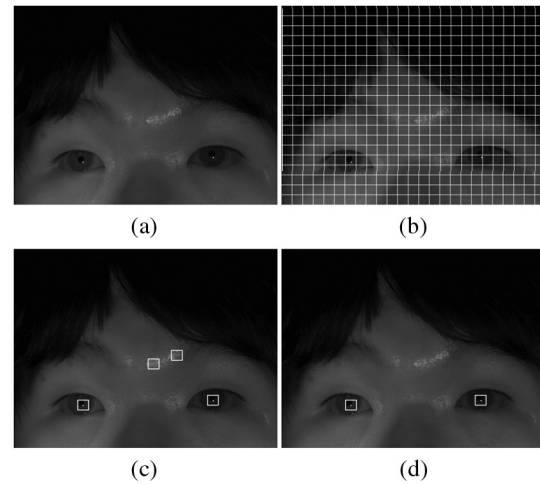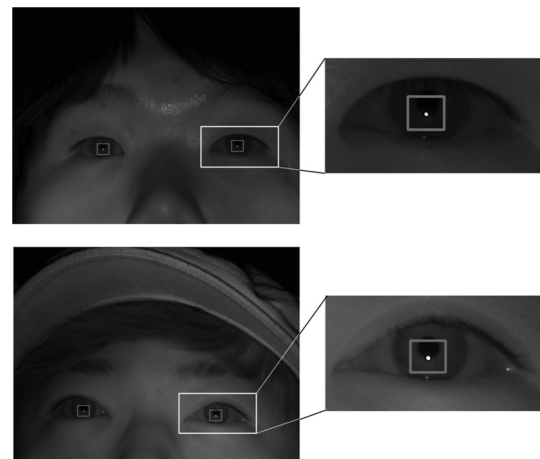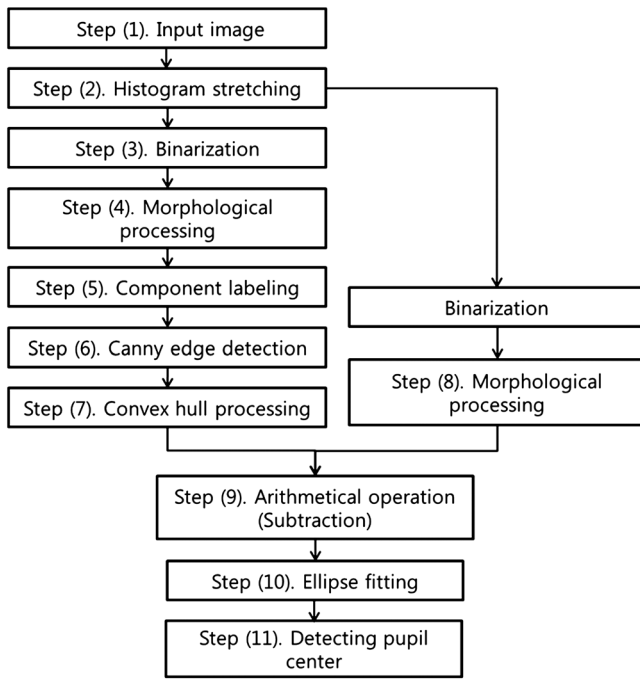
Figure 8 shows a flow chart of the proposed method used for pupil-center detection.

Based on the detected eye region, as shown in Fig. 7, the region of interest (ROI) for pupil detection can be defined as shown in Fig 9(a). The brightness of the ROI region is changed through a stretching of the histogram [step (2) of Figs. 8 and 9(b)]. For the stretching of the histogram, the average brightness of the input ROI region is calculated. Based on the average brightness, the brightness of the input ROI image is compensated. After the histogram is stretched, the ROI image is transformed through two binarization stages. The first stage is to separate the pupil from the other regions [binarization image 1, as shown in step (3) of Figs. 8 and 9(c)]. The second stage is to separate the SR from the other regions [binarization and morphology image 2, as shown in step (8) of Figs. 8 and 9(h)].[14,15,21] From the processed image achieved through the stretching of the histogram, we can obtain the maximum and minimum pixel values. The thresholds for binarization images 1 and 2 are determined based on the maximum and minimum pixel values, respectively, by which the binarized image can be robust
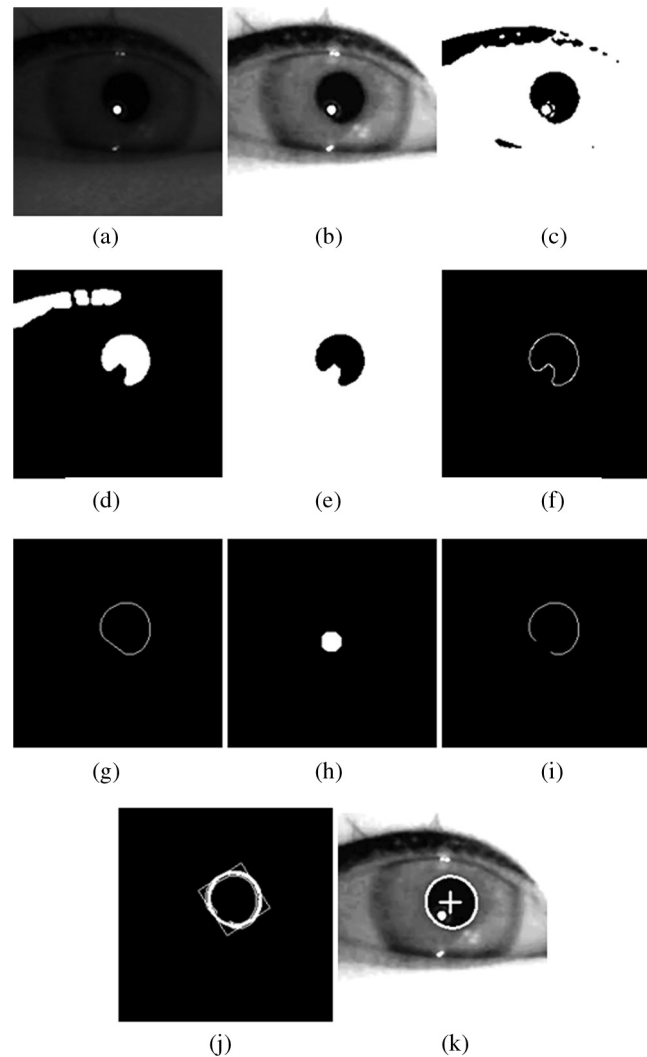
Step (1). Input image
Step (2). Histogram stretching
Step (3). Binarization
Step (4). Morphological processing
Step (5). Component labeling
Step (6). Canny edge detection
Step (7). Convex hull processing

Binarization
Step (8). Morphological processing

Step (9). Arithmetical operation (Subtraction)
Step (10). Ellipse fitting
Step (11). Detecting pupil center

Fig. 8 Flow chart of the proposed method used to detect the center of the pupil.

(a)　　　(b)　　　(c)
(d)　　　(e)　　　(f)
(g)　　　(h)　　　(i)
(j)　　　(k)

Fig. 9 Image results of pupil center detection: (a) the original image; the resulting images through (b) the stretching of the histogram, (c) the binarization of (b), (d) the morphological processing of (c), (e) the component labeling of (d), (f) the use of the canny edge detection algorithm on (e), (g) the convex hull of (f), (h) the SR binarization of (b), (i) the removal of the overlapped areas of (g) and (h) from (g), and (j) the application of the ellipse-fitting algorithm of (i); and (k) the final pupil detection image.

to variations in illumination. Regions other than the pupil region are then erased through morphological processing [step (4) of Figs. 8 and 9(d)]. With morphology processing, erosion and dilation are conducted four times each to erase any unnecessary parts in the eye image, such as the eyelashes. Figure 9(c) shows parts of the eyelashes. After the morphological processing shown in Fig. 9(d), the smaller parts of the eyelashes are merged together and erased. Next, the largest region is found through component labeling [step (5) of Fig. 8 and 9(e)]. The edge of the largest region is then detected using a canny edge detection algorithm [step (6) of Fig. 8 and 9(f)].[22] Most of the outer edge line of the largest region is extracted using a convex hull algorithm [step (7) of Figs. 8 and 9(g)].[21,23] Using the outermost edge line image [Fig. 9(g)] and the binarization and morphology image 2 [Fig. 9(h)], the image in Fig. 9(i) is obtained. Figure 9(i) is obtained by removing the overlapped area of Figs. 9(g) and 9(h) from Fig. 9(g). Figure 9(g) is obtained to compensate for the distortion of the pupil by the SR. The pupil center is then detected using an ellipse-fitting algorithm [step (10) of Figs. 8 and 9(j)]. The pupil shape is not usually a circular one.[24] In addition, when a user gazes at the monitor corner, the shape of the pupil is more distorted. Hence, we used the ellipse-fitting algorithm to detect the pupil boundary instead of circle fitting one considering the detection accuracy and speed. The result of the pupil detection is shown in Fig. 9(k).

Figure 10 shows the results of the pupil center detection. To obtain a more accurate gaze position, the detected pupil center is not represented as an integer value, but rather as a float value.

2.5 Corneal Specular Reflection Center Detection

In this section, we describe the method for detecting the corneal SR center in an eye region; this corresponds to step (3)

of Fig. 1. Based on the detected eye region, as shown in Fig. 7, the ROI for detecting the corneal SR is defined in Fig. 11(a). The ROI region is binarized, which discriminates the SR region from the other regions [Fig. 11(b)]. Because other regions, such as a reflection on the lachrymal gland, can have a similar gray level as that of the SR, component labeling is performed on the binarized image.[9,16] The region nearest to the detected pupil center is selected as the corneal SR region [Fig. 11(c)]. In addition, the geometric center of the selected SR region is determined as the SR center [Fig. 11(d)].[9,16]

To obtain a more accurate gaze position, the detected SR center is not represented as an integer value, but instead as a float value. As shown in Fig. 11(d), the geometric center of the selected SR region is determined as the SR center. That is, with the pixels of the selected SR region, the center positions of the X and Y axes are calculated. Since the X center position is calculated by dividing the summation of
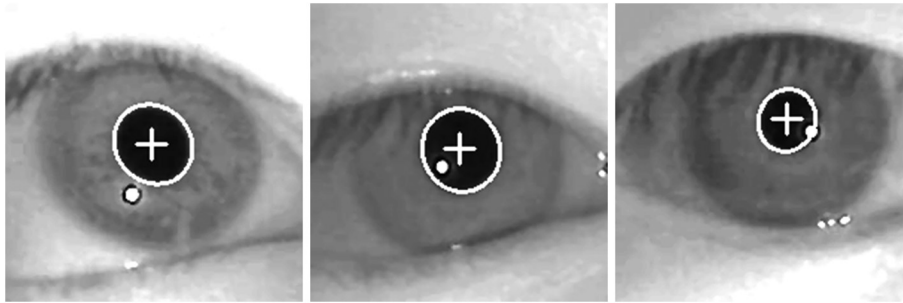
Fig. 10 Resulting images of pupil center detection.

all the *X* positions of the pixels (of the SR region) by the number of pixels, the calculated *X* center position can become a floating number (not an integer one). Like this, the calculated *Y* center position can also be a floating number. If we convert these floating numbers into integers by truncating the decimal fraction, the subpixel information of the SR position is not considered for calculating gaze position. Hence, in order to obtain a more accurate gaze position, we use the floating number without the truncation of the decimal fraction in the intermediate calculation. Consequently, the final calculated gaze position is represented as the floating number, and we represent it as the integer just at the final stage of calculation.

2.6 *Compensation for Pupil Distortion*

To calculate a detected gaze, it is necessary to accurately detect the pupil and the SR center position. The pupil is often covered by an eyelid, and pupil shape is frequently distorted, as shown in Fig. 12. This can degrade the accuracy of the pupil detection.

To compensate the distortion of the pupil shape, we compared two methods. The concept of the first method is shown in Fig. 13. We first find the horizontal axis that has the largest distance between two pixels of the pupil boundary. Hereafter, for convenience, we call this the horizontal major axis. The upper and lower pixels of the pupil boundary are then separated based on the horizontal major axis. By calculating the average distance between the center of the horizontal major axis and the upper (or lower) pixels, we can determine the occluded region (among the upper or lower pixels) that shows the smaller distance. We then replace the pixels of the occluded region (the upper pixels in Fig. 13) with the pixels of a nonoccluded area (the lower pixels in Fig. 13). The detailed steps of the first method are shown in Fig. 14.

First, the ROI region for the pupil detection is obtained as shown in Fig. 14(a). Through histogram stretching, binarization, morphological processing, and canny edge detection [Figs. 14(b) through 14(d)], the horizontal major axis in the image is detected using convex hull processing, as shown in Fig. 14(e).

When the horizontal major axis is found, the upper and lower pupil boundary pixels are divided as shown in Fig. 14(e). By calculating the average distance between the center of the horizontal major axis and the upper (or lower) pixels, we can determine the occluded region (among the upper or lower pixels) that shows the smaller distance. We then replace the pixels of the occluded region (the upper pixels in Fig. 14) with the pixels of the nonoccluded area (the lower pixels in Fig. 14), as shown in Fig. 14(f). In addition, we obtain the image of Fig. 14(g) for detecting
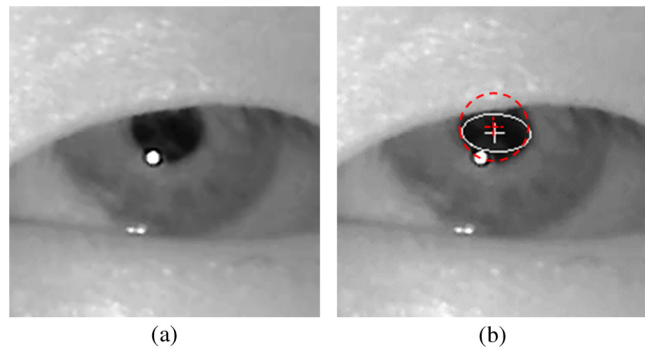
Fig. 12 Pupil distortion by an eyelid[25] and the results of detecting the pupil boundary and center: (a) the original image and (b) the resulting image after detecting the pupil boundary and center (the dotted lines show the correct pupil boundary and center, whereas the solid lines represent the detected pupil boundary and center).
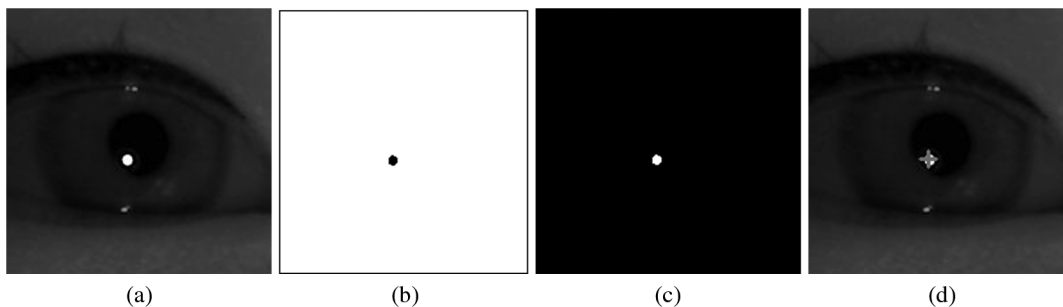
Fig. 11 Resulting images of corneal SR center detection: (a) the original image, the resulting images through (b) binarization and (c) component labeling, and (d) the final SR detection image.
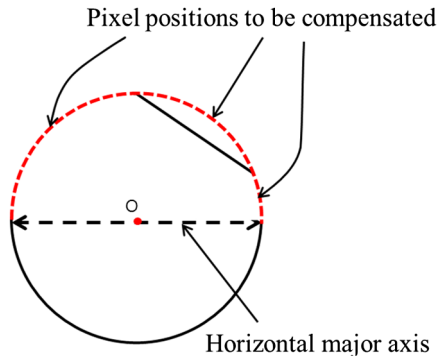
Fig. 13 Concept of compensation method 1.

the corneal SR region through binarization and morphological processing. First, we obtain the overlapped area of Figs. 14(f) and 14(g). Then, this overlapped region is removed from Fig. 14(f), and we can obtain Fig. 14(h). The reason why this overlapped region is removed is as follows. As shown in Fig. 14(d), the edge line detected by the canny operator includes the boundary of the corneal SR that is overlapped on the pupil boundary. Hence, the boundary of the corneal SR shows a concave shape attached on the pupil boundary, which causes the incorrect detection of a pupil boundary by the ellipse-fitting algorithm. To overcome this problem, we remove the overlapped region between the pupil boundary [Fig. 14(f)] and the SR region [Fig. 14(g)] and

used the remaining boundary of Fig. 14(h) for the pupil edge detection by the ellipse-fitting algorithm.

The pupil center is then detected using the ellipse-fitting algorithm, as shown in Fig. 14(i). The resulting images from compensation method 1 are shown in Fig. 15.

The second method for pupil shape compensation is similar to the first, except that, instead of replacing all pixels in the occluded area (the red-dotted pixels in Fig. 13), only the occluded pixels are replaced (the red-dotted pixels in Fig. 16). If a pixel (in the upper region) with a distance from the center of the horizontal major axis is smaller than the threshold, it is determined as a pixel to be compensated. Here, the threshold is determined as the average distance between the center of the horizontal major axis and the pixels of the lower region. The concept of the second method is shown in Fig. 16.

The detailed steps of the second method are shown in Fig. 17. First, the ROI for the pupil detection is obtained, as shown in Fig. 17(a). Through histogram stretching, binarization, morphological processing, and canny edge detection [Figs. 17(b) to 17(d)], the image in Fig. 17(d) is obtained. In addition, we obtain the image in Fig. 17(e) for detecting the corneal SR region through binarization and morphological processing. By removing the overlapped area of Figs. 17(d) and 17(e) in Fig. 17 from 17(d), we can obtain Fig. 17(f), and the horizontal major axis is found. Based on the horizontal major axis, the upper and lower pupil boundary pixels are divided, as shown in Fig. 17(f). By calculating the average distance between the center of the horizontal major axis and


(a) Original   (b) Histogram stretching   (c) Binarization, morpholgical processing, and component labeling   (d) Canny edge detection

(e) Find major horizontal axis after convex hull processing   (f) Replacing the upper pupil boundary pixels by the lower ones   (h) Removing the overlapped area of (f) and (g) from (f)   (i) Result

(g) Binarization and morphological processing for detecting corneal SR region
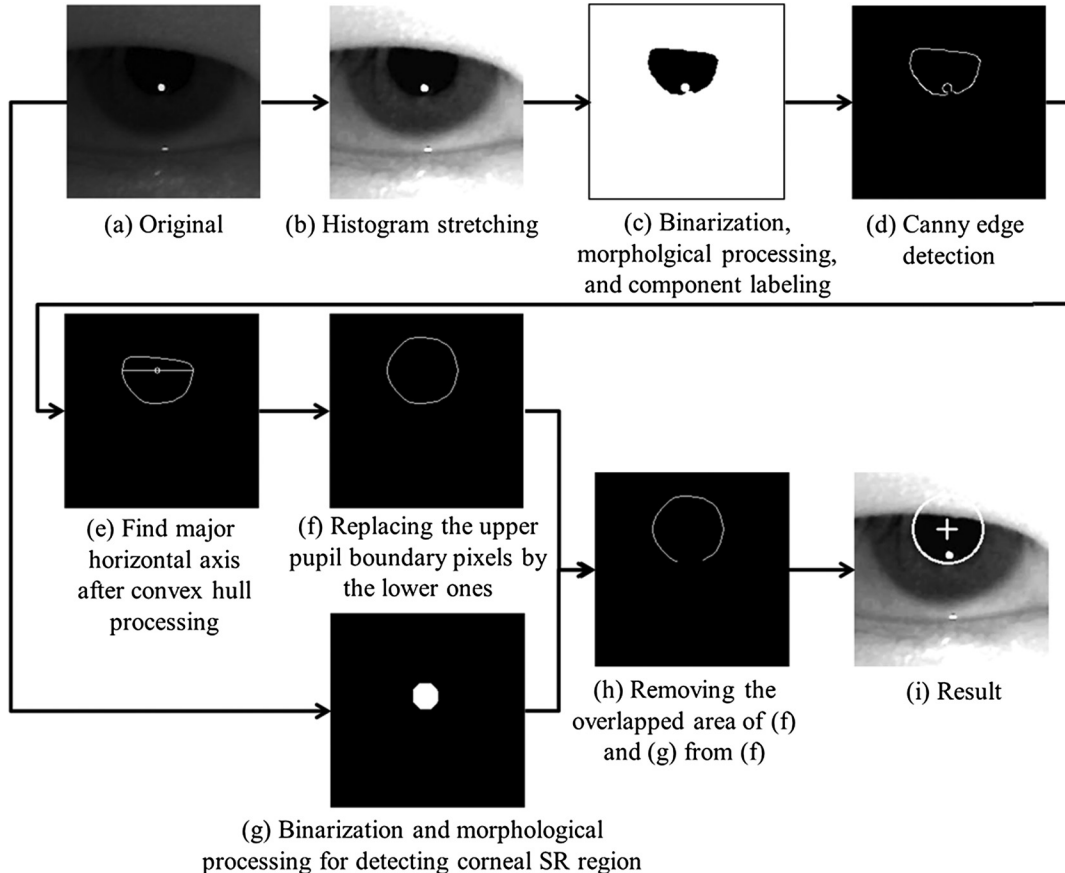
Fig. 14 Flow chart of compensation method 1 for the distortion of the pupil region by the eyelid.

Fig. 15 Resulting images from compensation method 1.

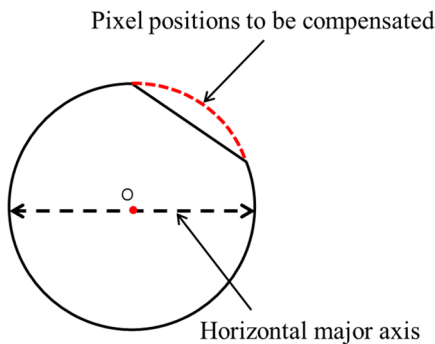Pixel positions to be compensated

Horizontal major axis
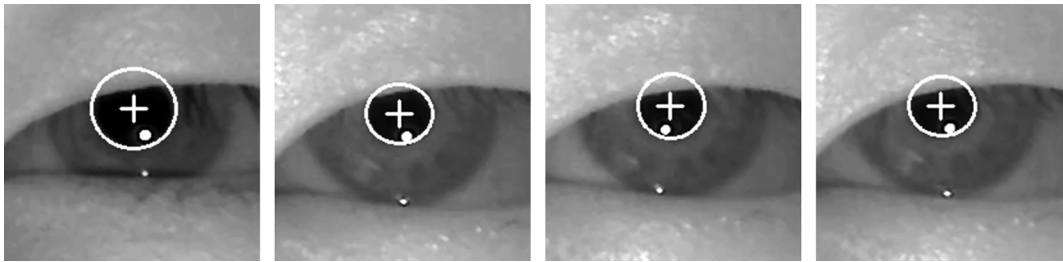
Fig. 16 Concept of compensation method 2.

the upper (or lower) pixels, we can determine the occluded region (among the upper or lower pixels) that shows the smaller distance. In addition, the average distance (D_1) between all the points of the lower region and the center of the horizontal major axis is calculated as the threshold. Only the lower region is used to obtain the threshold because the eyelid usually covers the pupil in the upper area. The distances between all the points in the upper region and the center of the horizontal major axis are calculated. If a distance is smaller than the threshold, the pixel point is compensated by D_1, as shown in Fig. 17(g). The outermost line is then obtained using the convex hull algorithm, as shown in Fig. 17(h). Additionally, the pupil center is detected using the ellipse-fitting algorithm, as shown in Fig. 17(i). The image results of compensation method 2 are shown in Fig. 18.
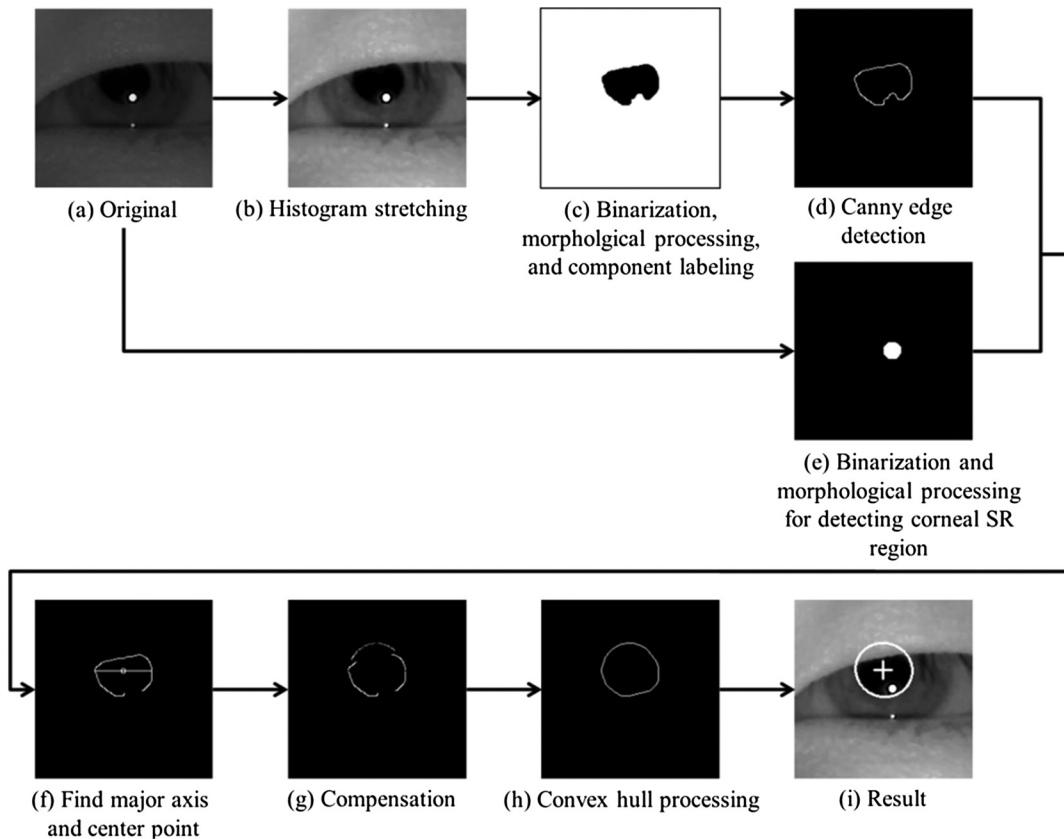

(a) Original

(b) Histogram stretching

(c) Binarization, morphlgical processing, and component labeling

(d) Canny edge detection

(e) Binarization and morphological processing for detecting corneal SR region

(f) Find major axis and center point

(g) Compensation

(h) Convex hull processing

(i) Result

Fig. 17 Flow chart of compensation method 2 for the distortion of a pupil region by the eyelid.
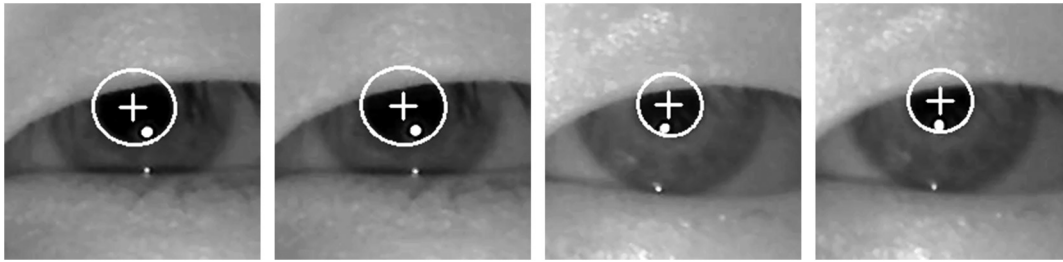
Fig. 18 Image results from compensation method 2.

2.7 Quality Score Measurements of Both Eyes

Our proposed gaze detection method uses a fuzzy algorithm to combine the gaze points of both eyes based on their quality scores. In this section, the method for obtaining the quality scores of both eyes is described, corresponding to step (4) of Fig. 1. We use four quality measurements: distortion (occlusion) by the eyelid, distortion (occlusion) by the SR, the level of circularity of the

pupil, and the distance between the pupil boundary and the SR center.

Factor 1 (the amount of distortion by an eyelid) is calculated as the ratio of pixels of the pupil edge occluded by an eyelid [Fig. 19(a) to the number of pixels of an undistorted pupil edge in the upper region (based on the horizontal major axis) [Fig. 19(b)]. Factor 1 can be expressed using Eq. (1)

$$\text{Factor 1} = \frac{\text{the number of pixels of pupil edge occluded by the eyelid}}{\text{the number of pixels of undistorted pupil edge in upper region}}. \tag{1}$$

The second quality measure is the amount of distortion caused by the SR. Factor 2 is calculated as the ratio of the number of pixels of a pupil edge occluded by the SR [the number of pixels along the dotted line of square B in

Fig. 20(b)] to the number of pixels in the distorted pupil edge in the lower part [the number of pixels along the solid line of square A in Fig. 20(a)]. Factor 2 can be expressed through Eq. (2)

$$\text{Factor 2} = \frac{\text{the number of pixels of pupil edge occluded by the SR}}{\text{the number of pixels of distorted pupil edge in lower region}}. \tag{2}$$

The third quality measure is the level of circularity of the pupil. Factor 3 is calculated as the ratio of the major to minor axes lengths of the pupil. The major and minor axes are detected using an ellipse-fitting algorithm, as described in Sec. 2.4 and Sec. 2.6. A pupil boundary is usually similar to a circular shape when a user gazes at the center of a monitor and becomes elliptical in shape when the user gazes at the corner of a monitor. In the latter case, factor 3 increases, and the consequent error when detecting an accurate pupil boundary and center can be increased. We, therefore, use factor 3 as the third quality measure. The closer to the circle

shape the pupil boundary is, the smaller the factor 3 becomes (consequently close to 1). Factor 3 can be expressed through Eq. (3)

$$\text{Factor 3} = \frac{\text{the length of major axis of pupil}}{\text{the length of minor axis of pupil}}. \tag{3}$$
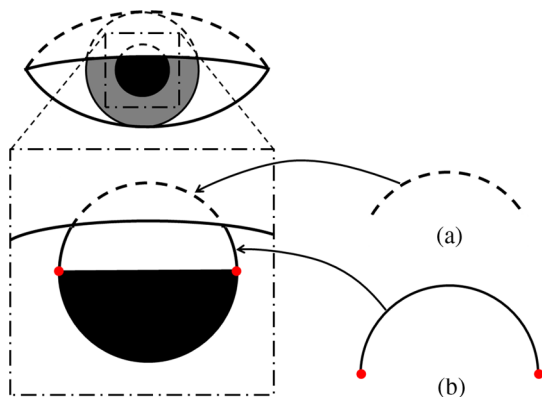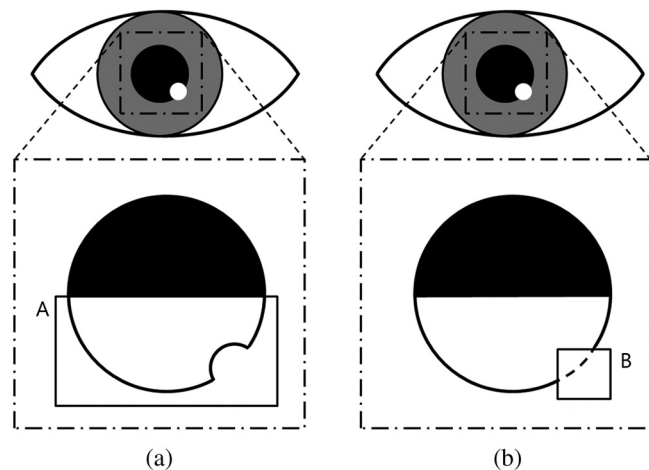

(a)

(b)

Fig. 20 Concept image of factor 2 (a) pupil edge region in the case where the distortion is occluded by the SR (solid line in square A) (b) occluded pupil edge by the SR (dotted line in square B).

(a)

(b)

Fig. 19 Concept image of factor 1.

The fourth quality measure is the distance between the SR center and the pupil boundary pixel nearest to the SR center. The longer the distance between the pupil boundary pixel and the SR center, the less distortion that occurs because of the SR, as shown in Fig. 21. Therefore, the larger the fourth quality measure, the more accurate the pupil center detection.

Using these four quality measurements, we can evaluate the accuracy of the detected pupil centers and the confidence levels of the calculated gaze points of both the left and right eyes. Based on the four quality measurements, the weight values for the gaze points of the left and right eyes are determined using a fuzzy algorithm (see Sec. 2.8).

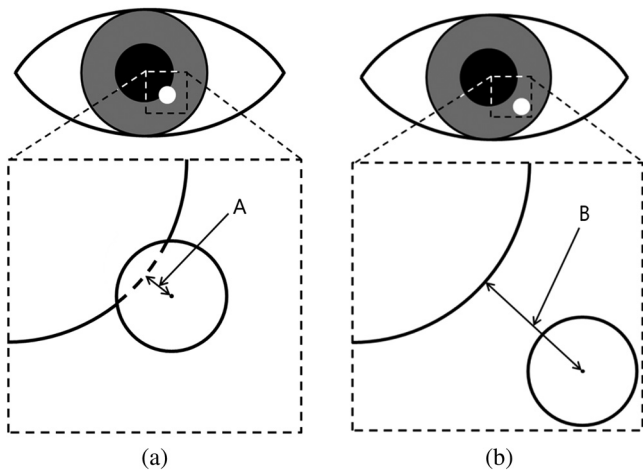## 2.8 Obtaining the Weight Value for the Gaze Position based on a Fuzzy Algorithm

In this section, the method for obtaining the weight values of the gaze points of the left and right eyes based on a fuzzy algorithm is described. The final gaze position can be calculated more accurately by combining the gaze points of both eyes when considering the weight value. The basic concept of combining the gaze points of both eyes is shown in Fig. 22.

Based on the four quality factors of each eye, as explained in Sec. 2.7, the gaze point score of each eye is obtained based on the fuzzy algorithm, as shown in Fig. 23. The weight values for the gaze positions of the left and right eyes are then determined using Eq. (4). The final gaze position is determined using the gaze positions of the left and right eyes when considering these weight values
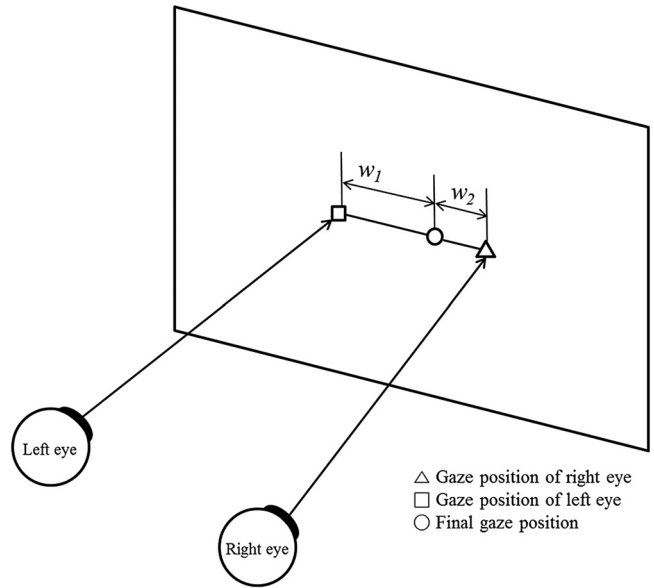
$$w_1 = \frac{\text{Score}_{\text{lefteye}}}{\text{Score}_{\text{lefteye}} + \text{Score}_{\text{righteye}}},$$

$$w_2 = \frac{\text{Score}_{\text{righteye}}}{\text{Score}_{\text{lefteye}} + \text{Score}_{\text{righteye}}}. \qquad (4)$$

To obtain the scores from Eq. (4), a fuzzy algorithm is used, as shown in Fig. 23. Using w_1 and w_2 and the gaze

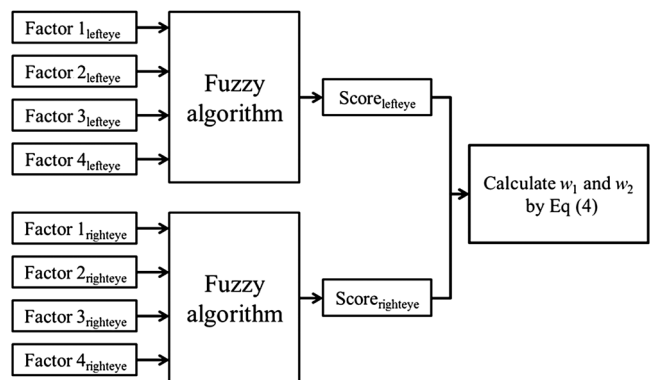**Fig. 22** Concept of combining the gaze points of both eyes.

points $[G_{\text{eye}}(x, y)]$ of each eye, the final gaze position $[G(x, y)]$ is obtained using Eq. (5)

$$G(x) = w_1 \times G_{\text{lefteye}}(x) + w_2 \times G_{\text{righteye}}(x)$$

$$G(y) = w_1 \times G_{\text{lefteye}}(y) + w_2 \times G_{\text{righteye}}(y). \qquad (5)$$

To use the fuzzy algorithm, it is important to determine the membership function. The membership function is designed according to the input value (factors 1 through 4 of Sec. 2.7), as shown in Fig. 24.

The larger the values of factors 1 through 3 [Eqs. (1)–(3)], the larger the distortions of the pupil shape that occur. This usually decreases the confidence level of the gaze detection accuracy. However, for factor 4, a larger value indicates that the distortion of the pupil shape is smaller. This generally increases the confidence level of the gaze detection accuracy. All factors for the input value of the fuzzy algorithm are normalized within the range of 0 to 1 and for factor 4, by subtracting the value from 1 after normalization. All factors, 1 through 4, with smaller values represent a higher confidence level of the gaze detection accuracy.

Fig. 21 Concept image of factor 4 (a) when the SR occludes the pupil edge (dotted line) (the shortest distance (A) between the pupil boundary pixel and the SR center), and (b) when the SR does not occlude the pupil edge (the shortest distance (B) between the pupil boundary pixel and the SR center).

Fig. 23 Procedure for calculating the scores using a fuzzy algorithm and the two weight values determined through Eq. (4).
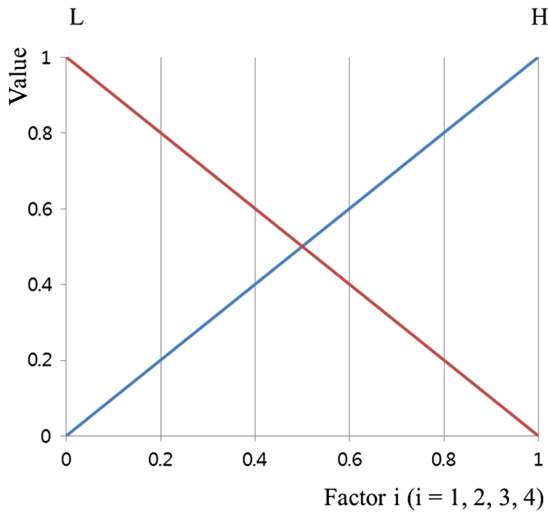
Fig. 24 Input fuzzy membership function.

The input values of the fuzzy algorithm can be categorized into two types: low (L) and high (H), as shown in Fig. 24. The fuzzy membership function for the input value is designed as shown in Fig. 24. The output membership function of the fuzzy algorithm is shown in Fig. 25. In addition, the output value (the score in Fig. 23) can be categorized as: low (L), middle (M), and high (H). The relationships between the input (factors 1 through 4) and output values (scores) are defined using the fuzzy rules shown in Table 2. As explained previously, factors 1 through 4 with smaller values represent a higher confidence level of gaze detection accuracy. Thus, if the value of all the factors 1 through 4 of one eye are low (L), for example, it means that the accuracy of the gaze point of the eye can be considered high, and we assign the output value (score of Fig. 23) as high (H). If the value of two of the factors 1 through 4 are low (L) and the other two factors are high (H), we assign the output value (score of Fig. 23) as middle (M). If the value of all the factors 1 through 4 of one eye is high (H), it means that the accuracy of the gaze point of that eye can be considered low, and we assign the output value (score of Fig. 23) as low (L).
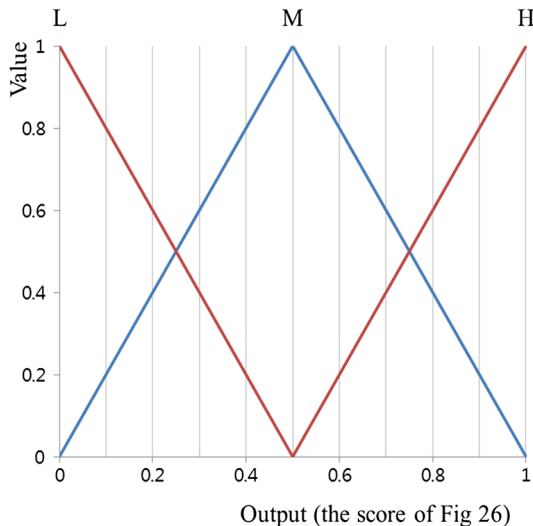

Fig. 25 Output fuzzy membership function.

Table 2 Fuzzy rules of the relationship between the input (factors 1 through 4) and output values (scores).

| Factor 1	Factor 2	Factor 3	Factor 4	Output
L	L	L	L	H
			H	H
		H	L	H
			H	M
	H	L	L	H
			H	M
		H	L	M
			H	L
H	L	L	L	H
			H	M
		H	L	M
			H	L
	H	L	L	M
			H	L
		H	L	L
			H	L

The output (score) can be obtained using the fuzzy membership function and fuzzy rules. With one input value, we can obtain two outputs using the input membership function shown in Fig. 24. Because the number of inputs is four (factors 1 through 4), eight outputs are obtained using the input membership function.

For example, as shown in Fig. 26, two outputs [0.3 (L) and 0.7 (H)] are obtained by factor 1. Another two outputs [0.3 (L) and 0.7 (H)] are obtained by factor 2. In this way, factors 3 and 4 also produce two outputs [0.3(L) and 0.7(H)], respectively. Based on these four pairs of outputs, we can obtain the combined set as {[0.3 (L), 0.3 (L), 0.3 (L), 0.3 (L)], [0.3 (L), 0.3 (L), 0.3 (L), 0.7 (H)], [0.3 (L), 0.3 (L), 0.7 (H), 0.3 (L)], ... [0.7 (H), 0.7 (H), 0.7 (H), 0.7 (H)]}. With one subset, we can determine one value (0.3 or 0.7) and one symbol (L, M, or H) based on the minimum or the maximum method and fuzzy rules of Table 2.[26,27]

For example, with one subset [0.3 (L), 0.7 (H), 0.7 (H), 0.7 (H)], we can select 0.3 based on the minimum method. In addition, we can obtain L as the output based on Table 2 (when factors 1 through 4 are L, H, H, and H, respectively, we can state that the output is L, as shown in Table 2). We therefore obtain 0.3 (L) from [0.3 (L), 0.7 (H), 0.7 (H), 0.7 (H)] and call 0.3 (L) the inference value (IV) in this paper. If we apply the maximum method, we obtain 0.7 (L) as the IV. Because the number of subsets is 16 ($2 \times 2 \times 2 \times 2$), the total number of IVs is also 16.

With one IV, we can obtain either one or two outputs (scores), as shown in Fig. 27. As shown in Fig. 27, if the IV is 0.7 (M), the corresponding outputs are S_1 and S_3. In this
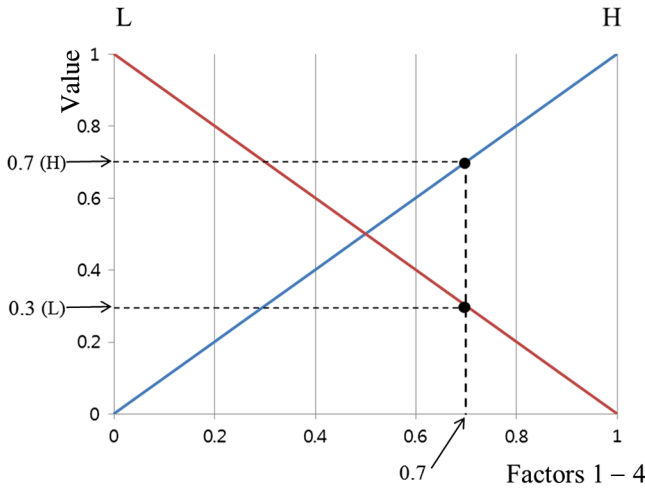
**Fig. 26** Example of obtaining two output values from one input using the input membership function.

way, we can obtain multiple outputs $(S_1, S_2, \ldots S_N)$ from 16 IVs. From the multiple outputs $(S_1, S_2, \ldots, S_N)$, we can determine one final output score based on a defuzzification method.[26,28] There are various defuzzification methods, of which five were chosen for this experiment: first of maxima (FOM), last of maxima (LOM), middle of maxima (MOM), mean of maxima (MeOM), and the center of gravity (COG)[26,28] FOM selects the first output IV calculated by the maximum. LOM selects the last output IV calculated by the maximum. MOM selects the middle of the first and last output IVs calculated by the maximum. MeOM selects the mean of the output IVs calculated by the maximum.[28]

In Fig. 27, when all the IVs are assumed to be 0.7 (H), 0.6 (M), and 0.3 (L), the maximum IV is 0.7 (H). In addition, the two outputs $(S_1$ and $S_3)$ are calculated using 0.7 (H). The final outputs (scores) by FOM, LOM, MOM, and MeOM are therefore S_1, S_3, $(S_1 + S_3)/2$, and $(S_1 + S_3)/2$, respectively. For the COG, the output (score) is determined as S in Fig. 27(b) from the geometrical center [G in Fig. 27(b)] of the union area of three regions $(P_1, P_2,$ and $P_3)$.

2.9 Calculating the Gaze Position by Combining Both Eyes with the Weight Values

With the output scores (Fig. 27) of the left and right eyes, we can obtain the weight values for the gaze positions of both eyes using Eq. (4). Using Eq. (5), we can obtain the final gaze position considering the weight values. To do so, we obtain the gaze positions of the left eye $[G_{\text{lefteye}}(x), G_{\text{lefteye}}(y)]$ of Eq. (5) and right eye $[G_{\text{righteye}}(x), G_{\text{righteye}}(y)]$ of Eq. (5) as follows. At the initial calibration stage, each user gazes at nine positions $(Q_1, Q_2, \ldots, Q_9)$ on a monitor, as shown in Fig. 28. The nine positions of the pupil centers are obtained as shown in Fig. 29. Because the user's head may move while gazing at the nine positions, the nine positions of the pupil centers are readjusted based on the SR positions in each image. We then obtain the nine positions of the pupil centers $(P_1, P_2 \ldots, P_9)$, and can define the relationship between each pupil subregion and monitor subregion. For example, pupil subregion 1 corresponds to monitor subregion 1. To obtain this relationship, we applied a geometric transform[9,16] and obtained the four matrices of the geometric transforms,

(a)

(b)
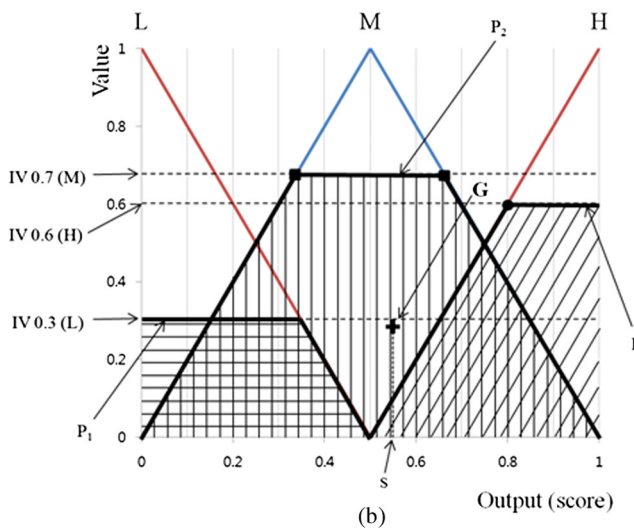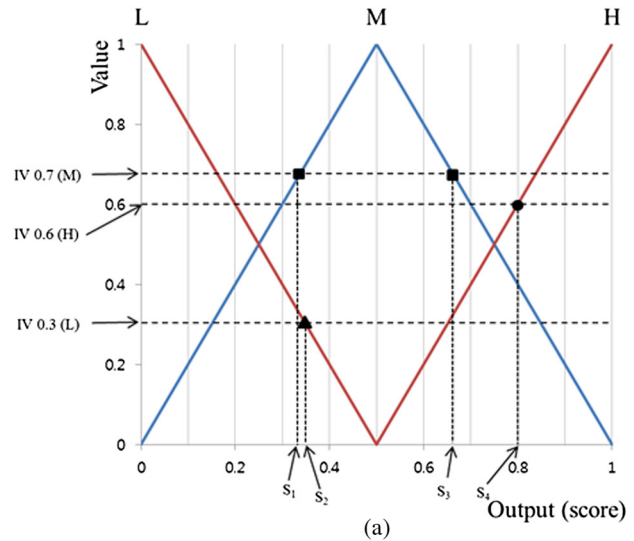
Fig. 27 Example of obtaining the output (score) from the IV and one final output (score) by the defuzzification method: (a) FOM, LOM, MOM, and MeOM, and (b) COG.
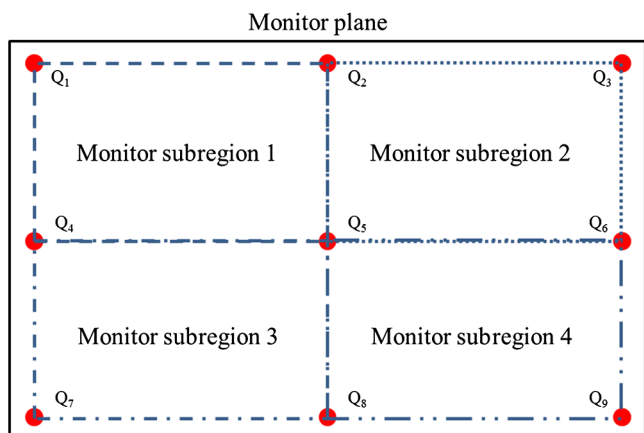
Fig. 28 Subregions of the monitor plane for a geometric transform.
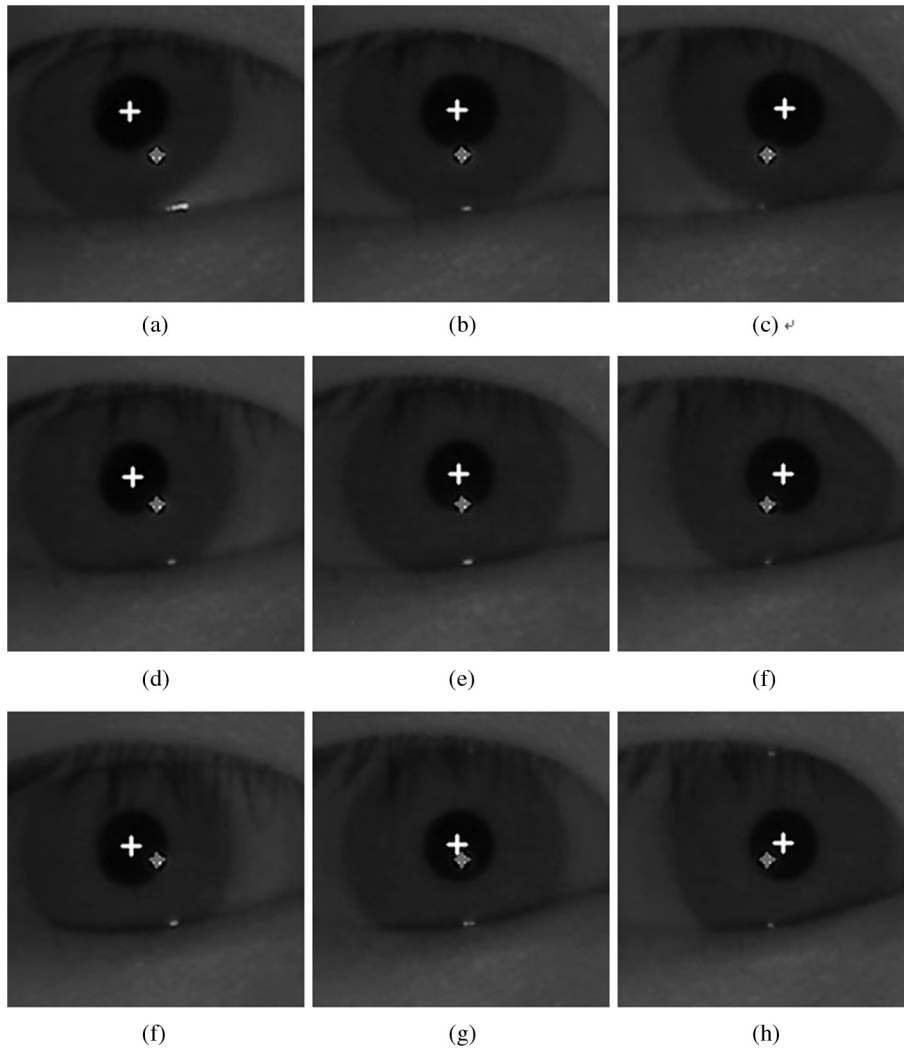
Fig. 29 Example positions of the pupil and the SR centers when the user gazes at nine calibration positions: (a) the upper-left calibration point (Q_1), (b) upper-center calibration point (Q_2), (c) upper-right calibration point (Q_3), (d) middle-left calibration point (Q_4), (e) middle-center calibration point (Q_5), (f) middle-right calibration point (Q_6), (g) lower-left calibration point (Q_7), (h) lower-center calibration point (Q_8), and (i) lower-right calibration point (Q_9).

as shown in Fig. 30.[9] If the measured pupil center belongs to pupil subregion 1, geometric transform matrix 1 is used to calculate the gaze position on the monitor. If it belongs to pupil subregion 4, geometric transform matrix 4 is used to calculate the gaze position on the monitor.[9]

Using this, we can obtain the gaze positions of the left eye $[G_{lefteye}(x), G_{lefteye}(y)]$ of Eq. (5) and the right eye $[G_{righteye}(x), G_{righteye}(y)]$ of Eq. (5). In addition, using Eq. (5), we can obtain the final gaze position on the monitor by considering the weight values.

3 Experiment Results

The proposed gaze detection method was tested on a desktop computer with an Intel i7 3.33 GHz CPU and 6 GB of RAM. The algorithm was implemented using Microsoft foundation class based C++ programming and the DirectX 9.0 SDK.

For the first experiment, we compared the accuracies of the two compensation methods of pupil distortion (occlusion) described in Sec. 2.6. We used 1400 images in the experiment. Seven hundred of the images included a pupil

not occluded by an eyelid; the others included an eyelid occluded pupil. Example eye regions of the experimental images are shown in Fig. 31.

The results are summarized in Tables 3 and 4. The error rate was measured as the ratio of the number of images (where the pupils were not correctly detected) to the total number of images. As shown in Tables 3 and 4, we can confirm that the compensation method 2 outperforms method 1. Figure 32 shows the image results of correctly detecting the pupil centers using methods 1 and 2 when the pupils are not occluded by an eyelid. Figure 33 shows the image results of incorrectly detecting the pupil centers using methods 1 and 2 when the pupils are not occluded by an eyelid.

The errors in Fig. 33 occurred because of the incorrect detection of the horizontal major axis, in which considerably many pixels of the upper part (based on the horizontal major axis) are replaced with pixels of the lower part. Figure 34 shows the image results of correctly detecting the pupil centers using methods 1 and 2 when the pupils are occluded by an eyelid. Figure 35 shows the image results when
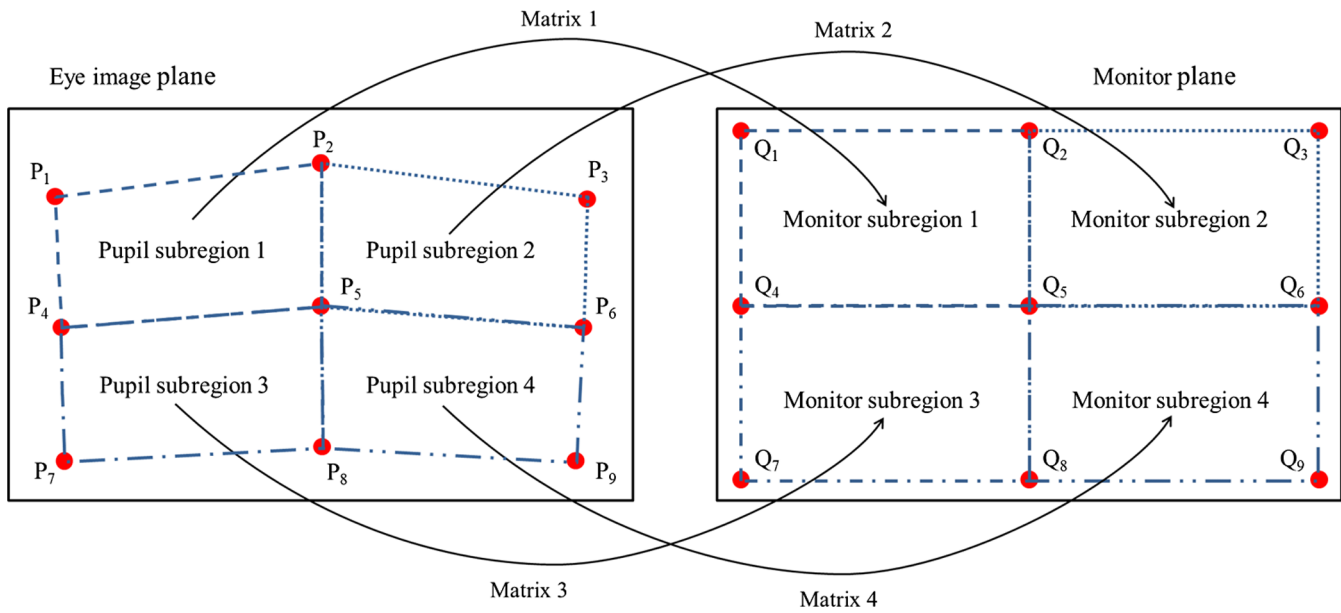
Fig. 30 Relationship between each pupil subregion and monitor subregion.
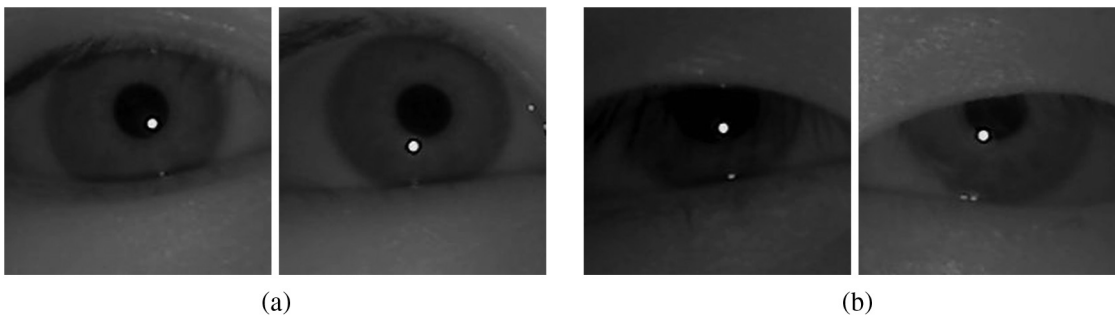

Fig. 31 Example eye images where the pupils (a) are not occluded and (b) are occluded by an eyelid.

incorrectly detecting the pupil centers using methods 1 and 2 when the pupils are occluded by an eyelid. Method 2 outperforms method 1 for the following reason. As shown in Figs. 13 and 16, all pixels of the upper region (where an occlusion occurs) are replaced by those of the lower region in method 1. However, only the occluded pixels of the upper region are replaced by the occluded pixels of the lower region in method 2. A more accurate compensation can, therefore, be achieved using method 2. When a severe occlusion by an eyelid occurs, the horizontal major axis (Figs. 13 and 16) can be incorrectly detected. This causes an inaccurate detection of the pupil boundary and center.

In the second experiment, we measured the changes in factors 1 through 4 of Sec. 2.7 according to the level of

distortion of the pupil and the SR position. As shown in Fig. 36, we can confirm that factor 1 [distortion (occlusion) of a pupil by an eyelid] is changed in accordance with the level of the eyelid occlusion. When the user's eye opens completely, as shown in Fig. 36(a), the factor 1 is 0. When the user's eyelid covers the pupil slightly, as shown in Fig. 36(b), the factor 1 is 0.334. Finally, when the user's eyelid covers the pupil more, as shown in Fig. 36(c), the factor 1 is 0.593. Based on these results, the more the eyelid covers the pupil, the higher factor 1 becomes. We can confirm that factor 1 represents the level of distortion of a pupil by an eyelid.

As shown in Fig. 37, we can confirm that factor 2 [distortion (occlusion) of the pupil by the SR] is changed based

Table 3 Comparison of error rates of compensation methods 1 and 2 (using images in which the pupils were not occluded) (unit: %).

	Method 1	Method 2
Error rate	0.29 (2/700)	0.14 (1/700)

Table 4 Comparison of error rates of compensation methods 1 and 2 (using images in which the pupils were occluded) (unit: %).

	Method 1	Method 2
Error rate	28.14 (197/700)	6.29 (44/700)

(a)

(b)
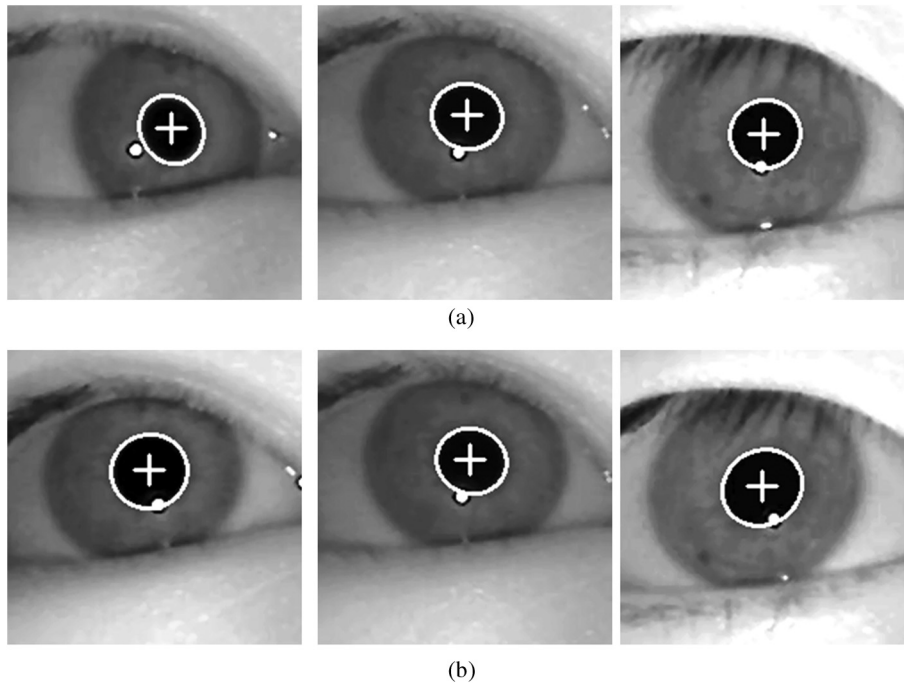
Fig. 32 Image results of correctly detecting the pupil centers using methods 1 and 2 when the pupils are not occluded by an eyelid using (a) method 1 and (b) method 2.

on the level of occlusion by the SR. When the position of the SR is inside the pupil, as shown in Fig. 37(a), the factor 2 is 0. When the position of the SR is on the edge of the pupil, as shown in Fig. 37(b), the factor 2 is 0.233. When the position of the SR is outside the pupil, as shown in Fig. 37(c), the factor 2 is 0. Based on these results, when the SR is on the edge of the pupil, factor 2 is larger than 0. This means that a higher factor 2 implies a more distorted (occluded) pupil. We can confirm that factor 2 represents the level of pupil distortion by the SR.

It was confirmed that factor 3 (the level of circularity of the pupil) is changed according to the user's gaze point. First, when a user gazes at the upper-left corner of the monitor, as shown in Fig. 38(a), the factor 3 is 1.223. When a user gazes at the center of the monitor, as shown in Fig. 38(b), the factor 3 is 1.073. When a user gazes at the upper-right corner of the monitor, as shown in Fig. 38(c), the factor 3 is 1.171. Based

on these results, we can confirm that factor 3 represents the level of pupil circularity.

Finally, the change in factor 4 (the distance between the SR center and the pupil boundary pixel nearest to the SR center) is measured according to the distance between the pupil boundary and the SR center. When the SR is inside the pupil, as shown in Fig. 39(a), the factor 4 is 26.178. When the SR is on the edge of the pupil, as shown in Fig. 39(b), the factor 4 is 1.516. When the SR is outside the pupil, as shown in Fig. 39(c), the factor 4 is 23.51. From these results, we can confirm that factor 4 represents the distance between the SR center and the pupil boundary pixel nearest to the SR center.

With Figs. 36 and 37, we show the changes of factors 1 and 2 according to the distortions of the pupil by an eyelid and the SR, respectively. In addition, with Figs. 38 and 39, we show the changes of factors 3 and 4 according to the gaze

(a)   (b)
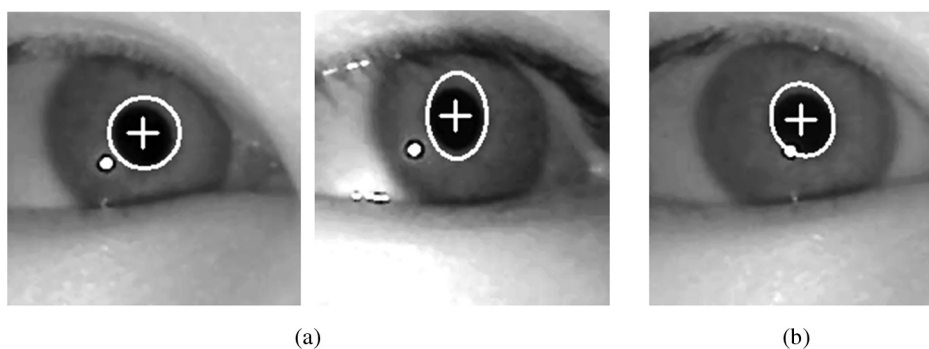
Fig. 33 Images of incorrect detection of the pupil center when the pupils are not occluded by an eyelid using (a) method 1 and (b) method 2.

(a)

(b)
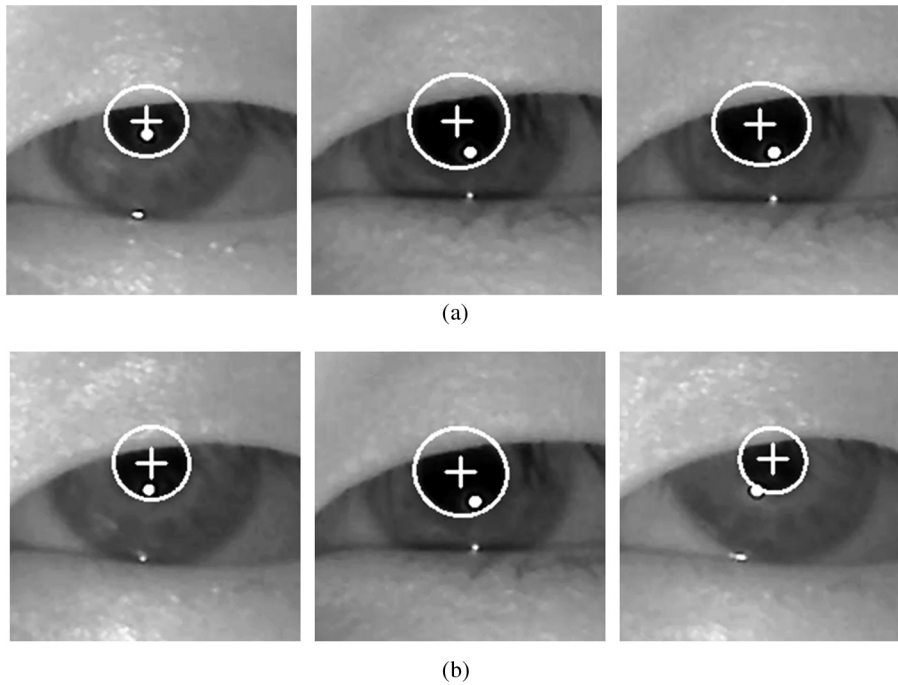
Fig. 34 Image result of correctly detecting the pupil centers when the pupils are occluded by an eyelid using (a) method 1 and (b) method 2.

positions on a monitor, respectively. From them, we can confirm that factors 1 through 4 represent the degree of quality of the eye image.

In the third experiment, we measured the accuracy of the gaze detection method. A 19-inch monitor with a resolution of 1280×1024 pixels was used for the experiment. Twenty subjects participated in the experiment, and each subject underwent six trials. As explained in Sec. 2.8, either the minimum or the maximum method can be selected for obtaining the IV. In addition, one of the five defuzzification methods, COG, FOM, LOM, MOM, or MeOM, can be used to obtain the final output score. We compared the gaze detection accuracies from both the minimum and the maximum methods and compared them based on the defuzzification method


(a)

(b)
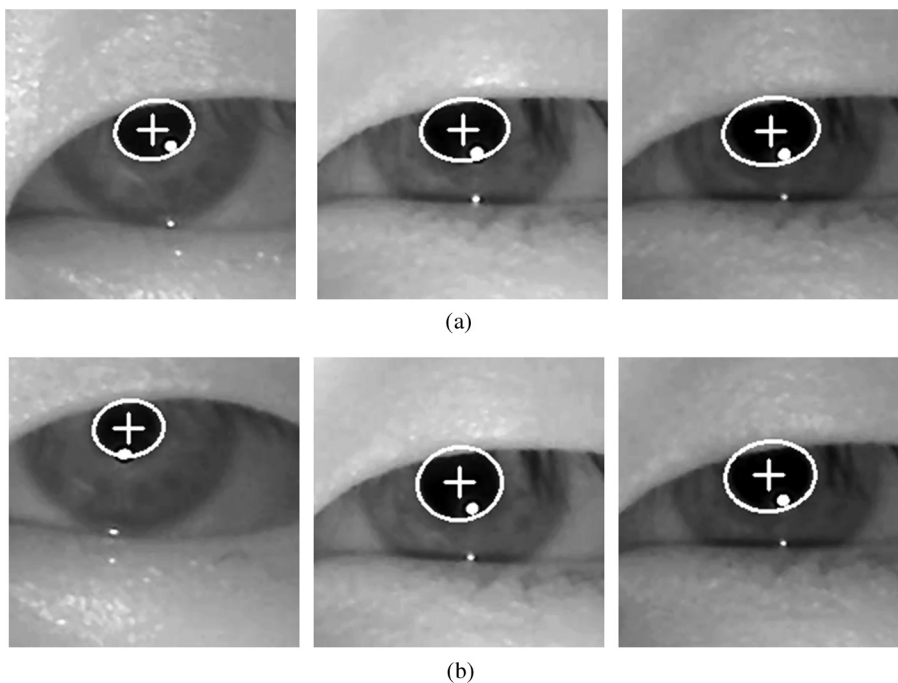
Fig. 35 Images of incorrect detection of the pupil center when the pupils are occluded by an eyelid using (a) method 1 and (b) method 2.

(a) (b) (c)

Fig. 36 Examples showing the change in factor 1 (distortion of a pupil by an eyelid) based on the level of occlusion by the eyelid when (a) the eye is open completely (factor 1 of 0), (b) the eyelid covers the pupil slightly (factor 1 of 0.344), and (c) the eyelid covers the pupil more than in (b) (factor 1 of 0.593).

(a) (b) (c)
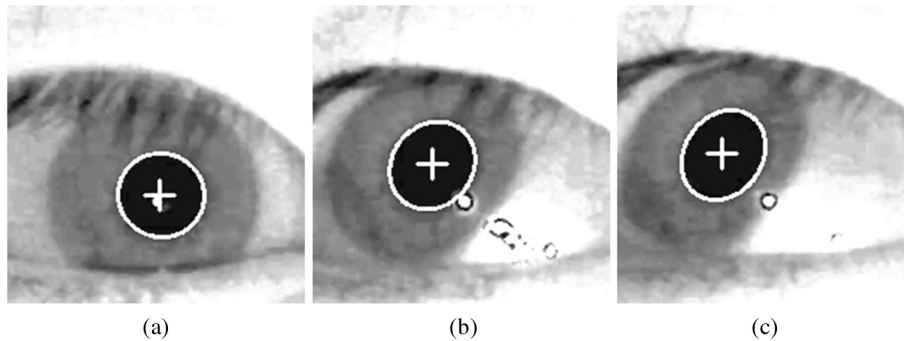
Fig. 37 Examples showing the change in factor 2 (distortion of a pupil by the SR) based on the level of occlusion by the SR when its position is (a) inside the pupil (factor 2 of 0), (b) on the edge of the pupil (factor 2 of 0.233) and (c) outside the pupil (factor 2 of 0).


(a) (b) (c)
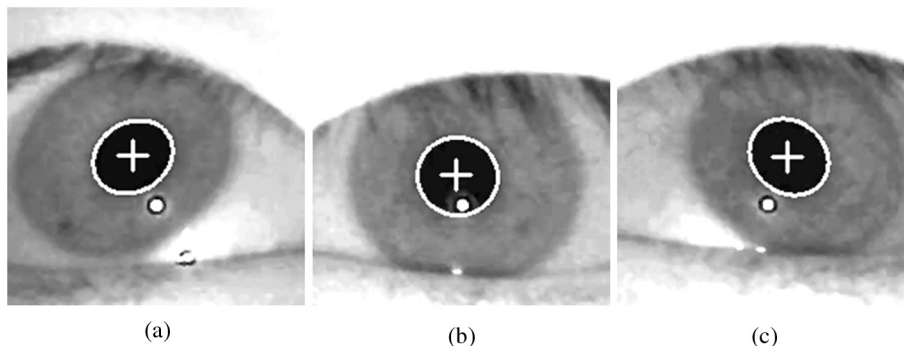
Fig. 38 Examples showing the change in factor 3 (the level of pupil circularity) when the user gazes (a) at the upper-left corner of the monitor (factor 3 of 1.223), (b) at the center of the monitor (factor 3 of 1.073), and (c) at the upper-right corner of the monitor (factor 3 of 1.171).

used. In addition, we compared the gaze detection accuracy of a previous monocular eye method[9] to that of the proposed method. The nine calibration points that each subject gazed at during the initial calibration stage, and the sixteen reference positions used for the experiment are shown in Fig. 40.
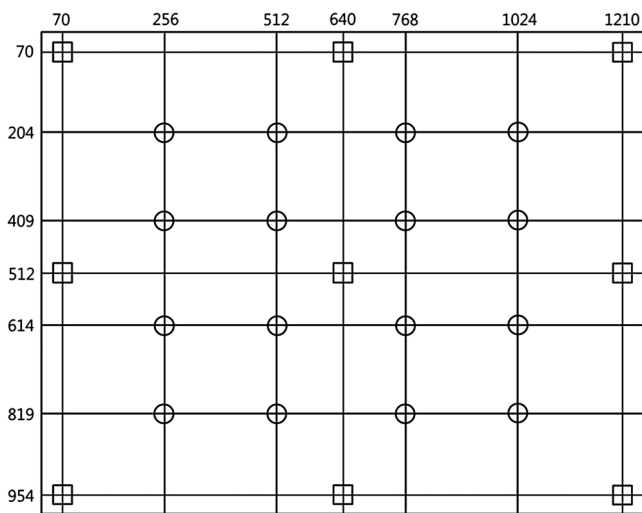
The accuracies of the gaze detection as the root-mean-square error (RMS) are shown in Tables 5 and 6. The gaze detection accuracy is measured based on the angular difference between the reference and calculated gaze point. As shown in Table 5, the methods based on the minimum or MOM show higher gaze detection accuracies compared to the other methods. A gaze detection error of 0.67518 deg

indicates about 30 pixels on a 19-inch monitor with a resolution of 1280×1024 pixels when the Z-distance between the user's eye and the monitor is 80 cm. As shown in Table 6, the proposed method shows higher gaze detection accuracy than monocular gaze detection and the method of averaging the gaze positions of two eyes.

Nevertheless, the enhancement of gaze detection accuracy by the proposed method is not much larger than that by averaging the gaze positions of two eyes. That is because the qualities of both eyes are similar in our experimental data for Table 6. Regarding the additional tests, we obtained data (from an additional five persons) when the quality of

(a)   (b)   (c)
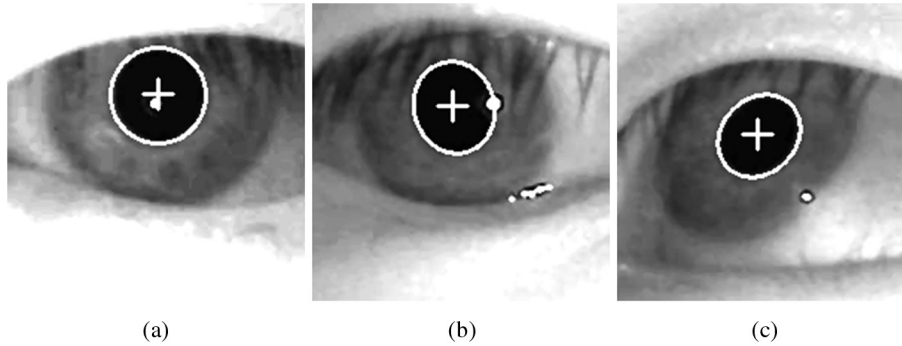
Fig. 39 Examples showing the change in factor 4 (the distance between the SR center and the pupil boundary pixel nearest to the SR center) when the SR is (a) inside the pupil (26.178), (b) on the edge of the pupil (1.516) and (c) outside the pupil (23.51).

Fig. 40 Nine calibration points and sixteen reference points ('□' is a calibration point and '○' is a reference point).

Table 5 Comparison of the gaze detection accuracies by the minimum and the maximum methods along with a defuzzification method (unit: degree).

	COG	FOM	LOM	MOM	MeOM
Minimum	0.67518	0.67976	0.67636	0.67703	0.67703
Maximum	0.68226	0.68273	0.68273	0.68273	0.67530

Table 6 Comparison of the accuracies of the proposed method to others (unit: degree).

	Monocular[9]		Binocular (by averaging the gaze positions of two eyes)	Binocular (Proposed method)
	Left eye	Right eye		
Accuracy	0.73097	0.85547	0.68389	**0.67518**

Table 7 Comparison of the accuracies of the proposed method to others in case that the quality of either left or right eye is very low (unit: degree).

	Monocular[9]		Binocular (by averaging the gaze positions of two eyes)	Binocular (Proposed method)
	Left eye	Right eye		
Accuracy	1.89974	1.89774	1.47042	**1.34719**

one of the two eyes is very low (i.e., one eye is more occluded by eyelid compared to the other). These data were acquired when the users naturally gazed at the references points of Fig. 40. Experimental results are shown in Table 7. We can confirm that the enhancement of gaze detection accuracy by the proposed method is larger than those by other methods when the quality of one eye is very low.

Figure 41 shows the example of gaze detection results by monocular and proposed gaze detections using the
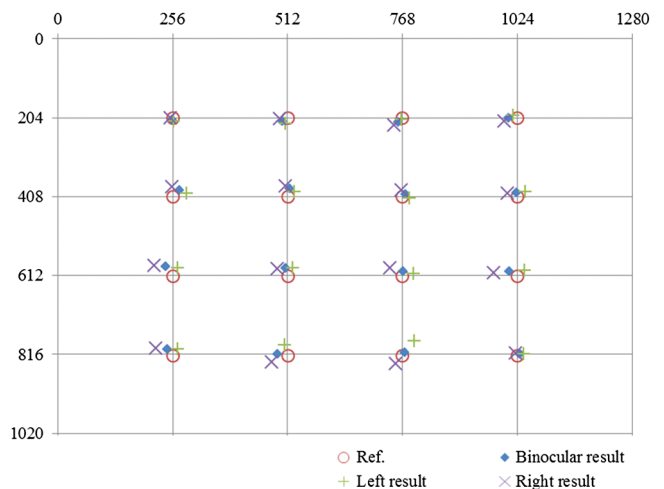
Fig. 41 Experimental results (the left and right results are the gaze positions of the left and right eyes, respectively, and the binocular result indicates the gaze position from the proposed method after combining both eye gaze positions using the MIN-COG).
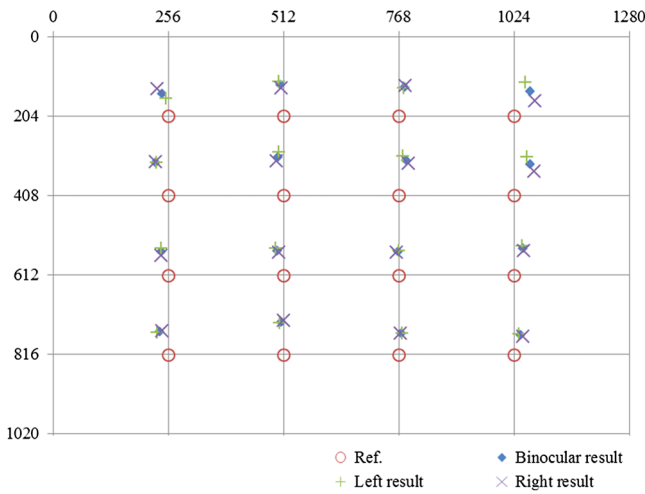
Fig. 42 The cases of gaze detection with low accuracy (the left and right results are the gaze positions of the left and right eyes, respectively, and the binocular result indicates the gaze position from the proposed method after combining both eye gaze positions using the MIN-COG).

MIN-COG. In Fig. 41, the open circle indicates a reference point where the user should focus their gaze, and the diamond indicates the resultant gaze position of both eyes when using the fuzzy algorithm (proposed method). In addition, "+" and "×" show the results when the gaze position is calculated by only the left or right eye, respectively.

Figure 42 shows the cases of gaze detection with low accuracy that occurred when the subjects did not correctly gaze at the nine calibration points during the calibration stage. Because such calibration information is used for calculating the final gaze position, as shown in Sec. 2.9, correct calibration is important for accurate gaze detection. To resolve this problem, it is necessary to verify the accuracy of the user's calibration information during the calibration stage and to ask the subject to gaze again at a point when the accuracy is low.

After estimating the gaze position using the proposed method, our system compensates for the estimated position considering the maximum movement of gaze position in order to prevent discontinuities in the estimated positions. That is, if the estimated x (or y) position using the proposed

(a)

(b)
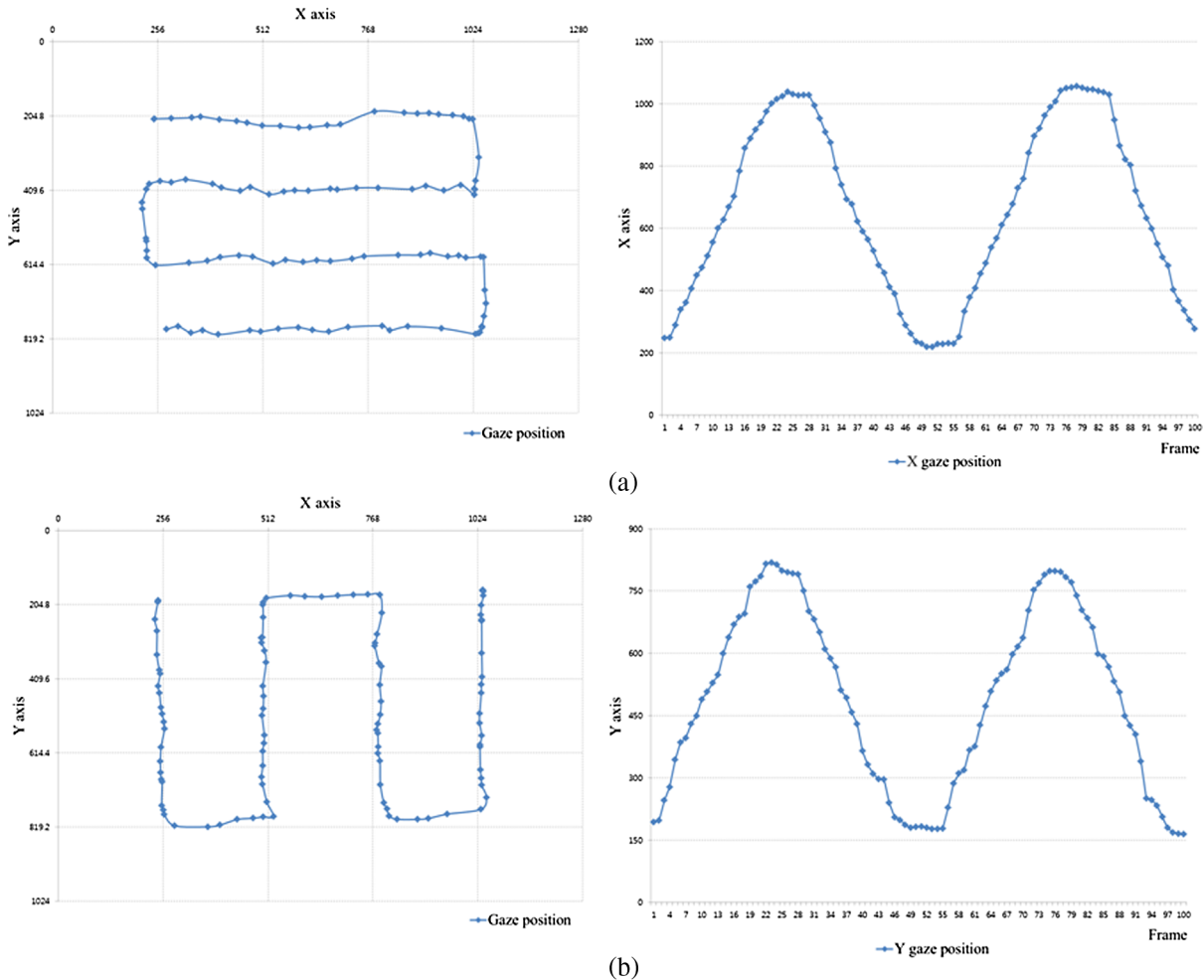
Fig. 43 Trajectories of the x and y gaze positions when a user gazes at the reference points of Fig. 40 in the horizontal and vertical directions, respectively: (a) the change of x gaze position according to frame number (right image) when a user gazes at the reference points horizontally (from left-upper position to left-lower one) (left image), and (b) the change of y gaze position according to frame number (right image) when a user gazes at the reference points vertically (from left-upper position to right-upper one) (left image).

method exceeds in the x (or y) maximum movement boundary from the position of the previous frame, the estimated x (or y) position is adjusted to the corresponding position of maximum movement boundary. The x (or y) maximum movement boundary is determined by experiments.

As shown in Fig. 43, we can confirm that the discontinuity of the estimated gaze positions does not occur, even in the case when the user gazes at the reference points of Fig. 40 in the horizontal and vertical directions, respectively.

Using the panning and tilting device can increase the operating range of the user, but the processing speed reduces due to the additional operation of panning and tilting. In addition, the consequent size and cost of the gaze detection system increases. Hence, we do not use the panning and tilting device for our gaze detection.

As shown in Fig. 3, both eyes should be included in the captured image because our method combines the gaze positions of the two eyes. In this case, the maximum allowed ranges of horizontal movement (rotation and translation) of the user's head are about ± 60 deg and ± 3 cm, respectively. In addition, the maximum allowed ranges of vertical movement (rotation and translation) of the user's head are -25 deg $\sim +60$ deg and ± 6 cm, respectively. However, even if only one eye is included in the captured image due to excessive head movement, our system can still operate by calculating the gaze position based on only one observed eye because the initial calibration information of both eyes is saved. In this case, the maximum allowed ranges of horizontal movement (rotation and translation) of the user's head are increased as ± 85 deg and ± 13 cm, respectively. The maximum allowed ranges of vertical movement (rotation and translation) of the user's head are -25 deg $\sim +60$ deg and ± 6 cm, respectively, which are the same as when both eyes are included in the captured image. These ranges sufficiently cover the natural movement of the user's head in a desktop computer environment. In addition, our system allows the natural movement velocity of the user's head when tracking the user's gaze position within these ranges of movements.

As the last experiment, we performed the additional experiments to include the gaze data from actual field imagery. As shown in Fig. 44, we performed the experiments

Table 8 Twenty sample words used for experiments.

the, and, that, have, for,
not, with, you, this, but,
his, from, they, say, her,
she, will, one, all, would

where each person types a word into our gaze detection system on a cyber keyboard displayed on a monitor.

A total of ten persons participated in the experiments, and a 19-inch monitor with a resolution of 1280×1024 pixels was used. We selected twenty sample words, based on the used frequency[29] as shown in Table 8, and used them for the experiments. In Table 8, the used frequency of the left-upper word ("the") is higher than that of the right-lower word ("would").[29]

If the user's gaze position belongs to the specific key button for the predetermined time duration (we call this dwell-time and set it at 2s), the corresponding character is selected and displayed as shown in Fig. 44. In addition, we set the time-out for typing one character at 10 s, and if the user is unable to type one character within this time-out, it is counted a failure case. We measured the performance in terms of average accuracy and execution time. As shown in Table 9, the average accuracy of typing twenty words of Table 8 by ten persons was about 92%. The average execution time to type one character and word was about 2.6 and 8.91 s, respectively. From the experimental results, we confirm that our gaze tracking system can be used for actual field application.

In order to obtain the optimal weights of Eq. (5), the fuzzy-based method can be designed heuristically without a training procedure contrary to the neural network-based system that requires an additional training process. Hence, our system of combining binocular gaze detection based on a fuzzy-based method has the advantage that its performance is less affected by the usage of different kinds of gaze detection systems, whereas the various weights of neural
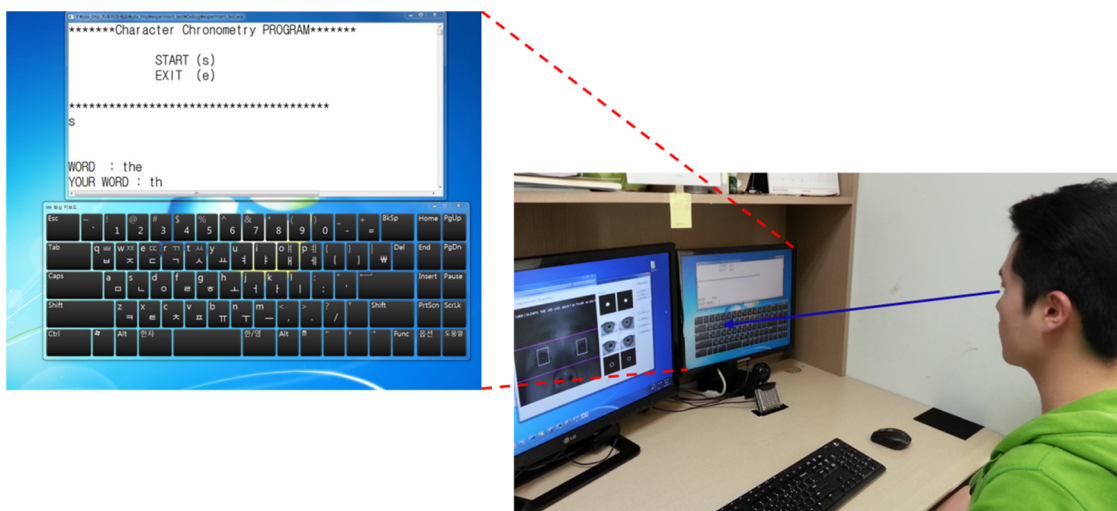
Fig. 44 Example of experiment where a user types words by our gaze detection system.

Table 9 Average accuracy and execution time for typing the sample word of Table 8.

User	Average accuracy (%)	Average execution time for typing one character (s)	Average execution time for typing one word (s)
User 1	95	2.313	7.891
User 2	90	2.713	9.213
User 3	90	2.497	8.602
User 4	95	2.819	9.645
User 5	100	2.761	9.688
User 6	90	2.5	8.599
User 7	85	2.513	8.725
User 8	100	2.446	8.438
User 9	85	2.595	8.632
User 10	85	2.846	9.668
Average	91.5	2.6	8.91

network-based gaze tracking system must be trained to suit the specific data.

In order to use the detection scheme of dark and bright pupil by the method of illuminatoron and off, an additional NIR illuminator (the distance to the camera is short) is necessary to produce the bright pupil in the image based on the principle of the red-eye effect.[25] In addition, it is required that the additional hardware device to turn on and off the two NIR illuminators (additional NIR illuminator for the bright pupil besides our NIR illuminator of Fig. 2), alternatively at fast speed, would need to be synchronized with the image frame. Hence, the size and cost of the system increase.

The pupil region can be located by using the two successive images that are captured by turning on and off these two NIR illuminators, alternatively. Hence, the frame rate of detecting the pupil center and consequent gaze position is reduced to half the frame rate of image acquisition. Due to these problems, we do not use the detection scheme of dark and bright pupils by the method of illuminator-on and off.

4 Conclusion

In this paper, we proposed a new binocular gaze detection method using a fuzzy algorithm with a quality measurement of both eyes. To combine the gaze points of the left and right eyes, we measured the qualities of both eyes based on the distortion from an eyelid and the SR, the level of pupil circularity, and the distance between the pupil boundary and the SR center. To obtain a more accurate pupil boundary, we compensated the distorted boundary of a pupil by an eyelid using the information from the lower half-circle of the pupil. The final gaze position was calculated using a fuzzy algorithm based on four quality score measurements. The experimental results indicate that the RMS error of the gaze estimation is about 0.67518 deg on a 19-inch monitor with a resolution of 1280×1024 pixels.

In the future, we will study a method for increasing the accuracy of the user calibration. This can be done by verifying the accuracy of the user's calibration information during the calibration stage and requesting the user to gaze again at a point when the accuracy is low.

References

1. D. H. Yoo and M. J. Chung, "A novel non-intrusive eye gaze estimation using cross-ratio under large head motion," *Comput. Vis. Image Underst.* **98**, 25–51 (2005).
2. J.-G. Wang and E. Sung, "Study on eye gaze estimation," *IEEE Trans. Syst. Man Cybern. B.* **32**(3), 332–350 (2002).
3. S.-W. Shih and J. Liu, "A novel approach to 3-D gaze tracking using stereo cameras," *IEEE Trans. Syst. Man Cybern. B.* **34**(1), 234–245 (2004).
4. E. Murphy-Chutorian, A. Doshi, and M. M. Trivedi, "Head pose estimation for driver assistance systems: a robust algorithm and experimental evaluation," in *Proc. IEEE Intelligent Transportation Systems Conf.*, pp. 709–714, Seattle, Washington (2007).
5. L. Piccardi et al., "WearCam: ahead mounted wireless camera for monitoring gaze attention and for the diagnosis of developmental disorders in young children," in *Proc. 16th IEEE International Symposium on Robot and Human Interactive Communication*, pp. 594–598, Jeju Island, Korea (2007).
6. X. Li and W. G. Wee, "An efficient method for eye tracking and eye-gazed FOV estimation," in *Proc. 16th IEEE International Conference on Image Processing*, pp. 2597–2600, Cairo, Egypt (2009).
7. C. W. Cho "Robust gaze-tracking method by using frontal-viewing and eye-tracking cameras," *Opt. Eng.* **48**(12), 127202 (2009).
8. Y. J. Ko, E. C. Lee, and K. R. Park, "A robust gaze detection method by compensating for facial movements based on corneal specularities," *Pattern Recognit. Lett.* **29**(10), 1474–1485 (2008).
9. J. W. Lee, H. Heo, and K. R. Park, "A novel gaze tracking method based on the generation of virtual calibration points," *Sensors* **13**(8), 10802–10822 (2013).
10. T. Nagamatsu, J. Kamahara, and N. Tanaka, "Calibration-free gaze tracking using a binocular 3D eye model," in *Proc. 27th Annual CHI Conference on Human Factors in Computing Systems*, pp. 3613–3618, ACM, New York, NY (2009).
11. C. Hennessey and P. Lawrence, "Noncontact binocular eye-gaze tracking for point-of-gaze estimation in three dimensions," *IEEE Trans. Biomed. Eng.* **56**(3), 790–799 (2009).
12. D. H. Yoo and M. J. Chung, "Non-intrusive eye gaze estimation without knowledge of eye pose," in *Proc. 6th IEEE International Conference on Automatic Face and Gesture Recognition*, pp. 785–790, IEEE Xplore Digital Library, USA (2004).
13. J. Zhu and J. Yang, "Subpixel eye gaze tracking," in *Proc. 5th IEEE International Conference on Automatic Face and Gesture Recognition*, pp. 124–129, IEEE Xplore Digital Library, USA (2002).
14. J. W. Bang, E. C. Lee, and K. R. Park, "New computer interface combining gaze tracking and brainwave measurements," *IEEE Trans. Consum. Electron.* **57**(4), 1646–1651 (2011).
15. J. W. Lee, "3D gaze tracking method using purkinje images on eye optical model and pupil," *Opt. Lasers Eng.* **50**(5), 736–751 (2012).
16. H. C. Lee, "Remote gaze tracking system on a large display," *Sensors* **13**(10), 13439–13463 (2013).
17. Y. He, "Key techniques and methods for imaging iris in focus," in *Proc. IEEE International Conference on Pattern Recognition*, pp. 557–561, Hong Kong, China (2006).
18. P. Magnan, "Detection of visible photons in CCD and CMOS: a comparative view," *Nucl. Instrum. Methods Phys. Res. Sect. A.* **504**(1–3), 199–212 (2003).
19. Webcam C600. http://www.logitech.com/en-us/support/webcams/5869 (5 November 2013).
20. F-number, http://en.wikipedia.org/wiki/F-number (19 March 2014).
21. R. C. Gonzalez and R. E. Woods, Eds., *DigitalImageProcessing*, 2nd ed., Prentice-Hall, Upper Saddle River, New Jersey (2002).
22. L. Ding and A. Goshtasby, "On the canny edge detector," *Pattern Recognit.* **34**, 721–725 (2001).
23. D. G. Kirkpatrick and R. Seidel, "The ultimate planar convex hull algorithm," *SIAM J. Comput.* **15**, 287–299 (1986).
24. J. Daugman, "New methods in iris recognition," *IEEE Trans. Syst. Man Cybernetics — Part B.* **37**(5), 1167–1175 (2007).
25. Red-eye effect, http://en.wikipedia.org/wiki/Red-eye_effect (19 March 2014).

26. G. P. Namand and K. R. Park, "New fuzzy-based retinexmethod for the illumination normalization of face recognition," *Int. J. Adv. Rob. Syst.* **9**, 1–9 (2012).
27. G. J. Klirand and B. Yuan, *Fuzzy Sets and Fuzzy Logic-Theory and Applications*, Prentice-Hall, New Jersey (1995).
28. W. V. Leekwijck and E. E. Kerre, "Defuzzification: criteria and classification," *Fuzzy Sets Syst.* **108**(2), 159–178 (1999).
29. Most common words in English, http://en.wikipedia.org/wiki/Most_common_words_in_English (19 March 2014).

Chul Woo Cho received his BS degree in electronics engineering from Dongguk University, Seoul, Republic of Korea, in 2009. He also received his PhD degree in electronics and electrical engineering at Dongguk University in 2014. His research interests include image processing and gaze tracking.

Hyeon Chang Lee received his BS degree in computer science from Sangmyung University, Republic of Korea, in 2009. He also received his PhD degree in electronics and electrical engineering at Dongguk University in 2014. His research interests include image processing, gaze tracking, and biometrics.

Su Yeong Gwon received her BS degree in computer science from Sangmyung University, Republic of Korea, in 2010. She is currently pursuing the combined courses of MS and PhD degrees in electronics and electrical engineering at Dongguk University. Her research interests include image processing and pattern recognition.

Jong Man Lee received his BS degree in computer science from Sangmyung University, Republic of Korea, in 2013. He is currently pursuing a MS degree in electronics and electrical engineering at Dongguk University. His research interests include image processing and pattern recognition.

Dongwook Jung received his BS degree in computer science from Sangmyung University, Republic of Korea, in 2014. He is currently pursuing a MS degree in electronics and electrical engineering from Dongguk University. His research interests include image processing and pattern recognition.

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 also received a 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.

Hyun-Cheol Kim received his BS and MS degrees in electronics engineering from Kyunghee University, Republic of Korea, in 1998 and 2000, respectively. In 2000, he joined Electronics and Telecommunications Research Institute (ETRI), Daejeon, South Korea, and now he is a senior researcher of ETRI. He has been engaged in the development of MPEG-4 authoring system and T-DMB terminal. His research interests include video processing, multimedia communications, interactive broadcast systems, and UI/UX.

Jihun Cha received his BS degree in computer science from Myongji University, South Korea, in 1993 and MS and PhD degrees in computer science from Florida Institute of Technology, USA, in 1996 and 2002, respectively. He is now serving as the director of the immersive media section in ETRI, Daejeon, Republic of Korea, and is a special fellow on standardization. His research interests include advanced user interaction, rich media technologies, interactive broadcasting system, feature extraction, and object detection/tracking in motion pictures.