e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:04/Issue:04/April-2022 Impact Factor- 6.752 www.irjmets.com

DRIVER'S DROWSINESS DETECTION SYSTEM USING DLIB

Prathamesh Kumthekar[*1], Suraj Patil[*2], Shounak Kulkarni[*3], Sumit Patil[*4], Sammed Chougule[*5], Dr. (Mrs.) A. A. Agashe[*6]

[*1,2,3,4,5]Student, Department Of Electronics Engineering, Walchand College Of Engineering, Sangli, Maharashtra, India.

[*6]Assistant Professor, Department Of Electronics Engineering, Walchand College Of Engineering, Sangli, Maharashtra, India.

ABSTRACT

Drowsiness is a condition where driver is not in fully alert mode (In this condition the driver is somewhat sleepy). Out of the road accidents that happen worldwide most of these occur due to this drowsy state and Fatigue of driver's. In this system, with the use of DLIB library, EAR and MAR algorithm we constantly monitor the driver's face through webcam or any other camera, this system will detect whether there is drowsiness or not (According to conditional logic) and thus will give alert alarm to the driver through speaker. To reduce the number of accidents due to drivers fatigue and hence increase the transportation safety; this system deals with automatic driver's drowsiness detection. We propose an algorithm to locate, track, and analyze both the driver's eyes and mouth.

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

I. INTRODUCTION

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. Despite the fact that 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 drivers by monitoring the eyes and mouth while they're driving. The technology recognizes the face in the video picture. To narrow down the system's ability to identify the eyes and mouth inside the face region, the face area is detected and used. Detecting the lips and eyes is the next step after finding the face. To detect the eyes and lips, the video stream is converted into picture frames per second. It is possible to tell if the eyes are open or closed by monitoring the eye aspect ratio (EAR) and to know whether the person is yawning or not by monitoring mouth aspect ratio (MAR).

II. METHODOLOGY

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.

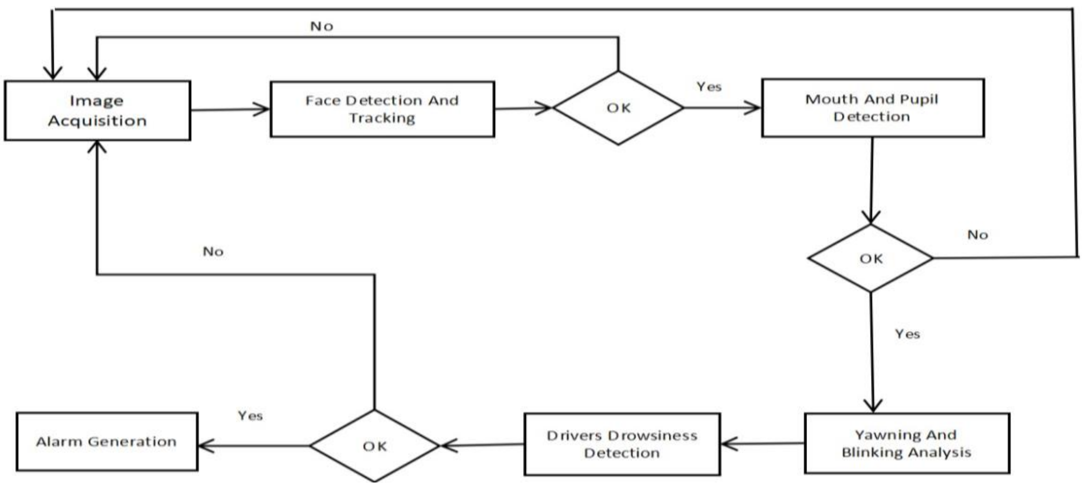

Figure 1: Workflow of Drowsiness Detection.

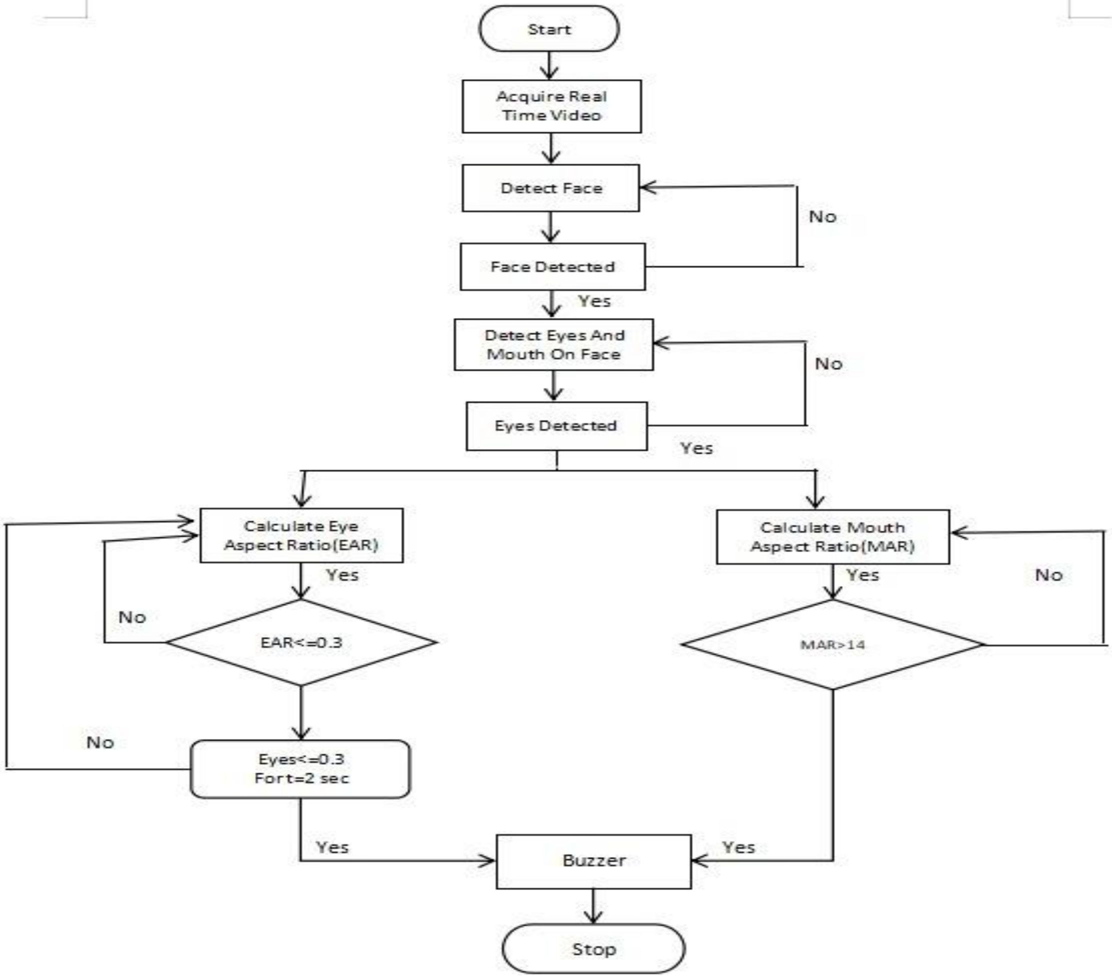

Figure 2: System Architecture.

Design Methodology

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 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 of time which is set, an alert sound is activated alerting the driver about the drowsiness.

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.

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:04/Issue:04/April-2022 Impact Factor- 6.752 www.irjmets.com

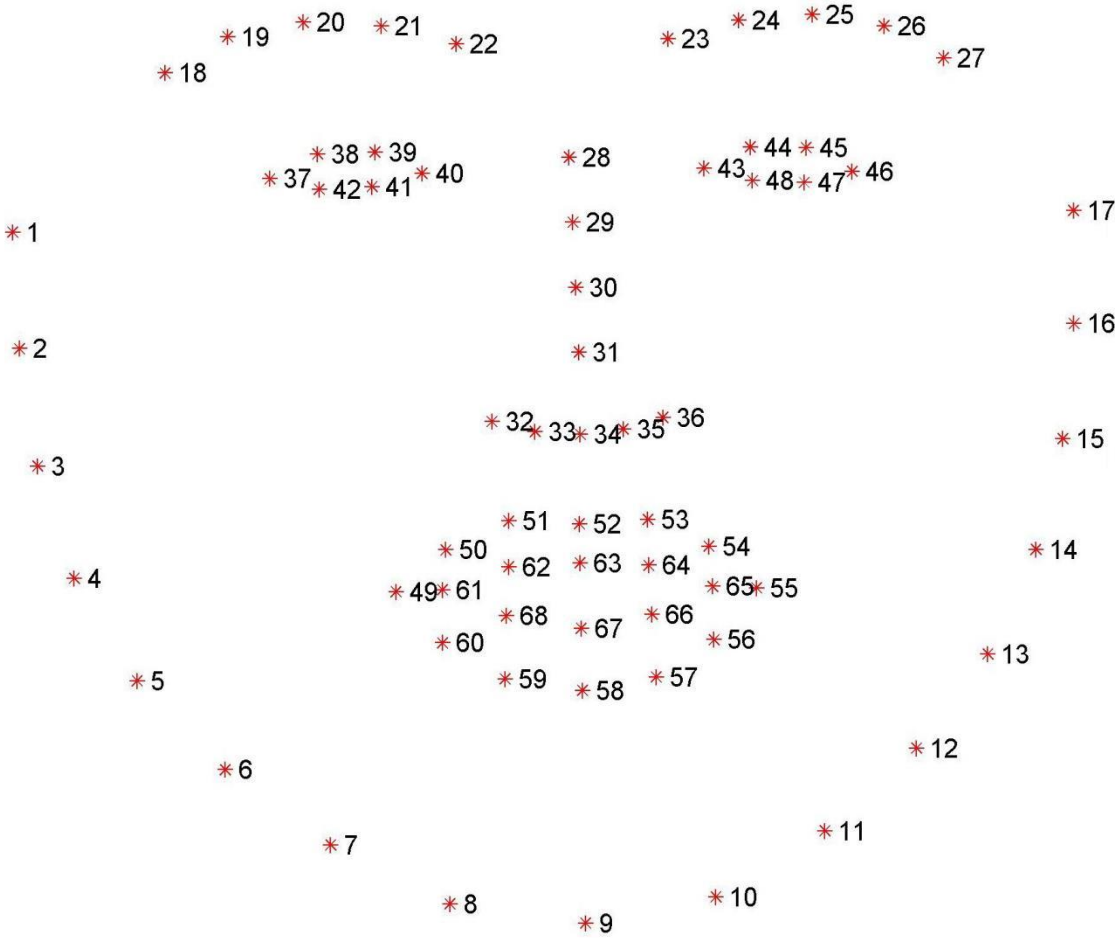

Figure 3: 68 Point View.

EAR Algorithm

Part	Landmark Points
Left Eye	[37-42]
Right Eye	[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. In order 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.

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.

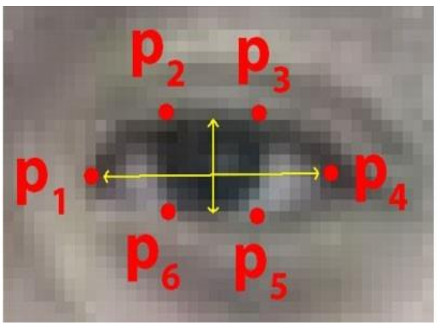

Figure 4: Eye Point View.

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

Equation 1

MAR Algorithm

Mouth Aspect Ratio (MAR) is a measure to find how wide open the mouth is extended from driver-drowsiness systems where it was used for detecting yawns.

The formula used to calculate 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
```

III. RESULTS AND DISCUSSION

False Acceptance Rate (FAR) : The percentage of identification instances in which wrong detections are incorrectly accepted.

False Rejection Rate (FRR) : The percentage of identification instances in which correct detections are incorrectly rejected.

Table 1. System Testing

Test Cases	Eye closure detected	Yawn detected	Result
Case 1	No	No	No result
Case 2	Yes	No	Alarm generation
Case 3	No	Yes	Alarm generation
Case 4	Yes	Yes	Alarm generation

Table 2. Performance Parameters

Sr. No.	Performance Parameters	Total Cases	Rate
1	FAR (False Acceptance Rate)	60	6.66%
2	FRR (False Rejection Rate)	60	5%
3	Response Time	20	0.4 Sec

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:04/Issue:04/April-2022 Impact Factor- 6.752 www.irjmets.com

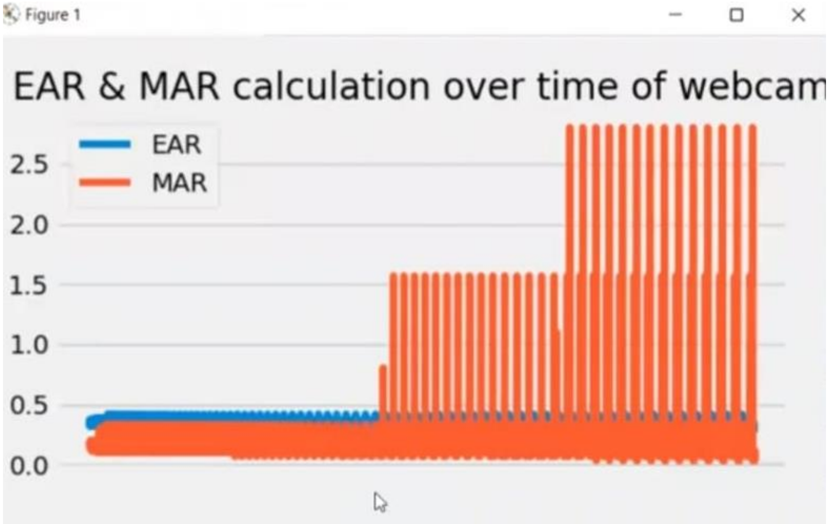

Figure 5: EAR & MAR calculation over time of webcam

Figure 6: Output 1

Figure 7: Output 2

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:04/Issue:04/April-2022 Impact Factor- 6.752 www.irjmets.com

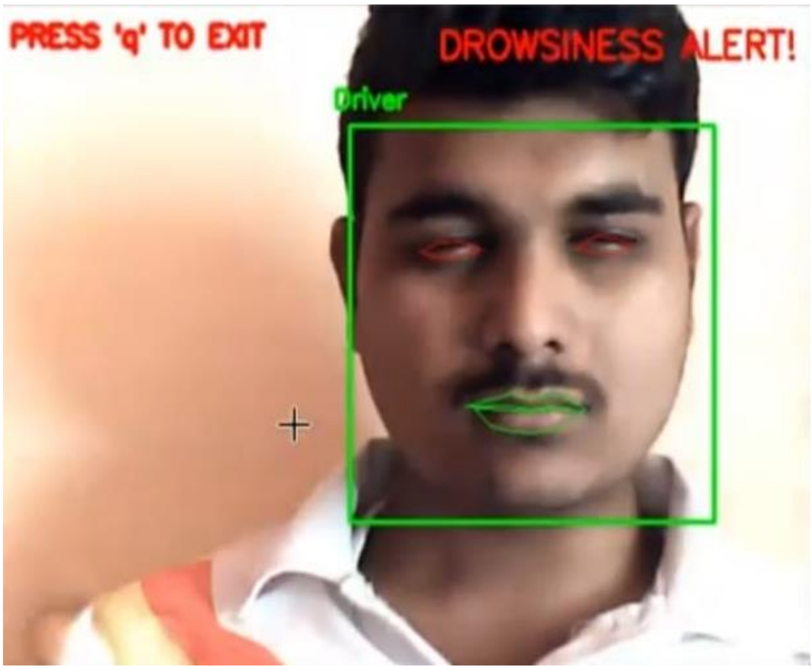

Figure 8: Output 3

IV. CONCLUSION

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

- DLIB
- Measurement of EAR & MAR
- Conditional logic

The FAR & FRR of our system is which is very low and also 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.

ACKNOWLEDGEMENTS

We are highly indebted to Dr. (Mrs.) A. A. Agashe and are obliged for giving us the autonomy of functioning and experimenting with ideas. We would like to take this opportunity to express our profound gratitude to them not only for their academic guidance but also for their personal interest in our project and constant support coupled with confidence boosting and motivating sessions which proved very fruitful and were instrumental in infusing self-assurance and trust within us.

V. 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/

[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