Fatigue Driver Detection System Using a Combination of Blinking Rate and Driving Inactivity

Wasan Tansakul and Poj Tangamchit
Control System and Instrumentation Engineering Department
King Mongkut's University of Technology Thonburi, Bangkok, Thailand
Email: wasan.tans@gmail.com, ipojchit@kmutt.ac.th

***Abstract*—We implemented a fatigue driver detection system using a combination of driver's state and driving behavior indicators. For driver's state, the system monitored the eyes' blinking rate and the blinking duration. Fatigue drivers have these values higher than normal levels. We used a camera with machine vision techniques to locate and observe driver's blinking behavior. Harr's cascade classifier was used to first locate the eye's area, and once found, a template matching was used to track the eye for faster processing. For driving behavior, we acquired the vehicle's state from inertial measurement unit (IMU) and gas pedal sensors. The principle component analysis (PCA) was used to select the components that have high variance. The variance values were used to differentiate fatigue drivers, which are assumed to have higher driving activities, from normal drivers.**

***Index Terms*—fatigue driving, blink detection, driving behavior**

I. Introduction

A study of car accidents indicates that almost 20% of the cause of crash results from non-readiness of drivers such as distraction, fatigue and lack of sleep. When drivers are asleep, the accidents tend to more severe because the drivers cannot react and maneuver the vehicle to avoid crashes. A prompt detection of sleepy drivers is therefore very useful. There are mainly two types of indicators used to detect fatigue drivers: driver's state and driving behavior. Driver's state is a direct indicator to driver's fatigue. However, it can be difficult to measure effectively, because it involves human factors which can be unpredictable. For example, a truck driver who wants to avoid being caught that he is asleep in his duty can easily take away or fake the sensors that measure his state. The driving behavior, on the other hand, can be measured with sensors installed in a vehicle, which cannot easily be tampered with. This paper combines both driver's state and driving behavior detection to get the benefits of both methods. This will make the detection system more practical than using only one indicator. For driver's state indicator, we use a clue from the driver's eyes. Eye behavior contains a useful clue for drowsiness. There are two approaches for detecting eye clues: Active and passive approaches. The active approach uses infrared light shining toward the eyes and detecting reflection. The passive approach relies on ambient light and detects eyes' behavior. The drawback for the active light is that the light source, although infrared, has to be strong so that its reflection is clearly visible. This will create eye strain when using it on the driver's eyes for a long period of time. Our work, on the other hand, chose the passive approach, which use ambient light or gentle light source. We use eye detection and tracking algorithm to detect blinking rate and duration. Fatigue drivers will have high blinking rate and longer duration than normal drivers. For driving behavior indicator, we install a 9-DOF inertial measurement unit (IMU) together with a gas pedal sensor. These sensors are used to measure the level of driving activity. The assumption we used is that fatigue drivers will have low level of activity, which will be reflected by the smoothness of sensor values. Since, there are many features from the sensors, we implement the principle component analysis (PCA) to reduce the dimension of data. Then, we measure the fluctuation of data by using the standard deviation to differentiate between normal drivers and fatigue drivers.

II. Related Work

Possible techniques for detecting drowsiness in drivers can be broadly divided into four major categories:

- Methods based on driver's current state, relating to the eye and eyelid movements [1]-[3].
- Methods based on driver performance and driving behavior [4]-[6].
- Methods based on physiological signals [7].
- Methods based on combination of the multiple parameters [8].

There has been lots of literature on detection of fatigue effects and the driver's current state specifically focused on changes and movements in the eye. This includes assessing changes in the driver's direction of gaze, blinking rate and actual eye closure. Generally, eyes detection consists of two steps: Locating face to extract eye regions and eye detection from eye's windows.

Manuscript received September 21, 2014; revised December 17, 2014.

doi: 10.12720/joace.4.1.33-39

Several researches use Haar-Like feature and AdaBoost algorithm for detecting face and eyes and use PERCLOS to evaluate driving fatigue. PERCLOS (Percent Eye Closure), a video based method that measures eye closure, is a reliable and valid determination of a driver's alertness level. PERCLOS is the proportion of total time that the driver's eyelids are closed 80% or more and reflects slow eyelid closures rather than blinks.

For example, W. Qing *et al.* [1] and Y. Kurylyak *et al.* [3] detected face and eyes region using Haar-Like feature and AdaBoost algorithm. W. Qing *et al.* use an improved template matching method to detect eye states and selected PERCLOS to evaluate driving fatigue, while Y. Kurylyak *et al.* used the frames differencing in combination with a thresholding method to detect the eyes closure and opening. They used the transition of eyes state to detect eyes' blink. B. Alshaqaqi *et al.* [2] designed an Advanced Driver Assistance System (ADAS) to reduce accidents due to drivers' fatigue. In this system, they proposed an algorithm to locate, track, and analyze the driver face and eyes to measure PERCLOS. Then they performed scientifically supported measure of drowsiness associated with slow eye closure.

Other measurements that are capable of measuring drivers' performance and physiological state are also proposed. Examples of these were: road boundary and tracking, fatigue driving detection system design based on driving behavior, EEG recording. A detail presentation is given below.

W. S. Wijesoma *et al.* [4] developed a road boundary and tracking system by using ladar sensing. They proposed a method based on the extended Kalman filtering filtering for fast detection and tracking of road curbs using successive range/bearing readings obtained from a scanning two-dimensional ladar measurement system.

W. Hailin *et al.* [5] and T. C. Chieh *et al.* [6] implemented a fatigue driving detection system design based on driving behaviors. W. Hailin et al. detected the changing signals of accelerate, brake, shift and steering to analyze driver's states. T.C. Chieh *et al.* detected driving fatigue by monitoring the driver's grip force on the steering wheel alone. The data was obtained by using two resistive force sensors attached to the steering wheel and connected to a computer and a data acquisition module.

For an EEG reading, L. Ming-ai *et al.* [7] studied the characteristic of EEG signal in a drowsy driving state by using a method based on the power spectrum analysis and FastICA algorithm in order to determine the fatigue degree. It can differentiate between two states: sober and drowsy. The multichannel signals were analyzed with the FastICA algorithm, and the power spectral densities were calculated after FFT and then the fatigue index F was determined.

According to the analysis above, the approaches that combine driver state and driver performance will improve the sensibility and reliability in fatigue detection. J. Wang *et al.* [8] developed a real-time driving danger level prediction system by using a level prediction system that uses multiple sensor inputs. They used a statistical model to predict the driving risk. They used three types of features: the vehicle dynamic parameter, the driver's physiological data, and the driver's behavior. In this system, they used hidden Markov model, conditional random field and reinforcement learning to model the temporal patterns that lead to safe/dangerous driving state.

III. System Overview

We designed our system to detect fatigue base on two clues: driver's state and driving behavior. Our system consists of two parts i) Fatigue detection from eyes movement, ii) Fatigue detection from driving behavior as shown in Fig. 1.

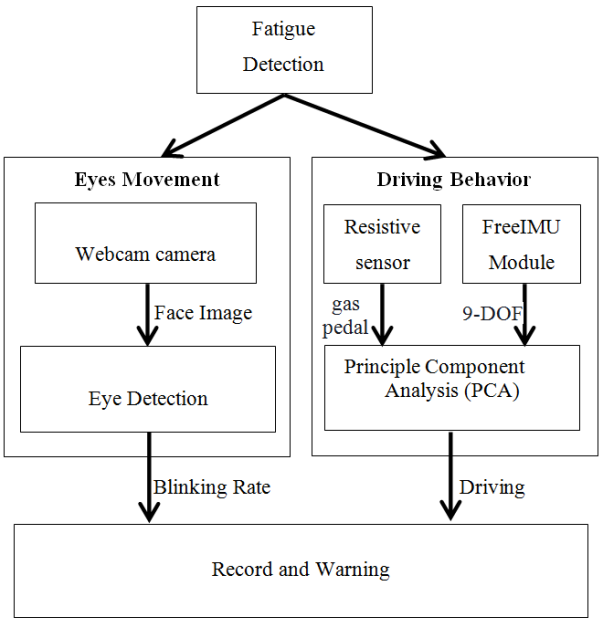

Figure 1. Fatigue detection overview

A. *Fatigue Detection From Eyes Movement*

The proposed blink detection procedure includes the following steps: i) face and eyes detection, ii) eyes tracking, iii) eye closure detection and evaluation of the blink rate by steps as shown in Fig. 2.

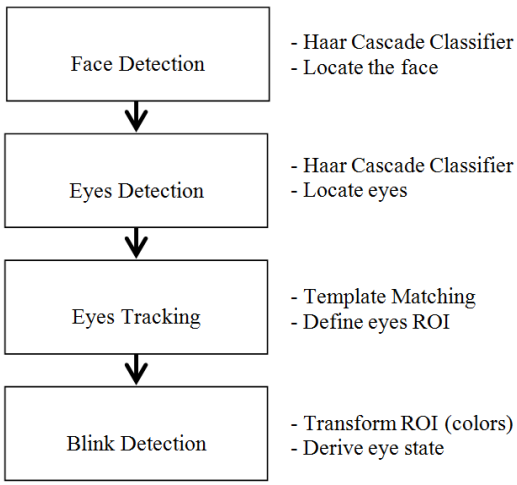

Figure 2. The proposed algorithm

1) Face & eyes detection

The first step in analyzing the blink of a driver is to locate the face and the eyes. We applied Haar Cascade Classifier for face and eyes detection. First, we detected the face to get the face location. Then, within the face region, we search for the eyes and get eyes' location. Fig. 3 illustrates the output of the cascade detector.

Figure 3. Output of the Harr cascade detector. Where green rectangles are the eyes and face location.

The drawback of Harr cascade is that it is computational expensive. It gave slow refresh rate compared to the speed of eyes' blink. Therefore, we applied template matching for eyes tracking. We got eyes template image from eyes location of cascade detector output and then system will track eye in the green rectangles as shown in Fig. 4.

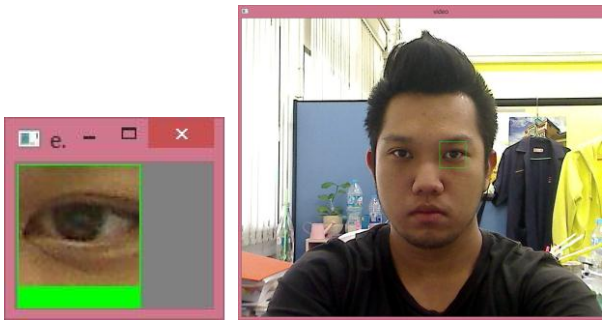

Figure 4. Eye template image and output of template matching.

Finally, we obtained a region of interest (ROI) around the eyes from output of template matching, and derive a measurement value which is used to determine the eye state.

2) Blink detection

We proposed the detection of blinking and the analysis of blink duration in this section. We utilized the color feature to detect eye closure by we will convert color space of eye ROI image from RGB to HSV and split image to V-channel to monitor the value change of lightness in ROI image. When user change state of eye from open to close, that will result in the lightness increases because eye-closed state have the size of black eye is smaller than the eye open state so lightness value in eye-opened state will less than eye-closed state. Fig. 5 illustrates the output image after convert colors space.

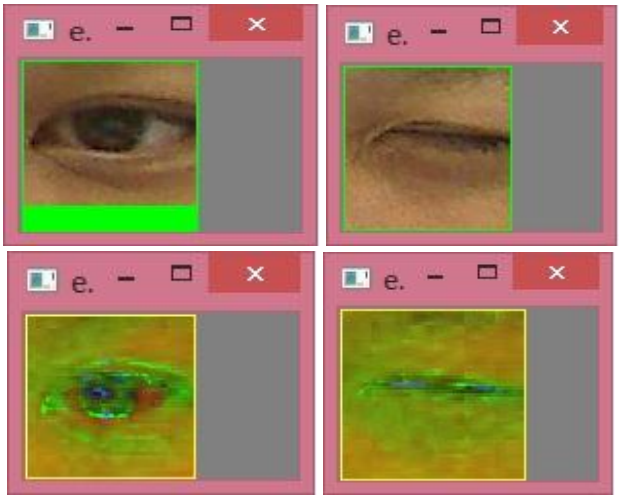

Figure 5. Eyes in different states in RGB (upper row) and the corresponding images transformed to the V-channel of the HSV-Color space (lower row).

After converting color space of eye ROI image from RGB to HSV. We can get the lightness value (V-value in HSV) and plot it on the time axis as in Fig. 6.

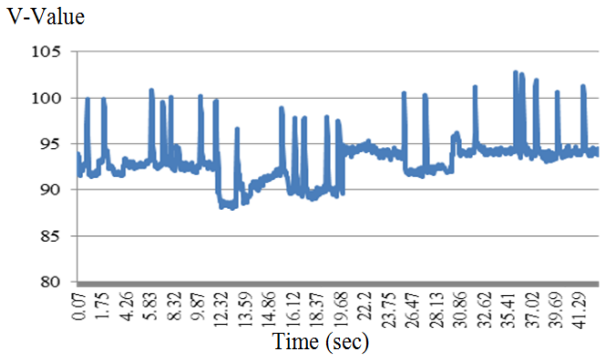

Figure 6. The plot shows V-Value compare with time

From lightness value in Fig. 6, we can detect blink and calculate blink duration by the flowchart in Fig. 7. The objective of the algorithm is to find the sharp slope of V-Value, and define it as the transition points. First, we will find positive slope. If the values of positive slope higher than "Th_close" system will judgment eye state to close and increase time of eye blink duration until the values of negative slope higher than "Th_open" and then system will stop increase time of eye blink duration and judgment eye state to open. Next, the system will count up blink and start find slope of V-Value again.

For the eye state, "$State_{eye}$" is defined by testing the value "V-Value" by a threshold and we can derive by the following (1):

$$State_{eye} = \begin{cases} Closed & if \ \dfrac{dV}{dt} > Th\ Close \\ Open & if \ \dfrac{dV}{dt} > Th\ open \end{cases} \quad (1)$$

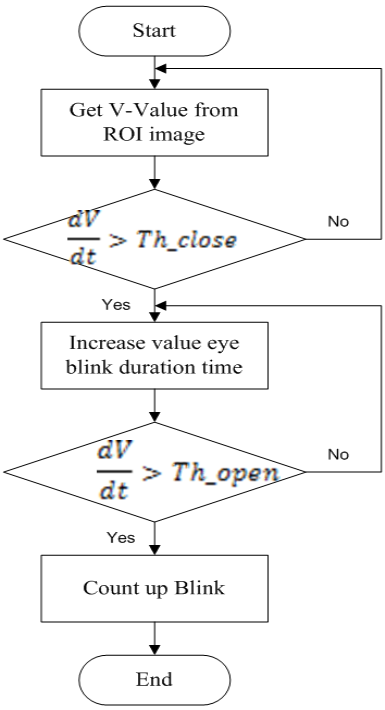

Figure 7. The proposed algorithm to detect blink and calculate blink duration

B. *Fatigue Detection from Driving Behavior*

The proposed driving fatigue detection based on driving behavior procedure includes the following steps: i) data acquisition from sensor, ii) analyze the driver's driving behavior by steps as shown in Fig. 8.

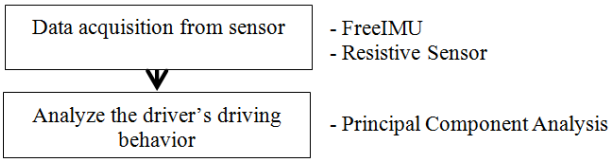

Figure 8. The proposed driving fatigue detection based on driving behavior.

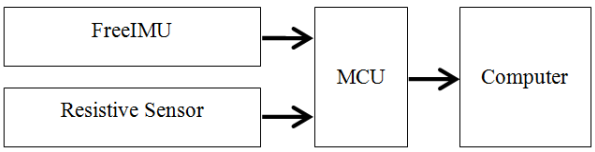

Figure 9. System overview

Fig. 9 shows all of the system hardware that includes the following devices i) FreeIMU module(GY-87), ii) Resistive Sensor, iii) MCU(Freeduino V1.16 Board), iiii) Computer.

1) *Data acquisition from sensor*

The force resistive sensor was mounted on the acceleration pedal. The sensor gives out a voltage value according to the force received. When driver does not apply pedal, the output voltage is 0V. When driver apply a full force to the pedal, the output voltage is 3.3V. The change of voltage reflects the pedal's movement. We use it as a input to construct the model of driving behaviors. Fig. 10 shows the resistive sensor and the position of it on an acceleration pedal.

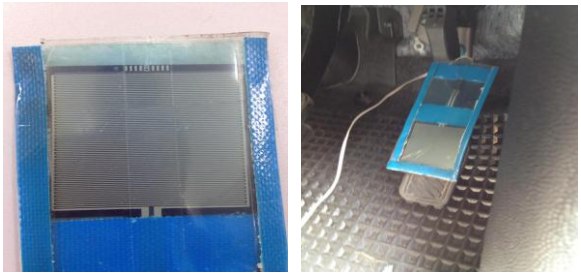

Figure 10. Force resistive sensor and its position on the pedal.

The car movement was measured from the FreeIMU module (GY-87), which is a combined accelerometer, gyroscope, and compass. Its signal includes the following 9 parameters i) 3-axis parameter from Gyro meter, ii) 3-axis parameter from Accelerometer, iii) 3-axis parameters from Magnetic Field. The protocol for sending and receiving data between IMU module and MCU was communicated by SPI protocol. Fig. 11 is shown FreeIMU module and setup position on console. The data from sensor was separated into two set are i) Training data, ii) Test data.

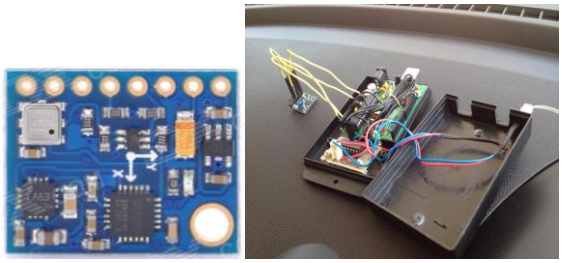

Figure 11. FreeIMU module and setup position on console.

2) *Analyze the driver's driving behavior*

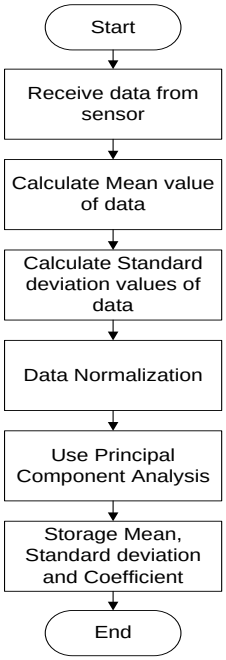

Figure 12. The proposed algorithm to calculate Mean, Standard Deviation and Coefficient values.

According to the data collected from resistive sensor and FreeIMU module (Sampling rate 1ms) we used principal component analysis to simplify the data and construct a fatigue driving identification model to analyze the driver's driving behavior. For analyze the driver's driving behavior includes two steps i) Calculate Mean, Standard Deviation and Coefficient value from training data for using to model the driver statement ,ii) Calculate Variances of Test data by using Mean, Standard Deviation and Coefficient of training data. We can calculate Mean, Standard Deviation and Coefficient value by the following flowchart in Fig. 12.

The fatigue driving identification is showed as in Fig. 13. First, the dataset from resistive sensor and FreeIMU module were normalization by using z-score and then the data were analyzed by Principal Component Analysis to reduce the dimensionality of a data set. Next, the variances were calculated from variable score from PCA (Principal Component Analysis) by using Mean, Standard Deviation and Coefficient value from training data. Finally, the combined variances of five columns were selected. If the combined variance is higher than a threshold, the algorithm will judge the driver state to normal state. But, if it is less than the threshold, the algorithm will judge the driver state to the fatigue state.

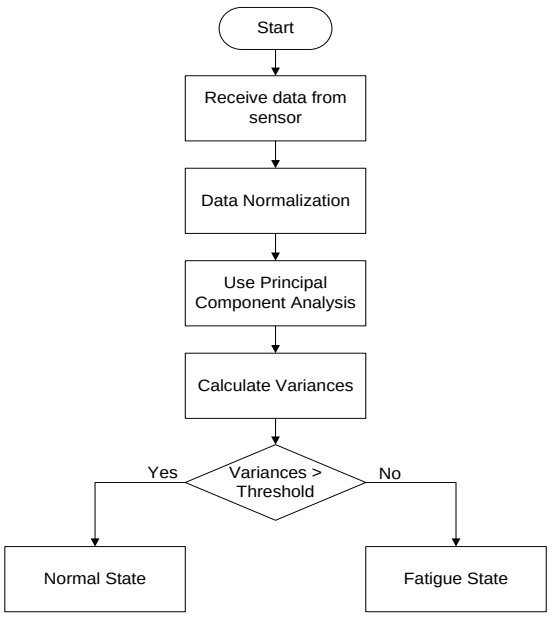

Figure 13. The proposed algorithm to identification driving state.

IV. Experiment and Result

For driving fatigue detection from eyes movement, we implemented all of the algorithms and tested in Microsoft Visual Studio 2010 in Windows 8 working on Computer with AMD A10 CPU and 8 GB RAM. Video was captured from Logitech Webcam Pro9000 and using OpenCV as an image processing and computer vision library.

To validate our system, we made experiments using 10 users at same location to evaluate the performance of the proposed system. First, we set up Logitech Webcam Pro9000 to Computer as shown in Fig. 14. Next, we set the user position is sitting in front of a computer screen as shown in Fig. 15 and then we detected eye blink from user by using our proposed algorithm and use driving simulation video to test and we use time interval to collect eye blink data for 60s. While collecting data we have observers to collect data of eye blink from looking with the eyes for comparing the results from our proposed algorithm and results from observers to calculate the accuracy of algorithm. Fig. 16 illustrates the experiment setup.

Figure 14. Set up logitech webcam Pro9000 to computer

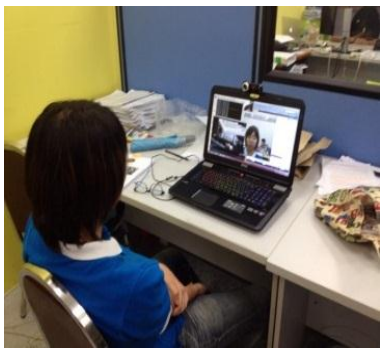

Figure 15. Set up user position

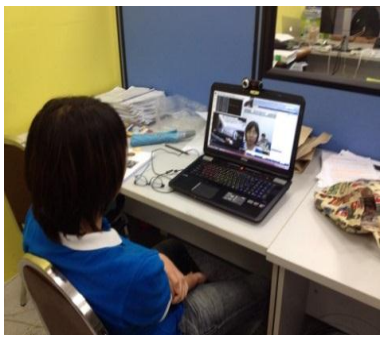

Figure 16. The experiment setup and observers

Fig. 17 shows the time intervals when there was detected closing and opening of eye in a test video for 60s. Such moments are shown as peaks.

The final result of blink detection is shown in Fig. 18 that can be calculated from (1), where high value represents the time interval when the eye was detected closed and low when detected opened. The duration of eye blink can be calculated as well. During the video capturing there were done 21 blinks, one of which between 21s and 25s has a blink lasted 4s.

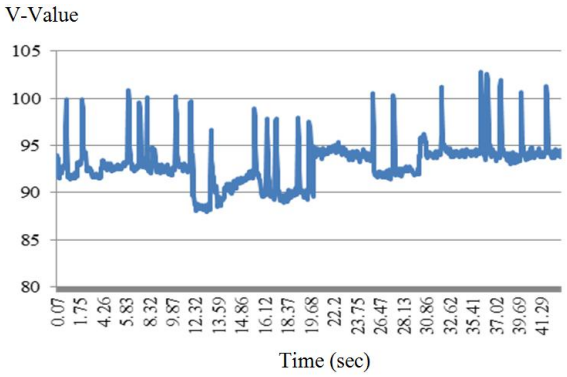

Figure 17. The time intervals when there was detected closing and opening

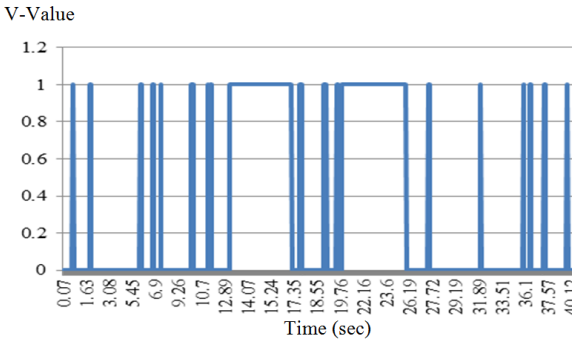

Figure 18. Detected blinks

The result of eye blink detection in the same location and driving simulation video are illustrated in Table I, We show the accuracy rates of our proposed eye blink detection methods.

TABLE I. THE RESULTS OF EYE BLINK DETECTION

User	*Blink Rate from algorithm (Number/minute)*	*Blink Rate from Observer (Number/minute)*	*Accuracy (%)*
1	15	16	93.75
2	12	14	85.72
3	14	17	82.36
4	15	18	83.34
5	19	21	90.48
6	13	14	92.86
7	13	15	86.67
8	10	11	90.91
9	11	13	84.62
10	12	12	100

The algorithm for driving fatigue detection from driving behavior were implemented and tested on Computer with AMD A10 CPU and 8 GB RAM and MCU (Freeduino V1.16 Board) to get data from sensor.

The experiment was performed by setting the resistive sensor to pedal, FreeIMU on console and Computer for getting data to analyze when we drive in real situation. The drivers were asked to acts as normal and sleepy. The driver states were assumed include following states i) Normal state, ii) Fatigue state. Fig. 19 shows the system installation. The final result of the driving behavior is shown in Table II. We can calculate variances from variable scores in PCA, and then the driver state was determined by thresholding the sum of variances.

Figure 19. All of device position

TABLE II. THE RESULTS OF DRIVING BEHAVIOR

Variances (Normal State)	***Variances*** (Fatigue State)
6.761058	1.424458
8.368608	0.889508
4.883198	3.318376
13.5185	4.582522
5.736565	1.942485
3.963947	1.74425
7.718399	6.460666
2.907499	2.500966
11.67473	2.02656
11.40921	2.383842

The driver state was determined by comparing the data from Table II with threshold value. The appropriate threshold value was set to 5, so, we have 3 values in normal state of Table II that value less than 5 are value in row 3, 6 and 8. The reason for this error is because it was the time period while the car was waiting for a traffic light or traffic jams.

V. DISCUSSION AND CONCLUSION

In this paper, we proposed an algorithm to detect driving fatigue, which consist of two parts: driving fatigue detection from eyes movement and driving fatigue detection from driving behavior.

For the fatigue detection from eye blinks, the algorithm detects eye blinks, calculates the blink rate, and the duration time. We performed an experiment using a video-based method. The results showed that our algorithm can work efficiently in nearly real-time, because our algorithm has high frame rate processing of approximately 20 fps. We compared the output from the algorithm with the blinks observed from a human observer. The results provide the average accuracy of approximately 89%.

According to the results, the graph analysis on ten sets of 1-minute samples show that the typical blink of human contains alternating inter-blink periods of shorter and longer durations depend on the driver's state. Although the results show that our method works well for the images taken under controlled environments and changes of illumination in less quantity, some issues are discussed below

1. The algorithm may be failed to detect eye blink when the rotation of the driver's head is over 30 degrees. Therefore, the installation of the camera is suggested to be in the front of the driver to prevent such a problem.
2. When illumination of the face has changed in large quantity, it causes the face and eyes area to become dark. The detection of eye blinks may fail and causes an error in the count of blinks.

For the fatigue detection from driving behavior, the experiment and results showed that our algorithm can effectively classify and analyze the driving behavior. However the results show that our method cannot work well under situations where traffic jam or waiting for traffic light for a long time. For the solution to improve the accuracy of the algorithm can be achieved by increasing the duration of data storage but the disadvantage of this solution is the algorithm will use a long time before it can measure the driver's state. This is the main reason why both methods have to be combined in order to support each other.

ACKNOWLEDGMENTS

This work was supported by the Higher Education Research Promotion and National Research University Project of Thailand, Office of the Higher Education Commission.

REFERENCES

[1] Q. Wu, B. X. Sun, B. Xie, and J. J. Zhao, "A perclos-based driver fatigue recognition application for smart vehicle space," in *Proc. 2010 Third International Symposium on Information Processing (ISIP)*, 2010, pp. 437–441.

[2] B. Alshaqaqi, A. S. Baquhaizel, M. E. Amine UOIS, M. Boumehed, A. Ouamri, and M. Keche, "Driver drowsiness detection system," in *Proc. 2013 8th International Workshop on Systems, Signal Processing and Their Applications (WoSSPA)*, 2013, pp. 151-155.

[3] Y. Kurylyak, F. Lamonaca, and G. Mirabelli, "Detection of the eye blinks for human's fatigue monitoring," in *2012 IEEE International Symposium on Medical Measurements and Applications Proceedings (MeMeA)*, 2012, pp. 1-4.

[4] W. S. Wijesoma, K. R. S. Kodagoda, and A. P. Balasuriya, "Road-boundary detection and tracking using ladar sensing," *IEEE Transactions on Robotics and Automation*, vol. 20, no. 3, June 2004.

[5] H. L. Wang, H. H. Liu, and Z. M. Song, "Fatigue driving detection system design based on driving behavior," in *Proc. 2010 International Conference on Optoelectronics and Image Processing (ICOIP)*, pp. 549 – 552.

[6] T. C. Chieh, M. M. Mustafa, A. Hussain, E. Zahedi, and B. Y. Majlis, "Driver fatigue detection using steering grip force," in *Proc. Student Conf. on Research and Development (SCORED 2003)*, Aug. 2003, pp. 45-48.

[7] M. A. Li, C. Zhang, and J. F. Yang, "An EEG-based method for detecting drowsy driving state," in *Proc. 2010 Seventh International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*, 2010, pp. 2164-2167.

[8] J. J. Wang, W. Xu, and Y. H. Gong, "Real-time driving danger-level prediction," *Engineering Applications of Artificial Intelligence*, pp. 1247–1254, 2010.

Wasan Tansakul received his B.Eng. Degree in Control System and Instrumentation Engineering from King Mongkut's University of Technology Thonburi, Bangkok, Thailand in 2012, where he is currently a Master's student. His research topic involves improvement of a driver fatigue detection device.

Poj Tangamchit received his Ph.D. degree in Electrical and Computer Eng. (2003) from Carnegie Mellon University, USA. He is currently an associate professor at the department of Control System and Instrumentation Engineering at King Mongkut's University of Technology Thonburi, Bangkok, Thailand. His research involves AI, robotics, and ITS.