

Real-Time Eye Tracking Using IR Stereo Camera for Indoor and Outdoor Environments

Sungsoo Lim¹ and Daeho Lee²

¹Department of Electronics and Radio Engineering, Kyung Hee University
Yongin, Gyeonggi, 17104, Korea
[e-mail: imsect@khu.ac.kr]

²Humanitas College, Kyung Hee University
Yongin, Gyeonggi, 17104, Korea
[e-mail: nize@khu.ac.kr]

*Corresponding author: Daeho Lee

*Received September 2, 2016; revised March 4, 2017; accepted May 2, 2017;
published August 31, 2017*

Abstract

We propose a novel eye tracking method that can estimate 3D world coordinates using an infrared (IR) stereo camera for indoor and outdoor environments. This method first detects dark evidences such as eyes, eyebrows and mouths by fast multi-level thresholding. Among these evidences, eye pair evidences are detected by evidential reasoning and geometrical rules. For robust accuracy, two classifiers based on multiple layer perceptron (MLP) using gradient local binary patterns (GLBPs) verify whether the detected evidences are real eye pairs or not. Finally, the 3D world coordinates of detected eyes are calculated by region-based stereo matching. Compared with other eye detection methods, the proposed method can detect the eyes of people wearing sunglasses due to the use of the IR spectrum. Especially, when people are in dark environments such as driving at nighttime, driving in an indoor carpark, or passing through a tunnel, human eyes can be robustly detected because we use active IR illuminators. In the experimental results, it is shown that the proposed method can detect eye pairs with high performance in real-time under variable illumination conditions. Therefore, the proposed method can contribute to human-computer interactions (HCIs) and intelligent transportation systems (ITSs) applications such as gaze tracking, windshield head-up display and drowsiness detection.

Keywords: Eye detection and tracking, multi-level thresholding, evidential reasoning, gradient local binary pattern, multiple layer perceptron

1. Introduction

An accurate method, which can detect eyes under variable illumination environments, is required for human-computer interactions (HCIs) and intelligent transportation systems (ITSs); however, most methods are designed for simple environments, and are based on face detection [1-11]. In these methods, eyes are detected within detected facial regions, and face detection needs a time consuming calculation because they use sliding windows for the whole image area. Moreover, these sliding window methods need various sizes of windows because the sizes of faces differ according to the distance from the camera to the real face.

Some methods use color or intensity information to detect faces [1,2]. Skin color distributions, however, are quite variable by illumination conditions. The methods using thresholding to segment humans from the background might also be inappropriate because the illumination of a human face is not always brighter than the background in outdoor conditions [10]. The method [3], which uses the clue that human skin exhibits an abrupt change in reflectance around 1.4 μ m wavelength light, seems to detect skin quite well; however the hardware is complicated because two separate bands of light are used. Thus, this method needs two bands of IR (infrared) light emitting diodes (LEDs) and a camera that can scan both bands of light.

After detecting faces, image projection, template matching algorithms [3][5-7] or various types of filters are used [1,4], where the methods using projection are vulnerable to rotation, and template matching-based and filter-based methods quite depend on their training data. Recently, some gaze tracking methods using IR reflections have been issued [12-16]. These methods require high-resolution cameras and placement of the IR LEDs in a designated position.

Methods using different images of reflection of two synchronized rings of IR LEDs have also been developed [15,16]. The two rings of IR LEDs are turned on and off alternately, the camera can thus scan interlaced images. After image acquisition, eyes are detected using the difference of the two images because the two rings of LEDs are not reflected in the same location. However, these methods also require complicated hardware and high-resolution cameras.

In this paper, we propose a robust eye tracking method, which uses evidential reasoning and geometrical rules for detected dark evidences binarized by fast multi-level thresholding, and two neural network classifiers based on multiple layer perceptron (MLP) using gradient local binary patterns (GLBPs). This method can reduce processing time and enhance accuracy because it does not use sliding windows; it also eliminates unnecessary candidates. Therefore, this method is less dependent on the training data set than other methods based on face detection, and can use low-resolution cameras.

2. Hardware Design

We use a set of IR LEDs and an IR stereo camera, where the IR LEDs have a half-intensity angle of $\pm 60^\circ$ and a peak wavelength of 830 nm as shown in Fig. 1, and IR filters are mounted on the lenses of each RGB camera. The two cameras have 640 \times 480 resolutions with frame rates of 30 fps, and the baseline of the two cameras is 120mm. We positioned IR illuminators consisting of 60 IR LEDs, three rows of 20 aligned LEDs, on the upper side of the IR stereo

camera as shown **Fig. 2**. In the IR illuminator part, there is an MCU (Micro Controller Unit) which controls the power of IR LEDs for serial communication with a computer.

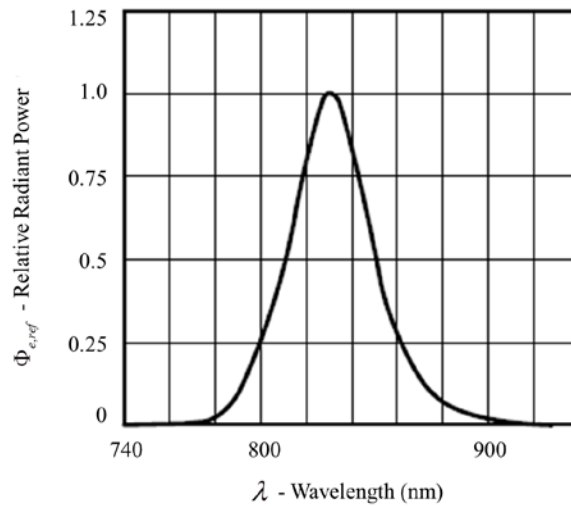


Fig. 1. Relative radiant power according to wavelength of IR LEDs.

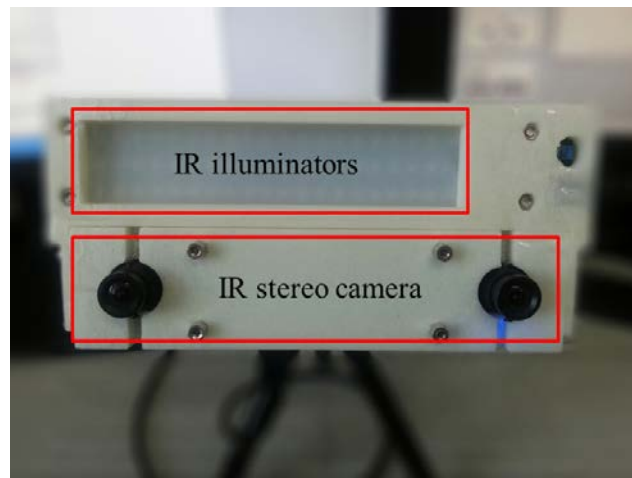


Fig. 2. Hardware setup of the stereo camera with active IR illuminators.

3. Eye Detection and Tracking Using IR Stereo Camera

To detect and track eyes, the overall scheme of the proposed method is shown in **Fig. 3**. To make acquired images suitable for detection, we first perform preprocessing, the details of which are shown in **Fig. 4**. This method detects dark evidences and eye pair candidates from the detected dark evidences. Two classifiers finally verify which the eye pair candidates is the real eye pair.

3.1 Preprocessing

In preprocessing procedure, stereo calibration and rectification are first performed to calculate camera parameters and to correct radial distortions. As the local illumination conditions may

be changed often, especially the intensity of IR light on the faces of people, we compensate the local illumination variation using a DC notch filter [17]. The rectified image $g(x, y)$, which is transformed from the original image $f(x, y)$, is filtered to the illumination compensated image $h(x, y)$ is calculated by

$$h(x, y) = g(x, y) - \bar{g}(x, y), \quad (1)$$

where $\bar{g}(x, y)$ is the local average image by a convolution mask. Fig. 5 shows the preprocessing results for different illumination conditions. After DC notch filtering, we set the region of interest (ROI) to reduce computation complexities.

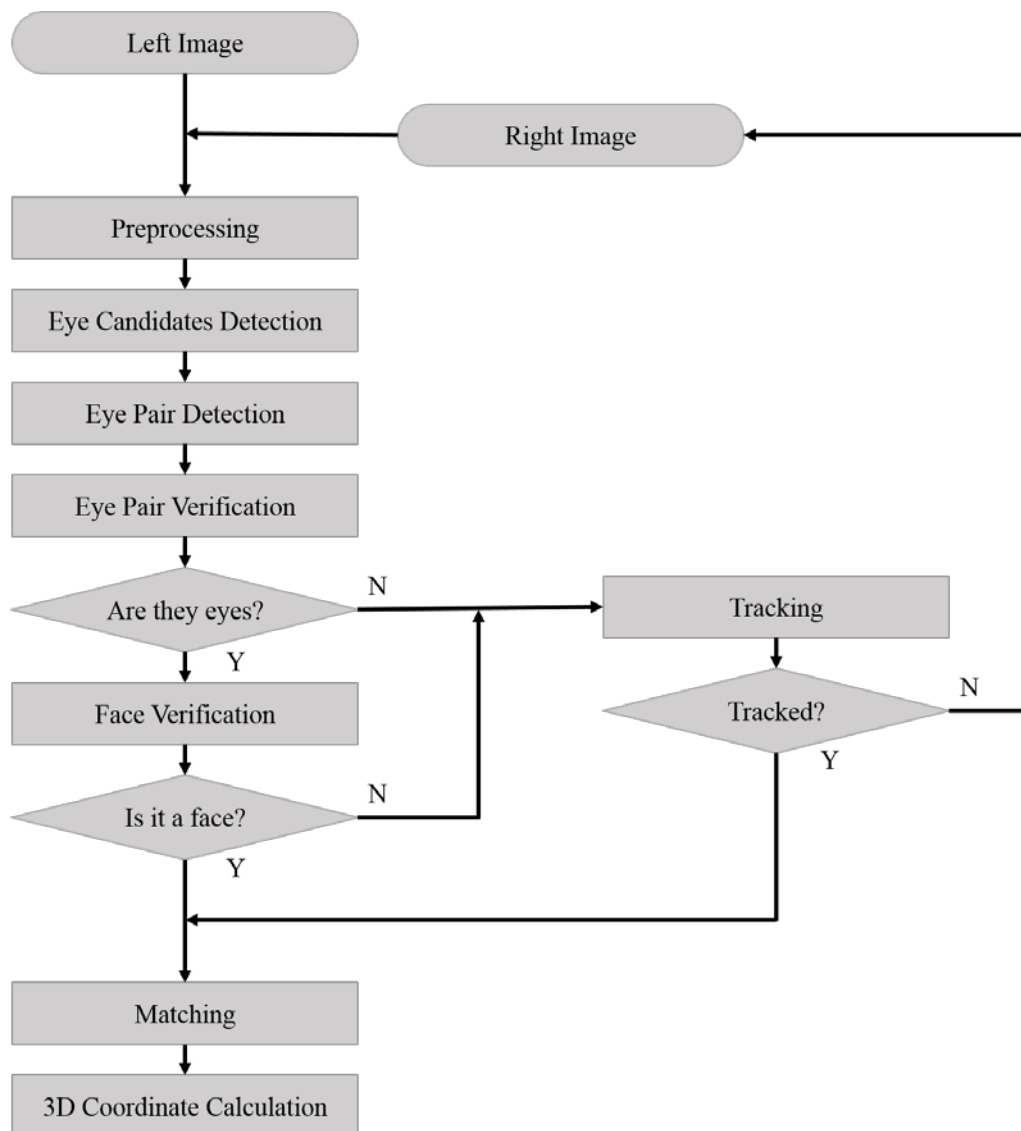


Fig. 3. Overall scheme of the proposed method.

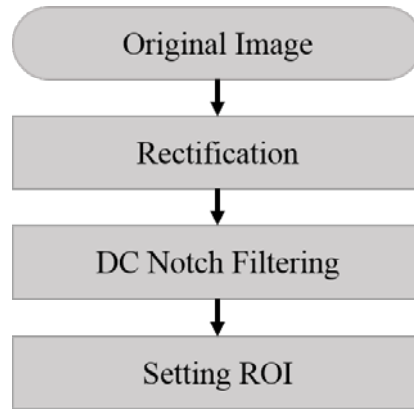


Fig. 4. Details of the preprocessing procedure.

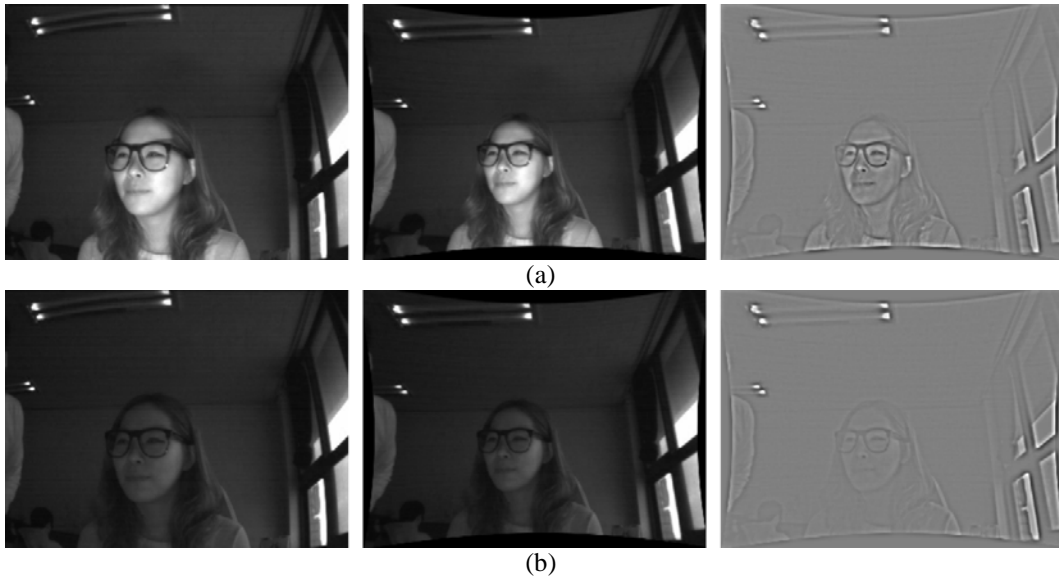


Fig. 5. Preprocessing results: (a) when under bright illuminations, and (b) when under dark illuminations (from left to right: $f(x, y)$, $g(x, y)$ and $h(x, y)$).

3.2 Dark Evidence Detection

After preprocessing, dark evidences such as eyes, eyebrows and mouths are segmented by binarization using multi-level thresholds, and they are labeled by connected component analysis. To accelerate multi-level thresholding, we use a novel multi-level thresholding, which uses label splitting by recursion as follows:

1. Labeling by an initial threshold (τ_0).
2. All the labels are inserted into a label list.
3. For each label of the label list.
 - A. Labeling with a lower threshold ($\tau_n = \tau_{n-1} - \tau_\Delta$) within the bounding box of the current label.
 - B. If the number of labels is more than one, the split labels are inserted into the label list, and go to step 3.

- C. If the new threshold ($\tau_{n+1} = \tau_n - \tau_\Delta$) is higher than the predetermined minimum threshold (τ_{\min}), go to 3.A. Otherwise, terminate the recursion.

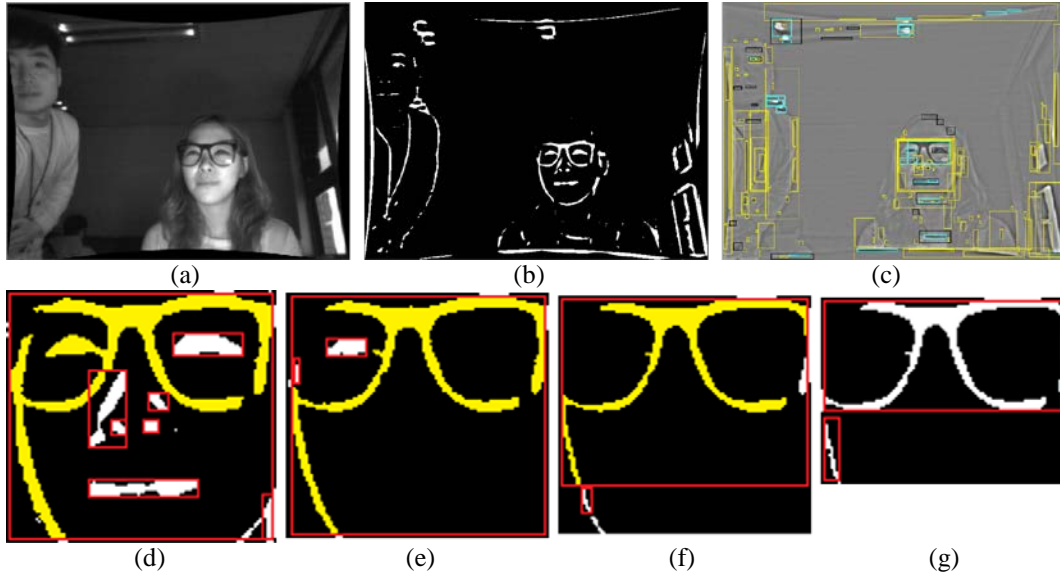


Fig. 6. Example of label splitting: (a) a rectified image, (b) binarized image by an initial threshold, (c) a segmentation result image, and (d) through (g) are the split labels.

An example of the label splitting method is shown in **Fig. 6**, where a binary result by an initial threshold of τ_0 is shown in **(b)**, and the zoom-in part of the face in **(b)** is shown in **(d)**. In **Fig. 6(d)**, each label marked with red rectangle is split by τ_0 . **Fig. 6(e)** is the label splitting result of the yellow label of **(d)** and it shows that the right eye label, which is not segmented by the initial threshold, is split. Likewise, **Fig. 6(f)** shows the label splitting result of the yellow label of **(e)**, and **(g)** shows the result of the yellow label of **(f)**. This splitting is performed recursively. The valid labels of this example are shown in **Fig. 6(c)**. The labels which segmented by the initial threshold are marked with black rectangles, and the split labels are shown with cyan rectangles. The labels marked with yellow rectangles in **Fig. 6(c)** are not dark evidences for facial components (eyes, eyebrows and mouths) by evidential reasoning in the following section (the width of the eyes is longer than the height), the eye pair detection.

3.3 Eye Pair Detection

After dark evidence detection, eye pairs are detected by evidential reasoning and geometrical rules, which is based on the face model shown in **Fig. 7**, as follows:

1. The width of eyes is longer than the height ($w_i > h_i$).
2. The sizes of paired eyes are similar as given by

$$\left(\frac{\min(h_1, h_2)}{\max(h_1, h_2)} > 0.7 \right) \wedge (h_i < T_h) \wedge (w_i < T_w) \wedge (h_i > \tau_h) \wedge (w_i > \tau_w), \quad (2)$$

where T_h and τ_h are the threshold of the maximum and minimum height of eyes respectively and T_w and τ_w are the threshold of the maximum and minimum width, respectively, and $i \in \{1, 2\}$.

- The absolute difference between angles of a pair of eyes should be less than the threshold as given by

$$|\theta_1 - \theta_2| < \tau_\theta, \quad (3)$$

where θ_1 and θ_2 are angles of each eye label and τ_θ is the threshold of the absolute difference between the two labels. The angles of each label are calculated by

$$\theta = \frac{1}{2} \tan^{-1} \left(\frac{\sum \sin(2\alpha)}{\sum \cos(2\alpha)} \right), \quad (4)$$

and α can be obtained by

$$\alpha = \tan^{-1} \left(\frac{y - \bar{y}}{x - \bar{x}} \right), \quad (5)$$

where (x, y) is a position of a label and (\bar{x}, \bar{y}) is a centroid of the label.

- The distance between the eyes, d_1 and the distance should satisfy the following:

$$\left(2 < \frac{d_2}{d_1} < 5 \right) \wedge \left(\tau_{d_1} < d_1 < T_{d_1} \right), \quad (6)$$

where τ_{d_1} and T_{d_1} are the threshold of the minimum and maximum distance, respectively.

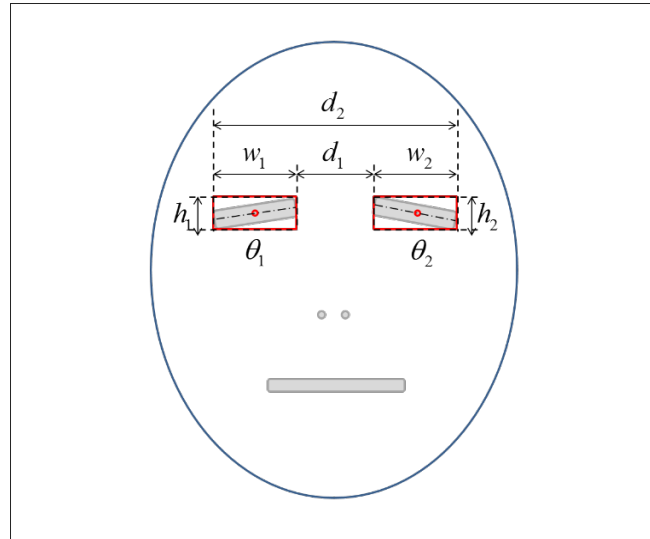


Fig. 7. Face model for eye pair detection.

The results of eye pair detection are shown in **Fig. 8**, where eye pair candidates are connected with green lines and the candidates are marked with magenta rectangles.

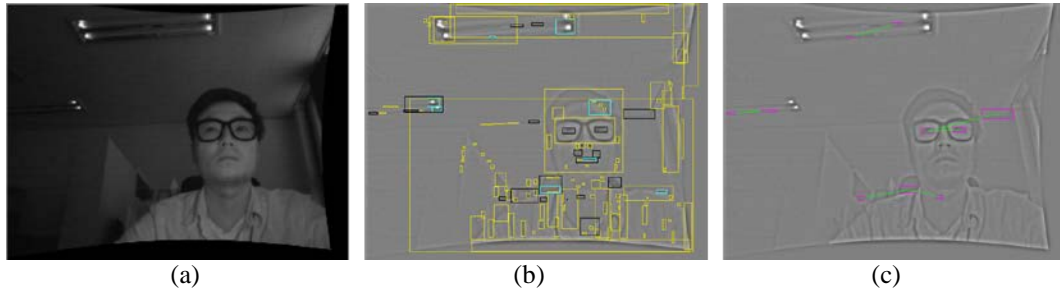


Fig. 8. Example of eye pair detection: (a) a rectified image, (b) the dark evidence image by multi-level thresholds, and (c) the eye pair candidate image.

3.4 Eye Pair Verification

As shown in **Fig. 8(c)**, some false alarms are detected because of simple evidential reasoning and geometrical rules. So we verify the candidates by classifying extended eye regions and pure facial regions, as shown in **Fig. 9**, where **(c)** is the extended eye region, and **(d)** is the extended pure facial region.

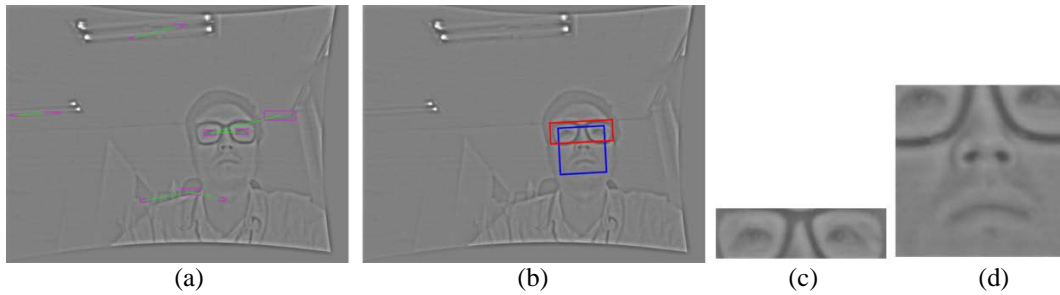


Fig. 9. Extracting eye regions and face regions: (a) An eye pair candidate image, (c) extended eye region and (d) extended pure facial region ((c) and (d) regions are segmented as shown (b)).

The input features of classifiers are an advanced version of LBPs (local binary patterns) [18,19]. Although LBPs have gradient information for spatial regions, we use GLBPs (gradient LBPs) [20] because GLBP is invariant for common illumination changes of texture images, as calculated by

$$GLBP_{P,R}(x,y) = \sum_{p=0}^{P-1} z \left(\frac{1}{2} (|h'_x(x,y)| + |h'_y(x,y)|), |h_p(x,y) - h(x,y)| \right) 2^p, \quad (7)$$

$$z(s,t) = \begin{cases} 1 & s \geq t \\ 0 & s < t \end{cases}$$

where P is the number of neighbor pixels on a circle of radius R ($R > 0$), h'_x and h'_y are gradients of h on the direction of x and y , respectively. So the gradients are calculated by $h'_x(x,y) = h(x-R,y) - h(x+R,y)$ and $h'_y(x,y) = h(x,y-R) - h(x,y+R)$. And, h_p is a neighbor pixel around the $h(x,y)$, e.g., h_p is the each 8 neighbor (clockwise or counterclockwise rotation) of 3×3 when $R = 1$ and $P = 8$. To reduce complex computations

for classifications, we use 3×3 windows and “+” shaped GLBP labels with $R = 1$ and $P = 4$, so the range of GLBPs is $[0, 15]$.

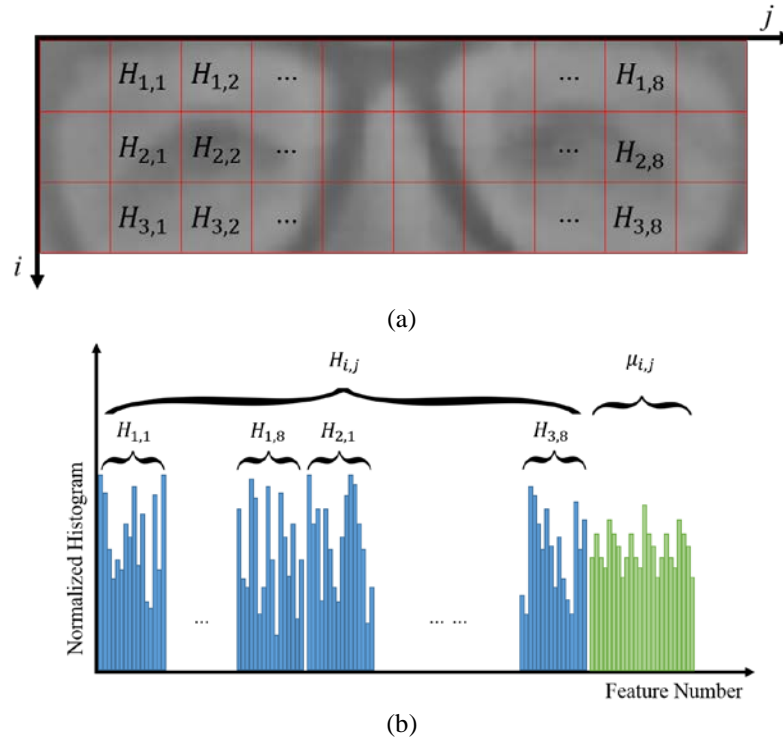


Fig. 10. Feature extraction for extended eye region: (a) the division of rectangles, and (b) the features for histogram of GLBP.

We divide the extended eye region into 3×10 equivalent rectangles, and calculate histograms of GLBP as shown in Fig. 10, where we do not use the left-end and the right-end rectangles because these areas may not include face as the case may be. Each pixel can have a decimal from 0 to 15 so that each rectangle can have a histogram ranging $[0, 15]$. The number of the GLBP features is $2^4 \times (3 \times 8)$, where 2^4 is the number of histogram ranges and (3×8) is the number of rectangles. GLBP is invariant to intensity reversals; however, eye, eyebrows and mouths are darker than face skin regions. Therefore, we use some additional features, which are average intensities for rectangles. Therefore, the total feature number is $2^4 \times (3 \times 8) + (3 \times 8) = 408$ and all the input features are normalized to $[0, 1]$. As shown in Fig. 10, $H_{i,j}$ and $\mu_{i,j}$ are the histogram and the average intensity of i, j rectangle, respectively.

To verify whether the eye regions are real eye regions or not, we use an artificial neural network (ANN) using an MLP. The neural network is composed with an input layer, a hidden layer and an output layer, where the number of nodes of the input layer is $408+1$, including a bias input, and 5 hidden nodes and 2 output nodes are used.

To reduce false alarms observed in eyebrows or glasses frames, we use another classifier for extended pure facial regions. The pure facial region is extended to include detected eye pair candidate, nose and mouth as shown in Fig. 9 (d). The input features of the second MLP are the histograms of GLBPs and average intensities of divided square regions shown in Fig. 11.

The number of input features is $2^4 \times (4 \times 4) + (4 \times 4) + 1 = 273$, where 2^4 is the number of histogram ranges, (4×4) is the number of squares, and the last term (4×4) is used for the average intensities of squares.

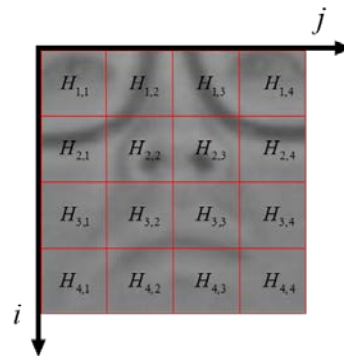


Fig. 11. Feature extraction for pure facial region.

Even though the classifiers are very accurate, single region classification may increase false positive rates or false negative rates. Therefore, we additionally classify eight neighbor regions as shown in **Fig. 12**, where the numeric on the left-top represents the classification result, 1 (positive) and 0 (negative). When the number of faces which are classified as true face is larger than the number of negatives, it confirms the region as a face; otherwise a non-face.

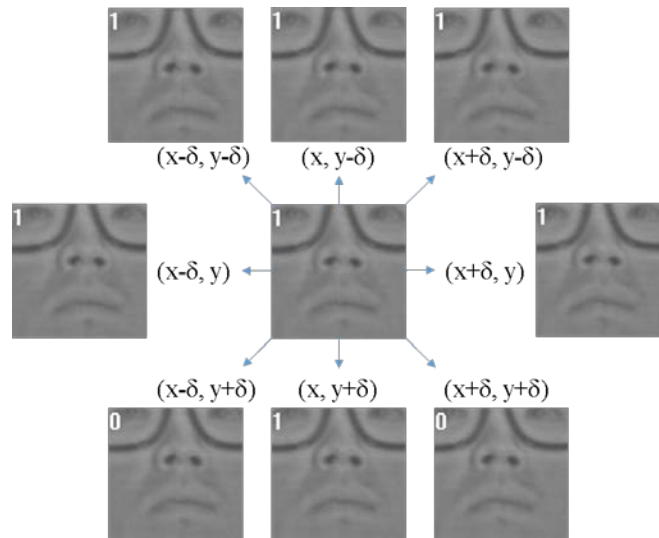


Fig. 12. Shifted faces and the outputs of neural network.

The results of the two classifiers are shown in **Fig. 13**. Eye pair candidates are indicated by magenta rectangles and green line in **Fig. 13 (a)**, where there are six eye pair candidates and the first MLP classifies two pairs as eye regions, which are marked with red rectangles and blue squares in **(b)** and **(c)**, respectively, and the classification results is shown in **(d)**. Finally, only the left region of **Fig. 13(e)** is classified as the pure facial region by the second MLP.

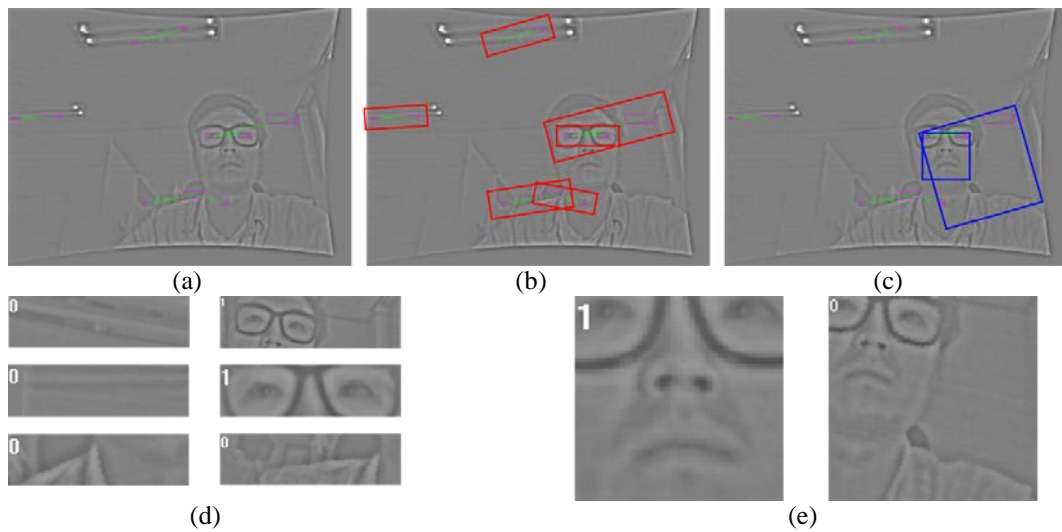


Fig. 13. Verification results: (a) eye pair candidates, (b) eye pair candidates in ROI, (d) eye areas of each eye pair candidates (e) face areas and (c) indicates face regions of (e).

3.6 Stereo Matching

When the true eye pair is detected by classifiers, the two eyes are independently detected in the opposite side image by region matching. Because two images are rectified by stereo calibration, region matching is performed with only horizontal searching. In **Fig. 14**, (a) shows the extended matching region of the detected left eye on the left image, the red line of (b) is the search range on the right image, and (b) shows the matching score graph calculated by SAD (sum of absolute difference), where the highest score position is indicated by the arrow.

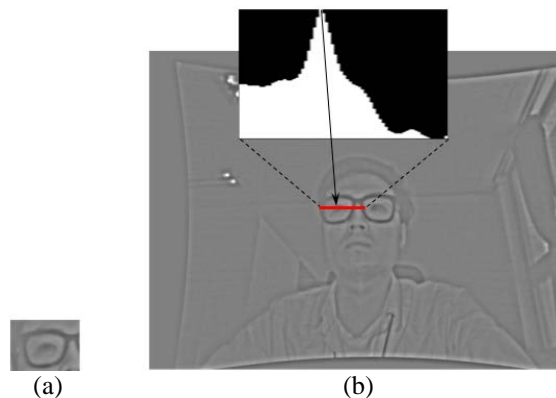


Fig. 14. Matching example: (a) detected right eye of the left image, and (b) the right image with a search range marked with a red line.

3.7 Three Dimensional (3D) Coordinate Calculation

To calculate the 3D real world coordinates of the detected eyes, it is necessary to locate the pupils. In this paper, the x coordinate of the pupil is determined by finding the maximum position of the vertical projection of darkness, and the y coordinate is determined using horizontal projection.

After detecting, the 2D coordinates on the image plane (x and y) of the pupils, 3D real-world coordinates X , Y and Z are easily calculated by

$$Z = B \times \frac{f}{d}, \quad (8)$$

$$X = (x - c_x) \times \frac{Z}{f}, \quad (9)$$

$$Y = (y - c_y) \times \frac{Z}{f}, \quad (10)$$

where B is the baseline, f is the focal length, (c_x, c_y) is the principal point, and d is the disparity of the corresponding pupil. These intrinsic parameters are estimated by stereo calibration, and we use Zhang's calibration [21]. Calculated 3D coordinates can be used for HCI and ITS applications such as gaze tracking.

3.8 Tracking

When we fail to detect eyes in a frame, this method tracks the eyes using the locations in the previous frame. This method searches the most similar objects to the eyes in the previous frame to track eyes, and the similarity is calculated by SAD. When we fail to detect and track eyes, the entire procedure is repeated with the right side image. If the system detects eyes in the right side image, it matches the eyes from the left camera image using also region matching.

4. Experimental Results

The proposed method was implemented in Visual C++ 2013 and tested on a PC (Intel® Core™ i7-3770 CPU @ 3.40GHz). The test image sequences were acquired from a stereo camera with 640×480 resolution.

We trained the neural network for extend eye region with 1559 true and 9667 false training data set of 49 different cases of 27 persons, and the training error rate was 0.181%. Likewise, we trained the neural network for pure facial region with 3856 true and 11874 false training data sets of 41 different cases of 24 persons, and the training error rate was 0.097%.

The test image sequences are shown in **Table 1**, and the results are shown in **Table 2** and **Fig. 15**. A few samples of the detection results of the test image sequences shown in **Table 1** are shown in **Fig. 15**, where the rectangles indicate the ROI, and the exact positions of pupils are marked with crosses. As shown in the first row images, the proposed system can detect eyes even when under dark illumination conditions. When the IR light reflects on the lenses of the glasses and it interrupts detecting eyes as shown in the second row images, this method can detect eyes because we have trained similar cases. Other results also show that this method can recognize eyes when various face expressions and postures. Moreover, **Fig. 15** shows that eyes can be successfully detected in spite of various subjects and various illumination conditions. Moreover, the proposed method can detect eyes even though the subjects are wearing glasses or sunglasses by including glasses as a condition in the neural network training data set.

Table 1. Sample images of the tested sequences


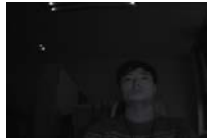


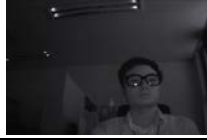

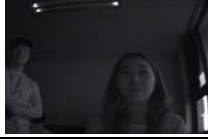

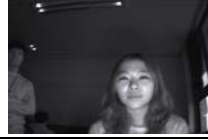
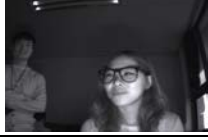




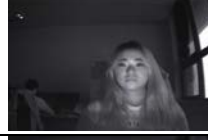
















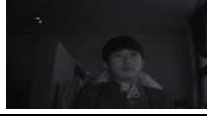

DB	Characteristic	Sample 1	Sample 2	Sample 3
1	Naked eyes, Movement, Trained, Dark environment			
2	Sunglasses, Movement, Rotation, Trained			
3	Naked eyes, Rotation, Facial expression, Trained, Brightness change			
4	Glasses, Rotation, Facial expression, Trained, Brightness change			
5	Naked eyes, Rotation, Trained, Brightness change			
6	Glasses, Rotation, Facial expression, Trained, Brightness change			
7	Naked eyes, Facial expression, Not trained, Bright environment			
8	Glasses, Not trained, Bright environment			
9	Naked eyes, Rotation, Facial expression, Not trained, Dark environment			
10	Glasses, Movement, Rotation, Facial expression, Trained			
11	Naked eyes, Movement, Rotation, Not trained Dark environment			

Table 2. Recognition rate and processing time

DB	Number of Frame	Average Processing Time [ms]	True positive	False negative	Accuracy [%]
			False positive	True negative	
1	2332	20.023	2332	0	100.00
			0	0	
2	2297	22.873	2297	0	100.00
			0	0	
3	896	20.415	895	1	99.89
			0	0	
4	460	24.314	457	3	99.35
			0	0	
5	422	20.462	418	1	99.05
			3	0	
6	540	17.789	540	0	100.00
			0	0	
7	1143	24.177	1141	2	99.83
			0	0	
8	1179	23.938	1179	0	100.00
			0	0	
9	1130	28.221	1112	5	99.56
			0	13	
10	1313	24.622	1299	0	100.00
			0	14	
11	2264	26.202	2230	0	99.91
			2	32	
Total	13976	23.350	13900	12	99.88
			5	59	

The recognition rates include the eye detection rates and tracking rates. As shown in **Table 2**, the proposed method can robustly detect eyes within 23.350ms averagely. Therefore, the method is fast enough to be applied to real-time systems. **Table 2** shows also that this method can detect eyes when users are wearing glasses or sunglasses (DB 2, 4, 6, 8, 10) and when they are under dark environments (DB 1, 9, 11). Additionally, the eyes of subjects who are not trained at all are also recognized well (DB 7, 8, 9, 11). The accuracies of all the DB are more than 99%.

In addition, we have experimented some other DBs, which are captured in cars under various illumination conditions to observe how this system operates in the real outdoor condition. Some results are shown in **Fig. 16**, where left six results show the results of bright conditions and the others show the results of dark conditions. One of the additional experimental results can be found on YouTube (https://youtu.be/A4_0R2pmMpw). This DB contains various conditions, when the car is going through off-road and on-road, and when the sunbeam directly lights on faces. Although all the training data sets are captured in indoor conditions and they are not that much various, this system detects eyes robustly. This system uses very simple algorithms for tracking and finding pupils so that the detected eye positions could be slightly different from real eye positions.



Fig. 15. Examples of result images (indoor conditions).



Fig. 16. Examples of result images (outdoor conditions).

Some examples of false positives and false negatives are shown in **Fig. 17**, where most false positives are observed in positions near to eyes, and most false negatives are caused by orientation of head, saturation of illumination or reflections of glasses.

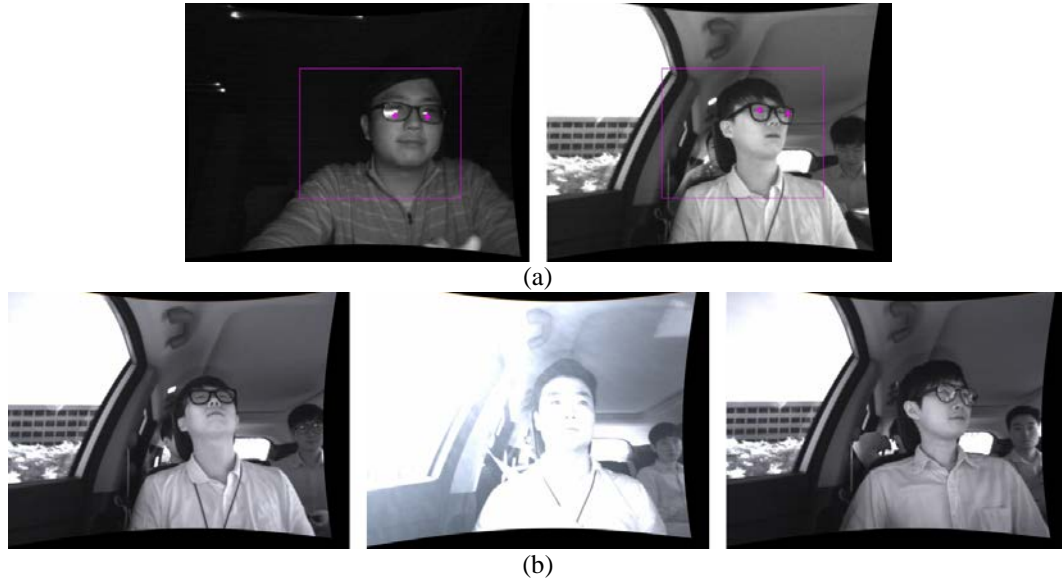


Fig. 17. Examples of false positives (a) and false negatives (b).

The accuracy comparison with other methods using RGB cameras [22,23], and IR cameras [24,25] are shown in **Table 3**, where the average accuracy of the proposed method is the highest for all environments.

Table 3. Accuracy comparison with other methods

Methods	Without Glasses	With Glasses	With Sunglasses	Dark Environment	Total Average
Kim, 2012 [22]	99.70 %	95.60 %	N/A	N/A	98.00 %
Petrisor, 2011 [23]	91.27 %	N/A	N/A	N/A	91.27 %
Zhao, 2006 [24]*	99.40 %	88.30 %	N/A	N/A	93.85 %
Bhowmick, 2009 [11]	90.34 %	N/A	N/A	N/A	90.34 %
Proposed	99.83 %	99.95 %	100.00 %	99.88 %	99.88 %

*The results of the Zhao's method are true positive rates.

A result of finding pupils and 3D coordinates of pupils is shown in **Fig. 18**, where the detected pupil positions are marked with the green crosses. In **Fig. 18**, the calculated 3D coordinates of the left eye are ($X=-5.327$ cm, $Y=-3.613$ cm, and $Z=78.111$ cm), and those of the right eye are ($X=12.310$ cm, $Y=-3.834$ cm, and $Z=77.262$ cm). The distance from the left eye to the right eye is 7.038 cm, and this is very similar to the actual distance, about 7.0~7.1cm.



Fig. 18. Result of finding pupils.

5. Conclusion

In this paper, we proposed a novel eye tracking method that requires only a low-resolution IR stereo camera and a set of IR illuminators. This method uses evidential reasoning and geometrical rules to detect eye pair candidates, and uses MLPs to verify the eye region and the pure facial region to reduce processing time and to increase detection performance. The average processing time was less than 24ms, which is applicable to real-time systems. The MLPs are well trained using various conditions including various glasses-on conditions, various eyes, faces and various facial expressions; therefore, it is robust to various conditions. This method can estimate 3D coordinates of eyes in order to indicate where people are looking at. Therefore, the proposed method can contribute greatly to ITSs, windshield head-up displays for vehicles, and even HCI applications.

References

- [1] S. A. Sirohey and A. Rosenfeld, "Eye detection in a face image using linear and nonlinear filters," *Pattern recognition*, vol. 34, no. 7, pp. 1367-1391, 2001. [Article \(CrossRef Link\)](#)
- [2] S. Tripathi, V. Sharma and S. Sharma, "Face detection using combined skin color detector and template matching method," *International Journal of Computer Applications*, vol. 26, no. 7, pp. 5-8, July, 2011. [Article \(CrossRef Link\)](#)
- [3] J. Dowdall, I. Pavlidis and G. Bebis, "Face detection in the near-IR spectrum," *Image and Vision Computing*, vol. 21, no. 7, pp. 565-578, July, 2003. [Article \(CrossRef Link\)](#)
- [4] G. C. Feng and P. C. Yuen, "Multi-cues eye detection on gray intensity image," *Pattern recognition*, vol. 34, no. 5, pp. 1033-1046, May, 2001. [Article \(CrossRef Link\)](#)
- [5] K. Peng, L. Chen, S. Ruan and G. Kukharev, "A robust algorithm for eye detection on gray intensity face without spectacles," *Journal of Computer Science & Technology*, vol. 5, no. 3, pp. 127-132 2005. [Article \(CrossRef Link\)](#)
- [6] A. Królak and P. Strumiłło, "Eye-blink detection system for human-computer interaction," *Universal Access in the Information Society*, vol. 11, no. 4, pp. 409-419, 2012. [Article \(CrossRef Link\)](#)
- [7] W. Zhong and Z. Huang, "The real-time eye detection for single user based on template matching," in *Proc. of IEEE Int. Conf. on Computational Science and Engineering*, pp. 831-834, December 19-21, 2014. [Article \(CrossRef Link\)](#)

- [8] M. Rezaei and R. Klette, "Adaptive Haar-like classifier for eye status detection under non-ideal lighting conditions," in *Proc. of 27th Conf. on Image and Vision Computing New Zealand*, pp. 512-526, 2012. [Article \(CrossRef Link\)](#)
- [9] A. Majumder, L. Behera and V. K. Subramanian, "Automatic and robust detection of facial features in frontal face images," in *Proc. of UKSim 13th Int. Conf. on Computer Modelling and Simulation*, pp. 331-336, March 30-April 1, 2011. [Article \(CrossRef Link\)](#)
- [10] D. W. Hansen and R. I. Hammoud, "An improved likelihood model for eye tracking," *Computer Vision and Image Understanding*, vol. 106, no. 2-3, pp. 220-230, May-June 2007. [Article \(CrossRef Link\)](#)
- [11] B. Bhowmick and K. S. C. Kumar, "Detection and classification of eye state in IR camera for driver drowsiness identification," in *Proc. of IEEE Int. Conf. on Signal and Image Processing Applications*, November 18-19, 2009. [Article \(CrossRef Link\)](#)
- [12] S. W. Shih and J. Liu, "A novel approach to 3-D gaze tracking using stereo cameras," *IEEE Trans. Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 34, no. 1, pp. 234-245, February, 2004. [Article \(CrossRef Link\)](#)
- [13] C. Ma, S. J. Baek, K. A. Choi and S. J. Ko, "Improved remote gaze estimation using corneal reflection-adaptive geometric transforms," *Optical Engineering*, vol. 53, no. 5, 053112, May, 2014. [Article \(CrossRef Link\)](#)
- [14] D. Beymer and M. Flicker, "Eye gaze tracking using an active stereo head," in *Proc. of IEEE Conf. Computer Vision and Pattern Recognition*, June 18-20, 2003. [Article \(CrossRef Link\)](#)
- [15] Z. Zhu, K. Fujimura and Q. Ji, "Real-time eye detection and tracking under various light conditions," in *Proc. of ACM Symp. on Eye Tracking Research & Applications*, pp. 139-144, 2002. [Article \(CrossRef Link\)](#)
- [16] Z. Zhu and Q. Ji, "Robust real-time eye detection and tracking under variable lighting conditions and various face orientations," *Computer Vision and Image Understanding*, vol. 98, no. 1, pp. 124-154, April, 2005. [Article \(CrossRef Link\)](#)
- [17] Y. Park and D. Lee, "3D vision-based security monitoring for railroad stations," *Journal of the Optical Society of Korea*, vol. 14, no. 4, pp. 451-457, December, 2010. [Article \(CrossRef Link\)](#)
- [18] O. Nikisins and M. Greitans, "Local binary patterns and neural network based technique for robust face detection and localization," in *Proc. of Int. Conf. of IEEE Biometrics Special Interest Group*, pp. 1-6, September 6-7, 2012. [Article \(CrossRef Link\)](#)
- [19] B. Jun, I. Choi and D. Kim, "Local transform features and hybridization for accurate face and human detection," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 35, no. 6, pp. 1423-1436, June, 2013. [Article \(CrossRef Link\)](#)
- [20] Y. He and N. Sang, "Robust illumination invariant texture classification using gradient local binary patterns," in *Proc. of Int. Works. on IEEE Multi-Platform/Multi-Sensor Remote Sensing and Mapping*, pp. 1-6, January 10-12, 2011. [Article \(CrossRef Link\)](#)
- [21] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vo. 22, no. 11, pp. 1330-1334, November, 2000. [Article \(CrossRef Link\)](#)
- [22] C. Kim, S. Choi, M. Turk and C. Choi, "A new biased discriminant analysis using composite vectors for eye detection," *IEEE Trans. Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 42, no 4, pp. 1095-1106, August, 2012. [Article \(CrossRef Link\)](#)
- [23] D. Petrişor, C. Foşalău, M. Avila and F. Măriuş, "Algorithm for face and eye detection using colour segmentation and invariant features," in *Proc. of 34th Int. Conf. on Telecommunications and Signal Processing*, pp. 564-569, August 18-20, 2011. [Article \(CrossRef Link\)](#)
- [24] S. Zhao and R. R. Grigat, "Robust eye detection under active infrared illumination," in *Proc. of Int. Conf. on Pattern Recognition*, August 20-24, 2006. [Article \(CrossRef Link\)](#)



Sungsoo Lim received the B.S. degree in Electronic Engineering and Biomedical Engineering from Kyung Hee University, in 2013. He also received the M.S. degree in Electronic Engineering from Kyung Hee University, in 2016. He is currently pursuing his Ph.D. degree in Electronic Engineering at the same university. His research interests include computer vision, image processing, ITS (intelligent transportation system), HCI (human computer interface), medical image processing, artificial intelligence, and digital signal processing.



Daeho Lee received M.S. and Ph.D. degrees in Electronic Engineering from Kyung Hee University, Seoul, Korea, in 2001 and 2005, respectively. He has been an Associate Professor in the Humanitas College at Kyung Hee University, Korea, since 2005. His research interests include computer vision, pattern recognition, image processing, computer games, ITS (intelligent transportation system), HCI (human computer interaction), EIT (electrical impedance tomography) analysis and digital signal processing.