

Research Article

Enhanced Image Processing and Fuzzy Logic Approach for Optimizing Driver Drowsiness Detection

Wasin AlKishri,¹ Abdallah Abualkishik ,² and Mahmood Al-Bahri²

¹Faculty of Computer Studies, Department of Information Technology and Computing, Arab Open University–Oman, Qurum, Madinat Sultan Qaboos-P. O. B: 1596, P. C: 130, Oman

²Faculty of Computing and IT, Sohar University, Sohar 311, Oman

Correspondence should be addressed to Abdallah Abualkishik; aabualkishik@su.edu.om

Received 20 October 2021; Revised 30 January 2022; Accepted 15 February 2022; Published 19 March 2022

Academic Editor: Soumya Ranjan Nayak

Copyright © 2022 Wasin AlKishri 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.

Driver drowsiness is a severe problem that usually causes traffic accidents, classified as more dangerous. The record of the National Safety Council reported that drowsy driving is caused by 9.5% of all crashes (100,000 cases). Therefore, preventing and minimizing driver fatigue is a significant research area. This study aims to design a nonintrusive real-time drowsiness system based on image processing and fuzzy logic techniques. It is an enhanced approach for Viola–Jones to examine different visual signs to detect the driver’s drowsiness level. It extracted eye blink duration and mouth features to detect driver drowsiness based on the desired facial feature image in a specific driver video frame. The size and orientation of the captured features were tracked and handled for determining image features such as brightness, shadows, and clearness. Lastly, the fuzzy control system provides different alert sounds based on the tracked information from the face, eyes, and mouth in separate cases, such as race, wearing glasses or not, gender, and various illumination backgrounds. The experiments’ results show that the proposed approach achieved high accuracy of 94.5% in detecting driver status compared with other studies. Also, the fuzzy logic controller efficiently issued the required alert signal of the drowsy driver status that helps to save the driver’s life.

1. Introduction

According to Peura et al. [1], driver drowsiness is one of the significant factors that cause many traffic accidents in the world. Annually, upwards of 100,000 vehicles are crashed, and around 1,500 people die. Annually, almost 12.5 billion dollars are the total loss cost for these types of accidents. Preventing such driver fatigue accidents is a high focus effort on many current types of research [2, 3]. Driver fatigue increases the need to develop a monitoring system that analyses driver status and provides different alert sounds based on his facial features. Most researchers concentrated on detecting drowsy driving through analyzing the eyes’ pupil’s parameters [4–6]. Through this research, driver drowsiness investigations are based on capturing the driver’s video and detecting the driver’s face using some technique. After that, they analyze the eye blinking frequency and

decide the driver fatigue status [7]. Some other researchers [8–10] include the mouth and features too. For some reason, the system may work inefficiently to detect driver drowsiness. Varying light conditions and vibration of the driver is one of the main challenges to identifying the driver status in real time [11]. However, using multiple visual features and cues is more efficient for detecting driver drowsy.

The proposed system detects driver fatigue in real time through observing various facial features and selecting the correct driver state. It provides different alert sounds based on a different level of drowsing for the driver. This has been carried out by using a fuzzy logic technique in addition to the face features’ detection process. This study is organized as follows. Section 2 explores significant reviews of previous studies and related works in the same field of research systems. Section 3 describes the model of the system and how the experiments were conducted. Section 4 displays the

results of the conducted experiments. Finally, present the conclusion and future work.

2. Background and Related Work

2.1. Driver Drowsiness. Drowsiness causes significant social and economic losses to the country about road accidents that often occur on highways. Drowsiness accidents happen when the driver delays responding to a specific situation and lose control of the vehicle. Moreover, it is difficult to determine the level of the driver's drowsiness because it cannot be measured after the accident. Drowsiness usually affects various attitudes such as vigilance, decision-making, and concentration for drivers [12].

2.2. Driver Fatigue Monitoring Techniques. Various techniques of monitoring vigilance and fatigue are used to measure driver performance, including the following.

2.2.1. Physiological Behaviors. Researchers have categorized this technique as the most accurate method to detect fatigue because it is based on physiological measures such as heart rate, eye movements, respiration, and brain waves. This technique is effective when the electrical activity of the brain and body muscles is recorded. These parameters are collected from various sensors placed on the driver's body or embedded in the car. The driver usually wears a wristband to measure heart rate and a helmet or special contact lenses to monitor eye or eye movements. Despite its effectiveness, the main drawback of this technique is intrusive because it requires attaching electrodes to the driver's body, which causes the driver discomfort.

2.2.2. Indirect Vehicle Behaviors. This technique requires a significant amount of time to analyze user behavior. It includes lateral position, steering wheel movements, and time crossings, which indicate the driver's vigilance and fatigue level. It is categorized as a nonintrusive technique, but it has several limitations: vehicle type, driver experience, engineering characteristics, and road condition.

2.2.3. Directly Observable Visual Behaviors. People with fatigue show many observable behaviors usually observed in facial features, such as eye movement, head movement, and facial expression. The technique is based on different typical visual characteristics that are detected from the captured image for a drive. The captured image includes parameters, such as a lower level of deflection, longer luminescence, slow eyelid movement, a smaller degree of eye-opening, eye closure, repetitive gestures, yawning, and slow motion [11].

2.3. Related Work. Various researchers use different methods and algorithms to measure driver fatigue. Anitha [13] proposed a novel twofold yawning detection system based on an expert system. In the first part of the system, they used the face detection algorithm's skin tone detection and defined the boundaries of the face; then, the blob dimensions for the mouth in the face containment are extracted. The

system verified the yawning through a histogram of blobs which is taken from the vertical projection of the lower face part. If the histograms values are satisfying with the threshold values, then yawning is confirmed. The proposed system achieved 94% performance for yawning detection.

Kurylyak et al. [14] proposed an efficient approach to detect driver drowsy based on eye blinking. They used a web camera to acquire the driver image as input to the classifier using the Viola–Jones algorithm with Haar-like features to detect the driver's face and extract the eye region. A Kalman filter works with a set of discrete-time equations to compute and track the changes of the eye state. They calculate the frame's mutation for the thresholding value to detect the eyes' closure and opening. The frame processing algorithm is pointed out to distinguish the involuntary blinks from the voluntary ones. Experimental results of this proposed system presented 94% system accuracy to detect and determine the state of the eye.

In another work by Jo et al. [15], researchers used the same method of Kurylyak et al. [14] to detect driver face, while they proposed a new eye drowsy detection method that combined two methods. Principal component analysis (PCA) is used to detect the eyes status in the daytime and linear discriminant analysis (LDA) is used in the night. They applied support vector machine (SVM) to classify the eye states to open and close through a specific interval of 3 minutes. Experimental results of this proposed system showed that 99% of the design work accurately to detect eyes drowsiness and driver distraction. However, the systems fail to recognize the eye in various high illuminations.

Abtahi et al. [16] proposed a new method of yawning detection based on the changes in the mouth geometric features and eye movement. They used color statistics for detecting skin color and texture. Therefore, they improved detection efficiency by using bounding rules for different color spaces (RGB, YCbCr, and HSV). They experimented on more than 500 images with varying reflections of light, skin color, haircuts, beards, and eyeglasses.

Danisman et al. [17] proposed an automatic drowsy driver monitoring system to detect the eye blink duration. The proposed algorithm can catch the eye blinks' movement in real time using a webcam. Initially, they recognized the driver's face using the Viola–Jones algorithm, which is available in the OpenCV library of Python. Then, they discovered the positions of the pupils by using the symmetry property of the eye detector. The main drawbacks of that system are the presence of glasses and the various high illuminations, which affect the calculation and detection of the driver's drowsiness level. The proposed method achieved a 94% accuracy and a 1% false rate.

Bergasa et al. [18] proposed a nonintrusive computer vision system for tracking driver's vigilance in real time. The proposed method tried to test six parameters: face position and eye (eye rate, closure duration, blinks frequency, nodding frequency, and position of gaze). They used the fuzzy logic approach to combine this feature and determined the driver drowsiness level. The system was an experiment in different driving environments (night and day) with other users. The system achieved 100% accuracy at night; however, the system did not work with glasses and bright days.

Jie et al. [19] proposed new a spontaneous dataset of driver yawning in different simulated driving scenarios conditions. They present three labelling of different cases related to yawning, namely, speaking and mouth (covered or uncovered). HOGs and LBPs are popular algorithms for describing appearance in computer vision and image processing that has been used successfully in order to detect the driver yawning. These algorithms work based on intensity gradients or edge directions of the image, where it counts the pixel in the grayscale image number of oriented gradient occurrences in a dense grid of uniformly spaced cells. These occurrences are represented as a histogram for each cell normalized in a larger block area and show the mouth states.

Tipprasert et al. [20] proposed a method to detect the driver's eyes closure and yawn for drowsiness analysis by an infrared camera. The camera can work in low light condition processing by MATLAB R2015a. They obtain a 7.5% error in yawning detection because some driver opened their mouth too much, and the camera could not capture the entire driver's face.

Al-sudani et al. [21] proposed a yawning-based fatigue prediction method that monitors driver drowsiness levels. They used a camera inside the car to record driving scenarios (yawning or nonyawning driver). They built a deep CNN model to classify the drivers' fatigue into three levels, alert, early, and fatigue. Experiments are conducted using the YAWDD dataset, achieving 96.2% accuracy.

3. Proposed Approach

This study displays a nonintrusive real-time drowsiness system based on the webcam video analysis. This section shows the detection algorithm used to detect the driver's drowsiness level by investigating and analyzing the different visual cues of the driver. The main two parameters are eye blink duration and mouth state information. Figure 1 presents a block diagram of the proposed monitoring system of the drowsy driver. Develop a fuzzy controller that helps to determine the driver state and issue a suitable alert sound. The detection and monitoring approach consists of six stages as follows:

- (1) Image acquisition
- (2) Face detection and tracking
- (3) Eye iris detection and tracking
- (4) Mouth detection
- (5) Mouth and eye information analysis
- (6) Analysis driver state

3.1. Image Acquisition. It provides images of a driver's face from the recorded video to observe and gather the visual cues and then determine the fatigue level. The MatlabR2016a environment provides an image acquisition toolbox that enables the user to connect to the scientific cameras. The proposed approach used a webcam tool that installs from support hardware properties in MATLAB to create a webcam object and snapshot function to acquire images in-

stream video and convert them to the frame. Then, we manipulated the webcam object properties to be efficiently correlated with the HP laptop webcam. The webcam object-specific properties shown in Table 1 are used for HP webcam.

3.2. Face Detection. Face detection is a computer technology that helps enhance human facial features in a digital image taken as input and used in various applications. This helps in processing the location and size of the human face and avoiding other objects [22]. There are many existing algorithms or methods for face detection technology, but the main difference is detection speed, accuracy, and purpose of use. Face detection algorithms work reasonably well with the detection of frontal and bright enough human faces images. It returns a sequence of analogous image coordinates where the human face is located and matched bit by bit. The proposal application assumes that the given input video detects only a single face (driver vehicle) in the camera view; otherwise, if there is more than one face, the system will detect the closest face to the center of the frame, as shown in Figure 2.

Implementing a face detection task is normalized to reduce and narrow down the domain of seeking the pupil and mouth detection. The eyes and mouth detector will not work if the desired face is not detected enough in the frame. However, pupil and mouth detection are located in the face area if the face is detected successfully by the face detector. After a comprehensive analysis, they suggest Viola and Jones's real-time detection technology, as presented in Figure 2. The Viola-Jones algorithm passes the image through 4 steps to detect the driver's face objects. We modified Viola and Jones's algorithm to detect the desired facial feature in a specific frame of the given video sequence of the vehicle driver instead of a static input image, as shown in Figure 3. After implementing the modified approach that extracted the driver image from the video and converted it to a binary bit, Figure 4 is obtained.

Step 1 (Haar feature method): initially, the desired face scanned with the Haar feature method, which contains scalar values representing two rectangles that can be horizontal or vertical in input image resolution 24×24 pixels with a possible number of rectangles' features 160,000.

For example, the area where the eyes and mouth are located then passed this feature as an argument to the calculating [23], where I is an image, P is a pattern, and $N \times N = \text{size}$:

$$\begin{aligned} & \sum_{1 \leq i \leq N} \sum_{1 \leq j \leq N} I(i, j) 1p(i, j) \text{ is white} \\ & - \sum_{1 \leq i \leq N} \sum_{1 \leq j \leq N} I(i, j) 1p(i, j) \text{ is black.} \end{aligned} \quad (1)$$

Step 2 (integral image): the system calculates the Integral image and defines either the image contains the face or not at a very low computational cost using cumulative distribution functions:

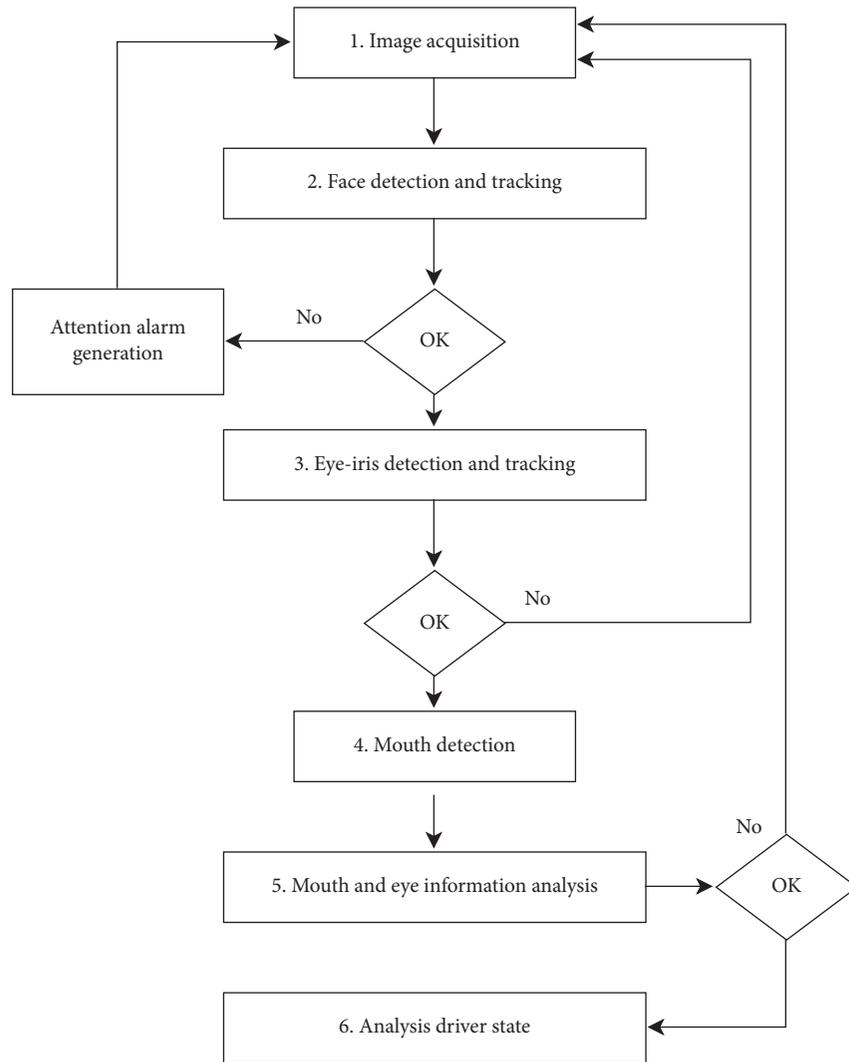


FIGURE 1: Block diagram of the proposed approach.

TABLE 1: Webcam object-specific properties.

Webcam Name:	HP Truevision HD
Quality rating	816
Frame rate	30 FPS
Stream type	Video
Image mode	rgb
Webcam megapixels	0.92 MP
Webcam resolution	1280 × 720
Video standard	HD
Number of colors	205668
Lightness	45.49%
Brightness	45.88%
Saturation	4.31%

```

Algorithm: Viola-Jones Face Detection Algorithm

1: Input: original test image
2: Output: image with face indicators as rectangles
3: for  $i \leftarrow 1$  to num of scales in pyramid of images do
4:     Downsample image to create image $i$ 
5:     Compute integral image, image $ii$ 
6:     for  $j \leftarrow 1$  to num of shift steps of sub-window do
7:         for  $k \leftarrow 1$  to num of stages in cascade classifier do
8:             for  $l \leftarrow 1$  to num of filters of stage  $k$  do
9:                 Filter detection sub-window
10:                Accumulate filter outputs
11:            end for
12:            if accumulation fails per-stage threshold then
13:                Reject sub-window as face
14:                Break this  $k$  for loop
15:            end if
16:        end for
17:        if sub-window passed all per-stage checks then
18:            Accept this sub-window as a face
19:        end if
20:    end for
21: end for
    
```

FIGURE 2: Viola-Jones detection algorithm.

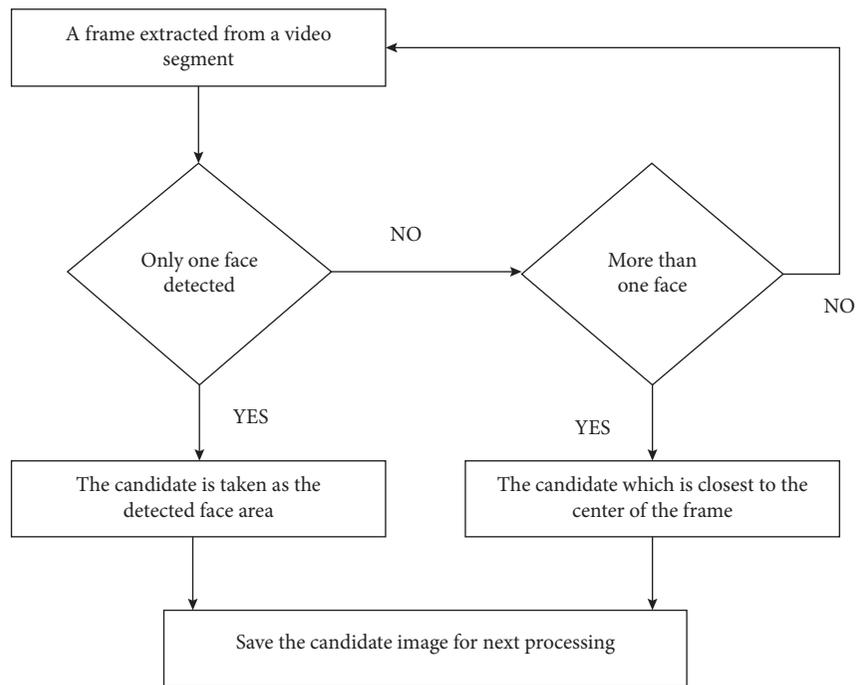


FIGURE 3: Flow diagram of the face detector.



FIGURE 4: (a) Original image, (b) converted to gray image, (c) converted to binary image, and (d) face detection.

$$II(i, j) := \begin{cases} \sum_{1 \leq s \leq i} \sum_{1 \leq t \leq j} I(s, t), & 1 \leq i \leq N \text{ and } 1 \leq j \leq N, \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

$$\begin{aligned} \sum_{N_1 \leq i \leq N_2} \sum_{N_3 \leq j \leq N_4} I(i, j) &= II(N_2, N_4) - II(N_2, N_3 - 1) - II(N_1 - 1, N_3 - 1), \\ \sum_{N_1 \leq i \leq N_2} \sum_{N_3 \leq j \leq N_4} I(i, j) &= II(N_2, N_4) - II(N_2, N_3 - 1). \end{aligned} \quad (3)$$

Step 3 (feature's selection Adaboost): this technique will remove all irrelevant features and combine only relevant features with their weight to evaluate and deciding either the image contains a face (1) or not (-1), where (X, Y) is a training example to the probability P , weights $w_i(1)$, $1 \leq i \leq n$, and ht = decision stump. As the empirical loss goes to zero with T , so do both false positive $P(f^T(X) \neq 1 \mid Y = -1)$ and false negative rates $P(f^T(X) \neq 1 \mid Y = 1)$:

$$\sum_{i=1}^n w_i(1) 1_{y_i \sum_{t=1}^T \alpha_t h_t(x_i) \leq 0} := P(f^T(X) \neq Y). \quad (4)$$

Step 4 (cascade method): this method increases processing power features by distributed every 10 features among single stages and subwindows. Each subwindow will evaluate according to its feature by the cascading method. In Figure 5, the classifier triggers the evaluation and checks the characteristics of each subwindow. If the subwindow is classified as positive (face), it will be passing through the steps. In the other case, the negative subwindow (not face) will immediately reject. This method will increase the performance power of the detected face by removing nonface-related windows at the beginning. Equation (5) defines the cascade decision rule to obtain empirical loss:

$$f_{\text{cascade}}(X) = 2 \left(\prod_{p=1}^i 1_{f_{p,p}^{T_p}(X) = 1} - \frac{1}{2} \right), \quad (5)$$

where $f^{(p)}$ is a classifier with false positive.

Face tracking is handling by the Kalman filter. The Kalman filter method is an efficient way to estimates the position of a moving object based on its historical values in the next time frame. It can predict the state of a dynamic system from a sequence of uncertainty measurements by using a recursive adaptive filter. In addition, the Kalman filter was implemented to predict the dynamic change rate of the moving object and reduce the location error [24]. The following equation is used to track and predict the face position, where x = target position, x_0 = initial position, v_0 = initial velocity, a = target acceleration, and Δ is the time interval (3 seconds in this example):

$$x = x_0 + v_0 \Delta t + \frac{1}{2} a \Delta t^2. \quad (6)$$

The face tracking method uses a particle filter based on face position, face speed, and location error. However, this

method fails when the brightness of the face is not enough due to background illumination and sudden move of the head. In Figures 5(a) and 5(b), face tracking is trained to track the front face image and its maximum rotation between ± 15 to ± 50 degrees. However, in Figure 5(c), the detector fails when the rotation is more than this and the alarm will rise.

3.3. Eye Iris Detection and Tracking. The iris is a significant parameter that can be considered to assess the fatigue level of a driver. Tracking the eyelid and eye movement can reveal the size of the iris. Therefore, it is easy to determine the eye state through geometric calculations. As shown in Figure 4, in appropriate circumstances, the system first detects the driver's face and then segments the upper half of the face to identify the eyes. Once this region has been detected in a cropped image, it can help the system reduce the computing cost of the drowsiness levels. The cropped image is then converted into a binary image using the adaptive threshold technique to detect and replace all the white pixels in the input image with the value 1 and black pixels with the value 0. Following this, we use image processing, which provides morphological operations to process images based on shapes. The morphology technique only works with the relative ordering of pixel values; the value of each pixel in the picture is modified based on its neighboring pixel in the input image. To successfully perform a morphological operation, the size and shape of the neighborhood image need to be specified.

Morphological image processing or calling a flat structuring element allows using functions such as segmentation, skeletonization, thinning, erosion, dilation, external boundary, and internal boundary. The system darkens the eye region by identifying the steel object to create a flat structuring element. This step is essential to eliminate false pixels and compute valid pixels. The eye detector focuses on the threshold and the rotation of the eye. Figure 6 illustrates a flat structuring element to detect the eyes. The iris is tracked successfully using a Kalman filter. This method uses a particle filter based on the iris' position, speed, and location error. However, this method fails when the brightness of the eye iris is not enough due to background illumination and sudden move of the head or camera resolutions. The eye size is different from one person to another; the system assumed at the beginning that the user is awake. Then, the threshold will be calculated and compared with the current eye ratio in the video frame. The system cannot predicate eye iris positions correctly due to sudden movement of the head or camera resolutions.

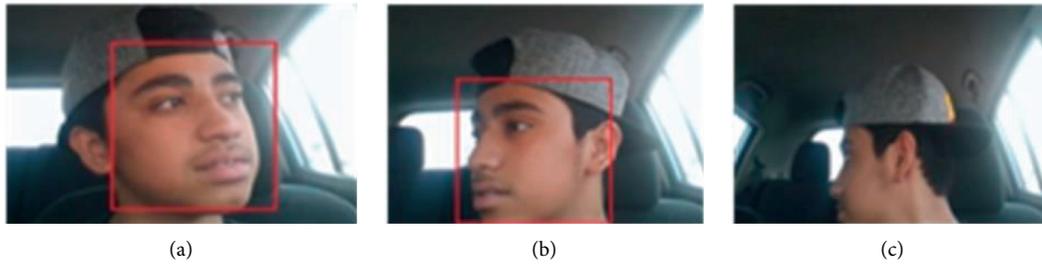


FIGURE 5: Face tracking. (a) Face detected in 30°. (b) Face detected in 45°. (c) Face undetected.

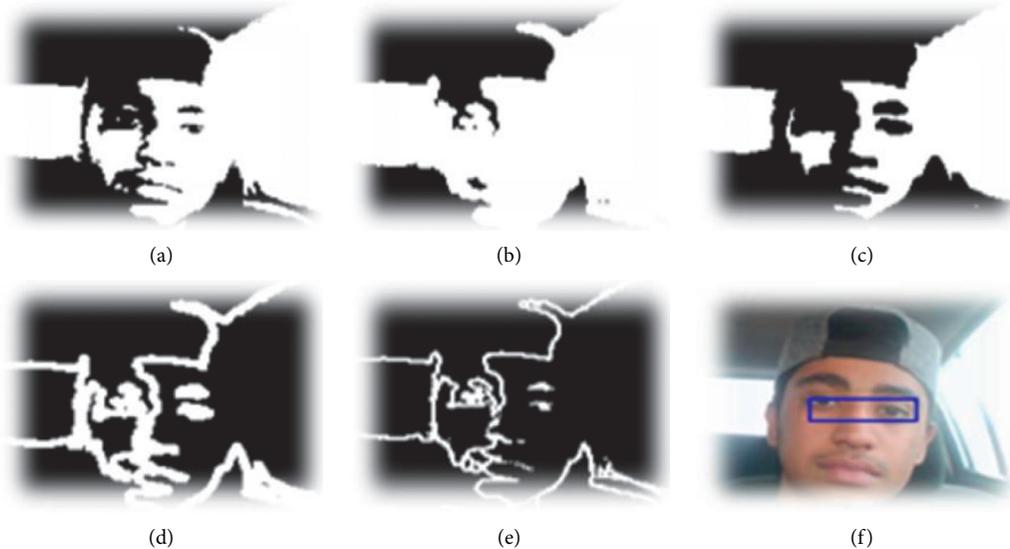


FIGURE 6: Flat structuring element to detect the eye region. (a) Original image. (b) Dilated image. (c) Eroded image. (d) Internal boundary. (e) External boundary. (f) Eyes detected.

3.4. Mouth Detection. Yawning is the most common sign of tiredness and drowsiness; hence, the system detects the driver's facial and eye movements. In the preprocessing stage, the system will crop the mouth region frame to determine the driver's state. Since the mouth is located in the lower part of the face, the mouth detectors will extract the mouth directly from the lower part of the face. After that, the system verifies mouth location using eye distance to detect the correct mouth segments. This is resolved by checking the boundaries of the mouth and eyes. Next, the system detects the mouth by calculating the connected object. The program will detect the upper and lower lips as two objects, assuming the mouth is open. When the mouth is closed, the program will calculate it as one object value; otherwise, the mouth would not be detected, and the program would return to zero value, as depicted in Figure 7. The monitored information will then pass to the fuzzy logic method to provide the driver's fatigue level.

3.5. Mouth and Eye Information Analysis. After successfully detecting and tracking the facial features of the mouth and eyes, the system will compute the mouth and eye states by defining the two input parameters, which are the total

number of black pixel areas and the ratio of black pixels compared to the ratio of white pixels.

Number of white pixels = sum (binary image).

Number of black pixels = sum (binary image) – number of white pixels.

Ratio = number of black pixels/numbers of white pixels.

The system uses the rules base to define the states of the eye and mouth. First, starting with the eye state, the system will compare the eye detection frame with the eye threshold to calculate the correct eye size ratio. Then, it will check if the ratio of the left eye pixel is more than the threshold; the eye state is considered open or closed. If the eyes are detected to be closed for more than three seconds, an alarm will sound. Similarly, the mouth state will be defined by checking the detected lips frame against the threshold to determine if the mouth is open or closed, as shown in Figure 8.

3.6. Develop a Fuzzy Model for Analysis Driver Behavior.

Fuzzy logic is a problem-solving algorithm that resembles human decision-making to provide a solution to a problem from vague or uncertain data. Fuzzy logic can describe fuzziness by representing a membership function and classifying the degree of truth of each element in a fuzzy set.

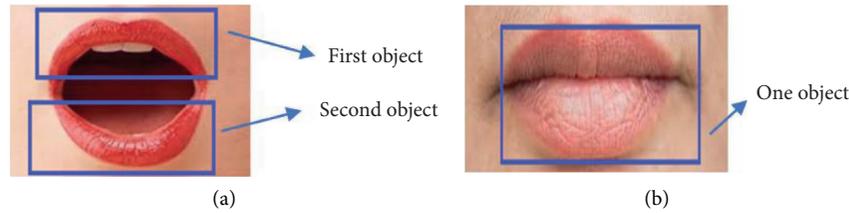


FIGURE 7: Mouth detection. (a) Open mouth. (b) Close mouth.



FIGURE 8: Mouth processing. (a) Open mouth sample. (b) Flat structuring element.

A membership function is used to present a graphical representation of the fuzzy set based on the principle of the fuzzy rule IF-THEN [25, 26]. Drowsiness is a type of fuzzy bodily state which is difficult to quantify objectively.

Therefore, developing a fuzzy model can be an easy way to analyze driver behavior and determine their level of fatigue. In this system, the fuzzy inference system involves three main steps:

Step 5 (fuzzification): define the two crisp input variables which are eye and mouth states with one output which is at the drowsy level, as shown in Figure 9.

Initially, we define the first input variable in which eye state is according to the ratio of eye closure as illustrated in Table 2. Similarly, for mouth state, we define the input variable according to the ratio of open, half open, and close. A drowsy level is defined by three terms: low, medium, and high. The inputs include eye state (open-half and open-close) and mouth state (open-half and open-close) values. The linguistic variables and terms for the inputs and outputs are given in Table 3.

Furthermore, researchers use fuzzy logic to describe the fuzziness of the variables by representing the values from 0 to 1 using a set of input membership functions of both inputs and the degree of truth of each element in a fuzzy set is classified as presented in Figures 10(a) and 10(b).

Step 6 (inference system): the if-then rules will be defined and evaluated in rule editor part of the fuzzy inference system. From the video frame sequence, the system can monitor the degree of the open/close eye and mouth frame. Then, the system uses knowledge rule-based fuzzy inference system and combines the two variables, which are eye and mouth state values. This process will help the system to classify the feeling of the driver, whether it has low, medium, or high drowsy, and proposed fuzzy rules based on the ratio

and threshold of open and close eyes. This is a set of rules for the knowledge base using IF-THEN logic defining in rule editor, as presented in Table 4.

Step 7 (de-fuzzification)

It converts the fuzzy input to the crisp value by using the output membership functions to determine the drowsy output level, as shown in Table 4. Use the rule viewer of MATLAB to interpret the entire fuzzy inference process at once and show the diagram of membership functions which influences the overall result. Figure 11 shows the rules' viewers of nine rows of plots rule and three columns of the input and output variables.

The first two columns of plots in yellow show the membership functions of the two input variables, and the last column represents the output membership function. The last plot in the last column shows the aggregate weighted decision for the given fuzzy rule system, which depends on the input values for the system.

Figure 12 presents the surface viewer of MATLAB three-dimensional curve of the entire fuzzy inference system. It is equipped with X (eye state input), Y (mouth state input), and Z (drowsy level output) to allow the calculation time reasonable for complex problems. Also, the IF-THEN rules appear in 9 colors as displayed in the diagram of the fuzzy module; if the eye state is closed at 0.3 degrees and the mouth state is opened at 0.9 degrees, then the drowsy level will be high.

4. Experimental Results

4.1. Offline Data Analysis and Training for Eye and Mouth.

One of the principal aims of the project was to conduct a nonintrusive real-time drowsiness system based on the webcam video analysis. The system is focused on different visual cues of the driver to collect data and detect the driver's

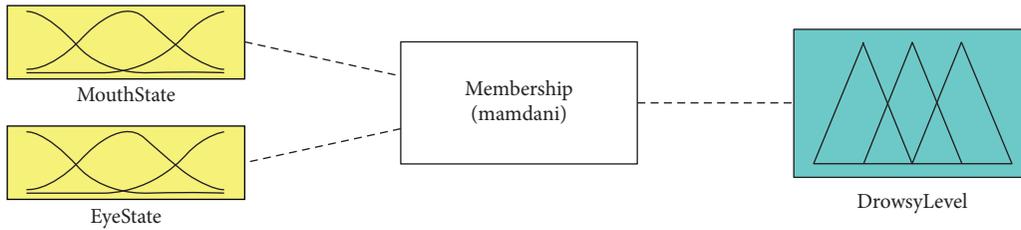


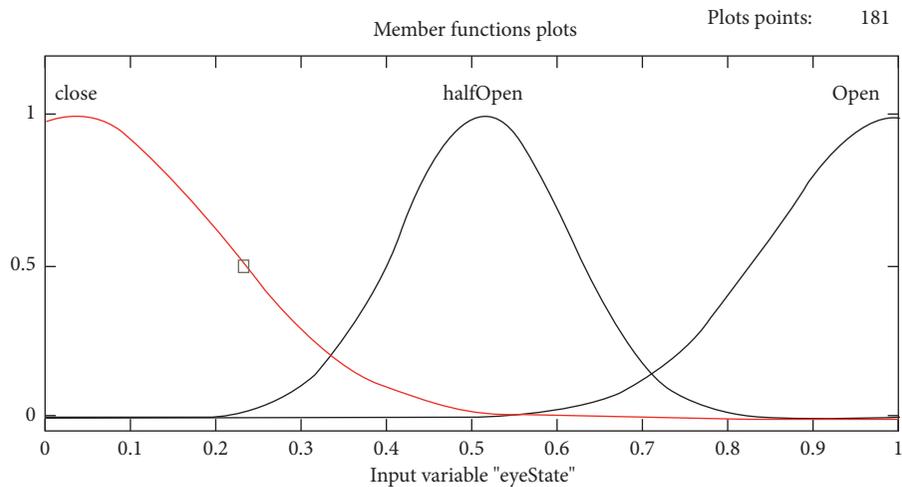
FIGURE 9: FIS editor showing the main simulation screen for driver drowsy.

TABLE 2: Eye closure.

			
Ratio of eye closure	100–74%	73–35%	34–0%
Area of black pixels	Small	Medium	Large
Drowsy level	Low	Medium	High

TABLE 3: Linguistic variables and terms for the inputs and output.

Variable	Type	Fuzzy set	Membership function
Eye state	Input	Open, half open, and close	Trapezoidal
Moth state	Input	Open, half open, and close	Trapezoidal
Drowsy level	Output	Low, medium, and high	Trapezoidal



(a)

FIGURE 10: Continued.

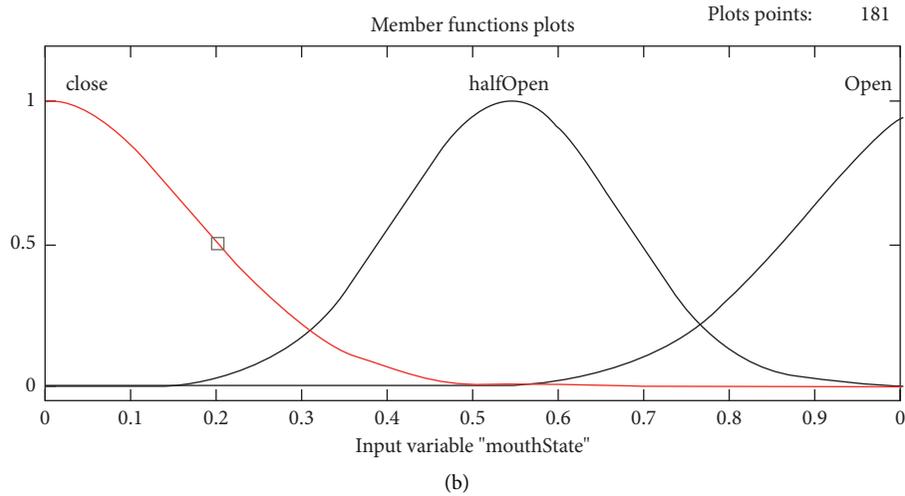


FIGURE 10: FIS editor showing the MF: (a) for eye state; (b) for mouth state.

TABLE 4: Rule editor.

Rule no.	Rule
Rule 1	IF eye state is open OR mouth state is close, THEN the drowsy level is low
Rule 2	IF eye state is close OR mouth state is open, THEN the drowsy level is high
Rule 3	IF eye state is half open OR mouth state is half open, THEN the drowsy level is high
Rule 4	IF eye state is open AND mouth state is half open, THEN the drowsy level is low
Rule 5	IF eye state is close AND mouth state is close, THEN the drowsy level is high
Rule 6	IF eye state is close AND mouth state is half open, THEN the drowsy level is high
Rule 7	IF eye state is half open AND mouth state is open, THEN the drowsy level is high
Rule 8	IF eye state is half open AND mouth state is close, THEN the drowsy level is medium
Rule 9	IF eye state is open AND mouth state is open, THEN the drowsy level is medium

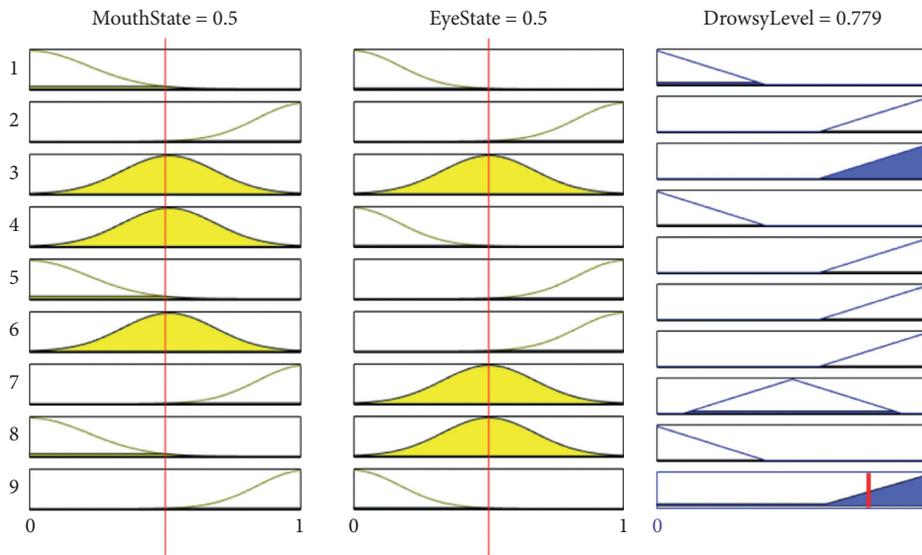


FIGURE 11: Rule's viewer.

level of alertness. The systems focused on two parameters used to identify the driver's status: eye status and mouth status information. Firstly, we conducted an offline test on the MRL eye dataset [27] of the open and closed eye, as presented in Figure 13. Similarly, for mouth training, the

OuluVS2 dataset contains different people's pictures with different ages and gender and is captured from different angles. The main objective of using the dataset is to figure out the threshold of open and closed eyes and to test the system's capability for detecting the eye state on different people or

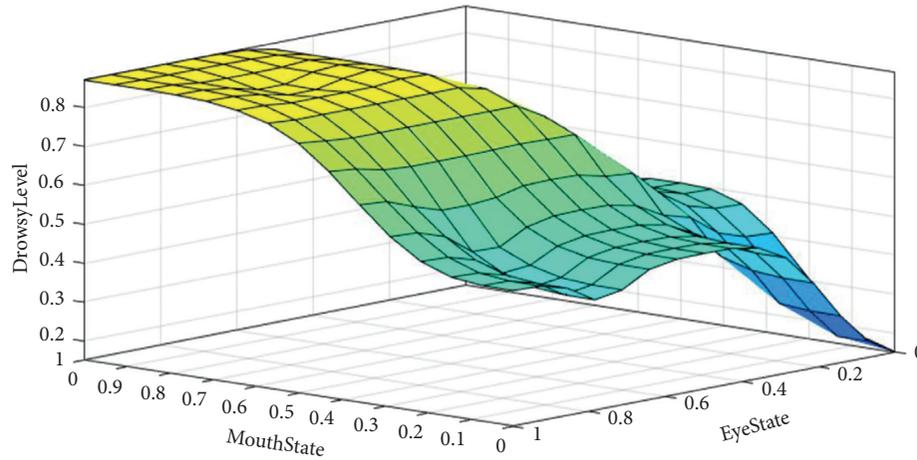


FIGURE 12: Surface viewer of fuzzy inference system.

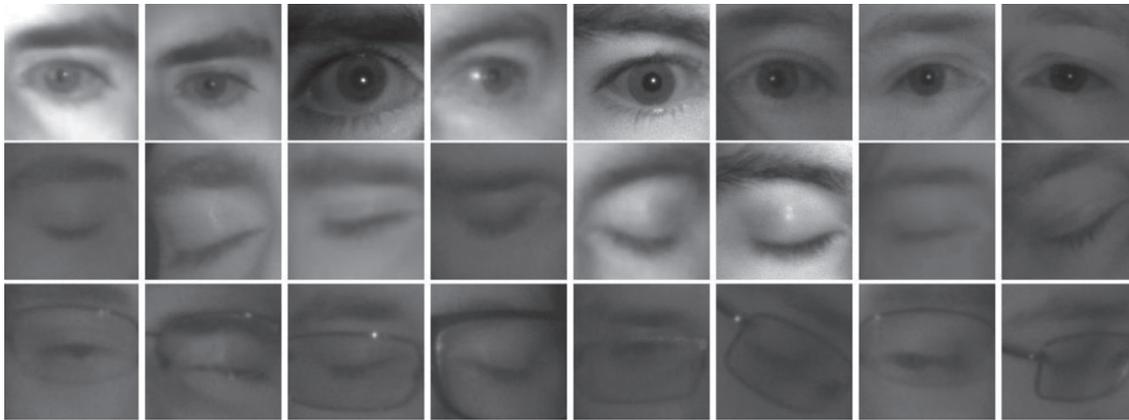


FIGURE 13: MRL eye dataset samples.

not. By testing different eye samples, we figured out that if the eye image ratio exceeds the threshold, the eye will be considered open, otherwise closed. Also, the results show that the program can detect all eyes in most pictures except the pictures with high reflection and lousy quality.

The system achieves promising results on detection eye status. Hence, it can conclude as results that the system can recognize the eye status successfully for both genders and wearing glasses or not. Besides, the system works effectively in case of offline detection due to the stability of the image. The error rate can be estimated due to the reflection and bad light of the pictures, see Table 5. However, the programs achieve 100% accuracy on offline detection mouth status. According to our results, the system can easily detect the mouth state from the pictures because no external obstacles affect how the system works, such as the sudden move, the distance between driver and camera, camera resolution, and background light.

Eye state visualization by using a histogram chart for open and close eye. MatlabR2016a environment provides an `imhist` function to represent an image histogram chart to show the distribution of information density in the grayscale image. It demonstrates the number of times where the density value in the image occurs. The digital image is a

grayscale image that includes some pixels with one scalar value called intensity. Therefore, the number of intensity levels refers to encoded images with $28 = 256$ intensity values, where 0 displays the black pixel and 255 display the white. The cropped of the open and close eye image is converted to a binary image by using an adaptive threshold to detect and compute the black and white pixel of the picture. Then, identify the steel object to create a flat structuring element essential for dilation and erosion in the binary images to eliminate false pixels and compute valid pixels. First, the system will crop open eye region from the sample face, as displayed in Figure 14(a). Then, to visualize the open eye in the histogram chart, the system needs to binarize the image to white and black pixels. The larger eye is opened based on the pupil; the darker pixels are found in the picture, as shown in Figure 14(b). By using binary image and `imhist` function, the histogram chart is presented in Figure 14(c).

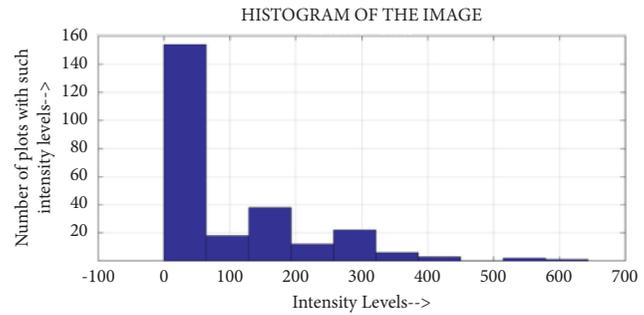
The histogram chart shows the distribution of the information density of the binary image and the threshold value, by calculating the ratio of black pixels (pupil) to white pixels (skin). Therefore, the system can determine whether the eye is close or open. According to the above histogram chart, the high distribution of the open eye image density is

TABLE 5: Offline MRL eye dataset analysis.

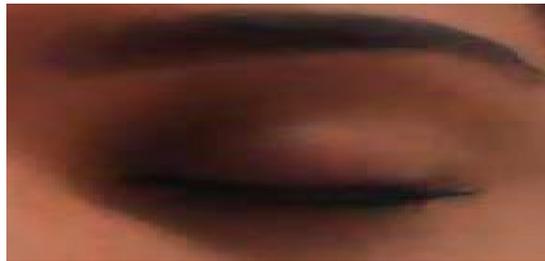
Subject	Gender	Glasses	Close	Open	No reflection	Low reflection	High reflection	Bad light	Good light	Accuracy of detection close eye (%)	Accuracy of detection open eye (%)
s0001	Male	555	406	361	267	30	470	523	244	82	80.7
s0029	Male	0	177	223	390	10	0	163	237	95.9	96.8
s0016	Female	0	455	173	417	183	28	120	508	89.3	90.1
s0036	Female	522	209	409	420	73	125	450	168	85	84



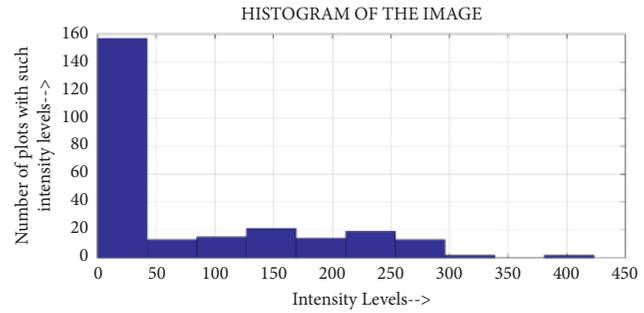
(a)



(b)



(c)



(d)



(e)



(f)

FIGURE 14: Results of implementation. (a) Sample of open eye. (b) Open eye sample flat structuring element. (c) Histogram of open eye. (d) Histogram of close eye. (e) Open eye sample flat structuring element. (f) Sample of close eye.

between 0 and 400, and then, the level of image density is gradually decreased. The histogram shows a peak of the dark pixels at around 155 when the density level is between 0 and 50. Similarly, for a close eye, the system will crop close eye area from the sample face, as displayed in Figure 14(d). Then, convert the cropped image to a binary image to visualize it in the histogram chart, as shown in Figure 14(e). According to the histogram of the close eye, shown density distribution of the close eye is between 0 and 300, and then, the level of image density is gradually decreased. The histogram shows a peak of the dark pixels at around 155 when the density level is between 0 and 50. The number of dark pixels slightly increased increasing to 20 in 150 and 250 intensity level with such intensity levels in Figure 14(f).

4.2. Online Data Analysis and Training for the Eye and the Mouth. In this section, the system focused on the analysis of real-time driver drowsiness detection. We conduct an experiment for training and testing on the 7 YaWDD videos' dataset [27] under different conditions such as gender, age, wearing glasses and, illumination conditions. Consecutively, Table 6 presents 7 video samples of monitoring drivers' behavior, where the acquired videos resulted in a resolution of 1280×720 of 30 frames per second. These videos were tested in 3 different situations: normal driving (without speaking) and speaking and yawning while driving. The experience indicates that the system could detect eye status for people who wear glasses and those who do not.

TABLE 6: Online YAWDD videos' dataset analysis.

7 YAWDD (videos)	Face detection (%)	Eye status (%)	Mouth status (%)	Yawning detection (%)
Detection accuracy	100	94	95	94.5

TABLE 7: Online YAWDD videos' dataset MRL eye image dataset analysis.

	YAWDD (videos)	MRL eye (images)
Total sample number	7	2413
Males	1	2
Females	2	2
Conditions' test	Normal driving (no talking) and talking and yawning while driving	Eye status (open/close) with and without glasses
Detection accuracy	95%	87.875%

On observation of the experiments on eye state detection, it was found that some cases of the eye state cannot be detected. However, these undetected eyes belong to the people who wear glasses. We can estimate the error due to the reflections of background light on the glasses' glass, which negatively affect the camera. Therefore, the eye detector is unable to determine the eye status. The programs achieve 94.56 for online data analysis and training for eye and mouth status, as shown in Table 7 [28].

5. Conclusions

This study proposes a real-time drowsy detection system that monitors the eye and mouth of the driver through driving and issue the suitable alert sound. The system was conducted under different experiments for detecting eye and mouth states with different people and conditions. It achieved high accuracy compared to other works, about 87.875% for eye detection and 100% for mouth detection. Also, the system was tested in real time with other videos recorded in a day and night, with different people wearing and not wearing glasses. The proposed method achieved 94.5% accuracy on real-time detection driver status. It also developed a fuzzy model for a mouth and eye variable input based on defined and evaluated IF-THEN rules to investigate driver fatigue levels. This control system is efficient in determining the driver status and issuing the needed alert. It is easy to modify and update based on new cases and factors. Finally, it visualizes the eye state using a histogram chart to check and examine the difference between open and closed eyes.

The contributions of this work are

- Optimize the current detection method (Viola-Jones) by adding new features that help fast and accurately detect the driver's face and mouth
- Develop a fuzzy logic controller that can determine the driver status quickly and efficiently issued the needed alert
- The proposed approach can work in offline and online systems and embed easily with any framework
- The proposed detecting approach includes several new features that enable it to detect the driver's face

and mouth in different conditions and a light contrast

- The proposed approach effectively detects the driver images and captures the driver image from a video

6. Future Work

Driver drowsy is a major cause of road accidents and economic risk. Using a Webcam tool to detect fatigue is still not efficient enough because the face cannot be detected when the driver moves their head quickly or suddenly. Also, facial features cannot be discovered in the video frame under various lighting conditions or while wearing sunglasses, which gives inaccurate data in the fatigue warning system. Hence, in the future, the system should use infrared camera-sensitive camera to allow the system work robustly in any lighting conditions. The proposed approach cannot detect the driver's mouth because a typically yawning gesture covers the mouth while yawning. Therefore, in the future, the system should include other parameters which are steering wheel angles. The steering wheel angles' sensor is widely used to measure the driver behavior and determine the fatigue level in real time. The system should improve the response time to get accurate results and avoid any wrong alarms of the drowsy detection system.

Data Availability

The offline data are adopted from MRL Eye Dataset, and the online data are adopted from A Yawning Detection Dataset.

Conflicts of Interest

The authors declare no conflicts of interest.

Authors' Contributions

WA. and AA. conceptualized the study; WA., MA., and JHY. developed methodology; WA. and MA. helped with software; WA., MA., JHY., and AA. validated the study; WA., AA., and MA. carried out formal analysis; WA. and JHY. investigated the study; MA. and AA. collected resources; WA. curated the data; WA., AA., MA., and JHY. wrote and prepared the original draft; WA. and JHY. reviewed and

edited the article; WA. and MA. visualized the study; AA. supervised the study; MA. administrated the project. All authors have read and agreed to the published version of the manuscript.

References

- [1] C. Peura, J. A. Kilch, and D. E. Clark, "Evaluating adverse rural crash outcomes using the NHTSA State Data System," *Accident Analysis & Prevention*, vol. 82, pp. 257–262, 2015.
- [2] K. Anjali, A. K. Thampi, A. Vijayaraman, M. F. Francis, N. J. James, and B. K. Rajan, "Real-time nonintrusive monitoring and detection of eye blinking in view of accident prevention due to drowsiness," in *Proceedings of the 2016 International Conference on Circuit, Power and Computing Technologies (ICCPCT)*, pp. 1–6, IEEE, Nagarcoil, India, March 2016.
- [3] S. Kaplan, M. A. Guvensan, A. G. Yavuz, and Y. Karalurt, "Driver behavior analysis for safe driving: a survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 6, pp. 3017–3032, 2015.
- [4] S. Mehta, S. Dadhich, S. Gumber, and A. Jadhav Bhatt, "Real-time driver drowsiness detection system using eye aspect ratio and eye closure ratio," in *Proceedings of the International Conference on Sustainable Computing in Science, Technology and Management (SUSCOM)*, Amity University Rajasthan, Jaipur-India, June 2019.
- [5] A. Madhan Kumar, M. Kabilan, S. Karthikeyan, and S. Sathiyapriya, "Drowsy driving warning and traffic collision information system using iot," *IJARIT*, vol. 5, 2019.
- [6] E. P. Herrera-Granda, J. A. Caraguay-Procel, P. D. Granda-Gudiño et al., "Drowsiness detection in drivers through real-time image processing of the human eye," in *Proceedings of the Asian Conference on Intelligent Information and Database Systems*, pp. 626–637, Springer, Yogyakarta, Indonesia, April 2019.
- [7] A. B. Roig, M. Morales, J. Espinosa, J. Perez, D. Mas, and C. Illueca, "Pupil detection and tracking for analysis of fixational eye micromovements," *Optik*, vol. 123, no. 1, pp. 11–15, 2012.
- [8] P. Shelke, S. Magar, and R. Jawkar, "Real time detection system of driver fatigue," *IJCERT*, vol. 3, 2016.
- [9] T. Azim, M. A. Jaffar, and A. M. Mirza, "Automatic fatigue detection of drivers through pupil detection and yawning analysis," in *Proceedings of the 2009 Fourth International Conference on Innovative Computing, Information and Control (ICICIC)*, pp. 441–445, IEEE, Kaohsiung, Taiwan, December 2009.
- [10] W. Liu, H. Sun, and W. Shen, "Driver fatigue detection through pupil detection and yawing analysis," in *Proceedings of the 2010 International Conference on Bioinformatics and Biomedical Technology*, pp. 404–407, IEEE, Chengdu, China, June 2010.
- [11] T. Azim, M. A. Jaffar, and A. M. Mirza, "Fully automated real time fatigue detection of drivers through fuzzy expert systems," *Applied Soft Computing*, vol. 18, pp. 25–38, 2014.
- [12] J. Berg, G. Neely, U. Wiklund, and U. Landstrom, "Heart rate variability during sedentary work and sleep in normal and sleep-deprived states," *Clinical Physiology and Functional Imaging*, vol. 25, no. 1, pp. 51–57, 2005.
- [13] C. Anitha, M. K. Venkatesha, and B. S. Adiga, "A two fold expert system for yawning detection," *Procedia Computer Science*, vol. 92, pp. 63–71, 2016.
- [14] Y. Kurylyak, F. Lamonaca, and G. Mirabelli, "Detection of the eye blinks for human's fatigue monitoring," in *Proceedings of the 2012 IEEE International Symposium on Medical Measurements and Applications*, pp. 1–4, IEEE, Lausanne, Switzerland, June 2012.
- [15] J. Jo, S. J. Lee, J. Kim, H. G. Jung, and K. R. Park, "Vision-based method for detecting driver drowsiness and distraction in driver monitoring system," *Optical Engineering*, vol. 50, no. 12, Article ID 127202, 2011.
- [16] S. Abtahi, B. Hariri, and S. Shirmohammadi, "Driver drowsiness monitoring based on yawning detection," in *Proceedings of the 2011 IEEE International Instrumentation and Measurement Technology Conference*, pp. 1–4, IEEE, Hangzhou, China, May 2011.
- [17] T. Danisman, I. M. Bilasco, C. Djeraba, and N. Ihaddadene, "Drowsy driver detection system using eye blink patterns," in *Proceedings of the 2010 International Conference on Machine and Web Intelligence*, pp. 230–233, IEEE, Toronto, Canada, August 2010.
- [18] L. M. Bergasa, J. Nuevo, M. A. Sotelo, R. Barea, and M. E. Lopez, "Real-time system for monitoring driver vigilance," *IEEE Transactions on Intelligent Transportation Systems*, vol. 7, no. 1, pp. 63–77, 2006.
- [19] W. Tipprasert, T. Charoenpong, C. Chianrabutra, and C. Sukjamsri, "A method of driver's eyes closure and yawning detection for drowsiness analysis by infrared camera," in *Proceedings of the 2019 First International Symposium on Instrumentation, Control, Artificial Intelligence, and Robotics (ICA-SYMP)*, pp. 61–64, IEEE, Bangkok, Thailand, 2019, January.
- [20] A. R. Al-sudani, "Yawn based driver fatigue level prediction," *Proceedings of 35th International Confer*, vol. 69, pp. 372–382, 2020.
- [21] Z. Jie, M. Mahmoud, Q. Stafford-Fraser, P. Robinson, E. Dias, and L. Skrypchuk, "Analysis of yawning behaviour in spontaneous expressions of drowsy drivers," in *Proceedings of the 2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*, pp. 571–576, IEEE, Xi'an, China, 2018, May.
- [22] R. Boda, M. J. P. Priyadarsini, and J. Pemeena, "Face detection and tracking using KLT and Viola Jones," *ARNP Journal of Engineering and Applied Sciences*, vol. 11, pp. 13472–13476, 2016.
- [23] P. Gejguš and M. Šperka, "Face tracking for expressions simulations," in *Proceedings of the International Conference on Computer Systems and Technologies*, June 2003.
- [24] R. R. Yager, "Expert systems using fuzzy logic," in *An Introduction to Fuzzy Logic Applications in Intelligent Systems*, pp. 27–44, Springer, Berlin, Germany, 1992.
- [25] O. V. Komogortsev and J. I. Khan, "Kalman filtering in the design of eye-gaze-guided computer interfaces," in *Proceedings of the International Conference on Human-Computer Interaction*, pp. 679–689, Springer, Beijing, China, July 2007.
- [26] J. Yousif and D. Saini, "Fuzzy and mathematical effort estimation models for web applications," *Applied computing Journal*, vol. 63, pp. 10–24, 2021.
- [27] M. M. Rahman, M. S. Islam, M. K. A. Jannat et al., "EyeNet: an improved eye states classification system using convolutional neural network," in *Proceedings of the 2020 22nd International Conference on Advanced Communication Technology (ICACT)*, pp. 84–90, IEEE, PyeongChang, Korea, February 2020.
- [28] S. Abtahi, M. Omidyeganeh, S. Shirmohammadi, and B. Y.D. D Hariri, "A yawning detection dataset," in *Proceedings of the 5th AC Multimedia Systems Conference*, pp. 24–28, Singapore, March 2014.