**Drowsiness detection using OpenCV**

# ABSTRACT

# The new way of security system which can be discussed during this project is predicated on machine learning and AI. Passenger security is that the main concern of the vehicles designers where most of the accidents are caused thanks to drowsiness and fatigue driving so as to supply better security for saving lives of passengers bags are designed but this method is beneficial after accident is accord. But main problem remains we see many accidents happening and lots of them are losing their lives. during this project we are using OpenCV library for image processing and giving input as user live video and training data to detect if person in video is closing eyes or showing any symptoms of drowsiness and fatigue then application will verify with trained data and detect drowsiness and lift alarm which can alert driver

**Keywords**

Eye aspect Ratio (EAR), Anaconda, Drowsiness, Python, OpenCV, Dlib library.

# INTRODUCTION

Driver fatigue could even be an enormous believe an outsized number of accidents. The motivation for this project comes from the increase in number of accidents during night time due to driver’s drowsiness while driving. The project is also useful to avoid accidents due to negligence in driving. This a technique used to detect the face of a person, animal and so on... Image processing plays an important role in identifying the drivers face and detect if the person is sleeping or not. Black box theory is an important concept in Image processing, which is used to localize the eye region of the face and then map with the trained neural network. The neural network is trained in such a way that it could be able to classify the different images of a person’s eye like closed eyes, opened eyes, partially closed eyes, etc. Other important parameter used in this project is the facial landmark recognition technique. This technique actually helps to locate the facial region of the person with geometrical points. The geometrical points of the person’s face are then separated from the whole face and then calculate the EAR (Eye Aspect Ratio) of the eye. If EAR is less then the optimum value that is calculated (indicates that the person is sleeping or feeling drowsy), a signal is passes to the buzzer by which the person who is driving awakes.

**2.Literature Review**

**2.1 Existing System:**

There is various methods like detecting objects which are almost vehicle and front and rear cameras for detecting vehicles approaching almost vehicle and bag system which may save lives after accident is accorded.

**2.2 Disadvantages of Existing System:**

Most of the prevailing systems use external factors and inform user about problem and save user after accident is accord but from research most of the accidents are thanks to faults in user like drowsiness, sleeping while driving.

**2.3 Proposed System:**

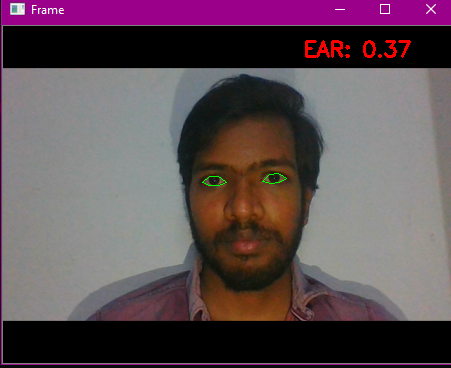
To affect this problem and supply an efficient system a drowsiness detection system are often developed which may be placed inside any vehicle which can take live video of river as input and compare with training data and if driver is showing any symptoms of drowsiness system will automatically detect and lift alarm which can alert driver and other passengers.

**2.4 Advantages of Proposed System:**

# This method will detect problem before any problem accord and inform driver and other passengers by raising alarm in this OpenCV based machine learning techniques are used for automatic detection of drowsiness.

**3.Algorithm**

Detection of blink can be evaluated by using Eye aspect Ratio (EAR) with OpenCV functions. Eye aspect Ratio is calculated using its formula with eye coordinates returned from OpenCV. Unexpected plunge in Eye aspect Ratio esteem against a set edge can be utilized for flicker location and microsleep recognition.



## **Figure-1:** Outcome of facial landmark detection and identifies the coordinates of eye.

## **3.1Measurement of EAR**

Each eye has 6 (x, y)- arranges in tourist spots returned Dlib indicator work, beginning at the left-corner of the eye and a short time later working clockwise around the remainder of the locale. There is an association between the width and the height of these bearings.

### EAR (Eye aspect Ratio) =

Where p1, p2, ..., p6 are 2D facial landmark locations.

Compute the eye aspect ratio to determine if the eyes are closed. If the eye aspect ratio indicates that the eyes have been closed for a sufficiently long enough amount of time, we will sound an alarm to wake up the driver. The eye aspect ratio is approximately constant while the eye is open, but will rapidly fall to zero when a blink is taking place. If the value falls but does not increase again, thus implying that the person has closed their eyes. At the point when the individual flickers the eye perspective proportion diminishes significantly, moving toward zero. Eye perspective proportion is steady, at that point quickly drops near zero, at that point increments once more, showing a solitary squint has occurred.

**Python Function for calculating EAR value.**

Def eye\_aspect\_ratio(eye):

# calculate the euclidean distances between the sets of vertical eye landmarks

# (x, y) coordinates

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

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

# compute the euclidean distance between the horizontal

# eye landmark (x, y)-coordinates

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

# compute the eye aspect ratio

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

# return the eye aspect ratio

return ear

**Algorithm for eye blink detection**

# check to see if the eye aspect ratio is below the blink threshold, and if so, increment the

# blink frame counter

if ear < EYE\_AR\_THRESH:

COUNTER += 1

# if the eyes were closed for a sufficient number of then sound the alarm

if COUNTER >= EYE\_AR\_CONSEC\_FRAMES:

# if the alarm is not on, turn it on

if not ALARM\_ON:

ALARM\_ON = True

# check to see if an alarm file was supplied, and if so, start a thread to

# have the alarm sound played in the background

if args["alarm"] != "":

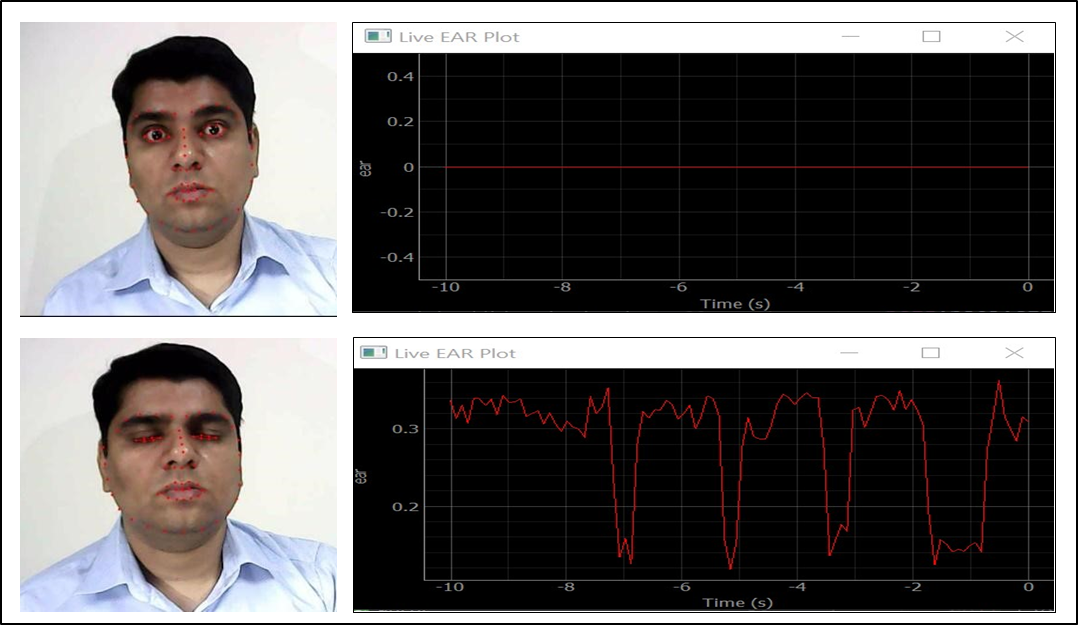
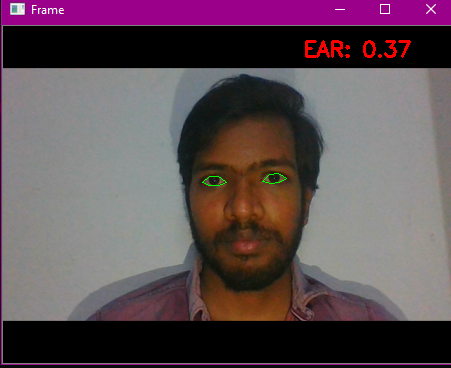
t = Thread(target=sound\_alarm,args=(args["alarm"],))

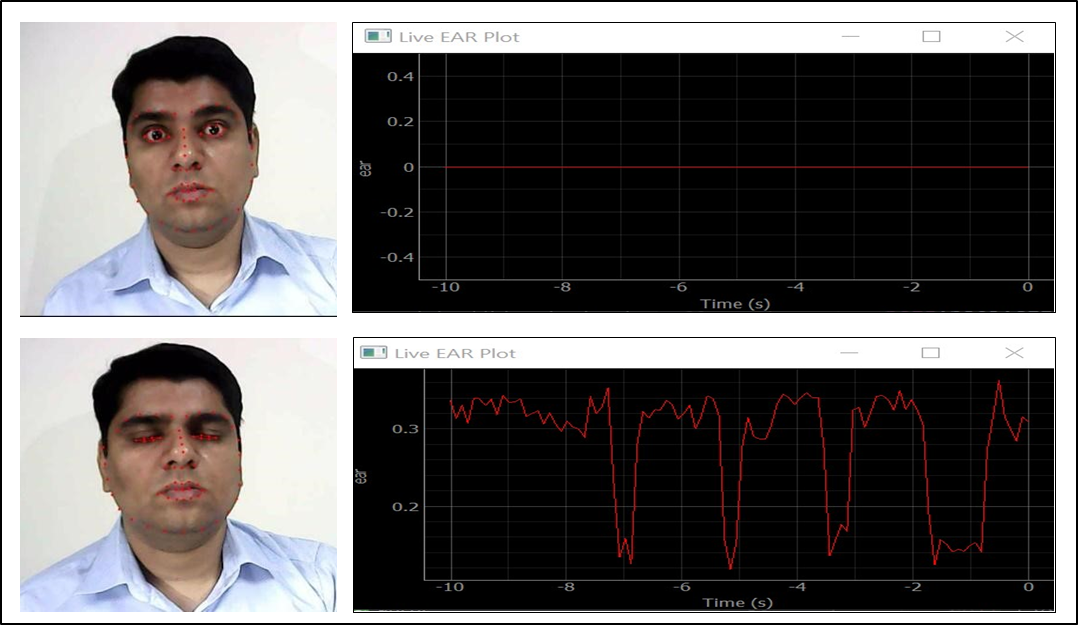
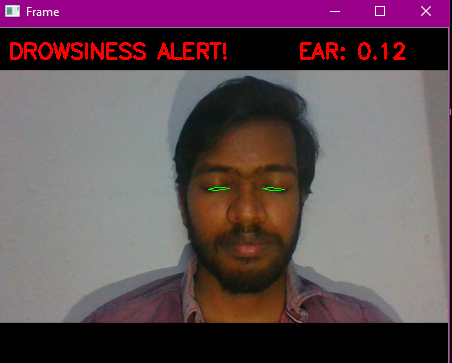
t.deamon = True

t.start()

In the event that regardless Eye aspect ratio value drops to lower than the set limit for a sufficient number then the thread is started for an alarm, which sound is played in background. If the number of frames is more than thirty then alarm is triggered. If Eye aspect Ratio value is restored to the set limit than the alarm if turned off. Legitimate rationale to stay away from bogus flicker identification is execute by creator.

# 3.2 EAR PLOT

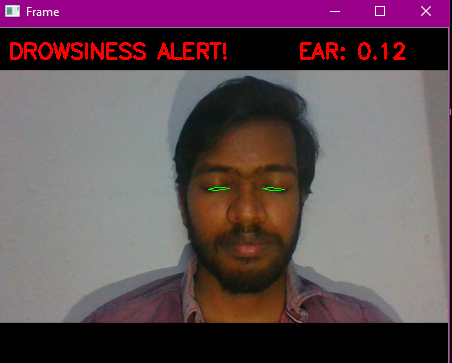
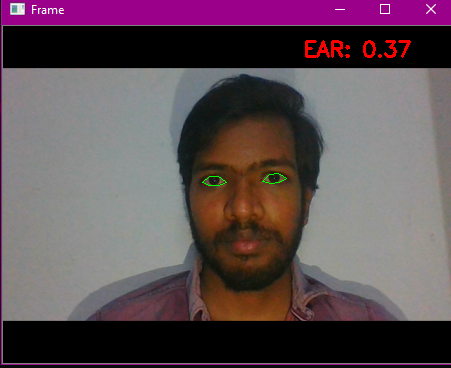




**Figure 2:** Simulation of EAR value for both open and closed eye.

**4. RESULTS**

It applies facial milestone restriction to separate eye areas from the face. Register the EAR and sound an alert if eyes have been shut for an adequately long enough time as appeared in figure 3



**Figure 3:** Test results

**5. CONCLUSION**

In this manner, we've effectively planned a model tiredne**ss** location framework with OpenCV programming.

**Future scope:**

The system so made was adequately attempted, its repressions recognized and a presumable course of action of movement made. In the constant driver weariness discovery framework, it's required to hamper a vehicle naturally when exhaustion level crosses a specific cutoff**.** Instead of edge languor level it's recommended to style perpetual scale driver exhaustion location framework. It screens the degree of laziness constantly and when this level surpasses a specific worth a sign is produced which controls the pressure driven stopping mechanism of the vehicle.

# 

# 6. 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. <https://en.wikipedia.org/wiki/Microsleep>
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](https://docs.opencv.org/trunk/d7/d8b/tutorial_py_face_detection.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.

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