International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 04 Issue: 08 | August -2020 ISSN: 2582-3930

Driver Drowsiness Detection System

Ajeet Singh [1], Ashish Srivastava [2], Divyam Srivastava [3], and Abhishek [4]

[1] B.Tech Student, Dept. Computer Science and Engineering, IMS Engineering College, Ghaziabad
[2] B.Tech Student, Dept. Computer Science and Engineering, IMS Engineering College, Ghaziabad
[3] B.Tech Student, Dept. Computer Science and Engineering, IMS Engineering College, Ghaziabad
[4] B.Tech Student, Dept. Computer Science and Engineering, IMS Engineering College, Ghaziabad

ABSTRACT

On road more than 30% of reported road accidents occur due to driver's fatigue and drowsiness. Driver's drowsiness state can be predicted by using various techniques. Such as by monitoring biomedical signals or by doing visual assessment of face mages and check driver's bio-behavior, also by putting an eye on drivers performance or by combining all above techniques. This algorithm depends on live measuring the EAR (Eye aspect Ratio) by utilisation of Image processing. Live video is input and taken as form of continues frames and it detects facial landmarks using pre trained neural netwok baese functions of Dlib. These functions are trained using HAAR Cascade algorithm. OpenCV is intel's Open source Iibrary. It is used image processing tasks. It is used primary tool for image processing. Python language is used for programming. Eye coordinates are detected and distance between them is measured using Euclidean distance. In the same way MAR is calculated using mouth coordinates. Later using machine learning algorithm KNN drowsiness

and Blinks are displayed on screen and audio warning is given for drowsy detection.

General Terms

Computer vision, Neural network, Deep learning, Image processing , Facial Landmark detection, Region of Interest (ROI).

1 Introduction

Alertness of driver is an important factor for safer drive. sleep deprivation or sleep disorders results to drowsiness of driver, an important factor for the increament in number of the accidents happening on roads nowdays. Driver drowsiness detection system can help in reducing these accidents happening due to driver drowsiness. There were 824 fatalities (2.2% of all fatalities) recorded in a single year in NHTSA's FARS database that were related to drowsy driving. A study was presented at the International Symposium on Sleep Disorders and driver's fatigue. And it says drowsi-driving is responsible for 30% of road accidents[1].Number of Different techniques are used in various driver-fatigue monitoring systems. There are four categories in which these techniques are divided. The first category contains intrusive techniques, these are mostly based on monitoring biomedical signals, and therefore physical contact with the driver is required for these techniques. The second category is non-intrusive techniques which are based on determining driver's bio-behavior using face images. The third category is of methods based on performance of driver, which monitor vehicle behavior such as moving course, braking, steering angle, speed, etc. and the fourth category is combination of techniques from the above mentioned three categories. The computer vision based techniques from the second category are much effective because drowsiness state of any person can be detected by observing it facial expressions, facial landmarks and visual bio-behavior such as gaze, eyelid movements, eye openness, head position, eyelid movements, and mouth openness. This algorithm is based on computer vision method. it detects blinks by estimating EAR(eye aspect ratio). It is done by monitoring eyes of person/driver throughout the entire video. Through a webcam live video of driver eyes will be captured in all light conditions and frames will be extracted for image processing

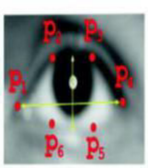
International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 04 Issue: 08 | August -2020 ISSN: 2582-3930

scheme. Algorithm can be divided broadly in following sub modules.

- Taking frame
- Applying grayscale and resizing
- Face detection
- Detecting facial landmarks
- Localization and tracking of eyes and mouth
- Extracting eye geometrical coordinates.
- Measuring Eye Aspect Ratio and mouth aspect ratio
- Monitoring of Eye aspect ratio for blinks detection and monitoring Mouth aspect ratio for yawn detection.
- Audio Visual warning

Eye Aspect Ratio:

EAR is define as per below formula , **where f1,f2….f6** is the eye coordinates as discussed in further text

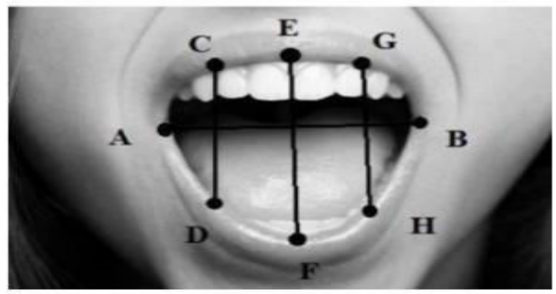


$$EAR = \frac{\|p_2 - p_6\| + \|p_3 - p_5\|}{2\|p_1 - p_4\|}$$

EAR (Eye aspect Ratio) = (|f2-f6|+|f3-f6|) /(2*|f4-f1|)

Mouth Aspect Ratio:

Mar is defines as per below formula, where A,B,C,D,E,F,G,H are the coordinate of mouth



MAR (Mouth aspect Ratio) = (|C-D|+|E-F|+|G-H|) /(2*|A-B|)

Driver drowsiness state : It is a state where driver is feeling of being sleepy and lethargic.

2 Literature Review

2.1 Face and Eye Detection by CNN Algorithms[3]

During this paper a awfully unique approach to critical parts of face detection problems is given, supported analogic cellular neural network (CNN) algorithms. The proposed CNN algorithms find and help to normalize human faces effectively which is a cause for cause for several accident related to the vehicles crashes. Driver fatigue their time requirement could even be a fraction of the previously used methods. The algorithm starts with the detection of heads on colour pictures using deviations in colour and structure of the external part which of the background. By normalizing the space and position of the reference points, all faces should be transformed into the identical size and position. For normalization, eyes function points of reference. Other CNN algo finds the eyes on any grayscale image by searching for the attribute of the eyes and eye sockets. Tests made on a typical database show that the algorithm works in no time and it's reliable[3].

2.2 Face Detection using Haar Cascades[4]

Object Detection using Haar feature-based cascade classifiers are often an genuine object detection method proposed by Paul Viola and Michael Jones in their paper, "Rapid Object Detection employing a Boosted Cascade of straightforward Features" in 2001. It is an approah based on machine learning where a cascade function is trained from many positive and negative images. it's then accustomed detect objects in other images. Here we'll work with face detection. Initially, the algorithm needs many positive examples (images of faces) and negative examples (images without faces) in order to point the classifier. Then we wish to extract features from it. For this, Haar features shown within the below image are used. they're almost like our convolutional kernel. Each feature could even be one value obtained by subtracting sum of pixels under the white rectangle from sum of pixels under the black rectangle[4].

2.3 Eye Detection Using Morphological and Color Image Processing[5]

Applications of Eye Detection include eye-gaze tracking, iris detection, video conferencing, auto-stereoscopic displays, face detection and face recognition. This paper pt an emphasis on very technique which can be used for eye detection using color and morphological image processing. it's observed that eye regions during a picture are characterized by low illumination, high density edges and high contrast as compared to other parts of the face.

the strategy proposed relies on assumption that a frontal face image (full frontal) is out there. Firstly, the technique which can be used for skin region detection is color-based and six-sigma which is employed on training algorithm and operates on RGB,HSV and NTSC scales. Further analysis involves morphological processing using boundary region detection and detection of sunshine source reflection by a watch, commonly called a watch fixed fixed fixed dot. this provides a finite number of eye candidates from which noise is subsequently removed. this method is found to be highly efficient and accurate for detecting eyes in frontal face images[5].

2.4 Algorithm on Gray Intensity Face for Eye Detection[6]

This paper presents a sturdy eye detection algorithm for grey intensity images. the concept of our method is to combine the respective advantages of two existing techniques, feature based method and template based method, and to beat their shortcomings. Firstly, after things of face region is detected, a feature based method are visiting be accustomed detect two rough regions of both eyes on the face. Then an accurate detection of iris centres are visiting be continued by applying a template based method in these two rough regions. Results of experiments to the faces without spectacles show that the proposed approach isn't only robust but also quite efficient[6].

2.5 Real-Time Face Detection Using EdgeOrientation Matching[7]

During this paper we describe our ongoing work on real-time face detection in grey level images using edge orientation information. we are visiting show that edge orientation are visiting be a sturdy local image feature to model objects like faces for detection purposes. we are visiting present a simple and efficient method for template matching and object modelling based solely edgy orientation information. We also show the thanks to obtain an optimal face model within the sting orientation domain from a gaggle of coaching images. Unlike many approaches that model the grey level appearance of the face our approach is computationally in no time. The time it’s take to process 320x240 image employing a multi-resolutin search with six resolution levels is 0.081 seconds. We exhibit the potential of our detection method on an image database of 17000 images taken from

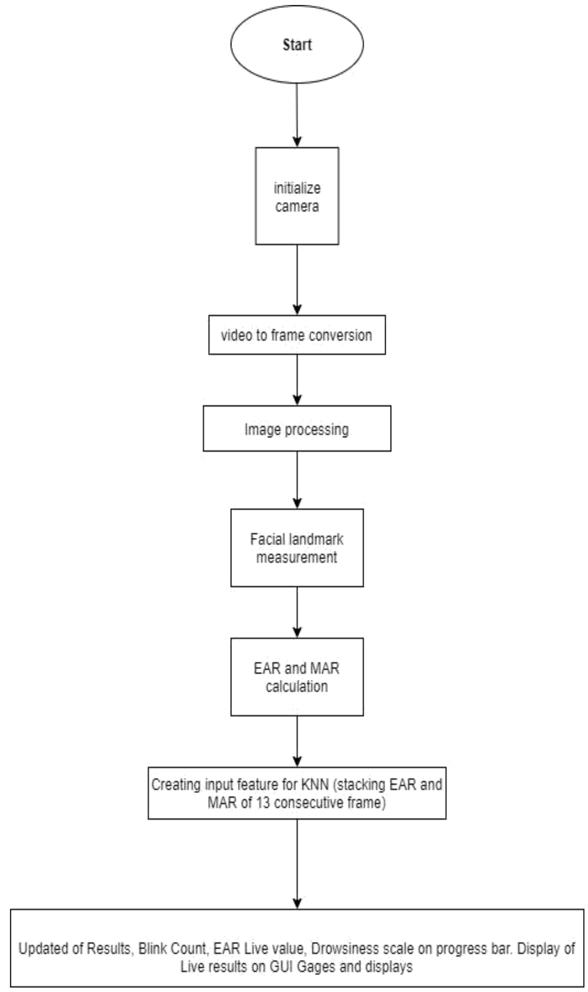
International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 04 Issue: 08 | August -2020 ISSN: 2582-3930

quite 2900 different people. The variations in head size, lighting and background are substantial. The obtained detection rate is sort of 93% thereon database[7].

3 Implementation

After further analysis, we finally arrived at the four main steps to realize our aims.

1. Real-time face attributes detection.
2. Calculate EAR and MAR
3. KNN
4. Drowsiness detection and audio-visual warning



3.1 Real-time face attributes detection

The first step towards goal in blink detection is face recognition and the localizing facial features. It is important to localize eyes. It is accomplished with the help of built-in shape predictor file in the Dlib library. The shape predictor identifies 68 landmarks on a face and assigns coordinates to each of them with respect to the bounding box of the face. These individual coordinates can be extracted from the created "shape" object as NumPy arrays. Using Dlib's coordinates for the left and right eyes, it is possible to highlight the eyes alone, so that a live stream will capture each frame and mark the contours of the eyes corresponding to the upper and lower lids.

3.2 Calculate EAR and MAR

Dlib predictor function returns 6(x,y) coordinates in landmarks representing each eye, taking the left-corner of the eye (as if you were looking at the person) as starting point, and then working in clockwise direction around the remainder of the region. There exists a relation between the width and the height of these coordinates.

Python Function for calculating EAR

def eye_aspect_ratio(self, eye):

A = dist.euclidean(eye[1], eye[5])

B = dist.euclidean(eye[2], eye[4])

C = dist.euclidean(eye[0], eye[3])

ear = (A + B) / (2.0 * C) return ear

Similarly mouth is represented by 8(x,y) coordinates in landmarks returned by Dlib predictor function. And calculate MAR using the above explained equation.

5. Drowsiness detection and audio-visual warning

Python Function for calculating EAR

def mouth_aspect_ratio(self, eye):

k = dist.euclidean(mouth[C], mouth[D],)

l = dist.euclidean(mouth[E], mouth[F])

m = dist.euclidean(mouth[G], mouth[H])

n = dist.euclidean(mouth[A], mouth[B])

mar = (k + l + m) / (2.0 * n)
return mar

3.2 KNN (K nearest neighbour)

Once EAR and MAR are calculated for each frame of real time feed a significant decrease in value of EAR is expected when person closes his eyes and regaining normal EAR value when eyes open.

Using the general approximation of the EAR value, for a still frame, if the aspect ratio is less than, the eyes are considered to be closed. Correspondingly, a calculated EAR of above 0.3 could indicate that the eyes are open. For a live stream however, it must be appreciated that the length each frame is not significant enough discern whether the eyes are open or not.

For a driver to be pronounced as drowsy, the eyes must remain closed for a good amount of time; roughly four to seven seconds. Therefore, it becomes essential to check whether the EAR remained below the threshold value for a fixed number of frames, depending on the frame rate of the camera used. We observed that there was a noticeable decrease in the original distance (seen in the open eyes) between the y coordinates of the two upper and two lower points in closed eyes. We set a fixed number of frames for which closed eyes must be detected above which the driver is assumed to be drowsy. Individual calibration is carried out through a simple and interactive interface. For each eye, the average vertical distance between the top and bottom eyelid is obtained. To conclude that the user is drowsy, both eyes must have their measured distance values below their respective thresholds.

But only this method is not as reliable as we though so we are also using a Machine learning algorithm **KNN(k nearest neighbor).**

Considering an approach where the consecutive frames EAR's were concatenated to create a Driver drowsiness state feature. This feature has three categories : open eyes(0) , blink(1) or closed (2).

Blink occurs when there is a slight touch of eyelids in the 6 central frames of driver drowsiness state feature. This KNN models uses 26 dimension input (13 consecutive EAR's and 13 consecutive MAR's) and returns a scalar (input feature) and output feature is drowsiness state.

K nearest neighbor (KNN) was used a classification learning algorithm, to predict the driver drowsiness state. For our training and test data, we used the Real life drowsiness data created by a research team from the University of Texas at Arlington specifically for detecting multi-stage drowsiness.

Later the model predicts driver drowsiness state

3.4 Driver drowsiness detection and audio-visual warning

After completing the several training processes when we start to get satisfactory results, real time monitoring can be done from than onwards. After every 6 new frames received from the video feed.

A new state feature is created which has 30 parameters and it is used by KNN classification algorithm to predict class of driver drowsiness.

And it helps in predicting whether driver feeling drowsy or not.

These conditions are considered as drowsiness state:

1. If within 30 seconds, 5 or more outputs are closed eye(2). It generally occurs when user's eyes close for a slow blink.
2. When 5 or more consecutive ouptput predicted by KNN is closed eye(2). This occurs if user's eye is closed for more than a second

Whenever drowsiness is detected, an alarm is rung to wake up the drowsy driver.

4 Result

Algorithm	Acc. Without glasses	Acc. With glasses	Total accuracy
EAR	59.8	45.4	52.6
KNN	85.3	80.2	82.75

We can see from the above results that KNN performs much better in driver dworsiness state prediction while compared to using consecutive frames EAR's.

A large number of high-end cars already have advanced drowsiness detection systems. As of now, we have implemented the algorithms as simple apps that can be run on a laptop. However, as this holds little significance in real world situations, the next steps to be taken involve the design of a cost-effective device that can be easily fitted in cars. Main application of propose algorithm is in detection of drivers drowsiness state during driving

5 Conclusion and Future Work

In present paper, author demonstrated how to build a blink and drowsiness detector using OpenCV, Python, and Dlib opensource Libraries by measuring EAR and MAR. The first step in building a blink detector is to perform facial landmark detection to localize the eyes and mouth in a given frame from a video stream. The eye aspect ratio and mouth aspect ratio for each eye and mouth respectively can be calculated using Euclidian distance functions of OPEN CV , which is a singular value, relating the distances between the vertical landmark points to the distances between the horizontal landmark points. The proposed algorithm has been tested on personal car driver for testing purposes. For authentic results, the camera position was focused on the driver’s face. The results are discussed in Result section and found satisfactory. To make our blink detector more robust to these challenges further following improvements can be implemented Computing the eye aspect ratio and mouth aspect ratio for the Nth frame, along with the both aspect ratios for N – 6 and N + 6 frames, the. N – 6 and N + 6 frames, then concatenating these aspect ratios to form a 26 dimensional feature vector. Training a Support Vector Machine (SVM) on these feature vectors. The combination of the temporal-based feature vector and SVM classifier helps reduce false-positive blink detections and improves the overall accuracy of the blink detector.

5 Acknowledgement

The project was a great opportunity for us as senior students to evaluate our capabilities and Engineering skills, but also to apply the concepts learnt during our bachelor at IMS Engineering College to a concrete example. We would like to express my sincere gratitude to the people who helped me during my final project. First of all, we would like to thank Dr.Soumi Ghosh ma’am who supervised us very carefully throughout our project. She was patient, comprehensive and very helpful. We are very grateful to her for sharing her knowledge with us and giving us the support that we needed. Her precious advice guided us from the beginning to the end.

REFERENCES

[1] Association for Safe International Road Travel (ASIRT), Road Crash Statistics.http://asirt.org/initiatives/informingroadusers/road-safety-facts/road-crash-statistics, 2016

[2] Driver Drowsiness Detection using Eye-Closeness Detection (2016 12th International Conference on Signal-Image Technology & Internet-Based Systems)

[3] Journal of VLSI Signal Processing 23, 497–511 (1999) c °1999 Kluwer Academic Publishers. Manufactured in The Netherlands.

[4] https://docs.opencv.org/trunk/d7/d8b/tutorial_py_face_de tection.html

[5] Eye Detection Using Morphological and Color Image Processing Tanmay Rajpathaka, Ratnesh Kumar and Eric Schwartzb

[6] A Robust Algorithm for Eye Detection on Grey Intensity Face without Spectacles- JCS&T Vol. 5 No. 3

[7] Froba Kebbuck: Audio- and Video-Based Biometric Person Authentication, 3rd International Conference, AVBPA 2001, Halmstad, Sweden, June 2001. Proceedings, Springer. ISBN 3-540-42216-1.

[8]Rajneesh, Anudeep Goraya and Gurmeet Singh. Real Time Drivers Drowsiness Detection and alert System by Measuring EAR. *International Journal of Computer Applications* 181(25):38-45, November 2018.

[9]Soukupová T., Cech J. "Real-Time Eye Blink Detection using Facial Landmarks". 21st Computer Vision Winter Workshop. (2016)

[10] Tereza Soukupova And Jan Cech. Real-Time Eye Blink Detection Using Facial Landmarks.

[11] @inproceedings{ghoddoosian2019realistic,
title={A Realistic Dataset and Baseline Temporal Model for Early Drowsiness Detection},

author={Ghoddoosian, Reza and Galib, Marnim and Athitsos, Vassilis},
booktitle={Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops},
pages={0--0},
year={2019}

12