Proceedings of the International Conference on Applied Artificial Intelligence and Computing (ICAAIC 2022)
IEEE Xplore Part Number: CFP22BC3-ART; ISBN: 978-1-6654-9710-7

Deep Learning based Driver Drowsiness Detection

PARTH P. PATEL
U.G. Student, Department of Information Technology
St. John College of Engineering and Management
Palghar, India
pparth661@gmail.com

CHIRAG L. PAVESHA
U.G. Student, Department of Information Technology
St. John College of Engineering and Management
Palghar, India
chiragpavesha01@gmail.com

SANTOSHI S. SABAT
U.G. Student, Department of Information Technology
St. John College of Engineering and Management
Palghar, India
santoshisabat2505@gmail.com

SHRADDHA S. MORE
Assistant Professor, Department of Information Technology
St. John College of Engineering and Management
Palghar, India
shraddham@sjcem.edu.in

***Abstract*— Driver drowsiness and distraction are now widely recognized as important contributors to deadly road accidents around the world. This typically occurs when a driver has not slept enough, but it can also occur as a result of untreated sleep problems, drugs, alcohol consumption, or shift employment. As a result, driver monitoring and identification are becoming increasingly important features of car safety systems. Head position, gaze direction, yawning, and eye state analysis are among the essential aspects. This research proposes a driver drowsiness detection system that uses eye blink, mouth opening and closing counts to detect drowsiness. When the driver's eyes are closed for an extended period of time, an alert sound is generated to notify him. Furthermore, the vehicle's owner is notified by e-mail if the driver is observed to be napping off more than a few times, in order to certify that the driver is taking certain steps to avoid falling asleep. The output of the system proposed in the paper on Deep learning technology of Dlib which uses CNN (Convolutional Neural Network) as its base algorithm for accurate detection, OpenCV, and Raspberry Pi environments with a mounted camera for the same, show that system achieves good result when it comes to drowsiness detection, reducing the overall number of accidents on the streets. For Realtime video input, the proposed method had a 96% of accuracy.**

Keywords—CNN, Deep Learning, Dlib, Drowsiness Detection, Eye blink, OpenCV, Raspberry Pi.

I. INTRODUCTION

One of the major contributing factors to motor vehicle accidents is driver drowsiness. Drowsy drivers are more likely to cause a serious accident because they are unable to react quickly enough to dangerous situations. To avoid such an accident, it's crucial to identify drowsiness as soon as possible and precisely [1]. According to the survey, 846 people died in 2021 as a result of sleepy driving. Between 2021 and 2018, sleepy driving was blamed for an estimated 2,363,031 crashes each year. On a yearly basis, car crashes account for around 996 deaths, 49,000 injuries, and 51,000 property damage [2]. Drowsy driving occurs when a driver is exhausted while driving, making it impossible for him to remain awake. This frequently happens when the driver doesn't get enough sleep. This can also happen if he/she has sleep problems like insomnia and even shift work sleep disorder (SWSD). This resulted in a collision with another vehicle for the drowsy driver. A human's drowsiness is identified by a few distinct motions and facial expressions such as the brief moment of eye closing, the mouth opens in a yawn, the jaw dips, and the neck tilts. This research concentrates on detecting tiredness and classifying a driver as drowsy by observing the eyes and mouth. The video input can be acquired by placing the camera on the vehicle's dashboard for real-time application of the model and can easily collect the driver's face, hands, upper torse and even through non-tinted spectacles in the video stream. The Dlib model has been taught to recognize 68 facial landmark points [3]. The features of drowsiness are retrieved, and the driver is notified if drowsiness is recognized. The model is not bound to have any prior knowledge about the person who will be using or testing the system.

This research provides deep learning strategy for observing driver drowsiness based on computer vision. This strategy adopts the input stream from driver's face and divides drowsiness into three categories (normal, yawn, and drowsy). This model's major benefit is that it is compact enough to be implemented on an handheld microcontroller with camera embedded on it while maintaining decent accuracy. A compressed model is important for integrating a driver's tiredness detection system into a normal car. Whereas a person who can fall asleep at any moment, having a real-time classifier for sleepiness detection that consumes little to no electricity and can be effortlessly installed on a car.

This paper conducted a literature review in Section II to learn about the various systems currently in use for detecting tiredness. The concept and execution of our algorithm are presented in Section III and in IV Section is all about components used in our system. In Section VI, the outcomes of real-time experiments are analysed and discussed. Section VII brings the paper to a close by addressing future options for our idea.

II. LITERATURE REVIEW

In 2020, Saifuddin Saif, et al. [4] suggested a Convolutional Neural Network (CNN) technique for detecting microsleep and tiredness using a machine learning algorithm. In the research done, the mounted camera detects the driver's face points, which is then fed to the detecting algorithm to determine a person's drowsiness status. The model gets trained on the dataset of open and closed eyes in different lighting conditions as it is only trained to detect the tiredness based on eye blink and other signs of drowsiness are missed. Given the system provides a live detection and feedback on the subject matter of eyes for sleepiness. It focuses on the development of instantaneous drowsiness detection system to avoid the catastrophic consequence of driver due to accident. This method creates a variety of driver faces and a model to detect sleepy driver with 93 percent enrolments on a single feature, while different symptoms of tiredness, such as yawning and even head tilting, should be monitored by the system.

In 2020, Rateb Jabbar et al. [5] presented a android application which detects the drowsy driver using deep learning trained model. The authors produced a model based on the recognition of face landmark points. The Dlib package was used to mine facial coordinate points after first extracting pictures from video frames. The features of eyes and mouth yawning were detected with model reaching the accuracy of only 80% which is less when dealt with the issue of safety. The multilayer perception classifier receives these facial coordinate points as input. Based on these points, the classifier does classification based on subject drowsy or subject not drowsy. When compared with current scenarios as mobile application handles a lot of work any small change or even a notification is able to distract the driver.

In 2019, Anshul Pinto et al. [6] projected a system that is built on convolutional neural network for a range of machine learning and predictive tasks, and it takes a picture of an eye as input and classifies it as open or closed. The performance of CNN was tested using eye blink detection algorithm and a formula called EAR as per results section. On average, the CNN technique was 93.3 percent accurate, while the EAR approach was 80.4 percent accurate. The system was developed using a yawn detection mechanism that might be linked with the status of the eyes to act as a metric for detecting the driver's alertness.

In 2018, Menchie Miranda et al. [7] presented an IoT based device made using Raspberry Pi and camera. Using camera and OpenCV python script the eyelid of the driver id detected and a random alarm turns on when the eyelids are closed for more than 3 seconds and the data is sent to a website which the vehicle owner can access later. Upon being subjected to different lighting conditions the accuracy got compromised as opposed to when in good lighting conditions.

In 2018, Fouzia et al. [8] presented a system sleepiness monitoring framework founded on a shape predictor algorithm that detects and counts the blink rate of the eyes before detecting drowsiness in real time. Image processing algorithms are used to gather information about the eye status, which deliver a non-invasive method of detecting drowsiness without causing any discomfort or inconvenience. The same framework needs to be used for implementing yawn detection which further would hail the drowsiness detection.

In 2020, Mariella Dreißig et al. [9] presented a system in which the eyelid movement was studied using classification kNN algorithm on dataset which contain 134 hours of recording and with Karolinska Sleepiness Scale (KSS) values it became easy to determine the drowsiness. The upper bound of k was selected as 1000 based on the distinct data points. The identification of appropriate features was a key component of the k-NN based classification. With a crucial decision being made on the value of k the accuracy of the model changes is the value of k is not chosen upto the mark. Different number of Head movements and eye features were detected which served as a basis for designing the model.

III. HAAR CASCADE ALGORITHM

HAAR cascade are very effective for object detection which is based on Viola-Jones Algorithm. This approach involves a lot of positive and negative images that are used to train the classifier. There are many features on the face, sum of pixels under white and black rectangles, so we can detect the feature even in low image quality. For training the images the threshold is calculated using the contrast of lighter to darker region. We select certain features for minimum error rate i.e. choosing such facial features which could help easily classify face and non-face image. As most of the part of image will be non-face a different classifier stages are used to identify based on which the face of the person is recognized if it passes through all the stages of the classifiers. In Fig. 1. A Haar kernel is a rectangle with all of the light pixels below and all of the dark pixels on the upper side of the rectangle. The difference between the average of the pixel values in the darker zone and the average of the pixel values in the lighter region is used to calculate the Haar. If the difference is close to one, then there is a edge detected by Haar.

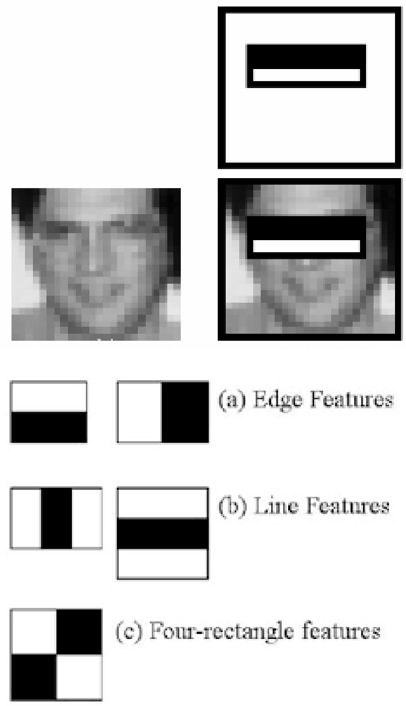

Fig. 1: HAAR cascade Algorithm

Every feature of a single value is calculated by taking subtraction of total pixels beneath white rectangle from total number of pixels beneath black rectangle.

Dlib library based on CNN: Deep Learning techniques is subfield of Machine Learning that is driven by the human brain. Dlib is a C++ deep learning library which uses CNN as its base algorithm. CNNs, also termed as ConvNets, are multilayered neural networks that are primarily use in image processing, video analysis as well as object detection. To execute the convolution process, CNN has convolution layer with several filters. CNN assigns every image a vector number which can be understood by library with fully connected dense layers of neural network. The CNN has responsibility of taking a picture of an eye as input and classifying whether it is closed or open.

- **Eye Blink Detection**: The eyes of the driver are identified using a neural network, and the model then checks for weariness using the eye close ratio values which are calculated using Eye aspect ratio which takes into account the ratio of distances between the eyelids to the distance of the eye corner. Upon using this method the accuracy increases as different eye shapes can easily be calculated for the drowsiness.

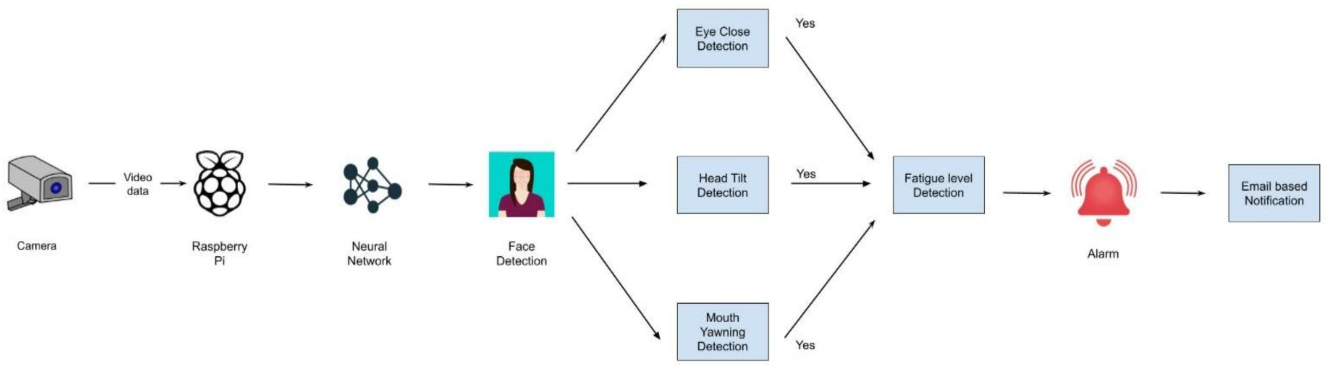

Fig. 2: Block Diagram of Proposed System

IV. PROPOSED SYSTEM

The block diagram of proposed system for detecting fatigue and sleepiness utilizing Deep learning and IoT technologies is shown in the Fig 2.

In Fig 2, a driver can activate the system simply by turning on the power. When the system is turned on, the camera's input is transmitted to the CNN algorithm in Python file. Where the person's face is processed by the corresponding libraries, and every facial feature in the diagram is compared to its threshold value and a prediction of tiredness is made using a series of if else statements and the counter variable count. Based on the counter value, an email is sent to the concerned person or organisation who is linked to the driver for alertness. The key components of the proposed system are as follows:

Raspberry Pi 4 Model B: A Raspberry Pi is single-board based computer that functions as the micro-computer which has Raspbian operating system built in.

Speaker: To produce sound to alarm the driver.

Web Camera: A webcam is a little digital recording device that connects to the Raspberry Pi. It captures high-definition videos with static photos at 720p 30fps.

GPS Module: The GPS (Global Positioning System) is satellite-based system that uses satellites and ground control station to accurately measure and track its geographical position on earth surface.

The driver's involvement with the system is enabled by the camera, which sends the visual stream to the Raspberry Pi microcontroller. Raspberry Pi is a small microprocessor with its own operating system that performs the same functions as a computer.

Proposed system is basically divided into four major parts:

$$\text{EYE Ratio}=\frac{\text{Distance between eyelids}}{\text{Distance between eye corner}} \tag{1}$$

- **Detection of yawning**: The trained model recognizes the driver's mouth, marks points while the mouth is closed, and calculates the person's tiredness based on the points and mouth open ratio. The approach helps in calculating clear prediction as the value obtained could be directly compared against the threshold value.

$$\text{YAWN Ratio}=\frac{\text{Upper-Lower Lip distance}}{\text{Distance between mouth corners}} \tag{2}$$

- **Detection of head position**: The Trained Model recognizes the drivers head based position through length of nose and compares with average length of nose for detection.
- **Alarming System:** Dependent on the driver's level of exhaustion, the sound system uses a microcontroller to alert the driver to take specific actions based on the amount of fatigue. In the event of an accident, an e-mail is sent to the appropriate employees as part of the alarming procedure.

V. IMPLEMENTATION

Facial Detection & Recognition:

Implementation of parameters for facial landmark detection is done in the planned work to identify the driver's state. 68 predetermined landmarks for predicting shape and identifying facial parts such as the eye, mouth, head and so on. Many differences in the dimensions of the distinguishing points reflect the person's diverse expressions.

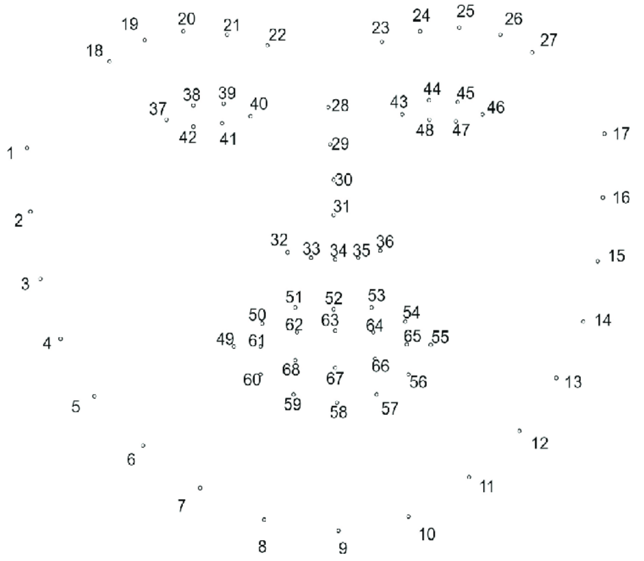

Fig. 3: Facial landmark points

Drowsiness and tiredness are detected using pre-existing features for face landmark recognition. We utilized the open-source Python library dlib to recognize 68 facial landmarks. These predetermined annotations assist in form of prediction-based recognition of numerous face region parts such as eyes, nose, eyebrows, mouth and others shown in above Fig. 3. The person's numerous expressions are represented by variations in the set parameters of these separate points.

These are the following landmark coordinates represented by the above coordinates referred from Fig. 3.

Table 1: Various facial region coordinates

Sr. no.	Facial Points	Landmark Coordinates
1)	Left Eye Region	[37, 42]
2)	Right Eye Region	[43, 48]
3)	Nose Region	[28, 36]
4)	Mouth Region	[49, 68]

The procedure for recognizing a facial landmark is as follows:

High-resolution cameras are used to monitor and capture images in process of extracting frames and then generate alerts. Each captured frame is evaluated to examine the pattern of features of the face, and EAR (Eye Aspect Ratio) and MAR (Mouth Aspect Ratio) at each frame is calculated using Haar Cascade Classifier. A blink and a yawn are considered when the Eye ratio and Mouth ratio values reach at their specific threshold levels. If eye gaze and yawns are detected for particular frames in a sequence, then system informs the driver via playing an alarm. The alarm is set off to get attention of driver and will continue to ring until driver gets wakes up.

A video is captured with a webcam and the frames are retrieved on a laptop. Following that, we have used image processing techniques such as open CV and dlib to extract the image. The video is recorded as soon as the driver sits in front of the web cam. Then eye closure, blinking, head tilt, and yawning are detected.

Eye Detection:

6 coordinates are marked on the eyes using the dlib landmarks predictor function, starting from the left corner and moving clockwise. The formula to obtain EAR (Eye Aspect Ratio), which is the height-to-width ratio of the eye. The EAR value is calculated using,

$$EAR = \frac{||d1 - d3|| + ||d2 - d4||}{2\,||d0 - d5||} \quad (3)$$

Where, d1, d2, d3, d4, d5, d6 are 6 facial landmark points to be extracted for eyes and $||di - dj||$ is Euclidean distance between points i and j. We determine whether the driver seems drowsy or not by monitoring eye closure and eye opening ratios using the above formula. EAR is often calculated independently for left and right eyes.

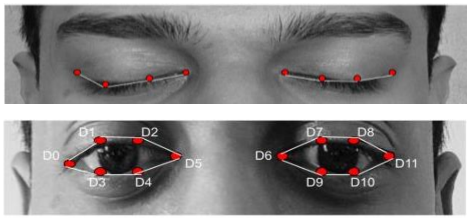

Fig. 4: Eye open and close coordinates

Mouth Detection:

The MAR (Mouth Aspect Ratio) is metric that determines the of wide your mouth. The mouth is represented by an 8-coordinate pair. The face landmarks are designated in clockwise manner, beginning at the left corner of your mouth. The MAR is obtained by taking the vertical distance between lower and upper lips by the horizontal distance between the lip corners.

$$MAR = \frac{||d2 - d8|| + ||d3 - d7|| + ||d4 - d6||}{2||d1 - d5||} \quad (4)$$

Whenever driver yawns, the gap between your lower and upper lips widens. The yawn count is increased when the MAR value surpasses a specified level.

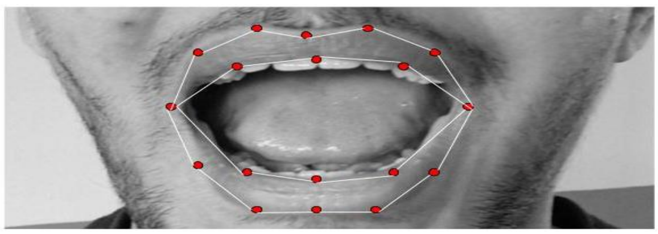

Fig. 5: Open mouth coordinates

Head Tilt Detection:

Drowsiness causes the driver's head to tilt forward, sideways or backward in relation to vertical axis. As a result, driver drowsiness can also be estimated based on the head bending angle. Since the simulated length of the nose on camera focal plane is proportional to the tilting, it could be used for determining head bending. Under normal circumstances, nose forms acute angle with the camera's focal plane. As the head travels upward, the angle increases, and as it travels downwards, it decreases. As a result, the ratio of the nose length to average of the nose length whereas awakened is measured for head bending, and if value is larger or less than specific range, it indicates both bending of head and drowsiness. The length of nose is calculated by using facial landmarks and defined as

$$\text{NLR}= \frac{\text{nose length (d31 - d28)}}{\text{average nose length}} \quad (5)$$

Dataset:

The dataset contains 80,000 images divided into train and test data where images indicate different facial features of eye open and close pictures, mouth opening and closing and lastly head tilt feature. The images are converted to grayscale for the algorithm to easily identify the landmarks.

Fig. 6: Dataset

Pseudo Code:

```
importing required libraries
sound = assign (alarm.wav)
EAR_threshold = 0.25
YAWN_threshold = 20

function alarm (message) :
var alarm
    s = "espeak" + message
    execute (s)

function eye_aspect_ratio (eye) :
    A,B,C = euclidean.distance ( eye[point] )
    ear  = (A+B ) / (2.0 * C)

function final_ear (shape):
    leftEAR = eye_aspect_ratio (leftEye)
    rightEAR = eye_aspect_ratio (rightEye)
    ear = leftEAR + rightEAR

function lip_distance :
    top_lip = shape[50:53]
       low_lip = shape [56:59]
```

In the above section of code the threshold for the predictions are set and the functions are created based on the formulae of EAR and MOR for detection.

```
    distance = abs (top_lip – low_lip )

    detector = cv2. Cascadefile("haarcascade_file")
predictor = dlib.shape_prediction(" shape_prediction.dat ")
    while True:
shape = rectangle ( x, y, width, height)
eye = final ear (shape)
lip = shape [48:60]
distance = lip_distance(shape)
     counter = 0

if ear < EAR_threshold:
                  alarm_status = True
                  sound.play ()
                  alarm ("Drowsiness Alert")
                  counter += 1

else if distance > YAWN_threshold:
                  alarm_status2 = True
                  sound.play ()
                  alarm (" Take a break ")
                  counter += 1
          else if  counter > 3:
                  send email
else :
                  alarm = False
                  if key = stop:
                              break
```

In this section of code based on the values obtained by the detector function the loop works based on constant cycle of comparison with the threshold value based on which the alarm signal is activated and email is sent based upon exceeding the counter value which increments which each consecutive detection based on the time constraint.

VI. Result and Discussion

A system is built with a single camera and microcontroller and alarm which is compact as some of the methods previously in use were able to cause distraction to the driver, where the drivers were vulnerable to an accident.

Fig. 7: Prototype

Upon activating system, the drivers face is recognized and the points are marked, if the EAR ratio of eye is lower than 0.25 unit which is the threshold value displayed in Fig 9 then the driver is alarmed.

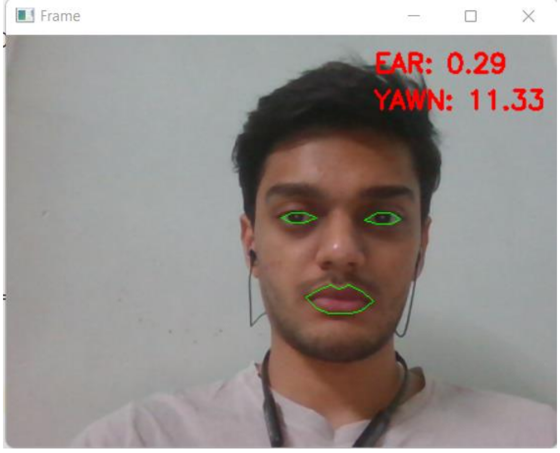

Fig. 8: Score when eyes are open and subject is not yawning

If the Mouth Open Ratio of the of the driver's face id found to be greater than 20 units the driver is alarmed with an alarm and with an appropriate message.

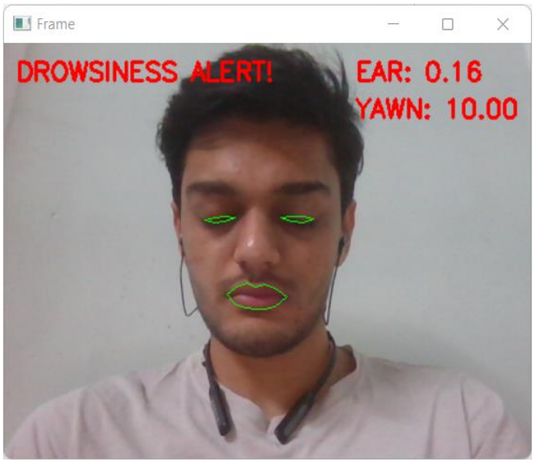

Fig. 9: Score when eyes are closed (Alarming Situation)

With all of the behavioral aspects combined in a single system based on the literature survey performed as all the proposed systems in the paper dealt with one or other signs of drowsiness but the combined result of all the aspects helps in proposing better alarming system which is developed in this paper.

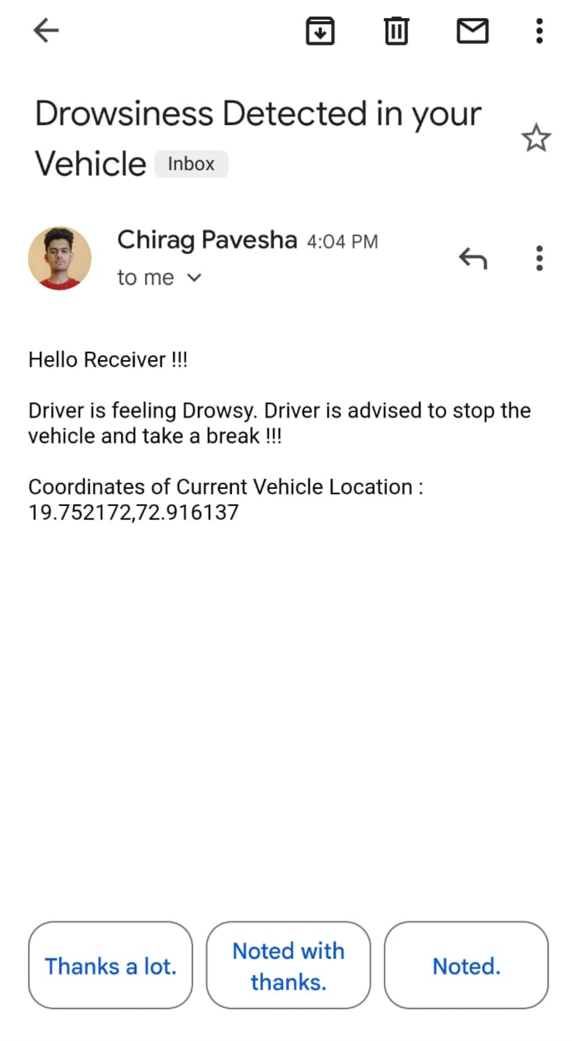

Fig. 10: Alert using Email

If the driver offend the drowsiness alert multiple time consecutively then an alert email is delivered to the concerned person alerting him of the drowsy driver and providing with geolocation of the vehicle.

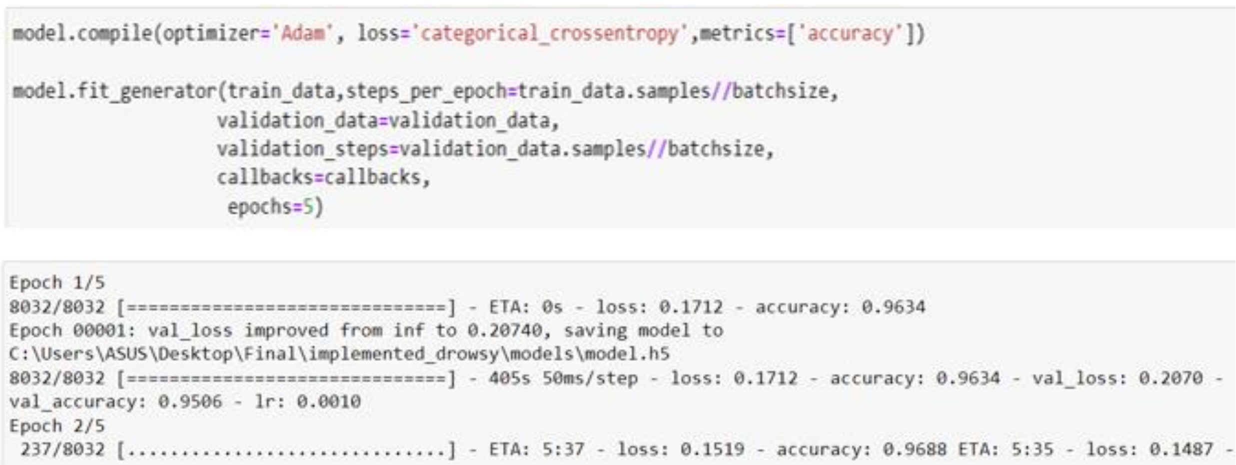

Fig 11: Model Accuracy

The model trained over the dataset images yields testing accuracy of 96.34% where model is tested on approximate 2000 images.

VII. Conclusion and Future Scope

The system designed achieves the objective of detecting drowsy driver and notifies the driver with alarming signal and appropriate audio message. With the help of CNN algorithm and Haar cascade files faster detection of face and extraction of facial features according to which the model predicts the mental state of the driver through which his behavioral features, include eye blinking, yawning and head tilt detection. With the help of deep learning technology even if the size is small, we get high accuracy, with the algorithm in place, the accuracy of 96% is achieved. As system is compact, and can be easily integrated in modern cars as it just requires camera and raspberry pi module. In case of consecutive violation of drowsiness threshold an email is sent to the concerned person with a message and current location of the driver is shared to that person. The systems accuracy gets little compromised in highly dark conditions.

In future more specialties could be added to the system for better prediction which can consist automatic calibration, which includes detecting drowsy drivers and automatically reducing the vehicle's speed for safety of driver on the road, determining vehicle states, such as calculating the distance of vehicle from the vehicle in the front, vehicle speed, and so on.

REFERENCES

[1] Jongseong Gwak, Motoki Shino and Akinari Hirao, "Early Detection of Driver Drowsiness Utilizing Machine Learning based on Physiological Signals, Behavioral Measures, and Driving Performance," 21st International Conference on Intelligent Transportation Systems (ITSC) Maui, Hawaii, USA, November 4-7, 2018

[2] Janki Chandiwala and Shrushti Agarwal, "Driver's real-time Drowsiness Detection using Adaptable Eye Aspect Ratio and Smart Alarm System,"7th International Conference on Advanced Computing & Communication Systems (ICACCS), 2021.

[3] Shruti Mohanty, Shruti Hegde, Supriya Prasad, J. Maniknadan, "Design of Real-Time Drowsiness Detection System using Dlib," 5th IEEE International WIE Conference on Electrical and Computer Engineering (WIECON-ECE), 2019.

[4] A. f.m saifuddin saif, Zainal rasyid mahayuddin, "Robust Drowsiness Detection for Vehicle Driver using Deep Convolutional Neural Network," International Journal on Advanced Computer Science and Applications (IJACSA), Vol.11, 2020.

[5] Rateb Jabbar, Mohammed Shinoy, Mohamed Kharbeche, Khalifa Al-Khalifa, Moez Krichen, Kamel Barkaoui, "Driver Drowsiness Detection Model Using Convolutional Neural Networks Techniques for Android Application," IEEE Conference on Informatics, IoT and Enabling Technologies (ICIoT), 2020.

[6] Anshul Pinto, Mohit Bhasi, Durvesh Bhalekar, Pradyoth Hegde, Shashidhar G. Koolagudi, "A Deep Learning Approach to Detect Drowsy Drivers in Real Time," IEEE 16th India Council International Conference (INDICON), 2019.

[7] Menchie Miranda, Alonica Villanueva, Mark Jomar Buo, Reynald Merabite, Sergio Paulo Perez, and John Michael Rodriguez, "Portable Prevention and Monitoring of Driver's Drowsiness Focuses to Eyelid Movement using Internet of Things," IEEE 10th International Conference on Humanoid, Nanotechnology, Information Technology,Communication and Control, Environment and Management, 2018.

[8] Fouzia, R. Roopalakshmi, Jayantkumar A. Rathod, Ashwitha S. Shetty, K. Supriya, "Driver Drowsiness Detection System Based on Visual Features," Second International Conference on Inventive Communication and Computational Technologies (ICICCT), 2018.

[9] Mariella Dreißig, Mohamed Hedi Baccour, Tim Schäck, Enkelejda Kasneci (2020). "Driver Drowsiness Classification Based on Eye Blink and Head Movement Features Using the k-NN Algorithm," IEEE Symposium Series on Computational Intelligence (SSCI), 2020

[10] Kim, W., Choi, H.-K., & Jang, B.-T, "Study on Training Convolutional Neural Network to Detect Distraction and Drowsiness," IEEE Region Ten Symposium (Tensymp), 2018.

[11] Lin, J. (2020). "Integrated Intelligent Drowsiness Detection System Based on Deep Learning," IEEE International Conference on Power, Intelligent Computing and Systems (ICPICS), 2020.

[12] Patnaik, R., Krishna, K. S., Patnaik, S., Singh, P., & Padhy, N., "Drowsiness Alert, Alcohol Detect and Collision Control for Vehicle Acceleration," International Conference on Computer Science, Engineering and Applications (ICCSEA), 2020.

[13] Wanghua Deng, Ruoxue Wu (20, "Real-Time Driver-Drowsiness Detection System Using Facial Features," IEEE Access, 2019

[14] Chaoyun Zhang, Rui Li, Woojin Kim, Daesub Yoon,And Paul Patras, "Driver Behavior Recognition via Interwoven Deep Convolutional Neural Nets With Multi-Stream Inputs," Electronics and Telecommunications Research Institute (ETRI), Daejeon, South K, 2020.

[15] Bhargava Reddy, Ye-Hoon Kim, Sojung Yun, Chanwon Seo, Junik Jang, "Real-Time Driver Drowsiness Detection for Embedded System Using Model Compression of Deep Neural Networks, " IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2017

[16] Kusuma, S., Divya Udayan, J., & Sachdeva, A., "Driver Distraction Detection using Deep Learning and Computer Vision, " 2nd International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT), 2019.

[17] Md. Yousuf Hossain and Fabian Parsia George, "IOT based Real-time Drowsy Driving Detection System for the Prevention of Road Accidents," IEEE, (ICIIBMS) Track 1: Image Processing, Computer Science and Information technology Bangkok, Thailand, 2018.