



Real Time Eyeball Movement Detection Based on Region Division and Midpoint Position

Renaldi Primaswara Prasetya¹Fitri Utamingrum^{1*}Wayan Firdaus Mahmudy¹¹*Faculty of Computer Science, Brawijaya University, Malang, Indonesia** Corresponding author's Email: f3_ningrum@ub.ac.id

Abstract: The development of technologies that utilize eye movements as interface for motion detection has an important role in human-computer interaction especially as a medium for controlling automated device such as electrical wheelchair. The reliability of eye movement detection can determine the system performance which can be implemented into an automated device. By tracking the movement of the eyeball, communication between users and automated devices can take easily and may be used by people with hand-foot impairment. In this paper, we propose a method by using the division of the eye region and checking the eyeball midpoint position for detecting the direction of eyeball movement which can be used as a navigation in real time condition. We also implement Haar Cascade method for detecting the eye region followed by tracking using Kernelized Correlation Filter to produce stable movement. This approach was proved reliable and could increase the precision and recall on eyeball movement detection significantly, reach the average to 0.93 and 0.84. Even, our proposed method enables a suitable level of sensitivity with low computational time.

Keywords: Eyeball movement detection, Region division, Eyeball midpoint, Tracking, Real time, Media control.

1. Introduction

The interaction of human and computer within a few years has developed quite rapidly. Nowadays, many devices have been controlled automatically. However, not everyone is able to interact with an automated device easily like people with impairment. In fact, automated device such as electrical wheelchair is one of the most essential requirements for people with impairment's social life, so they can do mobility easily. In its development, people with foot impairment is able to use a joystick or a remote to interact with a device. But these media are useless for people with hand-foot impairment. In order to overcome the problems, there should be other ways that can overcome these limitations. Eyes could be an alternative choice that used for communication between users and an automated device. Previous research utilized eye tracking that will give access to individuals with severe motor disabilities [1], the eyeball movement also used for controlling home appliance [2]. Moreover, eye movement detection

system also implement in vehicle industry field for detect the driver fatigue [3-5].

The process of detecting the eyeball movement consist of detecting the eye region that could be used as a focus region in movement detection phase, followed by the process of eyeball detection. This research is emphasized on developing methods in detecting the movement of the eyeball that can be used as a navigation tool of an automated device such as a wheelchair, smart motor, or replacement of remote functions.

In this paper, the vertical and horizontal division of the eye region box produced by the Haar Cascade method and the Kernelized Correlation Filter (KCF) tracker are proposed to generate accurate and reliable eyeball movement detection with low sensitivity level and low computational time. At first, Haar Cascade Classifier is performed for detecting the eye region with marked by a region box. After getting the eye region, we apply the KCF tracker to make the region box more stable and unchanged. Then the detected of eye region will be cropped, and

will be undergo a segmentation process by converting into binary image for getting the eyeball area. Finally, the direction of the eyeball movement will be determined inside the eye region box.

This paper is prepared as follows. In Section 2, explains related research that focuses on the method of eyeball movement detection and their drawbacks. In section 3, provides the proposed an architecture of the eye region to get proper eye movement detection. In Section 4, experiments will be conducted along with a discussion of the results. The final Section will be made a conclusion.

2. Literature review

Eye movement detection phase becomes an important part, especially if applied as media control of an automated device. Robust and rapidly detection of eye movements in an automated device can make the system run properly and in accordance with user commands. Previous research have been done by using x and y coordinates-based method to determine the direction of the eyeball movement [6]. At the beginning, this method determines the initial coordinate referred to Point of Gaze (PoG) of the iris when facing forward as a reference. Then, the new coordinates generated when the iris moves and will be compared to the PoG. If there is a change of x and y coordinate position from its reference position, it will be detected as movement. But coordinate-based method only accommodate two types of movement, and it has a high sensitivity without providing the appropriate range of movement, since a slight change in coordinate position has been considered as a movement, so it resulting an ambiguous direction.

K. Arai and R. Mardiyanto [7] presented a method using the angle value of each eyeball movement. The angular value calculation is obtained based on the \sin and \arcsin of a triangular formula. If the angle value is greater than the predefined threshold value, it will be detected as an eyeball movement. This method also has accuracy issues. The angle value generated from several directions of eyeball movement can be the same value and the threshold value can be changed, thereby reducing the reliability of the system.

Previous method have been used to detect the eyeball movement by utilizing Electrooculography (EOG). EOG is a method that uses electrodes placed on the skin surface around the eye [8]. The electrode is processed using an amplifier instrument and based on user's eye movements that generate voltage and then send an electrical signal to the microcontroller to determine the movement of an electric device

such as a wheelchair [9]. But the EOG-based method is constrained by some noise such as facial muscles, muscle movement of the body, head movement, and speech activity. In addition, this method is spending cost enough.

Another study was conducted by S. Hoppe and A. Bulling [10] by using Convolution Neural Network (CNN) and Fourier Transform (FFT) to detect the eye movement. Eye movement detection is obtained by observing the change in signal components generated by image pixels from eye movement. The signal features of eye movements are described in other forms into frequency components using FFT. Then, the feature will be identified using the CNN method to classify the type of eye movement. The approach by using FFT and CNN method as a deep learning algorithm can produce the eye movement detection which is quite detailed but it takes a long computational time.

General method has been offered by utilizing the calculation of the pixels amount in a binary image. Then, the movement of an eyeball is obtained by comparing the pixel amount of the eyeball [11]. However, this traditional method also can only accommodate two types of movement and requires a long computation time, because the system must calculate all section of pixels of the eyeball on each part. Even, comparing the number of pixels of the eyeballs enable the same pixel values which can produce an error detection.

Ghani *et al.* [12] have proposed the key-point position of the eye region which has an advantage in computational time in the process of detecting the eyeball movement. The process begins by detecting the eye area and then marks the point of coordinates of all parts of the eye including the position of the eye corner using edge detection approach. While, the eyeball detection was done by using the Hough Circle method to determine the pupil area. Furthermore, the eyeball movement will be detected if the pupil coordinates change to the position of the eye corner coordinates. The key-point-based methods also have high sensitivity issues, since a slight change in key-point has been detected as a movement. In addition, some noise around the eyes can lead to errors in the determination of coordinate points of the eye corner that can affect the process of eyeball movement detection.

To overcome the drawback of existing techniques, we propose method for detecting eyeball movement by divided the region of eye into nine parts, that is vertically and horizontally in which each region have different size area with certain range. Then the eyeball movement will be detected by checking the existence of eyeball midpoint

position in one of all parts of the region to determine the direction of eyeball movement. Hence, this approach improves accuracy and accommodate several movement direction that could be used to control the movement of an automated device. Moreover, minimize the level of sensitivity with low computational time, can be achieved by using this proposed method.

3. Proposed method

Our work scheme consists of four main stages before entering the main stage of the eyeball movement detection, such as getting the eye region box, stabilizing the eye region box, detecting the eyeball, and the division of the eye region box either, vertically and horizontally.

To clarify the contribution part of the proposed method, block diagram is given in Fig. 1.

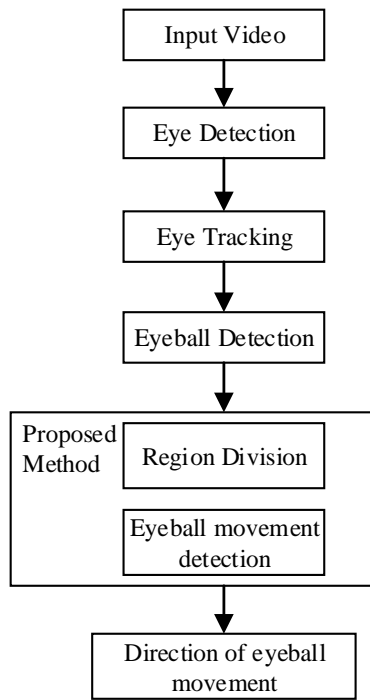


Figure.1 Block diagram of eyeball movement detection

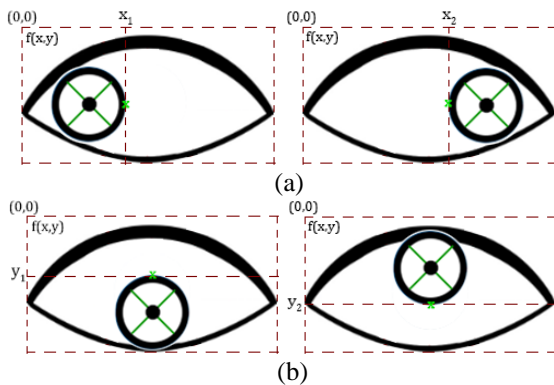


Figure.2 Calibration process to obtain coordinate points: (a) obtain x_1 and x_2 point (b) obtain y_1 and y_2 point

The division of the eye region box is obtained through the coordinate points of $x_1, x_2, y_1,$ and y_2 . As illustrated in Fig. 2, in order to obtain each coordinate point in the region division, the calibration process is done by moving the eyeball into four directions i.e. right, left, up, and down. For example, when the user moves the eyeball to the right, then by looking at the pixels that are on the outer edge which is marked with green points, a position of point x_1 is obtained. When the eyeball moves to the left, it will get x_2 point. The same thing happens to get the position of point y_1 and y_2 , respectively when the eyeball moves up and down.

Meanwhile, the values of $x_1, x_2, y_1,$ and y_2 , are the maximum (*max*) or minimum (*min*) pixel values of the eyeball image and can be described in Eq. (1) until Eq. (4).

$$x_1 = \max(x \in A) \tag{1}$$

$$x_2 = \min(x \in A) \tag{2}$$

$$y_1 = \max(y \in B) \tag{3}$$

$$y_2 = \min(y \in B) \tag{4}$$

Where A can be expressed as the set of pixel values of x coordinates in the eyeball image $f(x, y)$, while B is the set of pixel values of y coordinates in the eyeball image $f(x, y)$ as seen in Eqs. (5) and (6).

$$A = \{x \mid x \text{ is the pixel value of } x \text{ in the image } f(x, y)\} \tag{5}$$

$$B = \{y \mid y \text{ is the pixel value of } y \text{ in the image } f(x, y)\} \tag{6}$$

After obtained the coordinate points of $x_1, x_2, y_1,$ and y_2 , it will form the architecture model of the eye region box division as shown in Fig. 3. Based on the box region that has been design, each part of region will be checked whether there is the existence of the midpoint of the eyeball to determine the eye movement direction. Whether the eye moves to the right, left, up, down, or forward.

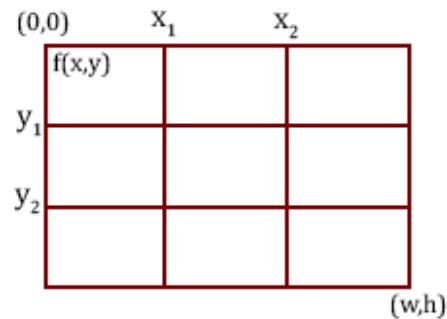


Figure.3 The architecture of the region box has been divided into nine parts.

3.1 Eye region detection

Before detecting the eye movement, the system must be able to detect the region of eye. In this case, we used the Haar Cascade classifier method that has the ability to detect an object appropriately [13]. The detection process in the Haar Cascade was done by changing the RGB image into grayscale image and using integral calculation with some image enhancement technique, considering good image quality, can improve the detection process [14].

Haar Cascade perform filtering process using selected haar feature template, inside a certain size of rectangular window to find the eye region [15]. This filtering was done using selected haar feature template. If the value of the haar features in a certain region below the threshold then it has to be qualify. However, if a haar feature value in a region is not qualified, so there is no object to be detected on it and its region area relocates to the others area. If the area meets the feature, then the rules are performed to the next feature. At least, if all requirements fulfilled feature, it means that there is an eye part in the area of the region. Considering region generated by Haar Cascade method still contains other unimportant parts, region of eye would be minimize by eliminate other parts such as eyebrow that interfere with the next detection process shows in Fig. 4, and could be determine the fixed eye region for use at the next phase.

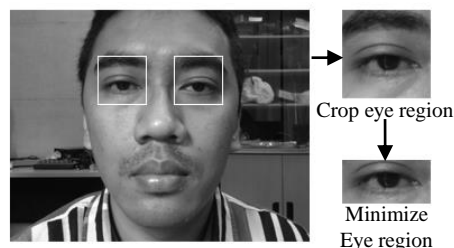


Figure.4 Cropping region of eye and minimize eye region to remove unimportant area

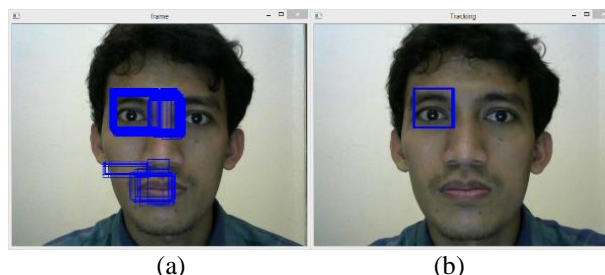


Figure.5 Eye box region without tracking process: (a) spread everywhere, while the detection followed by tracking process (b) is more focused

3.2 Tracking using Kernelized correlation filter (KCF)

Tracking is the process of searching for a particular moving object on a digital image in a frame sequence. In other words the tracking process determines a fixed label on objects that are tracked in different frames in a video [16]. There have been many studies that developed the tracking method one of them is the KCF method.

KCF tracker became the benchmark standard in visual tracking and ranked third in the 2014 Visual Object Tracking competition [17]. KCF solves tracking problems with simple linear regression and non-linear regression through double filters training data, enabling the use of multi-dimensional features such as Histogram of Gradient (HOG) and nonlinear kernels such as Gaussian Filter. As same as other detection-based trackers, KCF can be trained using a set of training template variations by utilizing cyclic shift modelling. Then to increase the processing speed of various template variations, KCF utilizes circulant matrix properties [18].

This tracking method is designed using the fact that some positive samples used on Multi Instance Learning trackers that proposed by [19], have overlapping areas that are too large to be fixed by the KCF method [20]. KCF tracker method has an advantage in terms of motion change and noise experiment that appropriate with our method approach. Therefore, the condition of the eye region box becomes stable and more focused. While the detection without tracking process leads to instability in the initialization process, because the eye detection process will occur in each frame, which will affect the distribution of the eye region box spread everywhere in each frame, even tending to detect errors around the eye area as shown in Fig. 5.

3.3 Eyeball detection

Detection of the eyeball area is obtained by converting the eye image into binary image and assembled by a morphological process. This was done to produce a more intact eyeball shape. In morphology, there are two fundamental operations of erosion and dilation. Closing is a process of dilation followed by erosion, the resulting effect is to fill small holes in objects, combine objects together, and generally made smooth boundary of a large object without changing the object area significantly [21].

Eliminating noise becomes an important thing in the processing of digital images [22-23]. Closing removes small objects from the foreground of an

image and used to find or detect something of the corresponding with a structuring element. Generally, the Eq. (7) shows the formulation of the closing operation, it seems that process of dilation (\oplus) of an image A followed by erosion process ($-$) based on structuring element B . Erosion is useful for constricting or attenuating objects in binary images, while dilation is useful for extending or bolding objects in binary images.

$$A \bullet B = (A \oplus B) - B \tag{7}$$

By using the closing operation, then the eyeball area will be clear as shown in Fig. 6 Afterward, the midpoint of the eyeball can be determined. Eq. (8) can be used for extract centroid data of an eyeball.

$$f(i, j) = f\left(\frac{width}{2}, \frac{height}{2}\right) \tag{8}$$

The midpoint coordinates $f(i, j)$ of the eyeball is the point intersection from width of eyeball region divided by two, with the height of eyeball region divided by two.

3.4 Eyeball movement detection

After getting the essential components of our scenario, such as getting the eye region box, stabilizing the detection of the eye region, detecting the eyeball and the midpoint of the eyeball, then we can detect the movement of the eyeball by using the division of the region box.

Region of eye divided into nine parts region that is vertically and horizontally with certain size as describe before. For horizontal partition, calibration is required with the movement of the eyeball to the right and to the left. While, for vertical partition, calibration is required with upward and downward gaze. Furthermore, the eyeball movement can be determined by detecting the midpoint $f(i, j)$ position of eyeball inside of the region box. Based on the illustration in Fig. 7, the rules of eyeball movement detection can be translated by the following Eq. (9) until Eq. (12):

$$Right\ f(i, j) = \begin{cases} i > 0\ and\ i < x_1 \\ j > y_1\ and\ j < y_2 \end{cases} \tag{9}$$

$$Left\ f(i, j) = \begin{cases} i > x_2\ and\ i < w \\ j > y_1\ and\ j < y_2 \end{cases} \tag{10}$$

$$Upward\ f(i, j) = \begin{cases} i > x_1\ and\ i < x_2 \\ j < y_1\ and\ j > 0 \end{cases} \tag{11}$$

$$Downward\ f(i, j) = \begin{cases} i > x_1\ and\ i < x_2 \\ j > y_2\ and\ j < h \end{cases} \tag{12}$$

$$Forward\ f(i, j) = \begin{cases} i > x_1\ and\ i < x_2 \\ j > y_1\ and\ j < y_2 \end{cases} \tag{13}$$

If midpoint coordinate of eyeball is included between range 0 to x_1 and between range y_1 to y_2 , it means the user moves the eye toward to the right. Conversely, if midpoint coordinate of eyeball is included between range x_2 until width (w) of the region box, and between range y_1 to y_2 , it means the user moves the eye toward to the left. As well as upward or downward movements, when midpoint coordinate of eyeball is included in range 0 to y_1 and between range x_1 to x_2 , it means the eye of the user moves upward. While, if midpoint of eyeball is included between range y_2 to height (h) of the region box, and between range x_1 to x_2 , it means the eye of the user moves downward. We also determine the initial eye gaze position that used as a reference and it's considered as facing forward. The detection of facing forward obtain when the midpoint of the eyeball is included between range x_1 to x_2 , and between y_1 to y_2 .

Referring to Fig. 7, region located in the right and left corner are not applied in detecting the movement of the eyeball. This is done in order to avoid the oblique direction, while the system only recognize four types of direction and facing forward as an initial gaze.

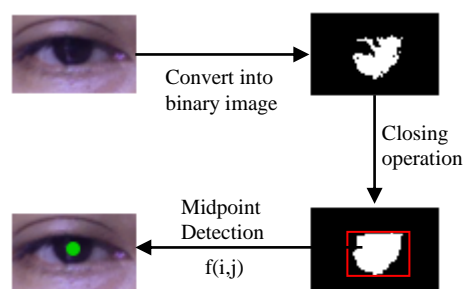


Figure.6 The eye region will undergo closing operation to determine the midpoint of the eyeball

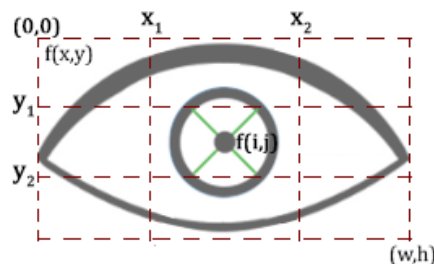


Figure.7 Midpoint position in each region would determine the direction of eyeball movement

Table 1. Calibration for eye region division

Data Testing	Right Gaze (x ₁)	Left Gaze (x ₂)	Downward Gaze (y ₁)	Upward Gaze (y ₂)
1	28	50	11	30
2	29	49	15	30
3	28	51	9	34
4	25	48	10	33
5	26	49	19	35
6	27	50	14	31
7	29	49	18	31
8	29	50	10	35
9	25	49	12	31
10	26	51	18	32

4. Experimental result and discussion

To test the proposed method that have been created, we did some experiments and tested by involving 20 users around the University, with the characteristics of rounded and narrow eyes, using a webcam with 720hd resolution in real time which is each users performs 2 times experiment for each eye gaze. This experiment implemented on an Intel Core i5-5200U processor 2,7GHz using Python 3.5 with OpenCV. As a limitation, this test is performed in bright enough lighting conditions, considering the lighting condition affect on the detection process. The data testing has been undergone scaling process into 480 pixel height x 640 pixel width, and the user is within 30 cm in front of the camera.

Before testing the eyeball movement, we did some experiment to assign the values of the coordinate points to be used in the region division involving ten persons by taking the most frequent values of each coordinate point as shown in Table 1. Based on the experiment, the coordinate value of x₁ is 29, the coordinate value of x₂ is 49, the coordinate value of y₁ is 18, and the coordinate value of y₂ is 31. The following Fig. 10 shows the results of eyeball movement detection using our proposed method with predefined region size.

To evaluate the performance of our proposed method quantitatively with four types of eye gaze and facing forward as initial gaze, we measure in terms of recall, precision calculation for multiclass problems as describe in Eq. (10) and Eq. (11), define by the number of True Positive (TP), False Positive (FP), and False Negative (FN).

$$\text{Precision} = \frac{TP}{TP + FP} \tag{10}$$

Table 2. Eye movement detection result using proposed method

	Right	Left	Upward	Downward	Total Predicted
Predicted Right	35	0	1	0	36
Predicted Left	0	38	0	1	39
Predicted Upward	4	2	35	0	41
Predicted Downward	1	0	0	28	29
Predicted Front	0	0	4	11	15
Total Eye Gaze	40	40	40	40	

$$\text{Recall} = \frac{TP}{TP + FN} \tag{11}$$

TP means the output system in accordance with the user command. FP means that, the number of detection errors involving all classes type of eye gaze, FN means the number of detection errors involving only one class type of eye gaze. Precision represent the success rate of an eye movement class type, of all types of test data classes, while recall represent the success rate of an eye gaze class type, from each class of testing data. Table 2 shows the result of our proposed method using multiclass calculation.

The left diagonal data that are marked bold, shows that our proposed method successfully detects the eyeball movement with reached total of 136 data as shown in Table 2. This data will be compared with the Angle Value method to prove that our proposed method is reliable and accurate in detecting real-time eyeball movement. Based on multiclass data calculation and each result of eyeball movement from Fig. 8, we get precision and recall of our proposed method compared with Angle Value method as shown in Table 3.

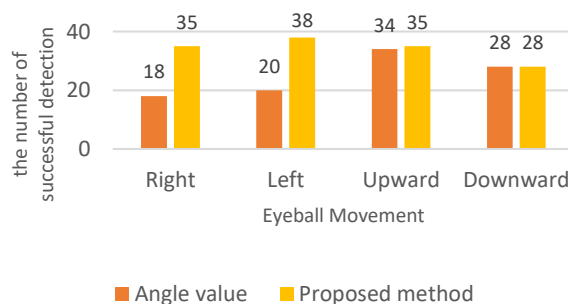


Figure.8 Eyeball movement detection result of each methods

Table 3. Comparison result of our proposed method with angle value-based method

Angle value based method [7]	Precision	Recall	Processing Time Average
Right	0.78	0.45	0.007 s
Left	0.86	0.5	0.015 s
Downward	0.7	0.7	0.012 s
Upward	0.47	0.85	0.006 s
Proposed method			
Right	0.97	0.87	0.0016 s
Left	0.97	0.95	0.0013 s
Downward	0.96	0.7	0.0017 s
Upward	0.85	0.87	0.0014 s

Since pixel based-method only accommodate right and left gaze, so in this comparison, we only compare our method with the right and the left gaze of eyeball movement. Graph data on Fig. 9 and the data in Table 4 shows the comparison between our proposed method and the Pixel value method. Meanwhile, on coordinate-based methods that have sensitivity problems, our method will be compared based on the accuracy value of the forward gaze which is the initial gaze position, where the forward gaze can indicate the sensitivity level of each methods. Table 5 shows the comparative results between our proposed method and the coordinate-based method, calculated from the total correct data divided by the total of the overall data.

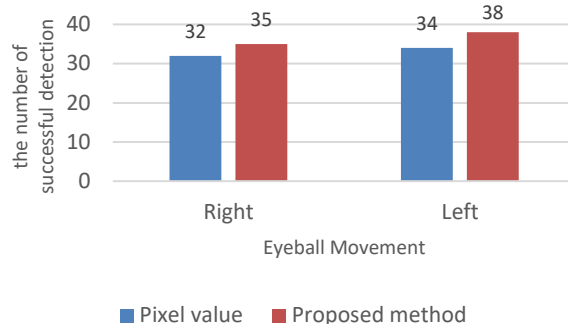


Figure.9 Eyeball movement detection result compared with pixel value methods

Table 4. Comparison result of our proposed method with pixel value-based method

Pixel value based method [8]	Accuracy	Processing Time
Right	80 %	0.034 s
Left	85 %	0.029 s
Proposed method		
Right	97 %	0.0016 s
Left	97 %	0.0013 s

Table 5. Comparison result of our proposed method with coordinate method

Methods	Front Gaze Accuracy
Coordinate Method [6]	20 %
Proposed method	80 %

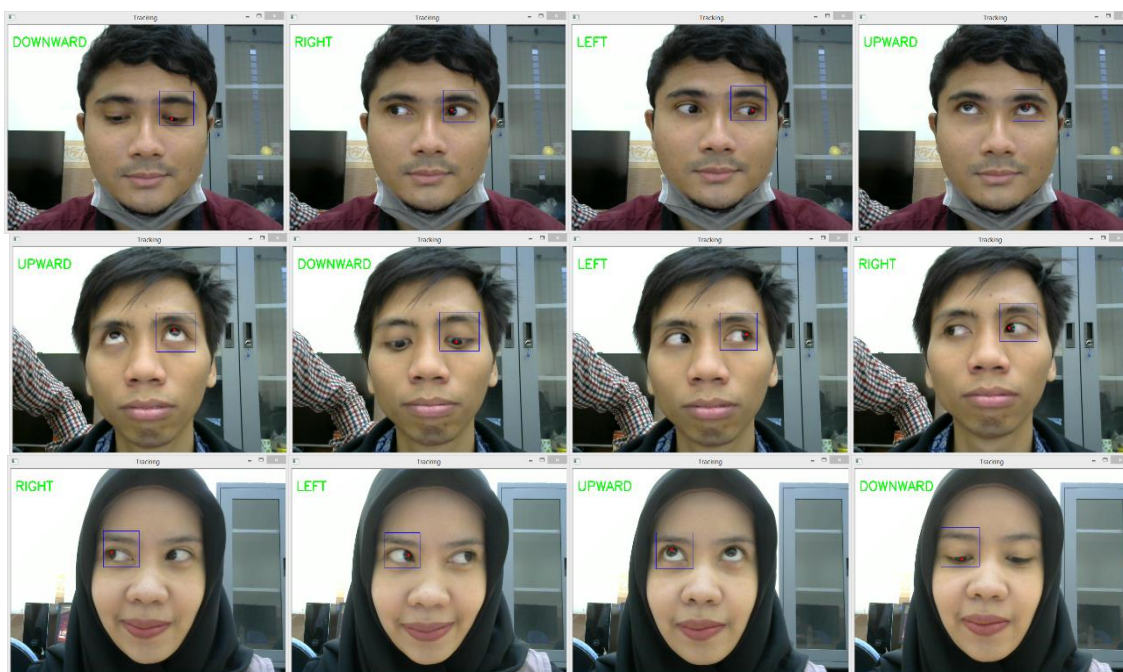


Figure. 10 Some of eyeball movement detection result using our proposed method

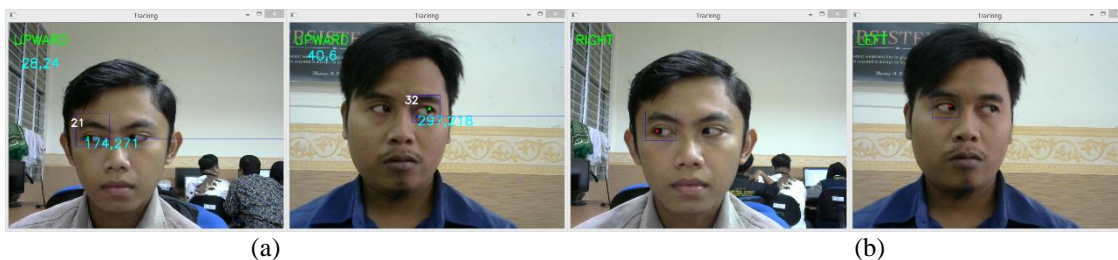


Figure. 11 Eyeball movement detection result from comparison method: (a) angle value method get wrong detection because of the threshold value and (b) our proposed method can fix the detection problems

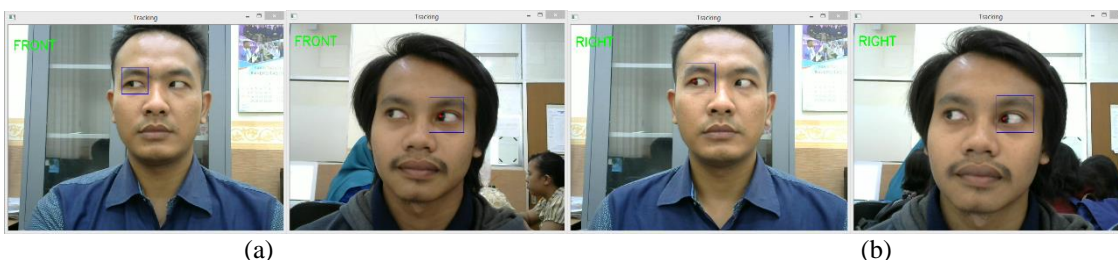


Figure. 12 Eyeball movement detection result from comparison method: (a) pixel of Region method get wrong detection and (b) our proposed method can fix the detection problems

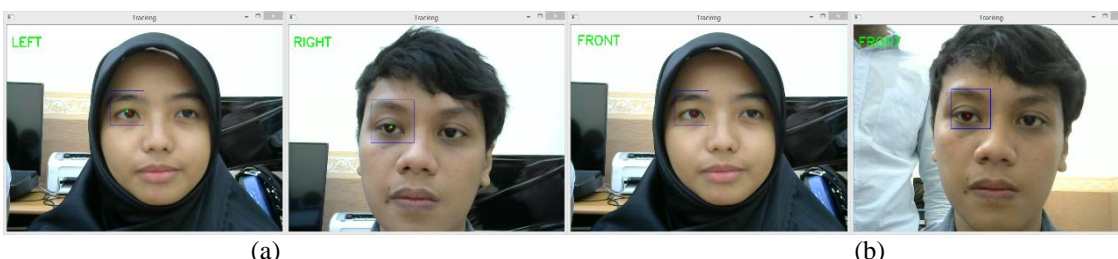


Figure. 13 Eyeball movement detection result from comparison method: (a) coordinate method produce an ambiguous direction and high sensitivity and (b) our proposed method can fix the detection problems

From the experimental results, it show that our proposed method generate reliable eyeball movement detection proven by each gaze of eyeball movement obtained better precision, recall and accuracy with low computational time, compared with previous method both Angle value method and Pixel value method as presented in Table 3 and Table 4. In addition, sensitivity issues from Coordinate method can also be handled by our proposed method as presented in Table 5.

The best level of precision and recall of our proposed method is obtained from left gaze of eye movements, which reaches the value of 0.97 and 0.95, followed by the right gaze with a precision value of 0.97 and recall of 0.87, as same as the accuracy value also get a high enough results.

Eyeball movement detection based on angle value method produces a detection error, due to an improper threshold value assignment. As same as detection using the Pixel of Region method allows the same number of pixels on each region which also causes detection errors, while our proposed method can fix the detection problems as shown in Fig. 11 and Fig. 12. On coordinate-based method,

the detection of eyeball movement has a high sensitivity level which slightly changes from the reference point (green point) is already considered a movement as shown in Fig. 13. Thus, produce an ambiguous movement direction, whether the movement of the eyeball already moving to the right, left, or still facing forward. Our techniques can overcome the drawback. The movement should be still detected in range of front gaze as generated by our proposed method.

5. Conclusion

This paper provides a reliable method for detecting the eyeball movement in real time condition using the design of the eye region box produced by the Haar Cascade method and the KCF Tracker which has been divided into nine parts either vertically or horizontally. The eyeball movement will be detected by checking the existence of the eyeball midpoint position in one of all parts of the region box to determine the direction of eyeball movement. By using these techniques, the eyeball movement can be detected properly. The

experimental results show the precision, recall, and accuracy of each direction of the eyeball movement using our proposed method is higher than the angle value based-method and the pixel value-based method.

In addition, the division of the eye region box by providing the appropriate range can overcome the sensitivity problems generated by the coordinate-based method. Moreover, our proposed method has a low computational time, because it only checks the midpoint position of the eyeball in each part of the region box without having to calculate and compare the total number of eyeball pixels as conducted by the pixel value based-method, and without having to calculate the angle value using trigonometric formula as conducted by the angle value-based method.

The high of accuracy with a suitable level of sensitivity and low computational time, makes the proposed method useful as an alternative medium for controlling automated devices such as wheelchairs and joystick replacements for people with hand-foot impairment. For future work, it can improve the performance of eyeball movement detection which can adjust the lighting conditions adaptively, considering the image processing of the eyeball area in real time is very dependent on the lighting conditions. Hence, we can add the image enhancement method in order to obtain the eyeball movement detection more quickly and accurately.

Acknowledgments

This work was conducted in Computer Vision Research Group, Faculty of Computer Science, Brawijaya University. We also would like thank to people who keen to provide pictures as dataset.

References

- [1] A. Sesin, M. Adjouadi, M. Ayala, M. Cabrerizo, and A. Barreto, "Eyeing a real-time Human-Computer Interface to Assist Those with Motor Disabilities", *IEEE Potentials*, Vol.27, No.3, pp.19-25, 2008.
- [2] H. Heo, J. M. Lee, D. Jung, J. W. Lee, and K. R. Park, "Nonwearable Gaze Tracking System for Controlling Home Appliance", *The Scientific World Journal*, Vol. 2014, 2014.
- [3] A. A. Lenskiy and J. S. Lee, "Driver's Eye Blinking Detection Using Novel Color and Texture Segmentation Algorithms", *International Journal of Control, Automation, and Systems*, Vol.10, No.2, pp.317-327, 2012.
- [4] P. Jimenez, L. M. Bergasa, J. Nuevo, N. Hernandez, and I. G. Daza, "Gaze Fixation System for The Evaluation of Driver Distractions Induced by IVIS", *IEEE Transactions on Intelligent Transportation Systems*, Vol.13, No.3, pp.1167-1178, 2012.
- [5] F. Vicente, Z. Huang, X. Xiong, F. De la Torre, W. Zhang, and D. Levi, "Driver Gaze Tracking and Eyes Off The Road Detection System", *IEEE Transactions on Intelligent Transportation Systems*, Vol.16, No.4, pp.2014-2027, 2015.
- [6] M. Gneo, M. Schmid, S. Conforto, and T. D'Alessio, "A Free Geometry Model-Independent Neural Eye-Gaze Tracking System", *Journal of Neuroengineering and Rehabilitation*, Vol.9, No.1, pp.82-96, 2012.
- [7] K. Arai and R. Mardiyanto, "A Prototype of Electric Wheelchair Controlled by Eye-Only for Paralyzed User", *Journal of Robotics and Mechatronics*, Vol.23, No.1, 2011.
- [8] A. Bulling, J. A. Ward, H. Gellersen, and G. Troster, "Eye Movement Analysis for Activity Recognition Using Electrooculography", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 33, No. 4, pp. 741-753, 2011.
- [9] R. Barea, L. Boquete, M. Mazo, and E. Lopez, "System for Assisted Mobility Using Eye Movements based on Electrooculography", *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, Vol.10, No.4, pp.209-218, 2002.
- [10] S. Hoppe and A. Bulling, "End-to-end Eye Movement Detection Using Convolutional Neural Networks", *ArXiv:1609.02452 e-prints*, 2016.
- [11] M. Singh, P. Jain, and S. Chopra, "Eye Movement Detection for Wheelchair Control Application", In: *Proc. of the International Conf. on Electrical, Electronics, Signals, Communication, Optimization*, Visakhapatnam, India, pp.4-8, 2015.
- [12] M. U. Ghani, S. Chaudhry, M. Sohail, and M. N. Geelani, "GazePointer: A Real Time Mouse Pointer Control Implementation based on Eye Gaze Tracking", In: *Proc. of the 16th International Multi Topic Conference (INMIC)*, Lahore, Pakistan, pp.154-159, 2013.
- [13] F. Utaminingrum, R. Primaswara, and Y. A. Sari, "Image Processing for Rapidly Eye Detection based on Robust Haar Sliding Window", *International Journal of Electrical and Computer Engineering*, Vol.7, No.2, pp.823-830, 2017.
- [14] S. D. Ganesan and M. A. R. Mohammed, "A Hybrid Face Image Contrast Enhancement Technique for Improved Face Recognition Accuracy", *International Journal of Intelligent*

- Engineering and Systems*, Vol.10, No.6, pp.106-115, 2017.
- [15] P. Viola and M. Jones, "Rapid Object Detection Using a Boosted Cascade of Simple Features", In: *Proc. of the IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, Kauai, USA, pp.511-518, 2001.
- [16] A. Yilmaz and O. Javed, "Object Tracking: A Survey", *Acm Computing Surveys (CSUR)*, Vol.38, No.4, pp.1-45, 2006.
- [17] M. Kristan, R. Pflugfelder, A. Leonardis, J. Matas, L. Cehovin, G. Nebehay, T. Vojir, G. Fernandez, A. Lukezic, A. Dimitriev, A. Petrosino, A. Saffari, B. Li, B. Han, C. Heng, C. Garcia, D. Pangercic, G. Haeger, F. S. Khan, F. Oven, H. Possegger, H. Bischof, H. Nam, J. Zhu, J. Li, J. Y. Choi, J.-W. Choi, J. F. Henriques, J. Van De Weijer, J. Batista, K. Lebeda, K. Oefjaell, K. M. Yi, L. Qin, L. Wen, M. E. Maresca, M. Danelljan, M. Felsberg, M.-M. Cheng, P. Torr, Q. Huang, R. Bowden, S. Hare, S. Y. Lim, S. Hong, S. Liao, S. Hadfield, S. Z. Li, S. Duffner, S. Golodetz, T. Mauthner, V. Vineet, W. Lin, Y. Li, Y. Qi, Z. Lei, and Z. Niu, "The Visual Object Tracking VOT2014 Challenge Results", In: *Proc. of Computer Vision-Eccv Workshops*, pp.191-217, 2015.
- [18] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "High-Speed Tracking with Kernelized Correlation Filters", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol.37, No.3, pp. 583-596, 2014.
- [19] T. G. Dietterich, R. H. Lathrop, and L. T. Perez, "Solving The Multiple Instance Problem with Axis Parallel Rectangles", *Artificial Intelligence*, Vol.89, No.1-2, pp. 31-71, 1997.
- [20] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "Exploiting the Circulant Structure of Tracking-by-Detection with Kernels", In: *Proc. of the 12th European Conference on Computer Vision*, Florence, Italy, pp.702-715, 2012.
- [21] A. Yuille, L. Vincent, D. Geiger, and P. Hall, "Statistical Morphology and Bayesian Reconstruction", *Journal of Mathematical Imaging and Vision*, Vol.1, No.3, pp.223-238, 1992.
- [22] F. Utaminingrum, K. Uchimura, and G. Koutaki, "Mixed Gaussian and Impulse Noise Removal based on Kernel Observation and Edge Direction", *International Journal of Innovative Computing Information and Control*, Vol.11, No.5, pp.1509-1523, 2015.
- [23] N. Kumar, H. Shukla, and R. Tripathi, "Image Restoration in Noisy Free Images Using Fuzzy Based Median Filtering and Adaptive Particle Swarm Optimization - Richardson-Lucy Algorithm", *International Journal of Intelligent Engineering and Systems*, Vol.10, No.4, pp.50-59, 2017.