International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 06 Issue: 04 | Apr 2019 www.irjet.net p-ISSN: 2395-0072

Implementation on Visual Analysis of Eye State Using Image Processing for Driver Fatigue Detection

Miss. Ankita M. Bhoyar[1], Prof. S. N. Sawalkar[2]

[1]student, Computer Science and Engineering, Sipna C.O.E.T., Maharashtra, India
[2]Assistant professor, Computer Science and Engineering, Sipna C.O.E.T., Maharashtra, India

---***---

Abstract - *Driver's fatigue is one of the major causes of traffic accidents, particularly for drivers of large vehicles (such as buses and heavy trucks) due to prolonged driving periods and boredom in working conditions. In this , propose a vision-based fatigue detection system for driver monitoring, which is easy and flexible for deployment in buses and large vehicles. The system consists of modules image acquisition , image resize, Haar Cascadec Classifier, dlib facial landmark detector, 68 landmark, eye region, eye region of interest, eye aspect ratio (EAR). A robust measure of eye aspect ratio (EAR) on the continuous level of eye openness is defined, and the driver states are classified on it. In experiments, systematic evaluations and analysis of proposed algorithms, as well as comparison with ground truth on EAR measurements, are performed. The experimental results show the advantages of the system on accuracy and robustness for the challenging situations when a camera of an oblique viewing angle to the driver's face is used for driving state monitoring.*

Key Words: Haar cascade classifier, dlib face detector , eye aspect ratio (EAR), openCV, fatigue detection.

1. INTRODUCTION

Fatigue, drowsiness and sleepiness are often used synonymously in driving state description. Involving multiple human factors, it is multidimensional in nature that researchers have found difficult to define over past decades. Despite the ambiguity surrounding fatigue, it is a critical factor for driving safety. Studies have shown that fatigue is one of the leading contributing factors in traffic accidents worldwide. It is particularly critical for occupational drivers, such as drivers of buses and heavy trucks, due to the fact that they may have to work over a prolonged duration of the driving task, during the peak drowsiness periods (i.e., 2:00 A.M. to 6:00 A.M. and 2:00 P.M. to 4:00 P.M.), and under monotonous or boredom working conditions. Drowsy driving is becoming one of the most important cause of road accidents. According to many surveys around 30% of road accidents is due to the driver fatigue and the percentage is increasing every year.

Drowsiness can be due to the adverse driving conditions, heavy traffic, workloads, late night long drive etc. Lack of sleep, absence of rest, taking medicines are also causes for drowsiness. When driver drives for more than the normal period fatigue is caused and the driver may feel tiredness which will cause driver to sleepy condition and loss of consciousness. This results road accidents and death of driver or serious injuries and also claims thousands of lives every year. Drowsiness is a phenomenon which is the transition period from the awake state to the sleepy state and causes decrease in alerts and conscious levels of driver. It is difficult to measure the drowsiness level directly but there are many indirect methods to detect the driver fatigue. In this the main techniques used for eye blink detection is Eye Aspect Ratio (EAR) method. The Ear method is done by calculating a quantity named EAR. In normal condition the value of EAR is almost constant. If the driver is in fatigue the EAR value will be approximately near to zero. Thus we can detect whether the driver is in fatigue or not.

2. LITREATURE REVIEW

Drowsiness detection can be mainly classified into three aspects such as:-

1. Vehicle based measures.
2. Physiological measures.
3. Behavioural measures.

Vehicle based measures is one of the method which is used to measure driver drowsiness. This is done by placing sensors on different vehicle components, which include steering wheel and the acceleration pedal. By analysing the signals from the sensors the level of drowsiness can be determined. Commonly using vehicle based measures for detecting the level of driver drowsiness are the steering wheel movement and the standard deviation in lateral position. A steering angle sensor which is mounded on the steering of vehicle is used to measure the steering wheel movement. The number of micro-corrections on the steering wheel reduces compared to normal driving when the driver is drowsy. Based on small SWMs of between 0.5° and 5°, it is possible to determine whether the driver is drowsy and thus provide an alert if needed. Another vehicle based measure used to measure the drowsiness of driver is SDLP. Here the position of lane is tracked using an external camera. The main limitation of this method is that it dependent on external factors such as road markings, lighting and climatic conditions. Therefore, these driving performance measures are not specific to the driver's drowsiness.

Physiological measures are the objective measures of the physical changes that occur in our body because of fatigue. These physiological measures can be utilized to measure the fatigue level and can provide alert for the drivers. These physiological changes can be simply measured by respective instruments such as (ECG), electrooculography (EOG), electroencephalography (EEG) and electromyogram (EMG). Electrocardiogram is one of the physiological measures which can be utilized to measure the fatigue of driver. Here ECG electrodes are used to collect ECG signals from the body which provides the critical parameters related to Heart Rate (HR), Heart Rate Variability (HRV) and respiration rate or breathing frequency. Each of these are related to drowsiness [4]. Electroencephalography (EEG) is one of the most reliable physiological measures for drowsiness detection. EEG electrodes are placed at correct place and get data from brain. After pre-processing the data, which is acquired from the EEG electrodes can be divided into different frequency bands. The pre-processing involves artefact removal and filtering. Commonly used frequency bands include the delta (0.5–4 Hz), theta (4–8 Hz), alpha (8–13 Hz), beta (13–30 Hz), and gamma (greater than 30 Hz) bands [5]. Power spectrum of EEG brain waves is used as an indicator to detect the drowsiness of the driver.

Here, EEG power of the alpha, theta bands increases and the power of the beta bands decreases. The EEG based drowsiness detection is not easily implementable. Because the driver has to wear an EEG cap while driving a vehicle. These devices are being distractive and this is the main disadvantage of this method. Behavioural changes take place during drowsing like yawning, amount of eye closure, eye blinking etc. In normal condition the rate of yawning will be less. When the driver is in fatigue the rate of yawning will be far higher than the normal. So by observing this yawning rate we can detect whether the driver is in fatigue or not. In eye closure method the count of eye blink of the driver is measured for obtaining the condition of the driver. The average duration of a normal eye blink is 0.1s to 0.4s. That means, in one second the eye will blink at least 2 or 3 times. This is observed for a few seconds. When the driver is in fatigue the count will be far less compared to the normal condition. Thus we can detect whether the driver is in fatigue or not .

3. Proposed System

To improve the accuracy as well as to reduce the execution time of fatigue, drowsiness detection system.

3.1. Webcam –

Webcam is used to take an input image.

3.2. Image Resize –

It is used to resize input image into standard image format.

3.4. Haar Cascade Classifier –

A Haar cascade classifier is an algorithm which is used to detect the object for which it has been trained for, from the source. The haar cascade is by superimposing the positive image over a set of negative images. The training is generally done on a server and on various stages.

3.5. Dlib face detector –

It is used to find and locate the face in the image. It initializes dlibs pre-trained face detector based on a modification to the standard histogram of oriented gradients (HOG).

3.6. Facial Landmark 68 R.O.I. –

The pretrained facial landmark detector inside the dlib library is used to estimate the location of 68 (x , y)-coordinates face that map to facial structures of the face. These annotations are part of 68 point shape predictor 68.dat which the dlib facial landmark predictor was trained on. The facial landmark detection is used to localize each of the important regions of face.

3.7. Eye Region of Interest –

Extracting exact eye locations takes place. Image cropping is employed to restrict the area of work nearer to the eyes since the activity of the eyes we concentrate on. The total area of picture is reduced by cropping the image to two fifth to three fifth of the total area of the picture on the upper region with the result that the separation of eyes is performed.

3.8. Eye localization and fatigue detection –

It is used to compute the ratio of distances between the vertical eye landmarks and the distances between the horizontal eye landmarks. The return value of the eye aspect ratio will be approximately constant when the eye is open. The value will then rapidly decrease towards zero during a blink.

Following are the steps takes place for fatigue detection –

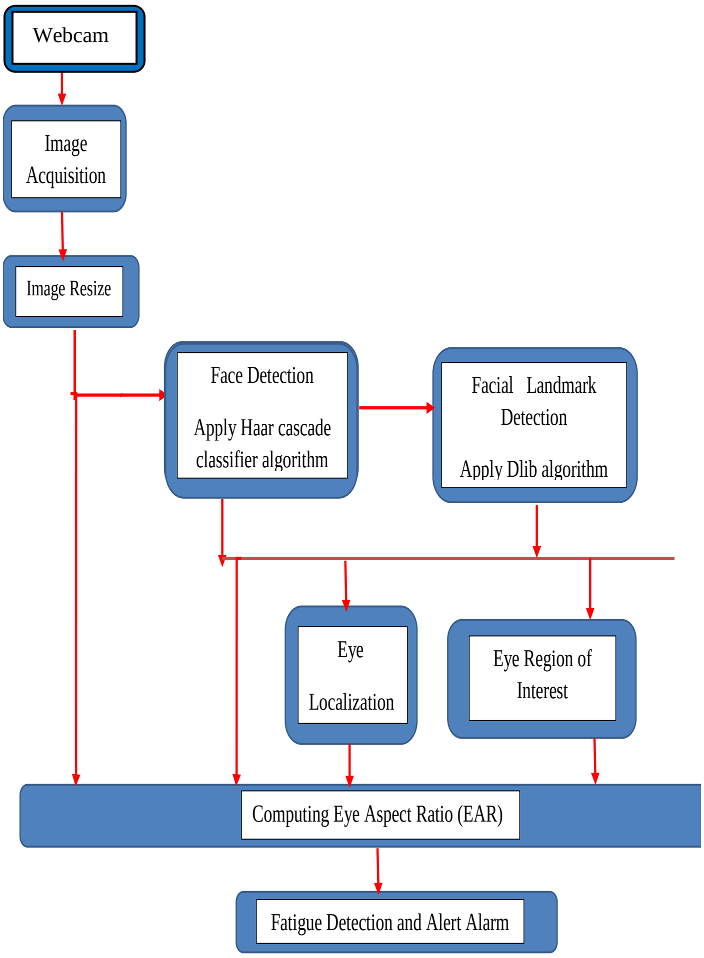


Fig -1: Framework Of proposed system

3.9. Eye Aspect Ratio(EAR)-

In this technique, we are using different landmarks to detect the opening and closing of eye. This landmark detector that capture most of the characteristic points on a human face image. It is used to compute the ratio of distances between the vertical eye landmarks and the distances between the horizontal eye landmarks. The return value of the eye aspect ratio will be approximately constant when the eye is open. The value will then rapidly decrease towards zero during a blink.

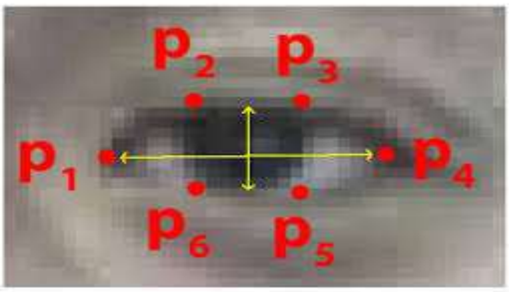


Fig .2 Landmarks obtained in EAR

The eye blink is a fast closing and reopening of a human eye. Each individual person has a little bit different pattern of blinks. The pattern differs in the speed of closing and opening of the eye, a degree of squeezing the eye and in a blink duration. The eye blink lasts approximately 100-400ms. From the landmarks detected in the image, we derive the eye aspect ratio (EAR) that is used as an estimate of the eye opening state. For every video frame, the eye landmarks are detected. The eye aspect ratio between height and width of the eye is computed. From the fig. 2 P1,P2,...,P6 are the landmarks on the eye.

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

where P1,.....,P6 are the 2D landmark locations on the eye. The EAR is mostly constant when an eye is open and is getting close to zero while closing an eye. Since eye blinking is performed by both eyes synchronously, the EAR of both eyes are taken and it is averaged. After getting the EAR value, if the value is less than the limit for 2 or 3 seconds the driver is said to be drowsy. After detecting fatigue , drowsiness with EAR threshold value the alert alarm runs and shows the message as “Drowsiness Detected”.

4. IMPLEMENTATION

Flow Diagram

Flow diagram of implementation is as follows:

4.1 Webcam

OpenCV is a Library which is used to carry out image processing using programming languages like python. This project utilizes OpenCV Library to make a Real-Time Face Detection using webcam as a primary camera.

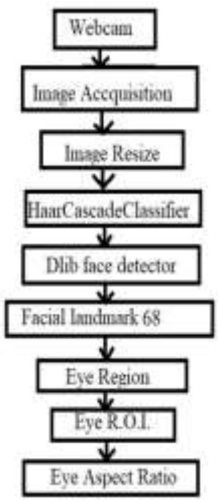


Fig.3 Flowchart of system

Following are the requirements for it:-

1. Python 2.7
2. OpenCV
3. NumPy
4. Haar Cascade Frontal face classifiers

Approach/Algorithms used:

1. This project uses LBPH (Local Binary Patterns Histograms) Algorithm to detect faces. It labels the pixels of an image by thresholding the neighborhood of each pixel and considers the result as a binary number.
2. LBPH uses 4 parameters :

(i) Radius: the radius is used to build the circular local binary pattern and represents the radius around the central pixel.
(ii) Neighbors : the number of sample points to build the circular local binary pattern
(iii) Grid X : the number of cells in the horizontal direction.
(iv) Grid Y : the number of cells in the vertical direction.
3The model built is trained with the faces with tag given to them, and later on, the machine is given a test data and machine decides the correct label for it.

How to use :
1. Create a directory in your pc and name it (say project)
2. Create two python files named create_data.py and face_recognize.py, copy the first source code and second source code in it respectively.
3. Copy haarcascade_frontalface_default.xml to the project directory, you can get it in OpenCV .

4.2 Image Resize –

Acquisition of image takes place. It is used to resize input image into standard image format.open cv haar cascade classifier supports either a single face detection or multiple faces detection (i.e., focusing on the people in a picture). We can easily create a face thumbnail of users, crop an image to focus on a person or resize an image to fill the required dimensions while making sure the person in the original picture appears in the resized version of the image.

4.3 Haar cascade classifier -

Object Detection using Haar feature-based cascade classifiers is an effective object detection method proposed by Paul Viola and Michael Jones in their paper, "Rapid Object Detection using a Boosted Cascade of Simple Features" in 2001. It is a machine learning based approach where a cascade function is trained from a lot of positive and negative images. It is then used to detect objects in other images. Here we will work with face detection. Initially, the algorithm needs a lot of positive images (images of faces) and negative images (images without faces) to train the classifier. Then we need to extract features from it. For this, Haar features shown in the below image are used. They are like convolutional kernel. Each feature is a single value obtained by subtracting sum of pixels under the white rectangle from sum of pixels under the black rectangle. Now, all possible sizes and locations of each kernel are used to calculate lots of features. For each feature calculation, we need to find the sum of the pixels under white and black rectangles. To solve this, introduced the integral image. However large an image, it reduces the calculations for a given pixel to an operation involving just four pixels. It makes things super-fast.

➢ **Haar-cascade Detection in OpenCV**

OpenCV comes with a trainer as well as detector. Its function run by Training. Here we will deal with detection. OpenCV already contains many pre-trained classifiers for face, eyes, smiles, etc. Those XML files are stored in the OpenCV/data/haarcascades/ folder. It create a face and eye detector with OpenCV. First we need to load the required XML classifiers.

Then load our input image (or video) in grayscale mode.

```
Import numpy as np
import cv2 as cv
face_cascade=cv.CascadeClassifier('haarcascade_frontalface_default.xml')
eye_cascade = cv.CascadeClassifier('haarcascade_eye.xml')
img= cv.imread('sa.jpg')
gray = cv.cvtColor(img, cv.COLOR_BGR2GRAY)
```

Now we find the faces in the image. If faces are found, it returns the positions of detected faces as Rect(x,y,w,h). Once we get these locations, we can create a ROI for the face and apply eye detection on this ROI ,since eyes are always on the face.

4.4 Dlib face detector –

dlib *is* a toolkit for making real world machine learning and data analysis applications in C++. While the library is originally written in C++, it has good, easy to use Python bindings. It is u*sed* dlib *for* face detection and facial landmark detection.The frontal face detector in dlib works really well. It is simple and just works out of the box. This detector is based on histogram of oriented gradients (HOG) and linear SVM. While the HOG+SVM based face detector has been around for a while and has gathered a good number of users. The CNN (Convolutional Neural Network) based face detector available in dlib. The HOG based face detector in dlib, is a good *"frontal"* face detector and it is, indeed. It detects faces even when they are not perfectly frontal to a good extend.

➢ **Dlib Facial Landmark Detector –**

We use dlib and OpenCV to detect facial landmarks in an image. Facial landmarks are used to localize and represent salient regions of the face, such as:

- Eyes

- Eyebrows
- Nose
- Mouth
- Jawline

Facial landmarks have been successfully applied to face alignment, head pose estimation, face swapping, blink detection and much more. The basics of facial landmarks, includes:

1. Exactly what facial landmarks are and how they work.
2. How to detect and extract facial landmarks from an image using dlib, OpenCV, and Python.

The pre-trained facial landmark detector inside the dlib library is used to estimate the location of 68 (x, y)-coordinates that map to facial structures on the These annotations are part of the shape predictor 68.dat file which the dlib facial landmark predictor was trained on. Regardless of which dataset is used, the same dlib framework can be leveraged to train a shape predictor on the input training data — this is useful if we would train facial landmark detectors or custom shape predictors.

➢ **Detecting facial landmarks with dlib, OpenCV, and Python**

The first utility function is rect_to_bb , short for "rectangle to bounding box":This function accepts a single argument, rect , which is assumed to be a bounding box rectangle produced by a dlib detector (i.e., the face detector).The rect object includes the *(x, y)*-coordinates of the detection. However, in OpenCV, we normally think of a bounding box in terms of *"(x, y, width, height)"* so as a matter of convenience, the rect_to_bb function takes this rect_object and transforms it into a 4-tuple of coordinates.

The dlib face landmark detector will return a shape object containing the 68 *(x, y)*-coordinates of the facial landmark regions. Using the shape_to_np function, we can convert this object to a NumPy array. We used these two helper functions, to detect facial landmarks in images.

4.5 Eye Aspect Ratio & Alert Alarm –

In this technique, we are using different landmarks to detect the opening and closing of eye. This landmark detector that capture most of the characteristic points on a human face image. The eye blink is a fast closing and reopening of a human eye. Each individual person has a little bit different pattern of blinks. The pattern differs in the speed of closing and opening of the eye, a degree of squeezing the eye and in a blink duration. The eye blink lasts approximately 100-400ms. From the landmarks detected in the image, we derive the eye aspect ratio (EAR) that is used as an estimate of the eye opening state.

For every video frame, the eye landmarks are detected. The eye aspect ratio between height and width of the eye is computed. From the fig. 2 P1,P2,…,P6 are the landmarks on the eye.It is used to compute the ratio of distances between the vertical eye landmarks and the distances between the horizontal eye landmarks. The return value of the eye aspect ratio will be approximately constant when the eye is open. The value will then rapidly decrease towards zero during a blink.

5. Modules

The main job of detection system is to detect fatigue, drowsiness of a driver, who is sitting on a driver sit. The System makes the detection easier. Modules play very essential role in every implemented system. Modules show flow of process in the system which include sub-module of the system. Following are the modules and sub-modules used in the system:

5.1 Frame Acquisition

In a frame acquisition, the acquisition of an input image through a webcam takes place. A webcam camera will be used for capturing live video of driver eyes in all visible conditions and frames will extracted for image processing scheme of video capturing.

5.2 Facial landmark detection

Using dlib face detector or a dlib facial landmark detector 68 facial landmark on face takes place. Another algorithm for a detection of face used is a haar cascade classifier which gives output, face in a bounding box form.

5.3 Eye Localization and tracking

Eye localization takes place to localize eye whether the eye state is open or not ,to track eye open and close state eye localization and tracking is used.

5.4 Extracting eye geometrical coordinates

Extraction of eye takes place to localize eye coordinates. Eye geometrical coordinates labelled as a p1,p2,p3,p4 ,p5 and p6.

5.5 Measuring EAR

From the landmarks detected in the image, we derive the eye aspect ratio (EAR) that is used as an estimate of the eye opening state.For every video frame, the eye landmarks are detected. The eye aspect ratio between height and width of the eye is computed. From the fig. 2 P1,P2,…,P6 are the landmarks on the eye.It is used to compute the ratio of distances between the vertical eye landmarks and the distances between the horizontal eye landmarks. The first step in building a blink detector is to perform facial landmark detection to *the eyes localize* in a given frame from a video stream. Once we have the facial landmarks for both eyes, we compute the *eye aspect ratio* for each eye, which gives us a singular value, relating the distances between the vertical eye landmark points to the distances between the horizontal landmark points.

Once we have the eye aspect ratio, we can threshold it to determine if a person is blinking **— the** eye aspect ratio will remain approximately constant when the eyes are open and then will rapidly approach zero during a blink, then increase again as the eye opens. To improve our blink detector, constructing a 13-dim feature vector of eye aspect ratios (N-th frame, $N - 6$ frames, and $N + 6$ frames).

5.6 Monitoring of EAR for blinks detection

In this, different landmarks are used to detect the opening and closing of eye. This landmark detector that capture most of the characteristic points on a human face image. The eye blink is a fast closing and reopening of a human eye. Each individual person has a little bit different pattern of blinks. The pattern differs in the speed of closing and opening of the eye, a degree of squeezing the eye and in a blink duration. The eye blink lasts approximately 100-400ms. So, the monitoring of EAR takes place to monitor a blink of eye whether a blink of eye takes place or not. It plays an important role for blink detection of eye.

5.7 Estimation of fatigue periods between blinking

Estimation of fatigue periods takes place during a blink detection of eye.

5.8 Audio Visual warning on fatigue detection.

In this step audio visual warning is run when the fatigue, drowsiness of a driver is detected. It shows a visual message after detecting fatigue , drowsiness with EAR threshold value the alert alarm runs and shows the message as “Drowsiness Detected”.

6. EXPERIMENTAL RESULT

Facial Landmark detection and identification of eye coordinates for face detection takes place. In this a dlib face detector used to locate a face in an image and dlib facial landmark predictor used to locate the landmark on face and eye coordinates.

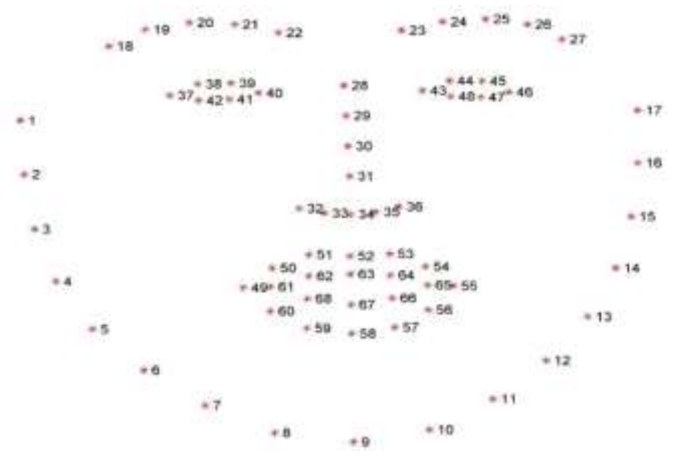
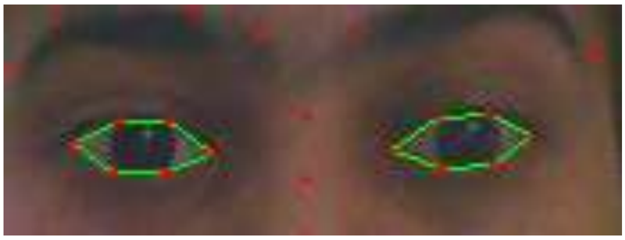


Fig.3 Facial 68 Landmark

6.1 Open Eye Coordinates



Eye coordinates p1,p2,p3,p4,p5 and p6 Typical calculated EAR with for open eye isapprox.0.24

In this open eye state takes place with the p1,p2,p3,p4,p5 and p6 eye coordinates.

6.2 Closed Eye Coordinates

EAR for close eye is approx. 0.15

In this section, we introduce experimental result on which the system is developed.

6.3 Eye Localization

In this localization of eye takes place for eye detection or eye status.

6.4 Graphical Analysis

In graphical analysis, we are considering EAR on X-axis and Time(sec) on Y-axis. The EAR measurements plots with respect to time. Each dip in EAR curve corresponds to eye blink and width of EAR dip represents duration of eye closed and formed basis of D detection. Algorithm performs well in repeated experiments on different type of faces. EAR threshold adjustment provided in GUI front end is useful for fine tuning of algorithm on different type of faces. No false detection and missed detection of blinks and closed eye for some period was observed.

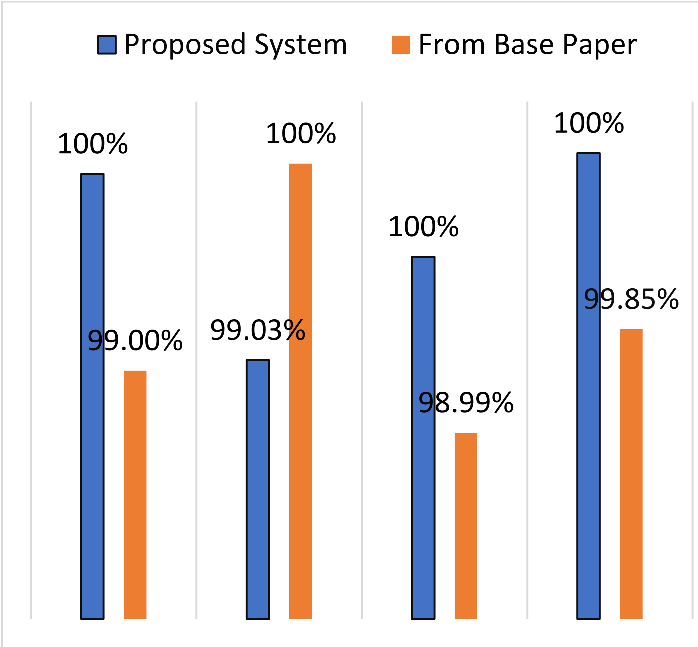


Fig.4 Graphical Analysis of for accuracy of a result

7. CONCLUSION

Driver Drowsiness Detection was built to help a driver stay awake while driving in order to reduce car accidents caused by drowsiness. This paper was concerned with drowsy drivers and their potential to cause car accidents. . The driver fatigue detection system calculates drowsiness level from the driver using a combination of webcam, haar cascade classifier, facial landmark detection is used to calculate whether or not a driver is drowsy. At the same time, it retrieves images from the camera, which is fast enough to detect a driver's features in real time. The system uses open source software called as open cv image processing libraries, the captures images are processed in this. Webcam and open cv makes the overall system to a low cost drowsiness detection system.

REFERENCES

[1] Bappaditya Mandal, Liyuan Li, Gang Sam Wang, and Jie Lin (2017) IEEE transactions on Intelligent transportation systems, Towards Detection of Bus Driver Fatigue Based on Robust Visual Analysis of Eye State.

[2] A Rosebrock (2017) facial landmarks with dlib, opencv and python.

[3] T. Soukupova and J. Cech. (2016, Feb. 3) Real-Time Eye Blink Detection using Facial Landmarks. Center for Machine Perception, Department of Cybernetics Faculty of Electrical Engineering, Czech Technical University in Prague. Prague, Czech Republic.

[4] Kazemi and J. Sullivan. (2014) One Millisecond Face Alignment with an Ensemble of Regression Trees. Royal Institute of Technology Computer Vision and Active Perception Lab. Stockholm, Sweden.