Design of Real-time Drowsiness Detection System using Dlib

[1]Shruti Mohanty, [1]Shruti V Hegde, [1]Supriya Prasad, [2]J. Manikandan
[1]Dept of ECE, PES University, 100-Feet Ring Road, BSK Stage III, Bengaluru - 85, Karnataka, India
[2]Dept of ECE, Crucible of Research and Innovation (CORI)
PES University, 100-Feet Ring Road, BSK Stage III, Bengaluru - 85, Karnataka, India
{email : shrutimohanty998@gmail.com, shrutihegde98@gmail.com, supriya.sprasad@gmail.com, manikandanj@pes.edu}

***Abstract*—Drowsiness while driving is a highly prevalent problem that leads to thousands of fatal accidents every year. A solution to prevent accidents and fatalities is the need of the hour and while there are complex systems developed that provide solutions for detecting drowsiness in drivers, this paper explores a simpler, yet highly effectual method of doing the same. In this paper, drowsy driver detection system is designed using Python and Dlib model. Dlib's shape detector is used to map the coordinates of the facial landmarks of the input video and drowsiness detected by monitoring aspect ratios of eyes and mouth. Performance evaluation of the proposed system designed is carried out by testing videos from a standard public dataset as well as real-time video captured in our lab. The proposed system gave a maximum recognition accuracy of 96.71% for dataset video input.**

***Keywords*— Python, face detection, drowsiness detection, computer vision, Dlib, OpenCV, HOG, facial landmark estimation**

I. Introduction

Facial expressions have the ability to offer deep insights into many physiological conditions of the body. The display of a myriad of emotions and reactions to stimuli have been a constant area of study and research, and also used in the development of intelligent systems such as facial emotion detection in [1], automatic pain detection in [2] and prediction of personality in [3] to name a few. There are numerous algorithms and methodologies available for face detection which is the fundamental first step in the process.

There has been extensive research and a number of papers that have put forth possible methodologies to detect inattentiveness and drowsiness in a driver in the last two decades. In [4], traditional techniques are elaborated which are based on physiological measurements including brain waves, heart rate, pulse rate and respiration. However, these techniques are intrusive in nature. Reference [5] is based on Rowley's eye detection code from the STASM library. However, the presence of glasses adversely affects the performance of the system. Reference [6] monitors only yawning patterns of the driver using two separate cameras to acquire information of the upper part of the body in order to track the driver's mouth. However, the hardware dependency is higher.

Drowsiness in humans is characterized by a few very specific movements and facial expressions- the eyes begin to close, mouth opens in a yawn, the jaw goes slack and the neck tilts. This paper focuses on tracking the eyes and mouth to detect drowsiness and classify a driver as drowsy. 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, hands, upper body and occlusions such as non-tinted spectacles.

The Dlib model is trained to identify 68 facial landmarks. As shown in Fig 1. the drowsiness features are extracted and the driver is alerted incase of drowsiness being detected. The model does not require prior information on the individual who is testing it. Main software requirements are Python and OpenCV.

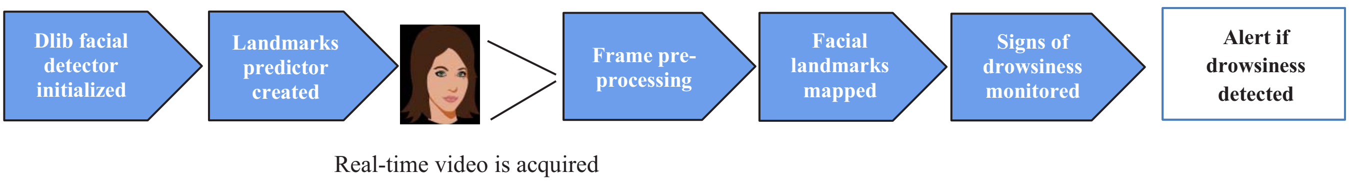


Fig. 1: Block diagram of proposed real-time drowsiness detection system

978-1-7281-4499-3/19/$31.0 ©2019 IEEE

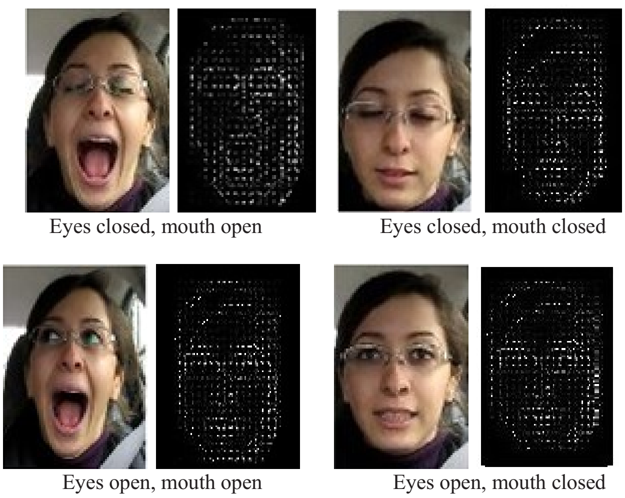
II. IMPLEMENTATION

For this approach, we implement the drowsiness detector using OpenCV and Python. The Dlib library is used to detect and localize facial landmarks using Dlib's pre-trained facial landmark detector. It consists of two shape predictor models [7] trained on the i-Bug 300-W dataset, that each localize 68 and 5 landmark points respectively within a face image. In this approach, 68 facial landmarks have been used (as shown in Fig. 2 below).

Fig. 2: Manner in which 68 facial landmarks are mapped on a detected face [8]

Histogram of Oriented Gradients (HOG) based face detector is used in Dlib. In this method, frequencies of gradient direction of an image in localized regions are used to form histograms. In many cases, it is more accurate than Haar cascades as the false positive ratio is small. Also, tuning at test time requires less parameters. It is especially suitable for face detection as firstly, it can describe contour and edge features exceptionally in various objects. Secondly, it performs operations on regional cells which allows motion of the subject to be overlooked. Moreover, Dalal and Triggs [9] discovered that HOG descriptor works well for human detection in images, which makes it appropriate for drowsiness detection.

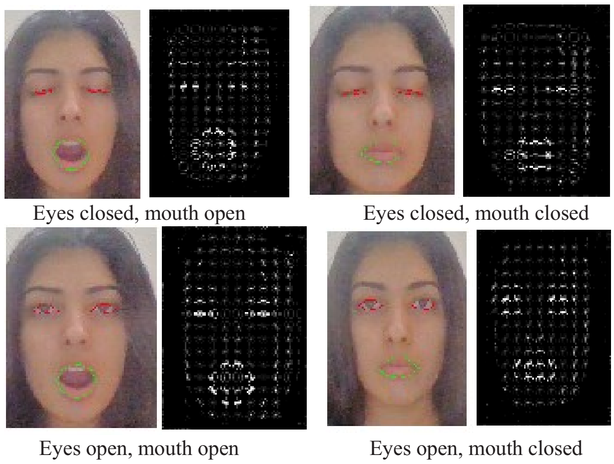
In our model, a HOG based detector is first instantiated to find the location of the face in each individual frame of the input video stream.



Eyes closed, mouth open   Eyes closed, mouth closed

Eyes open, mouth open   Eyes open, mouth closed

Fig. 3: HOG face features for four different dataset input cases



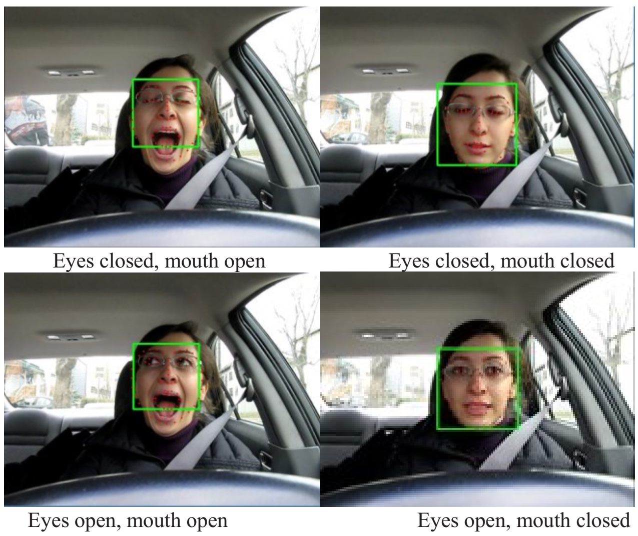
Eyes closed, mouth open   Eyes closed, mouth closed

Eyes open, mouth open   Eyes open, mouth closed

Fig. 4: HOG face features for four different real-time input cases

The outline of the facial features made by the oriented gradients makes it easy to discern the location and even the state of facial features. For example, in Fig. 3 and Fig. 4 we can see the difference in the HOG of an open mouth versus that of a closed mouth.

Upon finding the location of the face, the facial landmarks predictor is called to map the points of interest (eyes and mouth) and extract their coordinates.



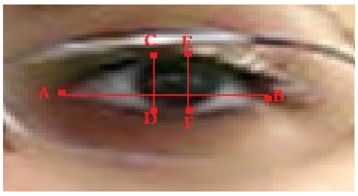
Eyes closed, mouth open   Eyes closed, mouth closed

Eyes open, mouth open   Eyes open, mouth closed

Fig. 5: Facial landmark mapping for four different dataset cases

Fig. 5 shows the facial landmark mapping on particular frames extracted from the video input for different cases such as eyes closed mouth open, eyes open mouth closed, eyes open mouth open and eyes closed mouth closed.

The coordinates of the right eye, left eye and mouth extracted at this stage are used to compute aspect ratio for the right eye, left eye and mouth based on Euclidean distance (as shown in Fig. 6 below).

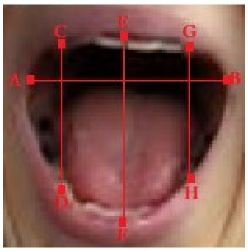


$$EAR = \frac{|CD| + |EF|}{2 * |AB|}$$

Fig. 6: Eye coordinates

From the formula in Fig. 6, the eye coordinates are obtained and eye aspect ratio (EAR) is calculated according to the formula. The aspect ratio of both the eyes is averaged as blinking is performed by both simultaneously.

Similarly, mouth aspect ratio is determined to detect yawning from the coordinates of the mouth and the formula shown in Fig. 7 below.



$$MAR = \frac{|CD| + |EF| + |GH|}{3 * |AB|}$$

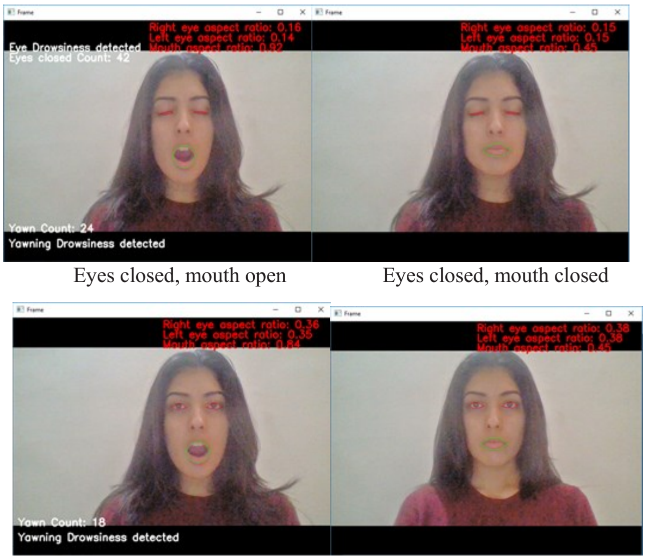
Fig. 7: Mouth coordinates

The final display of the drowsiness detection system shows the feed of the video input (from video dataset or real-time capture), along with the computed aspect ratio values and drowsiness detection alerts. If the aspect ratio of the eyes falls below the stipulated threshold, then the message “Eye Drowsiness detected” flashes on the screen along with the count of how many times the eye was noticed to be closed. If the aspect ratio of the mouth falls below the stipulated threshold, then the message “Yawning Drowsiness detected” is displayed on the screen along with the count of how many times yawning was detected.

Eyes closed, mouth open | Eyes closed, mouth closed

Eyes open, mouth open | Eyes open, mouth closed

Fig. 8: Drowsiness detection on a dataset video for 4 different cases (inset text: aspect ratios of both eyes and mouth and drowsiness count)



Eyes closed, mouth open | Eyes closed, mouth closed

Eyes open, mouth open | Eyes open, mouth closed

Fig. 9: Drowsiness detection on a real-time video for 4 different cases (inset text: aspect ratios of both eyes and mouth and drowsiness count)

The following steps are followed for the testing of the model:

Step 1: Input video (pre-recorded or real-time) is fed into the model. Individual frames are resized and converted to grayscale.

Step 2: Dlib’s HOG based face detector is initialised. The location of the face is pinpointed.

Step 3: The facial landmarks for the face region are determined by the predictor and mapped onto the face.

Step 4: Left eye, right eye and mouth coordinates are extracted, which are then used to compute aspect ratio for both eyes and mouth based on Euclidean distance respectively.

Step 5: The calculated aspect ratios are compared with fixed threshold values 0.15 and 0.83 for eye and mouth respectively to determine signs of drowsiness. If the average aspect ratio of left and right eye falls below the threshold, it is recognized as a sign of drowsiness. Similarly, if the mouth aspect ratio exceeds the set threshold, there is a possibility for it to be a yawn.

Step 6: When continuous signs of drowsiness is detected over a longer duration, the driver is alerted.

The real time-video is processed at 20 frames per second (fps), so each frame lasts for 0.05 seconds. Drowsy blinks typically last for 20 frames i.e., 1 second. Thus, a normal blink will not be identified as drowsy. Continuous eye blinks also last for lesser number of frames and are hence distinguishable from drowsy blinks.

III. EXPERIMENTAL RESULTS

A. Dataset Description

Table I provides a description of the 2 datasets used for testing purposes.

TABLE I: DATASET DESCRIPTION

Feature	YawDD Dataset[10]	MRL Eye Dataset[11]
Number of videos/images	29 Videos	84,898 Images
Number of males	16	33
Number of females	13	4
Actions performed	Without talking or talking/singing or yawning	Closed or open eyes
Resolution	640x480 RGB (24-bit true colour)	640x480, 1280x1024, 752x480

B. Results Obtained from Comparison

Table II below describes the recognition accuracy obtained from two approaches (eye closure and yawn detection), on using standard datasets and real-time.

Real-time computational results were calculated by taking the average of 5 trials each of 12 subjects (including 5 males and 7 females) recorded at different locations. Average result included cases with and without glasses. Video frames with instances of 2 states (sleepy and non-sleepy) for every trial. Highest percentage accuracy obtained is 96.71 % for yawn detection followed by 93.25% for drowsy blink detection.

TABLE II: RESULTS OBTAINED

Features	Recognition Accuracy	
	Real Time	***Dataset***
Eye	82.02 %	93.25 %
Yawn	85.44 %	96.71 %

The results of real-time detection are lower as the model currently works exceedingly well under good to perfect light conditions like those found in the dataset videos, whereas the real-time testing was performed under a variety of lighting conditions.

Future work will focus on enhancing the model to work under poor to mediocre lighting conditions, and including more drowsiness signs such as head nodding for the drowsiness detection model.

IV. CONCLUSIONS

In the Dlib approach, the library's pre-trained 68 facial landmark detector is used. The face detector which is based on Histogram of Oriented Gradients (HOG) was implemented. The quantitative metric used in 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. The average real-time test accuracies obtained using Dlib for eyes and yawn were 82.02% and 85.44% respectively and 93.25% and 96.71% respectively for pre-recorded videos.

REFERENCES

[1] M. Matsugu., K. Mori., Y. Mitari, and Y. Kaneda, "Subject independent facial expression recognition with robust face detection using a convolutional neural network," *Neural networks: the official journal of the Intern. Neural Network Society*, vol. 16, no. 5-6, pp. 555-559, June 2003.

[2] P. Lucey, J. Cohn, S. Lucey, I. Matthews, S. Sridharan and K. M. Prkachin, "Automatically detecting pain using facial actions," in *3rd Intern. Conf. on Affective Computing and Intelligent Interaction and Workshops*, Amsterdam, 2009, pp. 1-8.

[3] L. Liu, D. Preotiuc-Pietro., Z.R. Saman, M.E. Moghaddam, and L.H. Ungar, "Analyzing Personality through Social Media Profile Picture Choice," *Intern. AAAI Conf. on Web and Social Media*, Cologne, Germany, May 2016, pp. 214.

[4] P. K. Stanley, T. Jaya Prakash, S. Sibin Lal and P. V. Daniel, "Embedded based drowsiness detection using EEG signals," *IEEE Intern. Conf. on Power, Control, Signals and Instrumentation Engineering*, Chennai, India, Sept. 2017, pp. 2596-2600.

[5] T. Danisman, I. M. Bilasco, C. Djeraba and N. Ihaddadene, "Drowsy driver detection system using eye blink patterns," *Intern. Conf. on Machine and Web Intelligence*, Algiers, Algeria, Oct. 2010, pp. 230-233.

[6] L. Li, Y. Chen and Z. Li, "Yawning detection for monitoring driver fatigue based on two cameras," *12th Intern. IEEE Conf. on Intelligent Transportation Systems*, St. Louis, USA, Oct. 2009, pp. 1-6.

[7] Davis E. King, Dlib-models [Online]. Available: https://github.com/davisking/dlib-models

[8] L. Anzalone, *Training Alternative Dlib Shape Predictor models using Python,* Oct. 2018. Accessed on July 16, 2019. [Online]. Available: https://medium.com/datadriveninvestor/training-alternative-dlib-shape-predictor-models-using-python-d1d8f8bd9f5c

[9] N. Dalal, B. Triggs "Histogram of oriented gradients for human detection", *IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, San Diego, USA, June 2005, pp. 63-69.

[10] S. Abtahi, M. Omidyeganeh, S. Shirmohammadi, and B. Hariri. 2014. *YawDD: a yawning detection dataset, 5th ACM Multimedia Systems Conf.*, New York, USA, Mar. 2014, pp. 24-28.

[11] R. Fusek, *MRL Eye Dataset*, Accessed on May 28th, 2019. [Online]. Available: http://mrl.cs.vsb.cz/eyedataset