IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 2, June 2022
Impact Factor: 6.252

Driver Drowsiness Detection Using Machine Learning

Chilakala Anvitha[1], Mohammed Muzammil[2], Soundaraya. A[3], Raghu Kumar[4]
B.E. Student, Department of Computer Science[1,2,3]
Assistant Professor, Department of Computer Science[4]
Rao Bahadur Y Mahabaleshwarappa Engineering College, Ballari, Karnataka, India

Abstract: *In today's world, the number of road accidents that occur is increasing very rapidly. Some of these road accidents are minor whereas some of them may seriously injure people or even take their lives. These road accidents may involve collision of vehicles with each other or crashing the vehicles on to the buildings or others. Road accidents may even result in the death of people. It is estimated that the total deaths due to road accidents in India is around 1,50,000 per year which is approximately 400 accidents per day. Almost 3 lives are lost in every 10 minutes due to road accidents. one of the main factors that lead to road accidents are fatigue and drowsiness. Some popular methods that detect drowsiness use ECG and EEG, which are very complex. Even though these approaches are very precise, they require human involvement and have limitations that make them unsuitable for real-time driving. We're working on a technique to detect drowsiness in driver by monitoring the eyes and mouth while they're driving. The proposed method uses python along with two libraries OpenCV and dlib for facial landmark detection. A mathematical values Eye Aspect Ratio (EAR) and Mouth Aspect Ratio (MAR) is then calculated for determining whether the eyes are closed or not and weather the person is Yawning, The proposed systems detects these signs and plays an alarm.*

Keywords: Drowsiness, Alert Alarm, DLIB, EAR, MAR

I. INTRODUCTION

Nowadays, more and more professions require long-term concentration. Drivers must keep a close eye on the road, so they can react to sudden events immediately. Drowsiness is a condition where driver is not in fully alert mode (In this condition the driver is somewhat sleepy). If you have driven before, you have been drowsy at the wheel at some point. It's not something we like to admit but it's an important problem with serious consequences that needs to be considered. Drowsiness of the drivers is one of the key issues for majority of road accidents. Drowsiness threatens the road safety and causes severe injuries sometimes, resulting in fatality of the victim and economical losses. Drowsiness implies feeling lethargic, lack of concentration, tired eyes of the drivers while driving vehicles. Most of the accidents happen in India due to the lack of concentration of the driver. Performance of the driver gradually deteriorates owing to drowsiness. To avoid this anomaly, we developed a system that can detect the drowsiness nature of the driver and alert him immediately. This system captures images as a video stream through a camera, detects the face and localizes the eyes and mouth. The eyes are analyzed for drowsiness detection using Dlib library and whether the driver is yawning. Based on the result, the driver is alerted for drowsiness through an alarm system.

II. LITERATURE SURVEY

According to the author Esra, they have employed machine learning to data mine actual human behaviour during drowsiness episodes. Automatic classiers for 30 facial actions from the Facial Action Coding system were developed using machine learning on a separate database of spontaneous expressions. These facial actions include blinking and yawn motions, as well as a number of other facial movements. In addition, head motion was collected through automatic eye tracking and an accelerometer. The effect on drowsiness on the other facial expressions have not been studied thoroughly. This approach to drowsiness detection primarily makes pre-assumptions about the

According to Danisman et al., he developed a method to detect a drowsiness based on changes in eye blink rate. Viola Jones detection algorithm was used to detect face region from the images. To find the location of the pupils, neural network- based eye detector was used. The no of blinks per minute has been calculated if blinks increase indicated that driver becomes drowsy.relevant behaviour, focusing on blink rate, eye closure and yawning.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 2, June 2022
Impact Factor: 6.252

According to authors Venkata Rami Reddy et al, they have used the Viola-zones face detection algorithm to detect face and extract the eye region from the extracted facial. To extract features from dynamically identified, key frames from the camera sequences, stacked deep Convolution neural network is developed and used for learning phase. A SoftMax layer in CNN classifier is used to classify the driver as sleep or non-sleep. Usually, CNNs requires fixed size images as input so pre- processing is required. The pre-processing includes extracting the key frames from video based on temporal changes and store in database.

III. PROPOSED WORK

In the proposed system after the Artificial Intelligent have emerged in the market, Researchers and scholars have built several ML algorithms for detection of drowsiness. For real-time application of the model, the input video can be acquired by mounting a camera on the dashboard of the car and can accommodate the driver's face line the Dlib approach, the library's pre-trained 68 facial landmark detector is used. Face detector which is based on Histogram of Oriented Gradients (HOG) was implemented. The proposed algorithm was the Eye Aspect Ratio (EAR) to monitor the driver's blinking pattern and Mouth Aspect Ratio (MAR) to determine if the driver yawned in the frames of the continuous video stream.

DLIB:

Identification of faces in photographs or videos is easy, but we need additional information about the person's face, such as the person's posture, if the mouth is shut or open, whether the eyes are shut or open, if the person is gazing up and so on. There are 68 points (landmarks) on the face that may be obtained using DLIB software. From the 68 points, the coordinates of left eye, right eye and mouth are located and are further provided to the conditional logic for monitoring of EAR and MAR values.

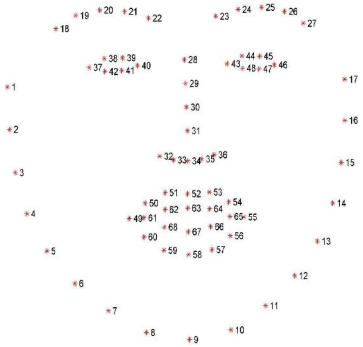


Figure 1: 68 Point View.

Eye Aspect Ratio (EAR):

1) Part	2) Landmark Points
3) Left Eye	4) [37-42]
5) Right Eye	6) [43-48]

So, we calculate the EAR based on these landmark point values. The distance between locations was calculated using facial landmarks collected by the app. To calculate the EAR value, these distances were employed. Equation 1 was used to determine EAR, which is the height-to-width ratio of the eye as the numerator indicates the eye's height, and the denominator represents its breadth, the picture shows all the ocular landmarks in their entirety.

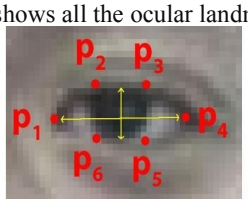


Figure 2: Eye Point View

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 2, June 2022
Impact Factor: 6.252

Referring equation 1, Calculates the distance between upper and lower eyelids using the numerator. The horizontal distance between the eyes. A higher numerator number indicates an increased EAR, whereas a lower numerator value indicates a lowered EAR. These numbers are used to detect driver fatigue in this situation.

$$EAR = \frac{\|p_2 - p_6\| + \|p_3 - p_5\|}{2\|p_1 - p_4\|}$$

Equation: 1

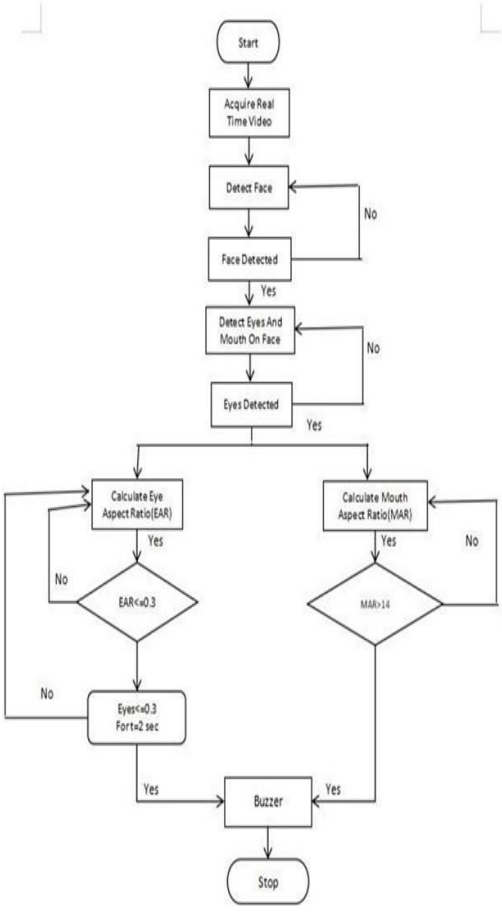
Mouth Aspect Ratio (MAR):

Mouth Aspect Ratio (MAR) is a measure to find how wide the open mouth is extended from the driver's drowsiness system where it was used to detecting yawns. The formula used to calculate the MAR is

```
def mouth_aspect_ratio(mouth):
        A = dist.euclidean(mouth[13], mouth[19])
        B = dist.euclidean(mouth[14], mouth[18])
        C = dist.euclidean(mouth[15], mouth[17])

        MAR = (A + B + C) / 3.0
        return MAR
```

IV. SYSTEM ARCHITECTURE



IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 2, June 2022
Impact Factor: 6.252

A web cam is used to capture the real time live video of the driver and then the video is disintegrated to video frames. The frames are then analyzed in Open CV through which faces are detected. Now the detected faces are again analyzed by the DLIB to detect the facial landmarks by using the 68-point approach. Eyes and Mouth are extracted as the main feature sets. The metrics of the facial landmarks are obtained and are used in Eye Aspect Ratio (EAR) approach-based formula to calculate the EAR of that face and compared with the following frames. If the EAR value is less than the threshold value for a certain period which is set, an alert sound is activated alerting the driver about the drowsiness. Same method followed for Mouth Aspect Ratio (MAR) if the driver Yawns, then the system warns him by playing an Alarm.

4.1 Flow Chart

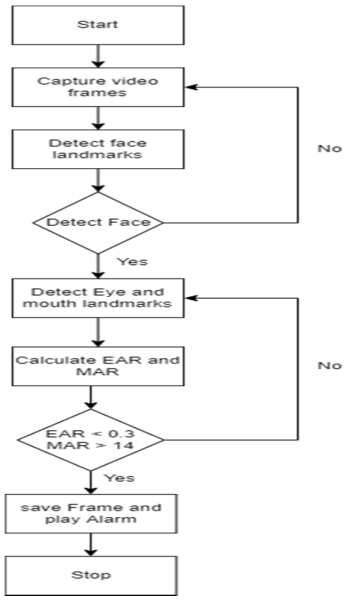


Figure 3: Data flow diagram

Here in the dataflow diagram, first the application captures the video frames from camera that is placed Infront of the driver, these video frames are captures synchronously and passed it to Dib frontal face detection model to detect the face landmarks and followed by detecting the eyes and mouth landmarks. Once these landmarks are detected we will be calculating EAR and MAR values, if these values cross the threshold values, then we are going to play an alarm and make the driver alert. In this project, we use a camera to obtain the real time video of the driver and convert that to video frames, these video frames are analyzed to detect faces and check if the driver is in a drowsy condition basing on the facial features. If the driver is drowsy, he will be alerted by the system. This method is an efficient way to detect the drowsiness compared to existing methods as there are no external conditions are considered. The hardware requirement is also minimal.

V. REQUIREMENTS

5.1 Hardware Configuration

- DEVICE –Computer system or laptop
- PROCESSOR –I5 OR I7 (with Graphic card preferable)
- RAM&HARD DISK SPACE–Minimum 8 GB RAM with 2 GB space in hard disk
- WEBCAMERA – Integrated or external (preferable)

5.2 Software Configuration

To develop or run the application, we need to have the following software installed:

- Python 3.8

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 2, June 2022
Impact Factor: 6.252

- VS Code
- OpenCV, Dlib

The following dependencies are also required in python language.

- OpenCV (for computer vision)
- OpenCV contribution python
- Numpy (for handling arrays and image matrices)
- Pandas(for handling of CSV files)
- Tkinter(for user interface)

5.3 Implementation Languages

Python is an interpreted, high-level, general-purpose programming language which was created by Guido Van Rossum and released in1991. Python is a multi-paradigm program language. Python is meant to be an easily readable language which uses English keywords where other languages use punctuation. It does not use curly brackets to delimit blocks and semicolons after statements are optional. Python uses white space indentation rather than curly brackets or keywords to delimit blocks. It can be used for developing desktop GUI applications, websites and web applications. Python features a dynamic type system and automatic memory management. Python is simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse.

Python has great support of libraries in all fields, and it is good choice for this project as it has OpenCV which is a largest and optimized library for computer vision. Python has all the libraries also for deep learning for image classification.

In order to develop the application, we chose Anaconda navigator as script editor to run and code files. Anaconda navigator has by default python interpreter, which also comes with a script editor named Spyder. Although PyCharm is also good editor, anaconda is famous for deep learning other libraries and data science toolbox as they are installed with anaconda itself. To detect gestures computer vision is necessary. So, in order to get that we used OpenCV library.

VI. CONCLUSION AND FUTURE SCOPE

To prepare a driver's drowsiness detection system we used different techniques including

- DLIB
- Measurement of EAR & MAR
- Conditional logic

The FAR (False Acceptance Rate) & FRR (False Rejection Rate) of our system is which is very low and response time is low so, from this we can conclude that our system is responsive and industry ready with some modifications (May be required as per compatibility). Using this software, the accidents which are caused by drowsiness condition can be significantly reduced.

This project also allows others to look forward for more improvisation with innovation for upcoming changes in technology and deploying in local environment.

The system can be improved from 128 shape predictor landmarks

Can be improved for night vision with good night vision camera.

ACKNOWLEDEGEMENT

We thank Mr. Ragu Kumar our guide for his immense support throughout the completion of the project and also our fellow team mates.

REFERENCES

[1]. Facial landmarks with dlib, OpenCV and Python: https://www.pyimagesearch.com/2017/04/03/facial-landmarks-dlib-opencv-python/

[2]. Eye blink detection with OpenCV, Python, and DLIB: https://www.pyimagesearch.com/2017/04/24/eye-blink-detection-opencv-python-dlib/

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 2, June 2022
Impact Factor: 6.252

[3]. Drowsiness Detection with OpenCV: https://www.pyimagesearch.com/2017/05/08/drowsiness-detection-opencv/
[4]. Real-Time Eye Blink Detection using Facial Landmarks:
[5]. http://vision.fe.uni-lj.si/cvww2016/proceedings/papers/05.pdf