Delta University Scientific Journal
Journal home page: https://dusj.journals.ekb.eg

Deep Learning-Based Driver Drowsiness Detection Using Facial Expression Analysis

Tarek M. Hassan[1], Walaa Yahia Ibrahim[2], Hesham A. Ali[3,4], Hanaa Salem Marie[5]

[1]Faculty of Artificial Intelligence, Delta University for Science and Technology, Gamasa 35712, Egypt, Email: tarek.hassan@deltauniv.edu.eg
[2]Faculty Artificial Intelligence, Delta University for Science and Technology, Gamasa 35712, Egypt.
[3]Department of Computer Engineering and Control Systems, Faculty of Engineering, Mansoura University, Egypt, Email: H_arafat_ali@mans.edu.eg
[4]Faculty Artificial Intelligence, Delta University for Science and Technology, Gamasa 35712, Egypt
[5]Faculty of Artificial Intelligence, Delta University for Science and Technology, Gamasa 35712, Egypt, Email: Hana.salem@deltauniv.edu.eg

Correspondence: Department of Communications and Computer Engineering, Faculty of Engineering, Delta University for Science and Technology, Gamasa, 35712, Egypt Mobile: +201024735025; Email: tarek.hassan@deltauniv.edu.eg (Tarek M. Hassan)

ABSTRACT
Driver Drowsiness Detection is technology that prevent the driver from accidents caused by driver fatigue that lead to fall asleep while driving. There are many factors that cause road accidents but driver drowsiness is the most contributor one to make a wide and deadly accidents. There are many features that make sure that the driver falls asleep like mouth opening, close eyes, yawn and head tilt. The objective of this paper is to introduce a Driver Drowsiness Detection alarming system based on a CNN (Convolutional Neural Network) for accurate detection and OpenCV to use camera for video and capture image.
In this paper, an algorithm is proposed to detect the Driver Drowsiness through eyes, which determine if the eye is closed or open. When the eyes are closed for period of time, the alarm turned on.
The experimental results show that the system achieves high accuracy, reducing the overall number of accidents on the streets. For Real-time video, the proposed method has achieved 97% of accuracy.

Keywords: *Facial Extraction, Drowsiness, Machine Learning, Eye extraction, Face Detection, CNN*

1. Introduction

Based on 2017 police and hospital reports, the National Highway Traffic Safety Administration (NHTSA) identified 91,000 car accidents as being caused by drowsy drivers around the world as shown in Figure 1. The statewide police reported crashes caused by driver drowsiness as being caused by drowsiness's drivers around the America as shown in Figure 2.

Every year, many people are injured or died because of fatigue or sleep while driving caused accidents. Therefore, driver drowsiness detection is an effective research due to its wide practical applicability. A basic drowsiness detection system consists of three modules: an acquisition system, a processing system, and a warning system. The video of the driver's face is captured in the acquisition system and transmitted to the processing block, where it is processed to detect drowsiness. The warning system sends a warning or alert to the driver if drowsiness is detected.

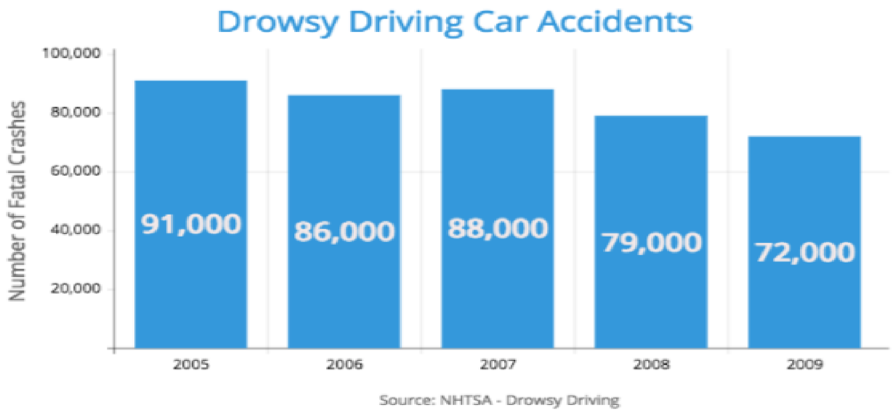


Figure1: Number of Drowsy driving crashes in the world from 2005-2009

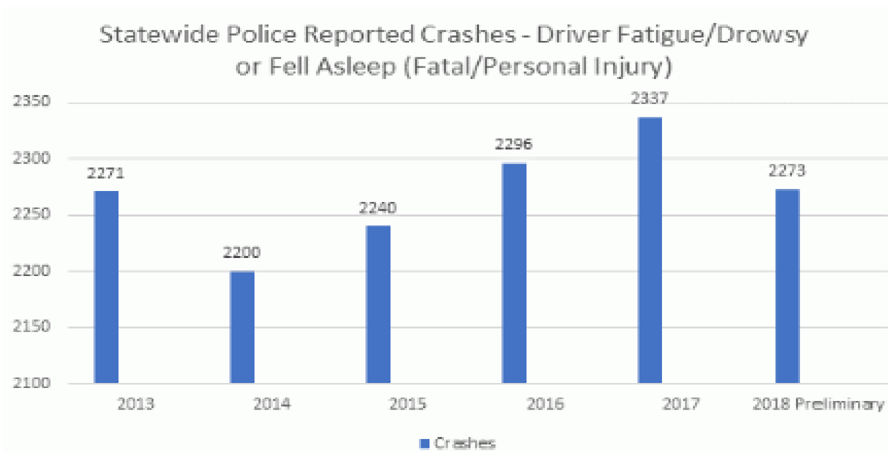


Figure 2: Number of crashes in America from 2013-2017

Table 1 shows the features of driving status.

Table 1: Features of driving status

Driving Status	Fatigue label	Features
Awake	1	Rich facial emotions are used, and the head remains upright. aware of their surroundings. Widely opened eyes with rapid blinking and vigorous eye movement
Drowsy	2	There is less focus on the outer world. In addition to head shakes and winks, drivers also scratch their faces, breathe deeply, yawn, and swallow. Eyes tend to shut, blink slowly, and have less activity in the eyeballs.
Very drowsy	3	Eyes continue to close while eyelids get heavier. Eyes are closing for a longer time. Drivers may nap, nod, slant their heads, and then lose the ability to drive.

The major contributions of this research paper are:

- Introduce a smart alert approach for intelligent vehicles that can prevent drowsy driving automatically.
- A driver drowsiness detection system based on Convolution Neural Network (CNN).
- Developing a real-time drowsiness detection system to monitor and prevent of fatigue with high accuracy.

The rest of this paper is organized as follows. Section 2 introduces the background and related work to the drowsiness detection system, followed by a comparison between the different approaches introduced recently. The proposed architecture and the different module that constitutes the framework is introduced in section 3. The dataset and experimental results are introduced in section 4. Finally, the conclusion of the paper is discussed in section 5

2. Related Work

Figure 3 shows the Different approaches for Driver Drowsiness Detection.

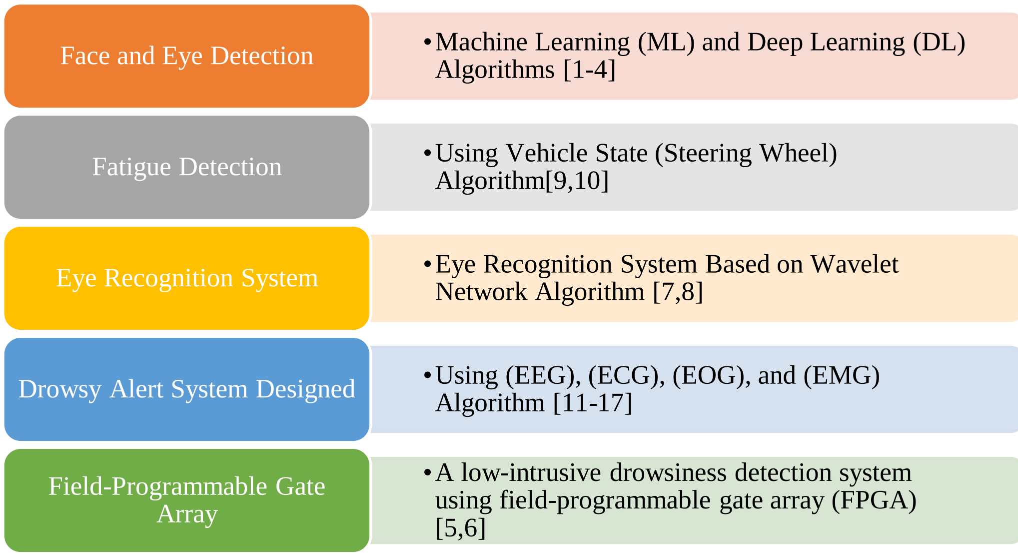


Figure 3: Different approaches for Driver Drowsiness Detection

2.1. Face and Eye Detection by Machine Learning (ML) and Deep Learning (DL) Algorithms:

CNN technique of the ML algorithm is proposed in [1] to detect microsleep and drowsiness. The detection of the driver's facial landmarks can be accomplished via a camera that is then passed to this CNN algorithm to properly identify drowsiness. Where the experimental classification of eye detection is prepared through numerous data sets like without glasses and with glasses in a day or night vision. So, it works for effective drowsiness detection with high precision with Android modules. The algorithm of Deep CNN was used to detect eye blink and its state recognition as provided by [2]. While in [3] developed an algorithm of LSTM and Recurrent Neural Networks (RNN) to categorize driver's behaviors through sensors. Other authors in [4] The RNN Algorithm examined the driver's habits. To prevent roadside accidents, it specifically focuses on building real-time fatigue-detecting systems. A variety of drivers' faces are created by this system, which uses multilayered 3D CNN models to identify sleepy drivers and has a 92 percent acceptance rate [20] proposed a drowsiness detection system that involves the Viola-Jones algorithm for detecting eye and face regions. The system detects the drowsiness state of the driver and sends an alarm to alert the driver. This system was tested it has shown 82% and 72.8% for indoor and outdoor environments

2.2. FPGA-Based Drowsiness Detection System:

Field-programmable gate array (FPGA) technology has been used to create a less intrusive sleepiness monitoring system [5]. This technology concentrates on eyeballs with bright pupils that are discovered by an inbuilt IR sensor light source. Due to this visual effect, the retinas could be detected up to 90% of the time, which helped locate drivers' eyes for assessing tiredness over several frames in order to prevent major accidents. Another study [6] used the Cyclone II FPGA to develop a real-time system for tracking human eyes.

2.3. Eye Recognition System Based on Wavelet Network Algorithm:

A method for a drowsy warning system employing wavelet networking is presented [7]. With the aid of classifying algorithms like Wavelet Network Classifier (WNC) that rely on Fast Wavelet Transform (FWT), which specifically results in a binary way decision (aware or not), that network follows the movement of the eyes. Heart rate and ECG are the physiological factors that are frequently retrieved for tiredness detection using wavelet transformation and regression. [8]. This theory was applied to the classification of heart rate data using a wavelet network, which can be used as a general sleepiness alarm system.

2.4. Fatigue Detection Using Vehicle State (Steering Wheel) Algorithm:

The authors in [9] introduce a neuro-fuzzy system with a support vector machine (SVM), particle swarm optimization method, and a non-interfering drowsiness detection system based on car steering data. This system addressed the issue of drowsiness utilizing the steering wheel algorithm which has been built by updating the concept and modification. It is primarily based on image formed steering movements or pictorial-based steering movements and the CNN algorithm for accurate drowsiness classification, which can also lower the rate of false drowsy detection. Another solution was introduced in [10], it observed vehicle-based methods. In the method, using sensors mounted to the car, they captured several types of sleepiness data, including yaw angles and steering wheel angles. On-time series data, the features derived from yaw and steering wheel angles are examined, and approximatively entropy features are computed. The accuracy of the back-propagation neural network classifier used in this paper's method to determine the driver's level of drowsiness was 87.21%. The system determines that the motorist is awake, sleepy, and extremely sleepy. Behavioral methods have been introduced in recent years to overcome the problems caused by physiological and vehicle-based methods. Behavioral methods are more reliable than vehicle-based methods because they focus on the driver’s facial expression rather than the vehicle’s behavior. On the other hand, Physiological methods produce highly accurate results but are not widely used due to their complicated nature

2.5. Drowsy Alert System Designed Using Electroencephalography (EEG), Electrocardiography (ECG), Electrooculogram (EOG), and Electromyogram (EMG) Algorithm:

A drowsiness detection system using the EEG approach was developed in [11] and includes several different components, including the AlexNet method, VGGNet method, and wavelet transform algorithm. Using the brain indication signal (EEG), camera, and sensors that are activated with the use of machine learning, this technique efficiently assesses the level of sleepiness in order to alert sleepy drivers. A method to detect sleepiness using the Heart Rate Variability (HRV) signal, which is collected using EEG sensors, was proposed in [12]. In order to build a fatigue alarm system that is also embedded with an Arduino controller board with Nearest Neighbors (KNN) classifier to improve the percentage of accuracy, [13] established an intrusive approach for tracking eyeball movement using EOG methodology. Table 2 summarizes the most important direction that employs the recent technology and advanced algorithm in AI to solve the problems.

Table 2: Comparison of the most recent technology and AI Algorithm

Approach	Technologies and Algorithms	Efficiency	Advantages	Disadvantages
Face and Eye Detection	(ML) (DL)	92%	High accuracy Speed	require large datasets for training, which can be time-consuming
FPGA-Based Drowsiness Detection	Field-programmable gate array (FPGA) technology	93%	Low power consumption Accuracy	expensive to develop, inflexible, and complex to design
Eye Recognition System	Wavelet Network Algorithm	94%	High accuracy Fast processing	require large datasets for training, time-consuming.
Fatigue Detection	Vehicle State (Steering Wheel) Algorithm	89%	Real-time detection	Cost, Limited functionality
Biometrics	(EEG), (ECG), (EOG), (EMG)	94%	High accuracy	Need for specialized hardware, Limited usability

3. The Proposed System Architecture

3.1. Architecture:

Figure 4 shows the proposed architecture for driver drowsiness detection using facial expression analysis. From this figure, the camera records the driver's face while driving and record a video stream. Then, the video is analyzed to look for signs of exhaustion and drowsiness as shown in table [1] and measures how much drowsiness is there. In the proposed architecture, the camera records a video stream and takes frames or images from the video when detects that the driver sleeps, then the system alerts. The system works on the simple principle of a counter, wherein a limit or threshold is set. If the driver is found sleepy (the eyes are closed), then an alarm will be turned on. The counter will increase based on the time for which the driver's eyes are closed. As a result, the driver will be alerted and can take control of the vehicle

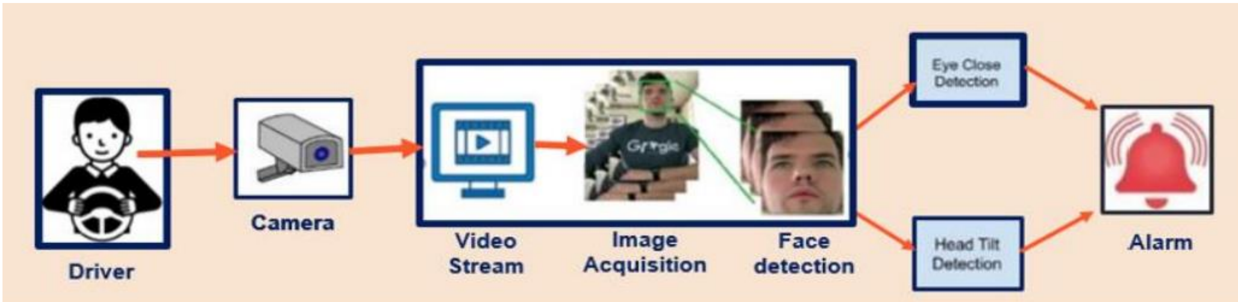


Figure 4: Proposed Architecture for Driver Drowsiness Detection using Facial Expression Analysis

3.2. Framework:

Figure 5 shows the proposed framework of the proposed system which consists of the following steps:

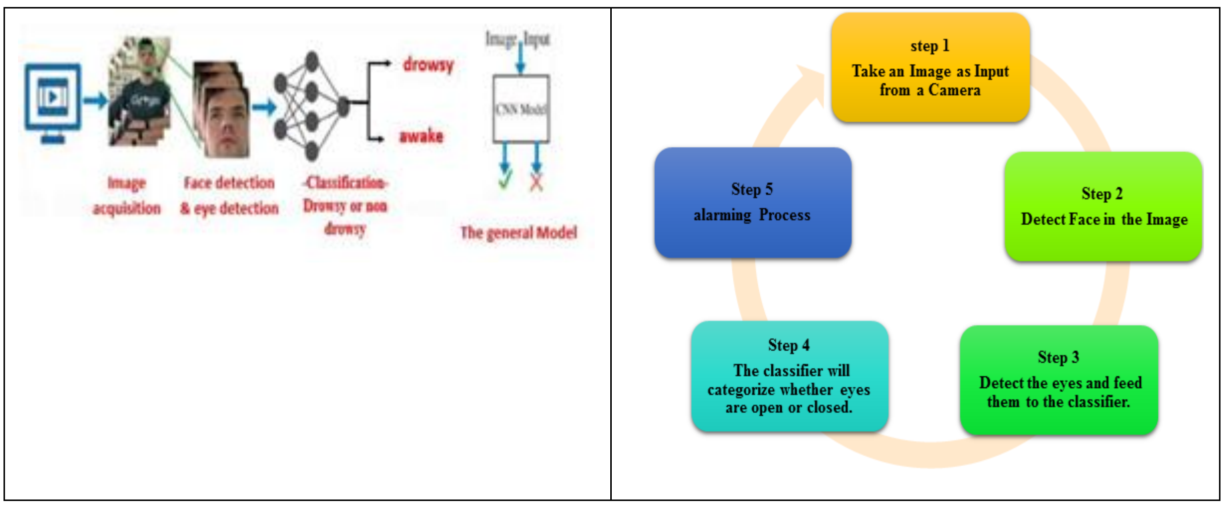


Figure 5 The Proposed Framework

Step 1: Take an Image as Input from a Camera

To access the camera, OpenCV technique is used, which read each frame and use it in next step.

Step 2: Detect Face in the Image

Before detect faces in images, the open cv algorithm and the Haar-casade classifier accepts the image in particular form it's grayscale image, therefore the first step after take images from camera is convert it into gray scale images, Haar-Cascade Classifier is used to recognize faces. Next, face recognition is performed and returns a list of detections with the object's bounding box's height and width as well as its x and y coordinates as shown in Figure 6.

Figure 6: Face Detection

Step 3: Detect the eyes and feed them to the classifier:

The cascade classifier is set for the left and right eyes and the same procedure which used in face is used to eyes. After which the eye's picture is extracted from the frame as shown in figure 7, it is then passed into the proposed CNN model.

Figure 7: Eye detection from the face

Step 4: The classifier will categorize whether eyes are open or closed:

The proposed CNN model is shown in figure 8, which classify the eyes and determines whether the eyes are open or closed.
The model used is built with Keras using Convolutional Neural Networks (CNN). A convolutional neural network is a special type of deep neural network which performs extremely well for image classification purposes. A CNN basically consists of an input layer, an output layer and a hidden layer which can have multiple numbers of layers. A convolution operation is performed on these layers using a filter that performs 2D matrix multiplication on the layer and filter.

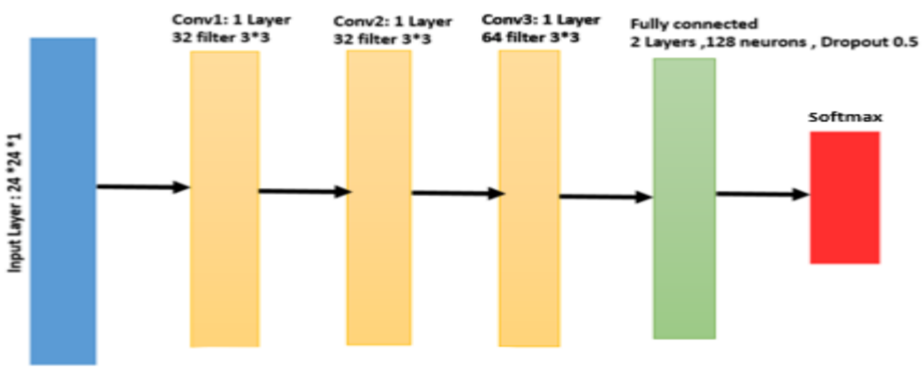


Figure 8 shown the CNN model, layers and its settings.

The CNN model architecture shown in Figure 8 consists of the following layers:
- Convolutional layer: 32 filter, kernel size 3
- Convolutional layer:32 filter, kernel size 3
- Convolutional layer :64 filter, kernel size 3
- Fully connected layer: 128 nodes

The reason for increasing the number of filters in deeper layers is to allow the network to learn more complex and abstract features, which may require more filters to accurately capture.
To feed our image into the model, certain operations are performed because the model needs the correct dimensions to start with. First, the image is converted from RGB into grayscale. Then, the image is resized to 24*24 pixels and used as the input of the model.
Note that, after step 3 (eye detection), we can use another method like EAR (Eye Aspect Ratio) when detect the eyes in image, then use land mark library to detect the points of eyes as shown in figure 9.

Figure 9: The landmark library applied on the eyes

And then apply an EAR equation:

$$\mathbf{EAR} = \frac{|d1-d3|+|d2-d4|}{2|d0-d5|} \qquad \textbf{(1)}$$

Based on equation 1, the eye aspect ratio is calculated then, if the value more than the threshold (can be any number, 0.5, 1, 3, 0.7, etc….), it states that eyes are open, otherwise, it states that eyes are closed.
Figure 10 shows the conversion of images from RGB to grayscale and the detection of face and eyes.

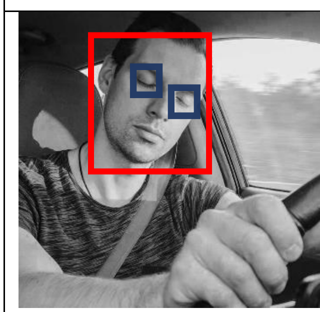
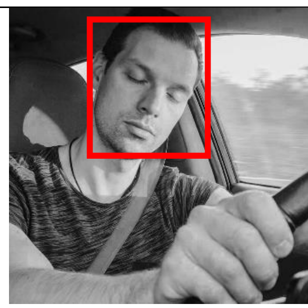
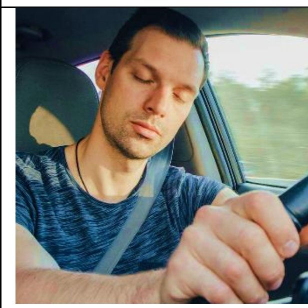
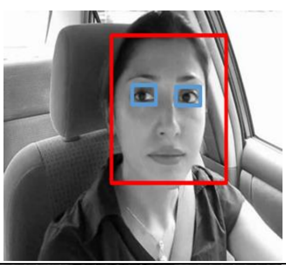
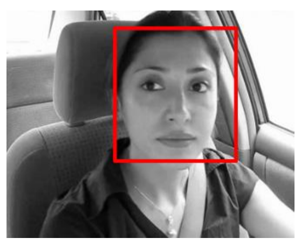
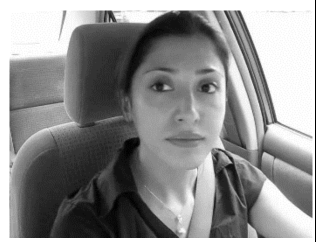
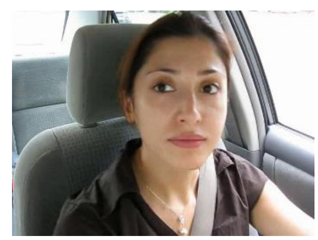
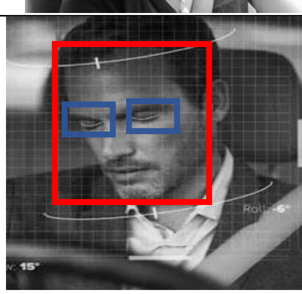
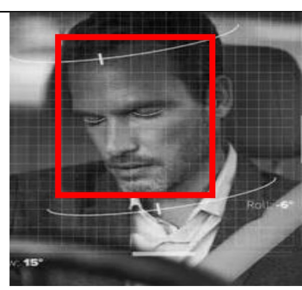
Eye detection	Image detection	Gray scale frame	Color Frame
			
			
			

Figure 10: Steps for Face detection and eye detection

4. Experimental Results and Analysis

The dataset used for the CNN model training system is the yawn eye dataset, which consists of 3000 images. Figure 11 shows samples of these images [22]. It is a free and open-source dataset available on Kaggle. The entire dataset has been divided into 2 parts; each part has 4 different feature values (yawn- non yawn- closed eyes-opened eyes) that will be taken into consideration while evaluation.

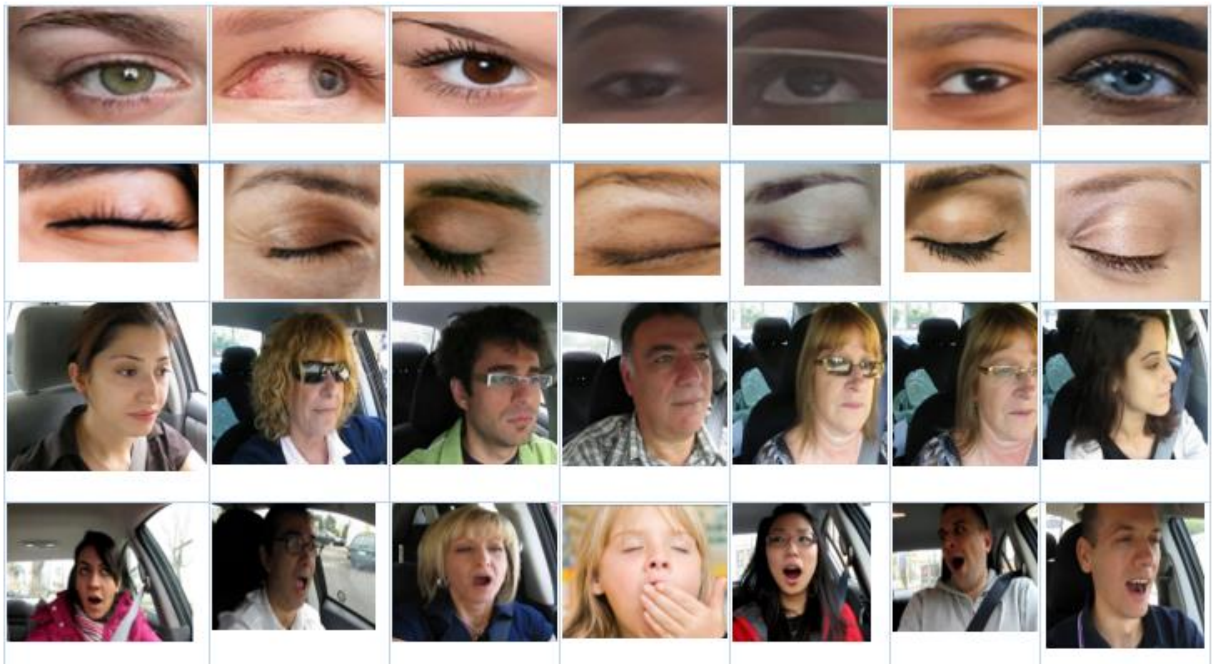


Figure 11: Samples of the Image within the used dataset

4.1. Performance evaluation parameters

Recent studies have used different performance metrics to analyze the model and provide information about classification performance. Four primary keys are used to test the classifier, true positive (Tp), true negative (Tn), false positive (Fp), and false negative (Fn) values. So, based on the four outcome values, the performance of the model is calculated using accuracy (ACC), sensitivity (Recall), specificity (SPC), precision (PPV), and F1-score.

Accuracy, as depicted in eq. (2), is the ratio of the number correctly predicted to the total predicted number.

$$\textbf{Accuracy}\ (\mathrm{ACC}) = \frac{\mathrm{Tp} + \mathrm{Tn}}{\mathrm{Tp} + \mathrm{Tn} + \mathrm{Fp} + \mathrm{Fn}} \qquad (2)$$

Sensitivity (Recall), as shown in eq. (3) is the number of samples predicted as positive from the total number of samples positive, also known as the true positive rate.

$$\textbf{Sensitivity}(\mathrm{Recall}) = \frac{\mathrm{Tp}}{\mathrm{Tp} + \mathrm{Fn}} \qquad (3)$$

Specificity (SPC, TNR) for the true negative rate, as given in the equation. (4) is the number of samples predicted as negative from the total number of samples negative.

$$\textbf{Specificity}\ (\mathrm{SPC}) = \frac{\mathrm{Tn}}{\mathrm{Tn} + \mathrm{Fp}} \qquad (4)$$

Precision, as in Equation (5), which is called positive predictive value, represents the number of samples predicted as positive from the total number of samples predicted as positive.

$$\textbf{Precision}\ (\mathrm{PPV}) = \frac{\mathrm{Tp}}{\mathrm{Tp}+\mathrm{Fp}} \quad (5)$$

The harmonic means of precision and recall, known as the F1-score, is shown in the equation. (6).

$$\textbf{F1}-\textbf{score} = \frac{2*\mathrm{Tp}}{2*\mathrm{Tp}+\mathrm{Fp}+\mathrm{Fn}} \quad (6)$$

4.2. Results and Setting

Table 3 shows the training results.

Table 3: Training result of the experiment

No. of epochs	Dropout	Batch size	Loss	Accuracy	Precision	Recall	AUC	F1	Specificity
10	without	16	0.0118	97.88576	0.980851	0.981848	0.985901	0.982845	0.979499
		32	0.0081	98.0527	0.982524	0.983523	0.986	0.984521	0.980288
		64	0.0203	97.88576	0.980851	0.981651	0.986	0.982746	0.969629
		128	0.1420	94.33092	0.943951	0.945896	0.977323	0.946363	0.882181
	0.2	16	0.0363	96.94304	0.971208	0.972885	0.984817	0.973478	0.968839
		32	0.0128	97.86612	0.980654	0.981848	0.985901	0.982746	0.97792
		64	0.0291	97.50278	0.976325	0.97781	0.985704	0.978506	0.965483
		128	0.0619	96.64844	0.968846	0.96993	0.984324	0.970914	0.93992
	0.5	16	0.0279	97.25728	0.974554	0.975544	0.985113	0.976534	0.97032
		32	0.0166	97.73846	0.979375	0.980666	0.985901	0.981464	0.975353
		64	0.0375	97.25728	0.973471	0.975643	0.985211	0.976041	0.959561
		128	0.1033	95.39148	0.95635	0.95417	0.980084	0.956716	0.916528
15	without	16	6.5023e-05	98.2	0.984	0.985	0.986	0.986	0.986901
		32	2.4942e-04	98.2	0.984	0.985	0.986	0.986	0.986704
		64	0.0043	98.2	0.984	0.985	0.986	0.986	0.982855
		128	0.0035	98.2	0.984	0.985	0.986	0.986	0.983644
	0.2	16	1.8109e-04	98.2	0.984	0.985	0.986	0.986	0.986803
		32	0.0069	97.98396	0.981835	0.982833	0.985704	0.983831	0.982657
		64	0.0098	97.98396	0.981146	0.98303	0.986	0.983634	0.979301
		128	0.0182	97.88576	0.981146	0.981356	0.985901	0.982746	0.97259
	0.5	16	0.0304	97.18854	0.973176	0.974559	0.985408	0.975351	0.970616
		32	0.0394	96.89394	0.971208	0.972195	0.984817	0.973182	0.964398
		64	0.0204	97.57152	0.977801	0.978499	0.985803	0.979591	0.973281
		128	0.0397	97.04124	0.974357	0.972294	0.985211	0.97476	0.958871
20	Without	16	4.0351e-05	98.2	0.984	0.985	0.986	0.986	0.987
		32	1.1094e-04	98.2	0.984	0.985	0.986	0.986	0.986901
		64	2.4002e-04	98.2	0.984	0.985	0.986	0.986	0.986803
		128	0.0096	98.1509	0.983508	0.984508	0.986	0.985507	0.978117
	0.2	16	6.1363e-05	98.2	0.984	0.985	0.986	0.986	0.986901
		32	2.2789e-04	98.2	0.984	0.985	0.986	0.986	0.986803
		64	8.7885e-04	98.2	0.984	0.985	0.986	0.986	0.98621
		128	0.0039	98.2	0.984	0.985	0.986	0.986	0.983052
	0.5	16	0.0014	98.13126	0.983311	0.984311	0.986	0.98531	0.985914
		32	0.0146	97.71882	0.979178	0.980174	0.985606	0.981169	0.978611
		64	0.0040	98.1018	0.983016	0.984508	0.986	0.98531	0.983743
		128	0.0051	98.1509	0.983508	0.984508	0.986	0.985507	0.982065

Table 4 shows the performance metric of the experiment with the testing dataset. The best testing accuracy is achieved at batch sizes of 32 without applying dropout, and the number of epochs is 20. The best average accuracy of the testing result was achieved at 16 batch size and without applying dropout, with 0.2 ratios and 32 batch size, and with 0.5 ratios and 32 batch size, with 98.15, 98.1, and 98.01, respectively.

Table 4: Testing result of the experiment

No. of epochs	**Dropout**	**Batch size**	**Loss**	**Accuracy**	**Precision**	**Recall**	**AUC**	**F1**	**Specificity**
10	without	16	0.001964	0.001964	98.36064	0.984606	0.982705	0.98599	0.986605
		32	0.00435	0.00435	98.32226	0.984222	0.985221	0.985783	0.98622
		64	0.011568	0.011568	98.34096	0.984419	0.985418	0.985901	0.986418
		128	0.111428	0.111428	95.12623	0.952406	0.952898	0.980932	0.954093
	0.2	16	0.031522	0.031522	97.48882	0.975711	0.976889	0.985191	0.977781
		32	0.006717	0.006717	98.20615	0.98306	0.984058	0.985773	0.985056
		64	0.008435	0.008435	98.22485	0.983266	0.984452	0.98598	0.985362
		128	0.039938	0.039938	97.72183	0.97846	0.979828	0.985448	0.980634
	0.5	16	0.04146	0.04146	97.06274	0.971634	0.97262	0.9846	0.973606
		32	0.004822	0.004822	98.34096	0.984419	0.985418	0.98599	0.986418
		64	0.04636	0.04636	96.26866	0.961833	0.966438	0.984659	0.965582
		128	0.064891	0.064891	96.24898	0.962936	0.965107	0.98386	0.965523
15	without	16	0.000255	0.000255	98.4	0.985	0.986	0.985901	0.987
		32	0.002121	0.002121	98.34096	0.984419	0.985418	0.98599	0.986418
		64	0.005892	0.005892	98.26421	0.983651	0.984649	0.98599	0.985648
		128	0.006579	0.006579	98.20615	0.983069	0.984067	0.985941	0.985065
	0.2	16	0.005057	0.005057	98.30258	0.984025	0.985024	0.985608	0.986023
		32	0.006383	0.006383	98.16679	0.982675	0.983673	0.985901	0.984671
		64	0.010782	0.010782	98.06544	0.981385	0.982381	0.985901	0.983378
		128	0.011804	0.011804	98.20615	983069.4	0.984067	0.985901	0.985065
	0.5	16	0.006845	0.006845	98.30258	0.984025	0.985024	0.985941	0.986023
		32	0.003437	0.003437	98.30258	0.984025	0.985024	0.985793	0.986023
		64	0.018295	0.018295	97.72104	0.978204	0.978674	0.985773	0.979894
		128	0.088085	0.088085	97.416	0.953884	0.950376	0.981405	0.95357
20	Without	16	0.00041	98.2	0.984	0.985	0.986	0.986	0.987
		32	0.00048	98.2	0.984	0.985	0.986	0.986	0.986901
		64	0.0006	98.2	0.984	0.985	0.986	0.986	0.986803
		128	0.0107	98.1509	0.983508	0.984508	0.986	0.985507	0.978117
	0.2	16	0.0008	98.2	0.984	0.985	0.986	0.986	0.986901
		32	0.00190	98.2	0.984	0.985	0.986	0.986	0.986803
		64	0.00573	98.2	0.984	0.985	0.986	0.986	0.98621
		128	0.00420	98.2	0.984	0.985	0.986	0.986	0.983052
	0.5	16	0.01413	98.13126	0.983311	0.984311	0.986	0.98531	0.985914
		32	0.00993	97.71882	0.979178	0.980174	0.985606	0.981169	0.978611
		64	0.0006	98.1018	0.983016	0.984508	0.986	0.98531	0.983743
		128	0.00280	98.1509	0.983508	0.984508	0.986	0.985507	0.982065

Table ٧ depicts Comparison between CNN Architecture & accuracy of our research and another researches. It is shown that the proposed system obtains high accuracy than other researches.

Table ٧: Comparison between CNN Architecture & accuracy of our research and another researches

	Characteristic	Accuracy
Predict driver Drowsiness ref.[26]	1)8 Convolutional layers 2) 3 max pooling 3) 6 dense layers 20 epochs	98%
DDS_R2 ref.[27]	1) 3 Convolutional layer 2) 1 max pooling 3) 1 average pooling 4) 4 dense layers 10 epochs	94.6%
Driver Drowsiness Detection The proposal	1) 3 Convolutional layer 2) 3 Max Pooling 3) 2 Dense layer 4) 2 Dropout 20 epochs	98.6%

Figures 9, 10, and 11 shows the model accuracy, model loss, and model accuracy/loss respectively with epochs used.

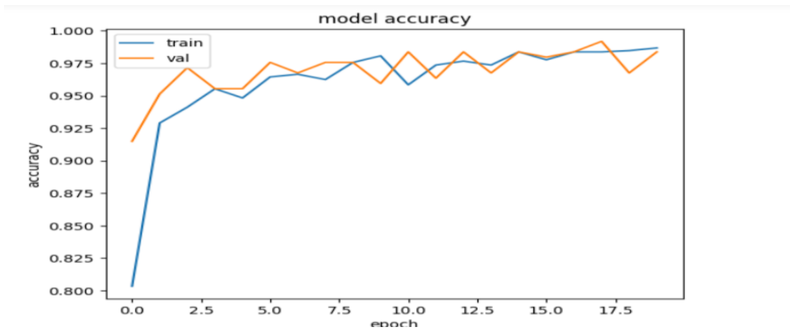


Figure 9: Relation between the accuracy and epochs

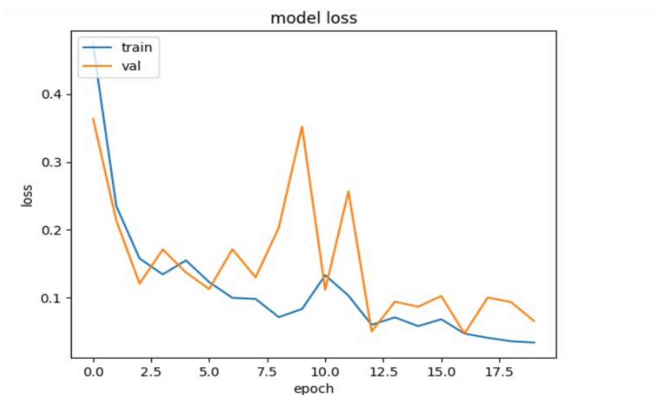


Figure 10: Relation between the loss function and epochs

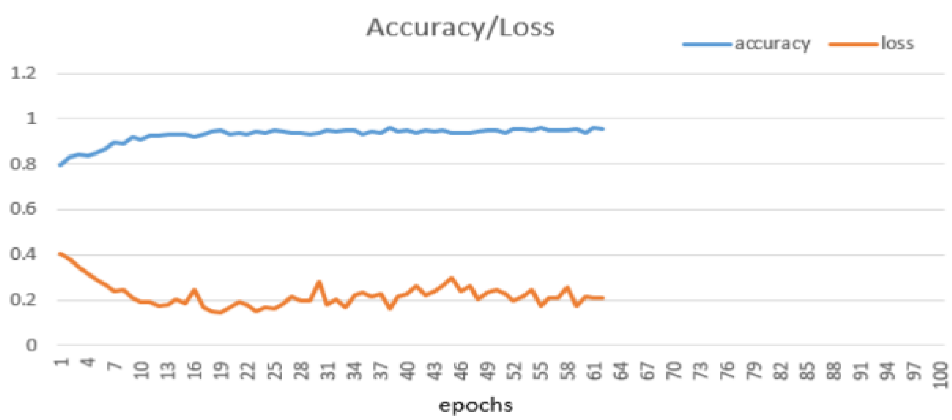


Figure 11: Relation between the accuracy / loss function and epochs

5. Conclusion

This paper proposed a framework for driver drowsiness detection using facial expression analysis based on deep learning and CNN architecture. In the proposed system, the CNN model was trained from inception with various dropout ratios. After reviewing the results, it is concluded that the proposed system offers a low-cost, real-time driver drowsiness monitoring system based on visual behavior. The average accuracy of training results, 97%, is achieved at 32 batch sizes and without applying dropout. The best average accuracy of the testing result was achieved at 16 batch size and without applying dropout, with 0.2 ratios and 32 batch size, and with 0.5 ratios and 32 batch size, with 98.15%, 98.1%, and 98.01%, respectively.

Disclosure
This research didn’t receive any specific grant from funding agencies in the public, commercial or not-for-profit sectors. The author reports no conflicts of interest in this work.

5. References

1. R. Jabbar, M. Shinoy, M. Kharbeche, K. Al-Khalifa, M. Krichen, and K. Barkaoui, “Driver drowsiness detection model using convolutional neural networks techniques for android application,” in 2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIoT), IEEE, 2020.
2. R. Sanyal and K. Chakrabarty, “Two stream deep convolutional neural network for eye state recognition and blink detection,” in 2019 3rd International Conference on Electronics, Materials Engineering & Nano-Technology (IEMENTech), pp. 1–8, Kolkata, India, 2019.
3. K. Saleh, M. Hossny, and S. Nahavandi, “Driving behavior classication based on sensor data fusion using lstm recurrent neural networks,” in *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, pp. 1–7, IEEE, 2017.
4. Y. Ed-Doughmi, N. Idrissi, and Y. Hbali, “Real-time system for driver fatigue detection based on a recurrent neuronal network,” *Journal of Imaging*, vol. 6, no. 3, p. 8, 2020.
5. S. Vitabile, A. De Paola, and F. Sorbello, “A real-time non-intrusive fpga-based drowsiness detection system,” *Journal of Ambient Intelligence and Humanized Computing*, vol. 2, no. 4, pp. 251–262, 2011.
6. S. Navaneethan, N. Nandhagopal, and V. Nivedita, “An FPGA-based real-time human eye pupil detection system using E2V smart camera,” *Journal of Computational and Theoretical Nanoscience*, vol. 16, no. 2, pp. 649–654, 2019.
7. O. Jemai, I. Teyeb, and T. Bouchrika, “A novel approach for drowsy driver detection using eyes recognition system based on wavelet network,” *International Journal of Recent Contributions from Engineering, Science & IT*, vol. 1, no. 1, pp. 46–52, 2013.
8. M. Babaeian, K. A. Francis, K. Dajani, and M. Mozumdar, “Real-time driver drowsiness detection using wavelet transform and ensemble logistic regression,” *International Journal of Intelligent Transportation Systems Research*, vol. 17, no. 3, pp. 212–222, 2019.
9. S. Arefnezhad, S. Samiee, A. Eichberger, and A. Nahvi, “Driver drowsiness detection based on steering wheel data applying adaptive neuro-fuzzy feature selection,” *Sensors*, vol. 19, no. 4, p. 943, 2019.
10. K. Mutya, J. Shah, A. D. McDonald, and J. Jefferson, “What are steering pictures are worth? using image-based steering features to detect drowsiness on rural roads,” *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 63, no. 1, pp. 2041–2045, 2019.
11. U. Budak, V. Bajaj, Y. Akbulut, O. Atila, and A. Sengur, “An effective hybrid model for EEG-based drowsiness detection,” *IEEE Sensors Journal*, vol. 19, no. 17, pp. 7624–7631, 2019.
12. A. A. Hayawi and J. Waleed, “Driver's drowsiness monitoring and alarming auto-system based on EOG signals,” in *2019 2nd International Conference on Engineering Technology and its Applications (IICETA)*, pp. 214–218, IEEE, 2019.
13. M. S. Song, S. G. Kang, K. T. Lee, and J. Kim, “Wireless, skin-mountable EMG sensor for human–machine interface application,” *Micromachines*, vol. 10, no. 12, p. 879, 2019.
14. D. Artanto, M. P. Sulistyanto, I. D. Pranowo, and E. E. Pramesta, “Drowsiness detection system based on eye-closure using a low-cost EMG and ESP 8266,” in *2017 2nd International conferences on Information Technology, Information Systems and Electrical Engineering (ICITISEE)*, pp. 235–238, IEEE, 2017.
15. L. B. Chen, W. J. Chang, J. P. Su et al., “A wearable-glasses-based drowsiness-fatigue-detection system for improving road safety,” in *2016 IEEE 5th Global Conference on Consumer Electronics*, p. 12, IEEE, 2016.
16. A. S. A. Bakar, G. K. Shan, G. L. Ta, and R. A. Karim, *Iot|eye Drowsiness Detection System by Using Intel Edison with GPS Navigation*, pp. 485–493, 2018.
17. Y. Ma, B. Chen, R. Li et al., “Driving fatigue detection from EEG using a modified PCANet method,” *Computational Intelligence and Neuro-science*, vol. 2019, pp. 1–9, 2019.
18. M. Ngxande, J.-R. Tapamo, and M. Burke, “Bias remediation in driver drowsiness detection systems using generative adversarial networks,” IEEE Access, vol. 8, pp. 55 592–55 601, 2020.
19. Z. Li, C. Liukui, J. Peng, and Y. Wu, “Automatic detection of driver fatigue using driving operation information for transportation safety,” Sensors, vol. 17, p. 1212, 05 2017.
20. S. Said, S. Alkork, T. Beyrouthy, M. Hassan, O. E. Abdellatif, and M. F. Abdraboo, “Real Time Eye Tracking and Detection- A Driving Assistance System,” Advances in Science, Technology and Engineering Systems Journal, vol. 3, no. 6, pp. 446–454, 2018

21. Driver Drowsiness Detection Using Face Recognition Peddagoni Sai Pavan Goud1 , Nenavath Aravind2 Students of Embedded Systems, Department of Electronics and Communication Engineering, Jawaharlal Nehru Technological University, Hyderabad
22. https://www.kaggle.com/serenaraju/yawn-eye-dataset-new
23. Deep Learning based Driver Drowsiness Detection serch paper St. John College of Engineering and Management Palghar, India
24. DDS_R2 | Kaggle
25. https://www.kaggle.com/code/ziadhamadafathy/predict-driver-drowsiness
26. CC Liu, SG Hosking, MG Lenné, "Predicting driver drowsiness using vehicle measures: Recent insights and future challenges" Journal of safety research, 2009 – Elsevier
27. CJ de Naurois, C Bourdin, A Stratulat, E Diaz , "Detection and prediction of driver drowsiness using artificial neural network models" Accident Analysis & …, 2019 - Elsevier