

## Research Article

# Smart Real-Time Video Surveillance Platform for Drowsiness Detection Based on Eyelid Closure

Muhammad Tayab Khan,<sup>1</sup> Hafeez Anwar,<sup>2</sup> Farman Ullah ,<sup>2</sup> Ata Ur Rehman,<sup>2</sup> Rehmat Ullah,<sup>3</sup> Asif Iqbal,<sup>4</sup> Bok-Hee Lee,<sup>5</sup> and Kyung Sup Kwak <sup>4</sup>

<sup>1</sup>The Incubator, Ghulam Ishaq Khan Institute of Engineering Sciences and Technology, Topi KPK, Pakistan

<sup>2</sup>Department of Electrical and Computer Engineering, COMSATS University Islamabad, Attock Campus, Pakistan

<sup>3</sup>Department of Computer Systems Engineering, University of Engineering & Technology, Peshawar, Pakistan

<sup>4</sup>Department of Information and Communication Engineering, Inha University, Republic of Korea

<sup>5</sup>Department of Electrical Engineering, Inha University, Republic of Korea

Correspondence should be addressed to Kyung Sup Kwak; kskwak@inha.ac.kr

Received 6 September 2018; Revised 23 November 2018; Accepted 22 January 2019; Published 19 March 2019

Guest Editor: Sarmadullah Khan

Copyright © 2019 Muhammad Tayab Khan et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We propose drowsiness detection in real-time surveillance videos by determining if a person's eyes are open or closed. As a first step, the face of the subject is detected in the image. In the detected face, the eyes are localized and filtered with an extended Sobel operator to detect the curvature of the eyelids. Once the curves are detected, concavity is used to tell whether the eyelids are closed or open. Consequently, a concave upward curve means the eyelid is closed whereas a concave downwards curve means the eye is open. The proposed method is also implemented on hardware in order to be used in real-time scenarios, such as driver drowsiness detection. The evaluation of the proposed method used three image datasets, where images in the first dataset have a uniform background. The proposed method achieved classification accuracy of up to 95% on this dataset. Another benchmark dataset used has significant variations based on face deformations. With this dataset, our method achieved classification accuracy of 70%. A real-time video dataset of people driving the car was also used, where the proposed method achieved 95% accuracy, thus showing its feasibility for use in real-time scenarios.

## 1. Introduction

We propose an image-based framework for the detection and recognition of drowsiness based on a person's eyes. Such a framework can be instrumental in a multitude of scenarios, such as driver drowsiness detection, and thus has the potential to save lives. Among other things, the most common causes of driver drowsiness are fatigue and excessive alcohol consumption. In such cases, it is extremely important to detect the condition of the driver and take appropriate steps to save lives on the roads. Our proposed framework is a step towards a solution to this public issue. To that end, our framework continuously monitors a driver's condition in real time by using a video camera installed in front of the person. From the video, we use an image-based noninvasive technique to detect the eyes of the driver and classify them as being open or not.

However, detection and classification of a driver's eyes constitute a nontrivial problem that has a set of challenges. First is localization of the driver's face from among other passengers. We utilize the Viola-Jones [1] algorithm for face detection due to its real-time performance and robustness to scale and location variations. In addition to the driver's face, the algorithm detects all faces in a video that may contain some falsely detected faces. After localization of the candidate face, the next challenge is detection of the eyes. As with face detection, the Viola-Jones algorithm detects many regions in the face as eyes, among which are falsely detected eyes. Lastly, the major obstacle in our approach is the removal of eyebrow in the detected candidate eye region. We propose using the curvature of the eyelids to determine the state of the eye as being open or closed. However, the curvature of the eyebrows is also detected with our specially designed curvature detection filter.

We propose an incremental approach to solve these problems. As a first step, we detect faces with the Viola-Jones algorithm. The candidate face from among the detected faces is then decided as being the one that has the largest area assuming that the driver is nearest to the camera. Only this face region is then processed in subsequent video frames, reducing the processing cost. The eyes in this face region are then detected and processed for eyelid curvature detection. Finally, to detect the curvature of the eyelids, we apply a filter only to that part of the candidate eye region that is more likely to contain the eyelids. Consequently, noise induced by the eyebrows is reduced to a reasonable level. The curvature of the eyelid is a compact feature that is feasible for real-time scenarios due to its quick computation and acceptable accuracy.

**1.1. Related Work.** Most of the methods proposed for image-based drowsiness detection use symptoms related to the level of the driver's drowsiness. Detecting eyes that are closed or open is used in many methods. For instance, Dong and Wu [2] proposed detecting the eye state via distance calculation between the upper and lower eyelids. Dasgupta et al. [3] used the amount of eyelid closure as a cue for drowsiness. The presence of the iris in the image indicates that the eye is open. Detection of the iris in an eye image using the circular Hough transform is used in [4] to classify the eye as being open or closed. The ratio between the eye's height and width, as well as its area, is used in [5] to determine the status of the eye. Other methods include template matching [6, 7], use of local image features [8, 9], and using a Hidden Markov Model (HMM) [10]. Head pose estimation is used by Teyeb et al. [11], where the level of driver alertness is measured by the head being inclined to a certain degree and for a specific time period. A threshold placed on the changing rate of the mouth contour to detect yawning is used by Alioua et al. [12] to detect drowsiness of the driver. However, expert knowledge can be instrumental for accurate and timely detection of drowsiness. Such knowledge is implemented by Rezaei and Klette [13] using a fuzzy control fusion system. Intense devoted driving condition of such a driver who has already faced an accident before is called hypovigilance. This condition leads to rapid exhaustion and could cause drowsiness. Smith et al. [14] proposed using a finite state machine (FSM) to detect hypovigilance and used it as a cue for drowsiness detection.

In addition to these techniques, specialized machine learning-based methods have been proposed with the recent emergence of deep learning. Park et al. [15] proposed a deep architecture, referred to as deep drowsiness detection (DDD). It is reliable for the exclusion of backgrounds and environment variations and achieved 73.06% accuracy. Weng et al. [16] introduced a novel hierarchical temporal deep belief network (HTDBN) for drowsiness detection. Their work highlights the detection of head positions and faces to detect drowsiness. Huynh et al. [17] used a three-dimensional convolutional neural network (CNN) to extract features in spatial-temporal domain. The method is designed to solve issues with extreme head poses, where it achieved

87.46% accuracy. Shih and Hsu [18] use a multistage spatial-temporal network (MSTN) with a CNN to detect various states of drowsiness. They achieved 82.61% accuracy. Lyu et al. [19] used random forest method to extract effective facial descriptors to describe drowsiness based on face alignment, and to classify the driver's facial states where the claimed accuracy is 88.18%. Although, a deep learning-based method achieves state-of-the-art performance, success still lies in the availability of large amounts of data to train deep nets. In contrast, our proposed method is purely based on image processing techniques that are suitable for real-time implementation. Our method is a stepwise procedure to determine the drowsiness of a driver. Following are the major steps:

- (1) face detection in the image,
- (2) driver's face extraction among the detected faces,
- (3) extraction of the region of interest (ROI) that contains the driver's face,
- (4) eye extraction and eyebrow removal from the detected driver's face,
- (5) extraction of the eyelids from the detected eyes,
- (6) determination of the eyelids as being concave up or down and classification of the eyes based on this concavity as being open or closed.

Explanation for each of the steps follows.

**1.2. Faces Detection in the Image.** Faces in the scene are detected via the famous Viola-Jones object detection algorithm [1] due to its fast calculation time. Following are the main steps of this algorithm:

- (1) integral image calculation,
- (2) feature detection,
- (3) AdaBoosting for redundant features rejection,
- (4) classification of the detected features using cascade of classifiers.

**1.3. Driver's Face Extraction among the Detected Faces.** The Viola-Jones framework is likely to detect multiple faces in the image as shown in Figure 1. Due to this reason, the major issue now is to extract the face of interest which is the driver's face. In our proposed setup, the camera is installed closer to the driving seat. Consequently, the face of the driver will occupy more number of pixels in the image than any other detected faces. From Figure 1 it can be observed that among all those detected faces, the driver's face has the highest area in the image.

**1.4. Extraction of the Region of Interest (ROI) That Contains the Driver's Face.** For fast processing as per demand of real-time application, we do not need to process the whole image in the consecutive frames but to process only the region of interest (ROI) that contains the driver's face. The parameters of the ROI are taken from the very first frame and then used in the rest of the driving session. The ROI is taken as the extension



FIGURE 1: Multiple faces detection in a driving-like scenario.

of all four sides of driver's face with 80% of its width as shown in Figure 2. In subsequent frames, we extract the ROI for the processing of driver's face, thus reducing the computational cost and time.

**1.5. Eye Extraction and Eyebrow Removal from the Detected Driver's Face.** The next step after face detection is to extract the driver's eyes from the detected face image. However, the bounding boxes for the eyes also contain eyebrows which act as noise, as shown in Figure 3. Since, our proposed method is based on the curvature of the eyelids and the eyebrows have a similar orientation, they are likely to generate falsely detected eyelids. Therefore, the eyebrows must be removed from the detected eye bounding boxes in order to achieve accurate eyelid detection. We propose combining the regions in the bounding boxes of both eyes in order to remove the eyebrows. To this end, the extreme points of both the bounding boxes are considered. Let  $S_L(x_1, y_1)$  and  $S_R(x_1, y_1)$  be the top left extreme points of both bounding boxes, as shown in Figure 3(a). Similarly,  $E_L(x_2, y_2)$  and  $E_R(x_2, y_2)$  are the bottom right extreme points of the left and right bounding boxes, respectively. Since, the combined bounding box for both eyes is the combination of both these bounding boxes, the extreme points of the combined bounding box are determined with the help of the extreme points of the individual bounding boxes. First, the  $x$ -coordinate of the upper left corner of the combined bounding box labeled as  $S_{Box}$ , as shown in Figure 3(b), is determined. Since  $S_L(x_1) < S_R(x_1)$ , the  $x$ -coordinate of  $S_{Box}$  is equal to  $S_L(x_1)$ . Similarly,  $S_R$  is lower than  $S_L$  along the  $y$ -axis, i.e.,  $S_R(y_1) > S_L(y_1)$ , so we take the  $y$ -coordinate of  $S_{Box}$  equal to  $S_R(y_1)$ . The bottom right extreme point, i.e.,  $E_{Box}$  of the combined region, is determined in a similar manner. Combining the bounding boxes of both the left and right bounding boxes helps to eliminate regions that contain eyebrows. Nonetheless, this combined region still contains a considerable part of the eyebrows, as shown in Figure 3(b). We propose eliminating the eyebrows by lowering the top left corner of the combined region by 20% of the height of the region along the  $y$ -axis as shown in Figure 3(c).

**1.6. Extraction of the Eyelids from the Detected Eyes.** The eyelids of the detected eyes are extracted via their curvature. We assume that the curvature of the eyelid's edge is greater

for open eyes than for closed eyes. However, before the measurement of the curvature, the edge of the eyelid should be detected. We propose modifying the Sobel operator so it can detect the curved edges in the detected eyes. This is shown in Figure 4, along with its response on an open eye. To get a pronounced response and to reject extra curvature detection like that of the iris, the filter is applied again to its own response. This is shown in Figure 5, where the first and second responses of the filter are depicted. Several contours are detected by the double response of the filter, where we consider the contour with the largest area as the eyelid.

**1.7. Eye Classification Based on Eyelid Curvature.** Once the eyelid is detected, the next step is to classify the curve as being upward or downward curve in order to determine the state of the eye. Concavity of the detected eyelid curve is determined with the help of its two extreme points and their mid points along both the  $x$ -axis and  $y$ -axis. If the curve of the eyelid is "concave down" then the eye is "open"; however, if the curve is "concave up" then the eye is "closed". The concavity approximation of the eyelid curve is explained as follows.

Let the two extreme points of the curve be  $A(x_1, y_1)$  and  $B(x_2, y_2)$ , as shown in Figure 6 for both concave up and concave down curves. Let the line segment passing through the mid of these extreme points with respect to their  $y$  coordinates be  $E'E''$ , as shown in Figures 6(a) and 6(d). Then, the  $y$  intercept of this line is given as follows.

$$E = \left| \frac{y_2 - y_1}{2} \right| \quad (1)$$

Similarly, the line segment that passes through the mid of the extreme points with respect to their  $x$  coordinates is  $C'C''$ , as shown in Figures 6(b) and 6(e). The  $x$  intercept of the line  $C'C''$  is given as follows.

$$C = \frac{x_2 - x_1}{2} \quad (2)$$

This line intersects the eyelid curve at point  $D$ . Let line segment  $D'D''$  that is parallel to the  $x$ -axis intersects the eyelid curve at point  $D$  as shown in Figures 6(b) and 6(e). Consequently, we get two "curve handling" line segments, namely,  $E'E''$  and  $D'D''$ , that can determine the concavity of the eyelid curve. We can see in Figures 6(c) and 6(f) that,



FIGURE 2: Region of interest (ROI) depicting driver's face.

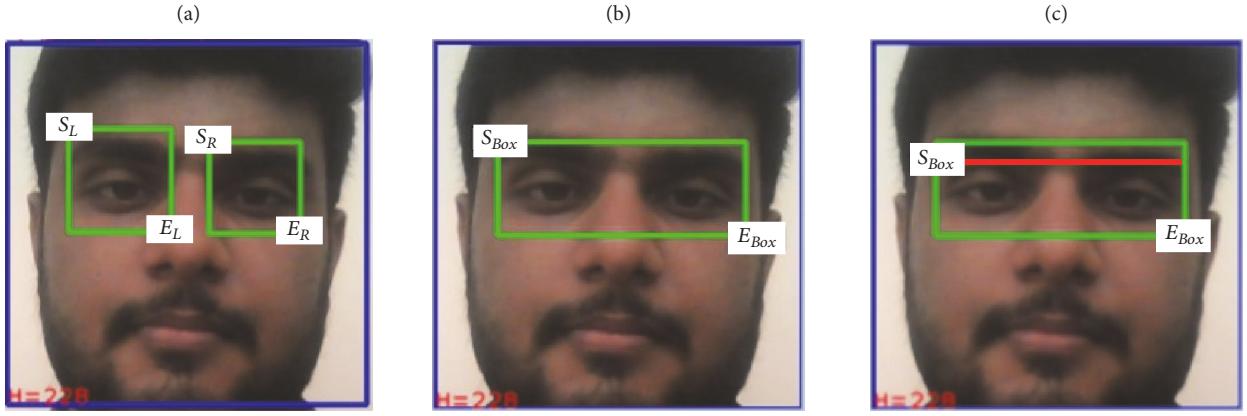
Top left corner of left bounding box:  $S_L(x_1, y_1)$ Top left corner of right bounding box:  $S_R(x_1, y_1)$ Top left corner of combined bounding box:  $S_{Box}$ Bottom right corner of left bounding box:  $E_L(x_2, y_2)$ Bottom right corner of right bounding box:  $E_R(x_2, y_2)$ Bottom right corner of combined bounding box:  $E_{Box}$ 

FIGURE 3: Removal of eyebrows.

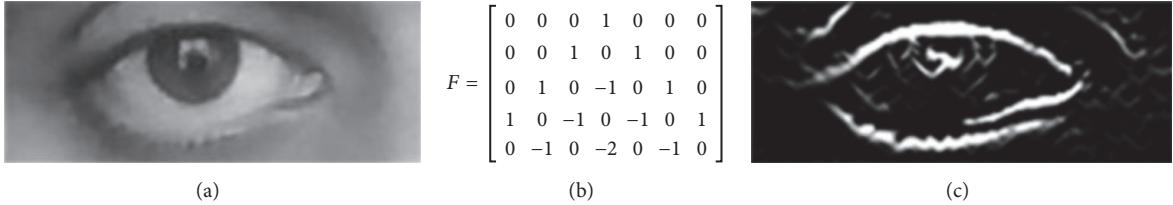


FIGURE 4: (a) Original image of the eye, (b) the proposed filter, and (c) response of the filter on eye image.

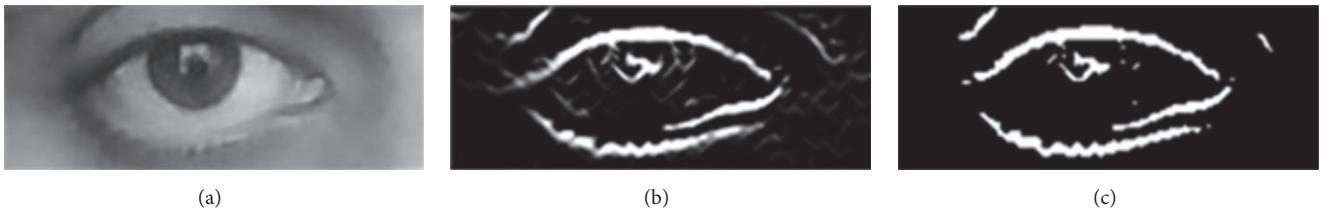


FIGURE 5: (a) Original image of the eye, (b) first response of the filter, and (c) second response of the filter applied to the first response.

if line segment  $E'E''$  lies above  $D'D''$  along the  $y$ -axis, then the curve is concave up, and the eye is closed. If line segment  $D'D''$  lies above  $E'E''$  along  $y$ -axis, then the curve is concave down and the eye is open.

However, such ideal curves are not always detected in a real scenario. Instead, half curves are mostly detected due to shade of the nose over the eyes which results in a low contrast region in the image. Consequently, our proposed

curve approximation method also works for such a condition which is shown in Figure 7.

## 2. Real-Time Drowsiness Detection and an Alarm Generation System

The proposed method is used to design an alarm generation system for the alertness of drowsy drivers. This system is built

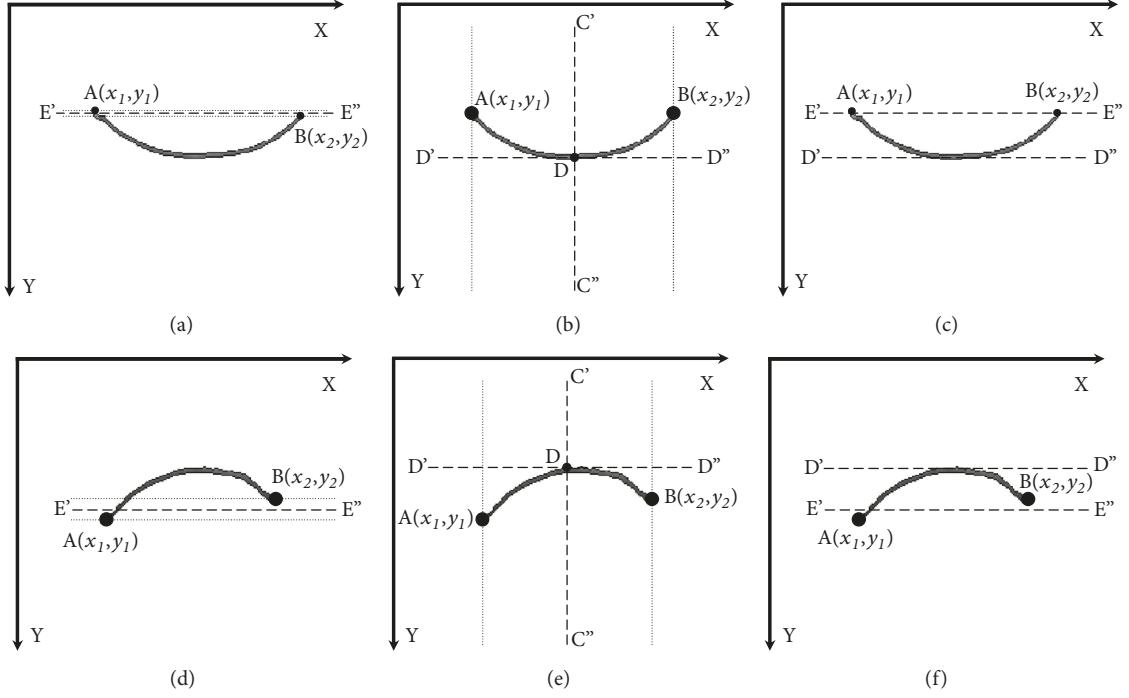


FIGURE 6: (a)-(c) Ideal eyelid curve for open eye, (d)-(f) ideal eyelid curve for closed eye.

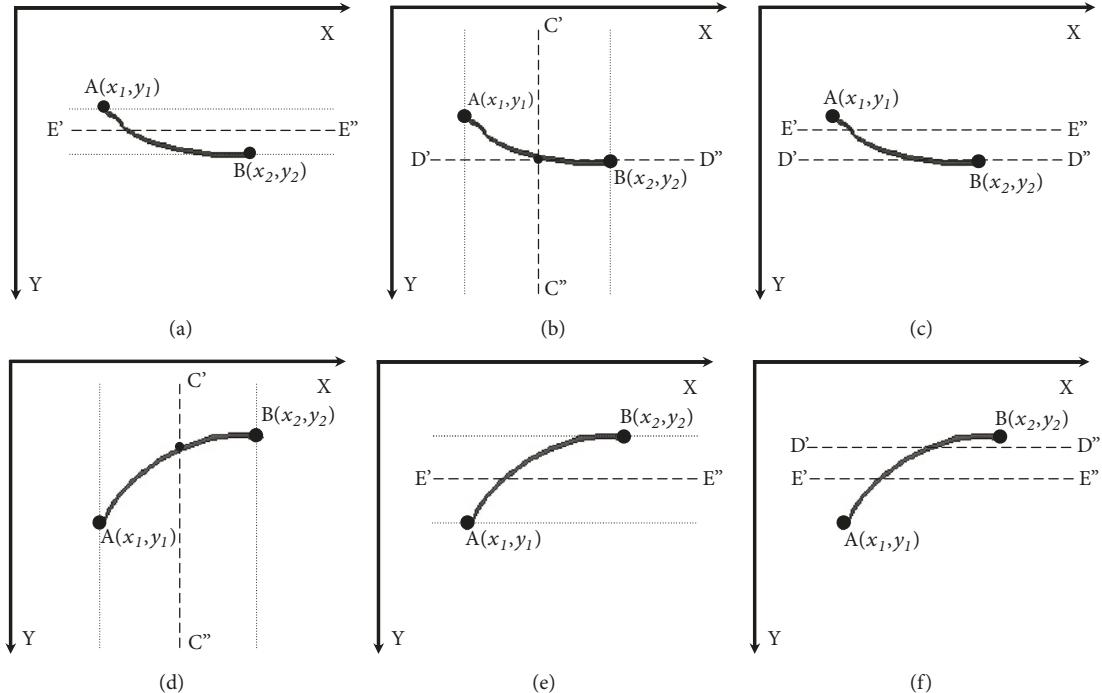


FIGURE 7: Ideal curves of the phenomenon when the eyelid curve is half detected.

on a Raspberry Pi with an interfaced camera for real-time video capture. A buzzer and a light-emitting diode (LED) are also interfaced to the Raspberry Pi to generate an alarm if the driver is detected as drowsy. The whole system is shown in Figure 8. A flow chart of the algorithm implemented in the system is shown in Figure 9. It consists of two main blocks.

The first is the preprocessing block where the face detection is performed on the real-time video captured by the camera. Once a face is found, its ROI is detected and processed in the second block to detect driver drowsiness via eyelid closure. The eyes of the driver are continuously monitored, and if found closed for a certain period of time, the alarm sounds.

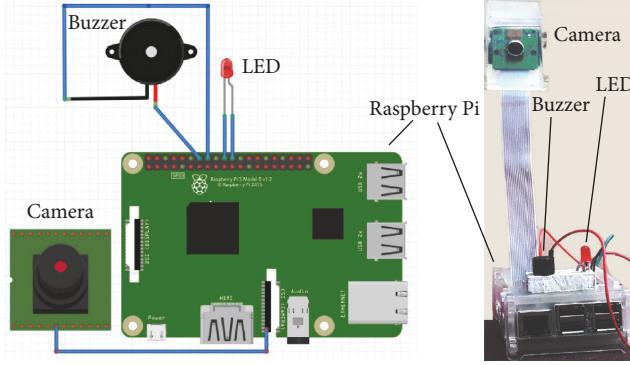


FIGURE 8: Real-time system for driver drowsiness detection and alarm generation.

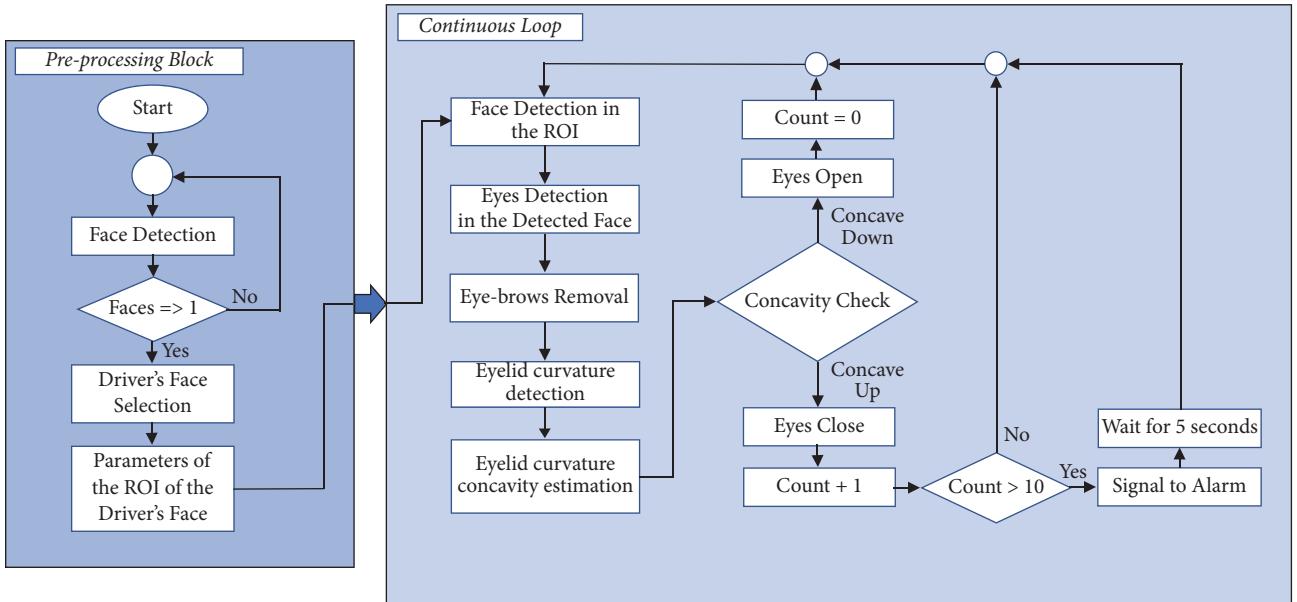


FIGURE 9: Flow chart of the real-time drowsiness detection and alarm generation system.

### 3. Results and Discussion

In order to evaluate our proposed method, we used image datasets that are incrementally difficult. For instance, as a first step, we generated our own image dataset where a person's face is imaged with uniform background. We acquired images of 319 persons with their eyes open and closed. Detailed statistics are shown in Table 1, where we outlined the accuracy of face detection, eye detection, and then classification of the eyes as open or closed. The detection results of both the open and closed eyes are shown in Figure 10. The Viola-Jones algorithm gives 100% face detection accuracy on our not-so challenging dataset due to the absence of background clutter and the uniform illumination from an in-door environment. Consequently, the eyes are detected with almost perfect accuracy (i.e., 98%). Our proposed algorithm for eyelid curvature detection and eye classification based on this detection gives almost 95% accuracy. However, this accuracy is dependent upon the eyes detection accuracy which is 98%. Therefore, the relative accuracy of our algorithm becomes 97%.

We also evaluated our proposed method on a bench mark dataset [20]. The images in the dataset are challenging due to image variations caused by face deformations, out-of-plane orientations, glasses, and irregular illuminations. Since, our basic assumption is that the subject will be looking towards the camera, we rejected images in the dataset with severe out-of-plane and in-plane rotations. The size of the images is  $100 \times 100$  that are scaled up to  $150 \times 150$  in order to meet the requirements of our proposed method. Table 2 shows the statistics achieved by the proposed method on the dataset. Some of the correct eye classification results are shown in Figure 11 while incorrect results are shown in Figure 12.

Lastly, the proposed method was evaluated with a real-time video of someone driving a car. The video was recorded in the day time with variations in illumination due to differences in the direction of sunlight. Two videos were recorded to evaluate the proposed method, for which statistics are shown in Table 3. Some exemplar frames where the eyes are correctly and wrongly classified are shown in Figures 13 and 14, respectively. The face and eye detection accuracies



FIGURE 10: Detection results of open and closed eyes from images with a homogeneous background.

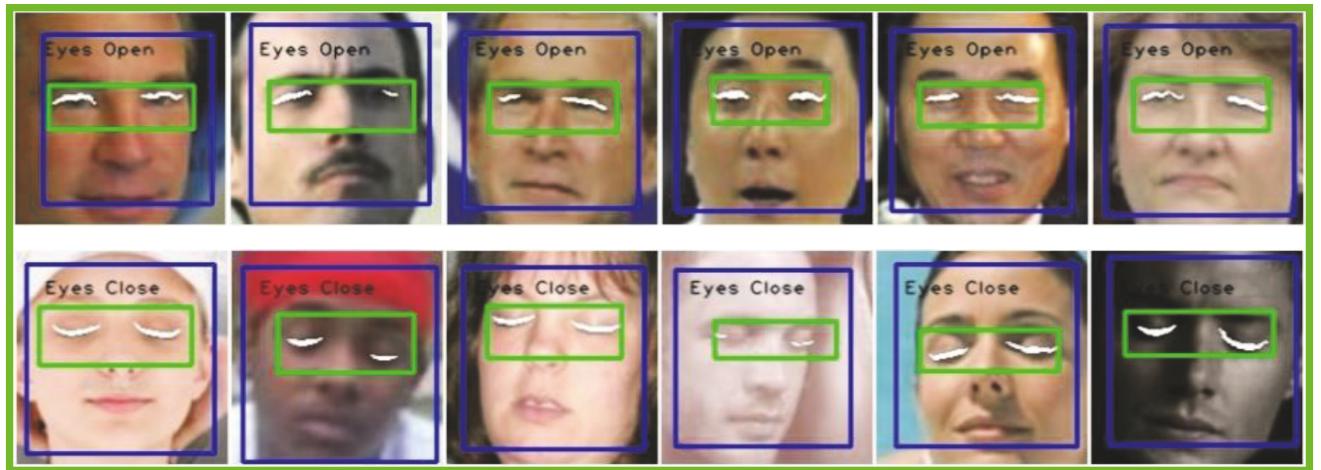


FIGURE 11: Correct detection of open and closed eyes on the bench mark dataset [20].

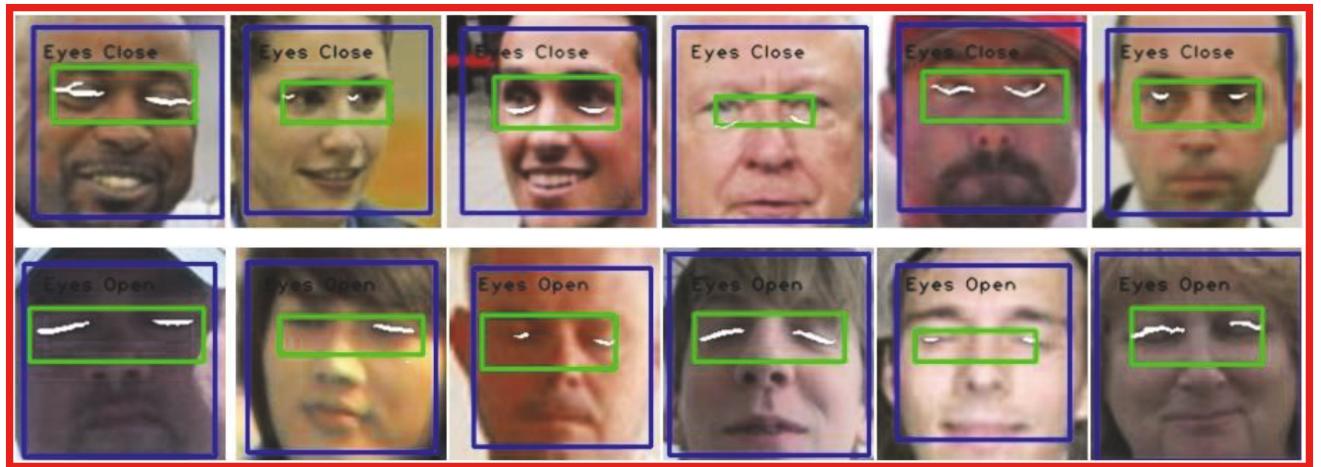


FIGURE 12: Incorrect detection of open and closed eyes from the bench mark dataset [20].

TABLE 1: Open and closed eye detection from images with uniform background.

Type of dataset	Number of images	Face Detection Accuracy	Eye Detection Accuracy	Eye Classification Accuracy
Close Eyes Dataset	319	100%	97.43%	90.31%
Open Eyes Dataset	319	100%	98.73%	99.26%
Total	638	100%	98.08%	94.78%

TABLE 2: Open and closed eye detection rates from images in a benchmark dataset [20].

Type of dataset	Number of images	Face Detection Accuracy	Eye Detection Accuracy	Eye Classification Accuracy
Close Eyes Dataset	869	90.45%	73.80%	68.70%
Open Eyes Dataset	807	98.80%	98.37%	73.17%



FIGURE 13: Correct detection of open and closed eyes on the video dataset.



FIGURE 14: Incorrect detection of open and closed eyes on the video dataset.

from both the videos are not encouraging. However, the main purpose of the proposed algorithm is to classify eyes in the detected face and eye images. To this end, it achieves over 95% eye classification accuracy from both videos. This shows the feasibility of the proposed method for driver drowsiness detection. It should be noted, however, that these results were achieved during the day time. For night time, our proposed method can be used on top of face and eye detection algorithms that work during night time.

#### 4. Conclusion

A method for image-based drowsiness detection in real-time driving surveillance videos is proposed. It is a four-step method that first detects the face of the driver in

the image from among several detected faces. Secondly, it extracts the eyes from the detected faces. In the third step, the curvature of the eyelids is detected using a modified Sobel operator. Finally, the eyes are classified as closed or open based on the curvature of the eyelids. The proposed method achieved an average classification accuracy of 95% on a simple image dataset with homogeneous backgrounds, an average classification accuracy of 70% on a complex benchmark image dataset, and greater than 95% classification accuracy on two real-time driving surveillance videos. However, the proposed method works only in the day time; its adaptation to night time will be explored in future work with more state-of-the-art face and eye detection algorithms. Similarly, more challenging face images where subjects might have glasses or phones will be used to evaluate the proposed method.

TABLE 3: Classification accuracy achieved on real-time video datasets.

Type of Dataset	Number of Frames	Face Detection Accuracy	Eye Detection Accuracy	Classification Accuracy
Closed Eyes video 1	1839	73.57%	73.17%	95.00%
Closed Eyes video 2	1387	96.75%	67.00%	95.00%
Open Eyes video 1	3601	44.32%	70.30%	90.50%
Open Eyes video 2	3201	78.01%	96.19%	95.20%
Total	10028	73.21%	76.66%	93.79%

## Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea Government (MSIT)-NRF-2017R1A2B2012337.

## References

- [1] P. Viola and M. J. Jones, "Robust real-time face detection," *International Journal of Computer Vision*, vol. 57, no. 2, pp. 137–154, 2004.
- [2] W. Dong and X. Wu, "Driver fatigue detection based on the distance of eyelid," in *Proceedings of the 2005 IEEE International Workshop on VLSI Design and Video Technology, IWVDVT 2005*, IEEE, pp. 365–368, China, May 2005.
- [3] A. Dasgupta, A. George, S. L. Happy, and A. Routray, "A vision-based system for monitoring the loss of attention in automotive drivers," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 4, pp. 1825–1838, 2013.
- [4] N. Alioua, A. Amine, M. Rziza, and D. Aboutajdine, "Eye state analysis using iris detection based on Circular Hough Transform," in *Proceedings of the 2011 International Conference on Multimedia Computing and Systems, ICMCS'11*, IEEE, pp. 1–5, Morocco, April 2011.
- [5] Y. Du, P. Ma, X. Su, and Y. Zhang, "Driver fatigue detection based on eye state analysis," in *Proceedings of the 11th Joint Conference on Information Sciences*, 2008.
- [6] X. Liu, X. Tan, and S. Chen, "Eyes closeness detection using appearance based methods," in *Proceedings of the International Conference on Intelligent Information Processing*, vol. 385, pp. 398–408, Springer, 2012.
- [7] G. Fa-deng and H. Min-xian, "Study on the detection of locomotive driver fatigue based on image," in *Proceedings of the 2010 2nd International Conference on Computer Engineering and Technology*, pp. V7-612–V7-615, Chengdu, China, April 2010.
- [8] L. Zhou and H. Wang, "Open/closed eye recognition by local binary increasing intensity patterns," in *Proceedings of the 2011 IEEE 5th International Conference on Robotics, Automation and Mechatronics, RAM 2011*, IEEE, pp. 7–11, China, September 2011.
- [9] M. Tafreshi and A. M. Fotouhi, "A fast and accurate algorithm for eye opening or closing detection based on local maximum vertical derivative pattern," *Turkish Journal of Electrical Engineering & Computer Sciences*, vol. 24, no. 6, pp. 5124–5134, 2016.
- [10] H. Qin, J. Liu, and T. Hong, "An eye state identification method based on the embedded hidden Markov model," in *Proceedings of the 2012 IEEE International Conference on Vehicular Electronics and Safety, ICVES 2012*, IEEE, pp. 255–260, Turkey, July 2012.
- [11] I. Teyeb, O. Jemai, M. Zaied, and C. B. Amar, "A drowsy driver detection system based on a new method of head posture estimation," in *Proceedings of the International Conference on Intelligent Data Engineering and Automated Learning*, pp. 362–369, Springer, 2014.
- [12] N. Alioua, A. Amine, and M. Rziza, "Driver's fatigue detection based on yawning extraction," *International Journal of Vehicular Technology*, vol. 2014, Article ID 678786, 7 pages, 2014.
- [13] M. Rezaei and R. Klette, "Look at the driver, look at the road: No distraction! No accident!" in *Proceedings of the 27th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014*, pp. 129–136, USA, June 2014.
- [14] P. Smith, M. Shah, and N. da Vitoria Lobo, "Determining driver visual attention with one camera," *IEEE Transactions on Intelligent Transportation Systems*, vol. 4, no. 4, pp. 205–218, 2003.
- [15] S. Park, F. Pan, S. Kang, and C. D. Yoo, "Driver drowsiness detection system based on feature representation learning using various deep networks," in *Proceedings of the Computer Vision – ACCV 2016 Workshops*, vol. 10118 of *Lecture Notes in Computer Science*, pp. 154–164, Springer International Publishing, Cham, Switzerland, 2016.
- [16] C. Weng, Y. Lai, and S. Lai, "Driver drowsiness detection via a hierarchical temporal deep belief network," in *Proceedings of the Computer Vision – ACCV 2016 Workshops*, vol. 10118 of *Lecture Notes in Computer Science*, pp. 117–133, Springer International Publishing, Cham, Switzerland, 2016.
- [17] X.-P. Huynh, S.-M. Park, and Y.-G Kim, "Detection of driver drowsiness using 3D deep neural network and semi-supervised gradient boosting machine," in *Proceedings of the Computer Vision – ACCV 2016 Workshops*, vol. 10118 of *Lecture Notes in Computer Science*, pp. 134–145, Springer International Publishing, Cham, Switzerland, 2016.
- [18] T. Shih and C. Hsu, "MSTN: multistage spatial-temporal network for driver drowsiness detection," in *Proceedings of the Computer Vision – ACCV 2016 Workshops*, vol. 10118 of *Lecture Notes in Computer Science*, pp. 146–153, Springer International Publishing, Cham, Switzerland, 2016.
- [19] J. Lyu, H. Zhang, and Z. Yuan, "Joint shape and local appearance features for real-time driver drowsiness detection," in *Proceedings of the Computer Vision – ACCV 2016 Workshops*, vol. 10118 of *Lecture Notes in Computer Science*, pp. 178–194, Springer International Publishing, Cham, Switzerland, 2016.
- [20] F. Y. Song, X. Y. Tan, X. Liu, and S. C. Chen, "Eyes closeness detection from still images with multi-scale histograms of principal oriented gradients," *Pattern Recognition*, vol. 47, no. 9, pp. 2825–2838, 2014.

