Eye Detection Based-on Color and Shape Features

Aryuanto Soetedjo
Department of Electrical Engineering
National Institute of Technology (ITN)
Malang, Indonesia

Abstract—**This paper presents an eye detection technique based-on color and shape features. The approach consists of three steps: a rough eye localization using projection technique, a white color thresholding to extract white sclera, and an ellipse fitting to fit the ellipse shape of eye. The proposed white color thresholding utilizes the normalized RGB chromaticty diagram, where white color objects are bounded by a small circle on the diagram. Experimental result shows that the proposed technique achieves a high eye detection rate of 91%.**

Keywords-eye detection; color thresholding; ellipse fitting.

I. INTRODUCTION

Nowadays researches on facial features detection and recognition attract attention for researchers, due to their significant contribution in the Human Computer Interface (HCI) applications. One of the interesting topics is an eye detection technique which is employed for activating mobile phone [1]; phoning, reading and browsing [2]; monitoring driver fatigue [3]; and improving human-computer dialogue [4].

Basically, eye detection technique is divided into three methods [5]: a) Shape-based, b) Appearance-based; c) Hybrid shape and appearance-based. In shape-based method, the eye shape such as eye edges, pupil, and eye corners are used as the features. In appearance-based method, the photometric appearance of eye is used for detection. Hybrid method combines both shape and appearance methods to exploit their advantages.

In [6], the eye is detected by extracting the pixels darker than surrounding and grouping them according to their geometric centers. The best eyeball is selected by applying a set of geometric constraints. In [7], an ellipse fitting method is employed to detect the potential eye region. The antropological characteristics of human eye, such as the width and height ratio of the eye and the orientation of the major axis are used to verify the detected eye. The horizontal projection is employed in [8] to find the eyebrows. Further the region below is extended to locate the pupil. The generalized projection function is proposed in [9]. It combines the integral projection, variance projection, and hybrid projection functions. In the method, pupil is located by applying the vertical and horizontal projections.

In [10], a cascaded classsifier trained by AdaBoost algorithm is employed to detect multi-view face and eye. To detect eye, they proposed four-step detection, i.e. eye's candicate area restriction, eye's candidates detection using an eye classifier, eye pair candidates using an eye pair classifier, and an eye pair decision based on features extracted from previous steps. Instead of rectangle feature which is used in [10], a pixel-pattern-based texture feature is used in [11]. They employed Adaboost and Support Vector Machine classifier to classify the eye and non-eye image patches. In [12], wavelet and Neural Network is used to classify an eye or non-eye region.

In [13] and [14], shape and appearance are combined to generate an eye model which is invariant to scale and rotation changes. In [13], a color model is employed for coarse-scale eye tracking and grayscale appearance is used for precise localization.

Previous works as decribed above mostly utilize intensity image for detecting eye. Color is usually employed for skin color modeling [13]. In eye detection, the Infrared light (IR) is the dominant use of color [15], [16]. This method relies on the fact that when IR light falls on the eye, it produces the bright pupil effect. However, it has a drawback that a special geometric arrangement of IR light emitting diodes (LEDs) should be prepared for producing the desired bright pupil effect.

In this paper, a color thresholding technique is proposed to detect eye. It takes an advantage of the white color of eye's sclera [17]. Further the shape feature, i.e. ellipse shape, is utilized to find the precise location of eye. The method assumes that a face region has been obtained by a face detection technique. The proposed eye detection consists of three steps: a) Once a face is detected, the rough location of eye is localized by horizontal projection of the gradient image; b) White color thresholding is employed to extract eye's sclera; and c) An ellipse detection technique is applied to detect the precise eye location.

The rest of paper is organized as follows. Proposed technique is described in Section 2. Section 3 discusses the experimental results. Finally conclusion is covered in Section 4.

II. PROPOSED TECHNIQUE

A. Overview

Fig. 1 illustrates the overview of proposed system. Prior to eye detection, the face detection is applied to locate a face. It employs the popular Viola-Jones face detector [18]. Since the Viola-Jones detector requires a grayscale image as the input, it needs to convert the RGB color image to the grayscale image as shown in the figure. After face is detected, the eye detection algorithm starts by localizing the rough eye location using horizontal projection technique of the gradient image. By analyzing peaks of the horizontal projection, the rough location of eye is obtained. Then the original color image is cropped according to the rough eye location. In the cropped image, white color of eye's sclera appears as the significant clue for detecting eye. Thus white color thresholding is applied to detect the sclera. It creates a blob image (binary image) contains the white and non-white objects. Since the shape of eye (sclera) is ellipse then an ellipse detector is applied to detect the boundary of eye's sclera.

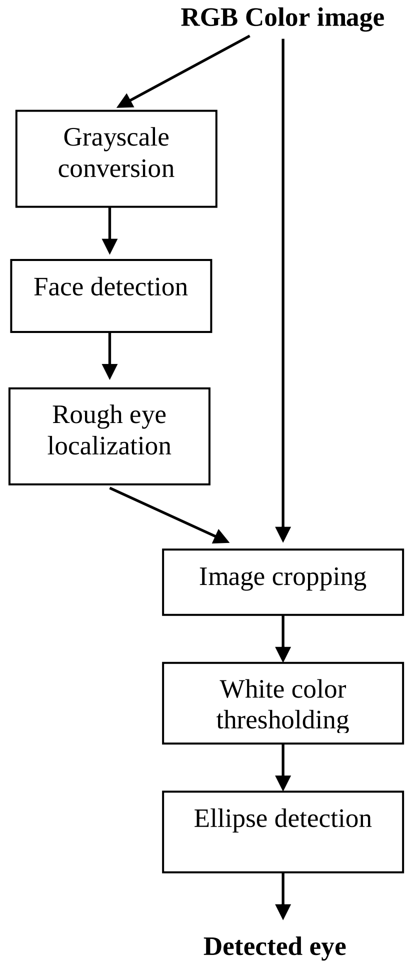
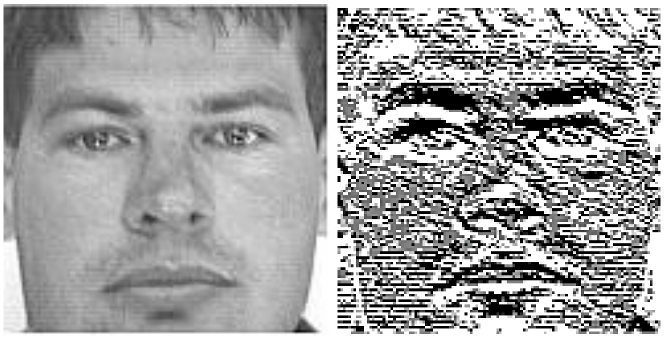


Figure 1. Overview of proposed eye detection.

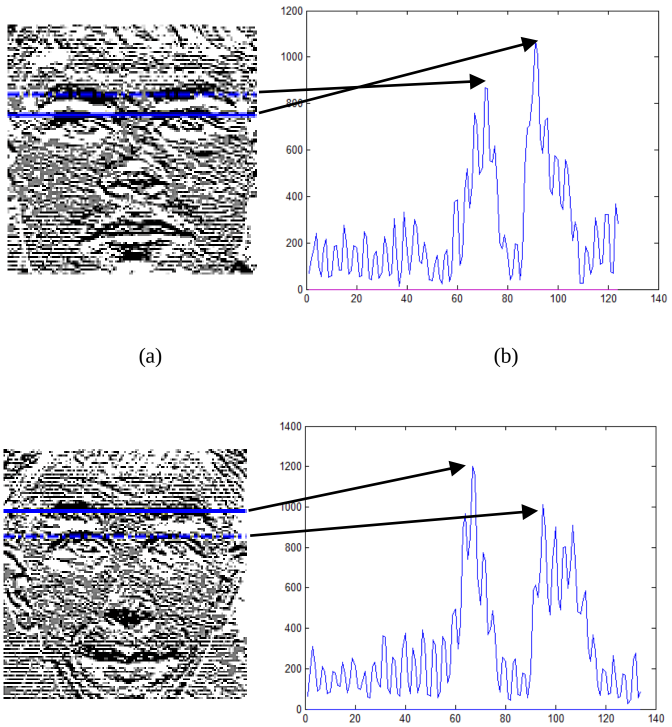
B. Rough Eye Localization

It is observed from the face image that vertical gradient of the image is dominated with horizontal edges, due to the appearance of the facial's components such as eyebrow, eye-lid, and lip as shown in Fig. 2(b). Therefore, if the gradient image shown in the figure is projected onto vertical axis (horizontal projection), then there will exist peaks on the projection represent the location of eyebrow, eyelid, and lip. Since the aim here is to localize eye, thus the projection is applied to the upper-half face only. Fig. 3 shows the gradient images with their horizontal projections of the upper-half part. Fig. 3(b) shows that maximum peak of the projection denotes the location of the eyelid. However this situation does not always hold. In Fig. 3(d), maximum peak of the projection denotes the location of eyebrow. Fortunately, from the experiments, the location of eyelid could be found as follows. It first finds the two maximum peaks of the projection. From these two maximum peaks, find the one which the vertical position is lower than the other one, and assign it as the vertical position of eyelid.



(a) (b)

Figure 2. (a). Grayscale image; (b) Vertical gradient image.



(a) (b)

(c) (d)

Figure 3. (a), (c) Gradient image; (b), (d) Horizontal projections of the image (a) and (b) respectively.

Once the vertical position (*vp*) is found, the upper boundary (*vu*) and lower boundary (*vl*) of eye are defined as

$$vu = vp - (0.1 \times hface) \quad (1)$$

$$vl = vp + (0.1 \times hface) \quad (2)$$

where *hface* is the height of face image. The boundary of left and right eyes are defined as

$$Left\_x = 0.2 \times wface \tag{3}$$

$$Left\_y = vu \tag{4}$$

$$Left\_w = 0.2 \times wface \tag{5}$$

$$Left\_h = vl - vu \tag{6}$$

$$Right\_x = 0.6 \times wface \tag{7}$$

$$Right\_y = vu \tag{8}$$

$$Right\_w = 0.2 \times wface \tag{9}$$

$$Right\_h = vl - vu \tag{10}$$

where *wface* is the width of face image, *Left*(*Right*)*_x*, *Left*(*Right*)*_y*, *Left*(*Right*)*_w*, *Left*(*Right*)*_h* are the *x*-coordinate of left-top corner, *y*-coordinate of left-top corner, width and height of bounding box of the left(right) eye respectively.

C. White Color Thresholding

Color thresholding is usually used for skin color segmentation [13], [19]. Previous work [19] shows that the normalized RGB color space is effectively used for skin color segmentation. Skin region is defined on the normalized RGB chromaticity diagram as shown in Fig. 4. In the figure, skin region is bounded by five lines i.e.: a) *line g=2,* b) *line g=r-0.4*, c) *line g=0.4*, d) *line g=-r+0.6*, e) *line-c*. A small circle denoted as *line-c* is used to exclude white pixels. Contrary, in the proposed eye detection, the aim is to extract the white color of eye's sclera. Therefore the white color thresholding could be considered as extracting the pixels which the *r* and *g* components are inside on the small circle. The proposed method is described in the following.

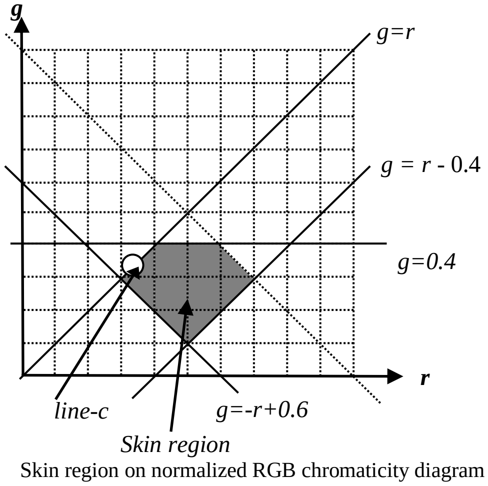


Figure 4. Skin region on normalized RGB chromaticity diagram [19].

Let the red (*r*), green (*g*) and blue (*b*) components of the normalized RGB color are expressed as:

$$r = \frac{R}{R+G+B} \tag{11}$$

$$g = \frac{G}{R+G+B} \tag{12}$$

$$b = \frac{B}{R+G+B} \tag{13}$$

The binary image of extracted white pixels is denoted as *Ithr* and obtained by the following equations:

$$Ir(x,y) = (r(x,y) - 0.33)^{2} \tag{14}$$

$$Ig(x,y) = (g(x,y) - 0.33)^{2} \tag{15}$$

$$Ic(x,y) = Ir(x,y) + Ig(x,y) - 0.0009 \tag{16}$$

$$Ithr(x,y) = \max\left(sign\left(Ic(x,y)\right), 0\right) \tag{17}$$

where

R is red component of *RGB* color

G is green component of *RGB* color

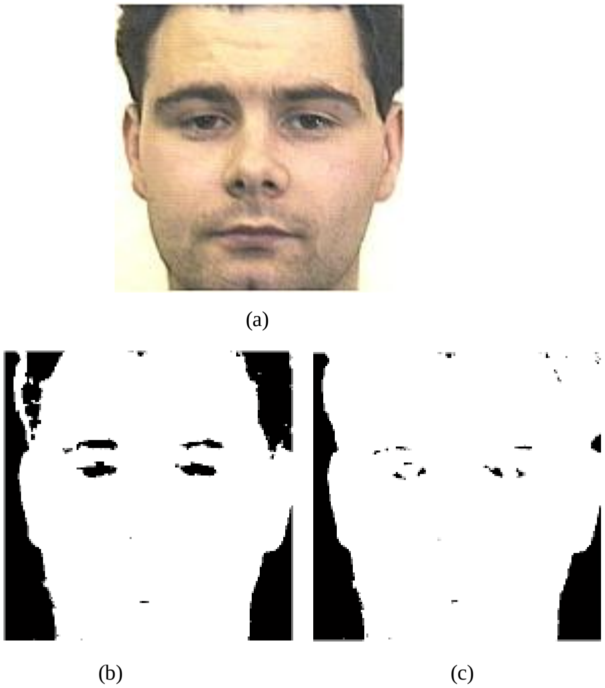
B is blue component of *RGB* color

I(*x*,*y*) is the intensity value at pixel (*x*,*y*)

sign(*a*) is equal 1 if *a>0*, 0 if *a=0*, -1 if *a<0*

max(*a*,*b*) is the maximum value of *a* and *b*

Fig. 5 shows the thresholded image obtained by above equations. Using Eqs. (14)-(17), the binary image (*Ithr*) is obtained as the shown in Fig. 5(b), where the black color represents the extracted objects, in this case white objects, while white color represents non-extracting object. From the figure, it could be seen that the method also extracts non-white objects outside the face. It is confirmed by [19] and could be observed from Fig. 4 that the above formulas will not extract skin color. Thus the skin-face region shown in Fig. 5(b) is in white color (not extracted). Since the white color thresholding is applied on rough eye region where only eye's sclera is in white color, the thresholding method will work appropriately.



(a)

(b)

(c)

Figure 5. (a) Color image, (b)Thresholded image using Eqs. (14)-(17), (c) Thresholded image using Eqs. (14)-(18)

The normalized RGB color has drawback that *r* and *g* component of the dark and bright color may have the same value such as addressed in [19]. Suppose a dark pixel with the value of *R*=0.1, *G*=0.1, and *B*=0.1. The chromaticity values of this pixel are *r*=0.33 and *g*=0.33. In other hand, a bright pixel with the value of *R*=0.9, *G*=0.9, and *B*=0.9 will have the value of *r*=0.33 and *g* =0.33.

Therefore those two pixels will be considered as the same color in the chromaticity diagram. From Fig. 5(b), it is observed that the black color of the hair is also extracted. To overcome the problem, the following rule is introduced. The pixels extracted by Eqs. (14)-(17) are assigned as the white pixels if the following formula is satified.

$$\frac{R(x,y)+G(x,y)+B(x,y)}{3} > 0.4 \qquad (18)$$

Using Eq. (18) the black hair is not extracted as shown in Fig. 5(c).

D. Ellipse Detection

The proposed white color thresholding extracts the white color of eye, thus the extracted blob image does not make solid ellipses, but they contain holes (i.e. pupils) as shown in Fig. 6. To find the precise boundary of the eye, an ellipse detection technique is employed. The ellipse detection works with edge images, thus it needs to detect edge pixels of the blob. A Sobel edge detector is employed to find the edge pixels. After edge pixels are extracted, ellipse fitting method [20] is applied to detect the ellipses.

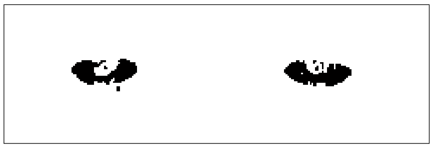


Figure 6. Blob image of extracted eye's sclera

The ellipse fitting method [20] uses the least square criterion to fit an ellipse from edge pixels. Let an ellipse is expressed as

$$a_1x^2 + a_2xy + a_3y^2 + a_4x + a_5y + f = 0 \qquad (19)$$

The five coefficients a_1, a_2, a_3, a_4, a_5 are estimated by the least square estimator from the given edge pixels (points) $(x_1,y_1),...,(x_N,y_N)$ by minimizing the squares sum of an error between the edge pixels and the ellipse using the following formula

$$err = \sum_{i=1}^{N}(a_1x_i^2 + a_2x_i + a_3y_i^2 + a_4x_i + a_5y_i + f)^2 \qquad (20)$$

The estimated five coefficents are then converted to the parameters of ellipse, i.e.: orientation, center coordinate, semi-major axis, and semi minor-axis.

Fig. 7 shows the detected ellipses on the left and right eyes superimposed on the blob image. From the figure, it is shown that the method is able to detect the boundary of eye. It is worth noting that blob image contains hole inside it, thus the edge pixels contain the outer and inner part of the blob. However the ellipse fitting method fits the outer ellipse appropriately.

Figure 7. Detected ellipse on the left and right eyes.

III. Experimental Results

To verify the proposed eye detection method, the algorithm is implemented using MATLAB and tested on two hundred face images taken from [21]. The dataset contains frontal face images with varying lighting.

In the experiments, three approaches are evaluated: a) Eye detection based-on Haarcascade Classifier [18] (**HC**); b) Proposed rough eye localization (**PL**); c) Proposed eye detection (**PD**). It is noted here that eye detection based-on Haarcascade Classifier detects the bounding box of eye, but the proposed eye detection detect the precise location of eye (eye's sclera) acoording to the shape. Therefore these two methods could not be compared directly. In addition to the proposed eye detection, the proposed rough eye locatization is evaluated here for fair comparison to the eye detection based-on Haarcascade Classifier.

The true detection rate and false detection rate are calculated for comparison. The true detection rate is defined as the total number of detected eye's pair divided by the total number of tested images, while the false detection rate is defined as the total number of detected non-eye objects divided by the total number of tested images. By this definition, if only one eye (not a pair of eye) is detected, then it is considered as not detected. Table I shows the comparison results.

The true detection rate of **HC** is the highest, but the false detection is also high. It is a typical result obtained by the Haarcascade classifier, i.e. a high true detection rate, but a false detection occurs. The true detection rate of **PL** and **PD** are lower compared to the **HC**, but they have no false detection. From the observation, the lower true detection of **PL** is caused by the misdetection of eye due to the position of left and right eyes which are not in a horizontal line. This result relates to the approach employed in the proposed rough eye localization which assumes that the position of two eyes should be in a horizontal line as described in Section B.

TABLE I. COMPARISON RESULTS

No	Method	True Detection	False Detection
1.	Eye detection based-on Haarcascade classifier (**HC**)	96.5%	43%
2.	Proposed rough eye localization (**PL**)	93.5%	0%
3.	Proposed eye detection (**PD**)	91%	0%

From the table, there is a deviation between **PL** and **PD**. It implies that the some eye images obtained by the rough localization technique could not be detected by white color thresholding and the ellipse detection techniques properly. From the observation, it is caused by two cases : a) The white color thresholding fails to extract eye pixels, thus there are no ellipse detected; b) The white color thresholding extract the spurious pixels which yield the wrong ellipses. The high true detection rate of **PL** is comparable to the one of **HC**. It shows that the proposed white color thresholding works effectively. The advantages of the proposed method compared to **HC** are: a) There is no false detection; and b) The eye is detected precisely according to the eye's shape.

Fig. 8 illustrates some of the detection results, where the figures in the left, center, and right columns represent the results of **HC**, **PL**, and **PD** respectively. In the first and third rows, all three methods detect or localize the eye properly. But the **PD** localizes eye precisely, in the sense that it detect the ellipse boundary of eye, while the detected boundaries of **HC** and **PL** are in rectangle boxes. From the third row, it is shown that the detected eyes obtained by the **HC** are not in a horizontal line due the skew position of the left and right eyes. However the detected boundaries of left and right eyes are always in the horizontal line. In the second and fourth rows, the non-eye objects are detected by **HC** (false detection), but not detected by **PL** and **PD**. In the fourth row, there is an eye glasses in the image. The **HC** and **PL** locate the eyes properly, but **PD** could not detect eye properly, i.e. the detected ellipses do not fit the eye properly. As shown in the figure, the glasses create the white shadows which result the wrong ellipse detection.

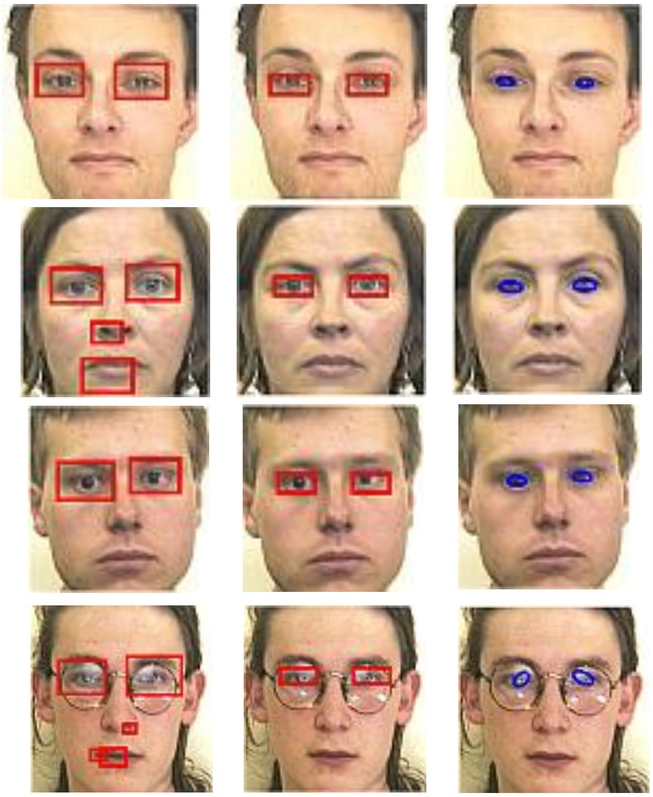


Figure 8. Eye localization and detection results.

IV. CONCLUSION

In this paper, an eye detection technique is presented. It first localizes the rough position of eye using projection technique. Then a white color thresholding based-on the normalized RGB chromaticity diagram is employed. It extracts the white color of eye's sclera. Finally the boundary of eye is found by an ellipse fitting method. The method shows a high performance in rough eye localization of 93.5 % and the true eye detection rate of 91% is achieved.

In future, the performance of proposed method will be improved and will be extended for detecting the driver's fatigue by measuring the degree of openess of the eye. Further the real time implementation using video camera will be conducted.

REFERENCES

[1] E. Miluzzo, T. Wang, and A.T. Campbell, "EyePhone: activating mobile phones with your eyes," Worshop on Networking, Systems, Applications on Mobile Handhelds (MobiHeld). ACM, 2010.

[2] K. Arai and R. Mardiyanto, "Eye-based human computer interaction allowing phoning, reading e-Book/e-Comic/e-Learning, Internet browsing, and TV information extraction," International Journal of Advanced Computer Science and Application, Vol. 2, No. 12, pp. 26-32, 2011.

[3] Q. Wang, "Eye location based on structure feature for driver fatigue monitoring," World Academy of Science, Engineering and Technology, Vol. 72, pp. 368-372, 2010.

[4] J. Machrouh, F. Panaget, P. Bretier, and C. Garcia, "Face and eyes detection to improve natural human-computer dialogue," Proceedings of the Second IEEE-EURASIP International Symposium on Control, Communication, and Signal Processing, Morocco, 2006.

[5] D.W. Hansen and Q. Ji, "In the eye of the beholder:a survey of models for eyes and gaze," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 32, No. 3, pp. 478-500, 2010.

[6] J. Ren and X. Jiang, "Fast eye localization based on pixel differences," Proceedings of the 16th International Conference on Image Processing, Cairo, pp. 2733-2736, 2009.

[7] D. Sidibe, P. Montesinos, and S. Janaqi, " A simple and efficient eye detection method in color images," Proceedings of International Conference Image and Vision Computing New Zealand, Great Barrier Island, 2006.

[8] M.A. Kashani, M.M. arani, M.R. Ramezanpour, "Eye detection and tracking in image with complex background," Journal of Emerging Trends in Computing and Information Sciences, Vol. 2, No. 4, pp. 182-186, 2011.

[9] Z.H. Zhou and X. Geng, "Projection functions for eye detection," Pattern Recognition, Vol. 37, No. 5, pp. 1049-1056, 2004.

[10] J.B. Kim, S.C. Kee, and J.Y. Kim, "Fast detection of multi-view face and eye based on cascaded classifier," Proceedings of IAPR Conference on Machine Vision Applications, Tsukuba, 2005.

[11] H. Lu, W. Zhang, and D. Yang, "Eye detection based on rectangle features and pixel-pattern-based texture features," Proceedings of International Symposium on Intelligent Signal Processing and Communication System, Xiamen, pp. 265-268, 2007.

[12] M. Motwani, R. Motwani, and F.C. Harris, "Eye detection using wavelets and ANN," Proceedings of Global Signal Processing Conferences and Expos for Industry, California, pp. 27-30, 2004.

[13] D.W. hansen, J.P. Hansen, M. Nielsen, A.S. Johansen, and M.B. Stegman, "Eye typing using markov and active appearance models," Proceedings of IEEE Workshop Applications on Computer Vision, pp. 132-36, 2003.

[14] T. Ishikawa, S. Baker, I. Matthews, and T. Kanade, "Passive driver gaze tracking with active appearance models," Proceedings of 11th World Congress Intelligent Transportatiom Systems, Nagoya, 2004.

[15] Q. Ji and X. Yang, "Real-time eye, gaze, and face pose tracking for monitoring driver vigilance," Real-Time Imaging, Vol. 8, pp. 357-377, 2002.

[16] D.W. Hansen, "Using colors for eye tracking," Color Image Processing: Method and Applications, pp. 309-327, CRC Press, 2006.

[17] Aryuanto and F.Y. Limpraptono, “A vision-based system for monitoring driver fatigue,” Proceedings of TEKNOIN, Yogyakarta, 2009.

[18] P. Viola, M. Jones, “Robust real-time face detection,” International Journal of Computer Vision, Vol. 57, No. 2, 2004.

[19] A. Soetedjo and K. Yamada, “Skin color segmentation using coarse-to-fine region on normalized RGB chromaticity diagram for face detection,” IEICE Transactions on Information and System, Vol. E91-D, No. 10, pp. 2493 -2502, 2008.

[20] http://www.mathworks.com/matlabcentral/fileexchange/3215-fitellipse

[21] http://pics.psych.stir.ac.uk/