Drowsiness Detection using Opencv

KCS-752 Mini Project/Internship Assessment

Presented By- Sunny Yadav
Section- CSE- A
Roll No- 1803210154
Year- 4th Year
Semester- 7 Sem
Submitted to- Mrs. Aditi Arora

Outlines

- Objective
- Motivation
- Requirement
- Idea
- Algorithms
- Flow Diagram
- Working
- Conclusion
- References

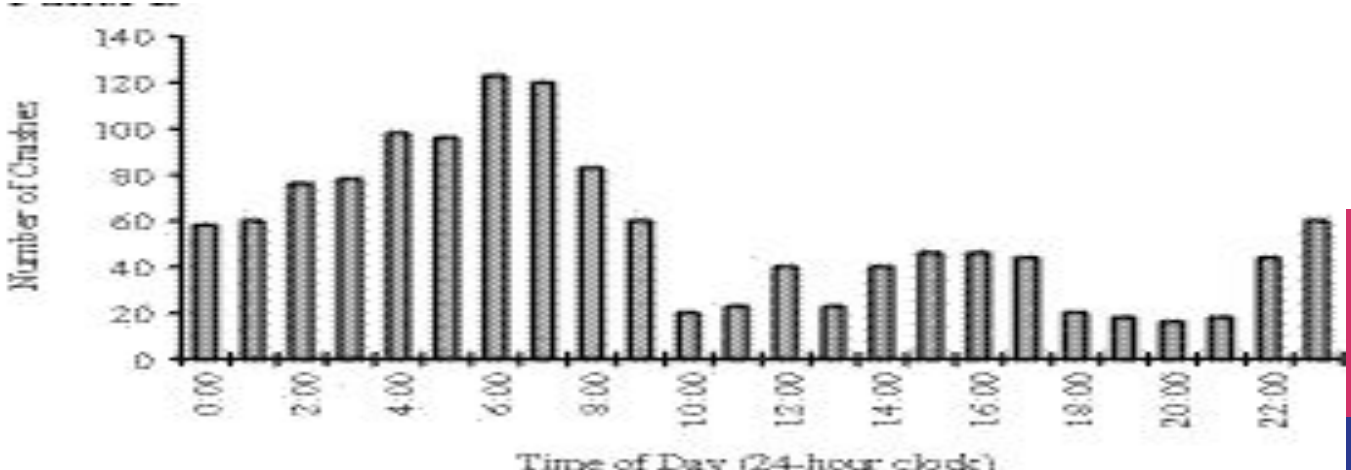
Introduction

- Our aim is to provide a driver drowsiness system which will alert not only the driver but also the co-passengers with a loud alarm in the car. By using face tracking and video processing.
- The system works in spite of driver wearing spectacles and in various lighting conditions.

Motivation

- *A study (In the U.S) showed that **37% of drivers** surveyed admitted to falling asleep at the wheel.*
- *An estimated **1.35 million** drivers have been involved in a drowsy driving related crash in the past five years.*
- ***Fall-asleep crashes are likely to be serious.** The morbidity and mortality associated with drowsy-driving crashes are high, perhaps because of the higher speeds involved (Horne, Reyner, 1995b) combined with delayed reaction time.*

- *Accidents study in the U.S (1990-92):*
  - *Time of occurrence of crashes in drivers at ages 26 to 45 in which the crashes were attributed by the police to the driver being asleep (but in which alcohol was not judged to be involved).*
  - *The X axis is the time of day and the Y axis is the number of crashes.*



Requirement

Hardware Requirements:-

- Camera
- Laptop (intel i3 ,4gb RAM)

Software Requirements:-

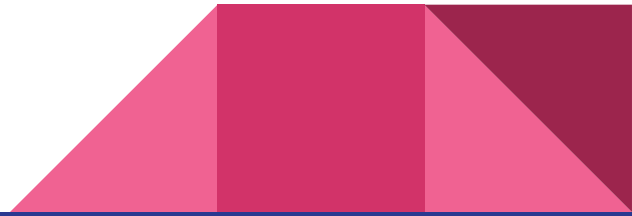
- Pycharm
- Python
- opencv

Idea

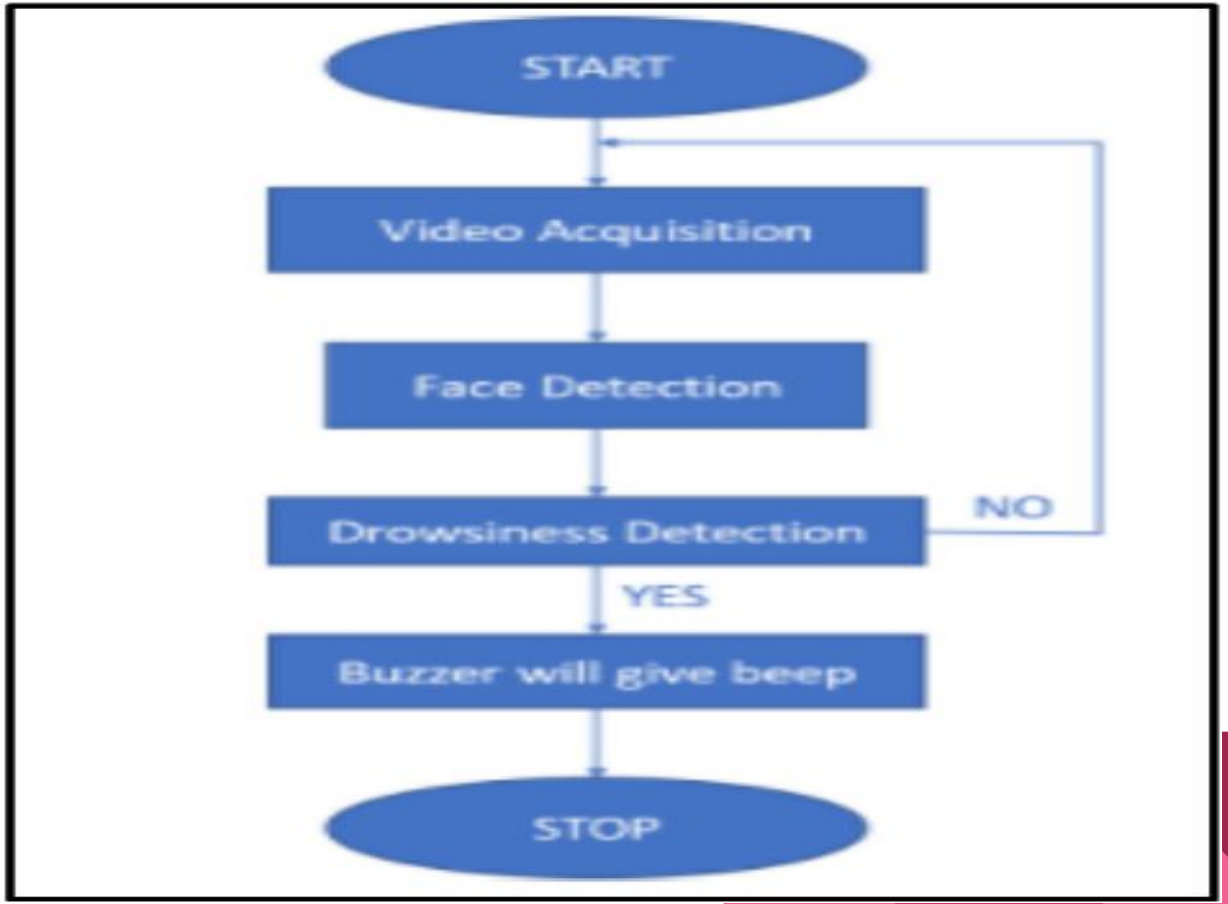
- *A video camera placed inside the car is continuously filming the driver's face during the ride.*
- *A detection system analyses the movie frame by frame and determines whether the driver's eyes are open or shut.*
- *If the eyes are shut for more than 1/4 a second (longer than a normal blink period) then the systems beeps to alert the driver.*

Algorithms

- Haar Cascade Algorithm
- Eye Aspect Ratio [EAR]
- Lips Aspect Ratio [LAR]



Flow Diagram



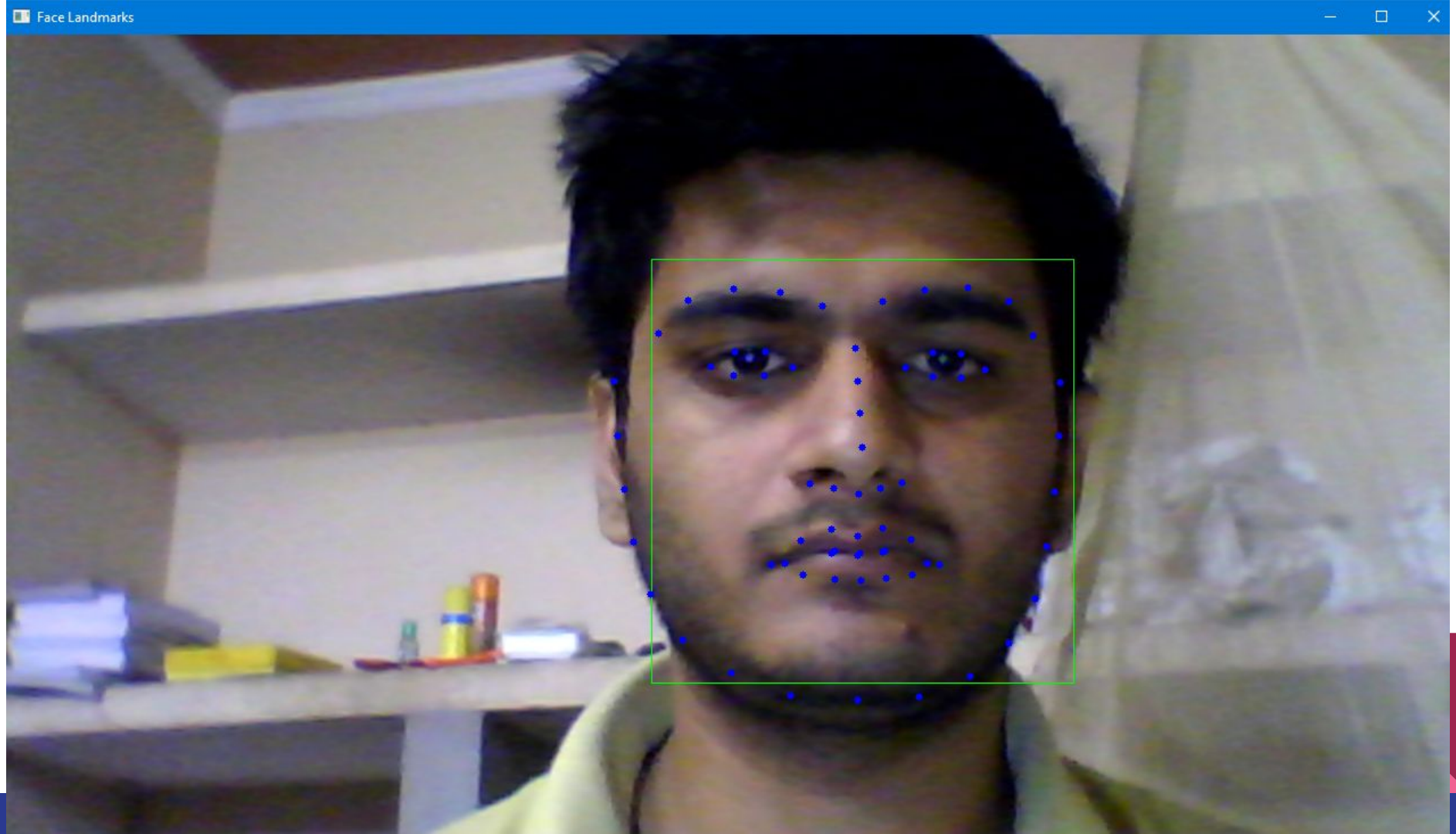
Working

- Drivers face is continuously monitored using a video camera.
- In order to detect the drowsiness the first step is to detect the face using the series of frame shots taken by the camera.
- Then the location of the eyes and lips is detected
- If eye is closed more than a specific time or number of yawn is more than specific number than alert signal generated.
- If the signal crosses the threshold, then the alarm produce loud sound and the driver can wake up by that.

30 frames per second

- dlib library is used for detecting faces and Yawn
- facial landmarks of a person is shown in figure

Fig. 1: Facial Landmarks


Face Landmarks

EAR algorithm:

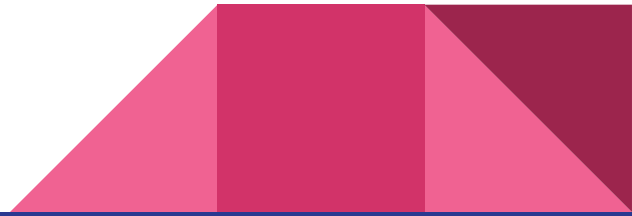
Step 1: Use the Detected Eye region from the EAR algorithm

Step 2: Compute the Eye Aspect Ratio to determine if the eyes are closed

Step 3: If EAR satisfies the drowsy condition then move to step 5

Step 4: If EAR is normal then go to Step 1

Step 5: Sound Alarm



Eyes

As shown in the facial landmarks image we can define two lists with numbers corresponding to left and right eye.

```
left_eye = [36,37,38,39,40,41]
right_eye = [42,43,44,45,46,47]
```

Blink Detection

In order to build a function that will help us to determine if there is a blink or not we could use the Eye Aspect Ratio (E.A.R.) value as defined on the paper *Real-Time Eye Blink Detection using Facial Landmarks [Soukupova, Cech].*

Let's enumerate the facial landmarks of a single eye with points P_1 to P_6 as shown below.

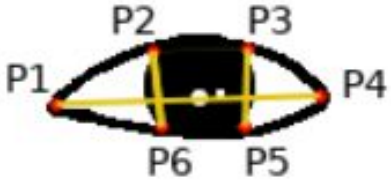


Fig. 4: Vertical and Horizontal distances

We will calculate the two vertical distances between points $P_2 - P_6$ ($dist1$) and $P_3 - P_5$ ($dist2$), and the horizontal (largest) distance between points $P_1 - P_4$ ($dist3$). So the E.A.R. (from now on **ear**) is defined as :

$$ear = \frac{(dist1 + dist2)}{2 \cdot dist3}$$


You're getting sleepy.
Very sleepy.
At the count of three,
you will not wake up.

LAR Algorithm:

Step 1: Use the Detected lips region from the LAR algorithm

Step 2: Compute the Lips Aspect Ratio to determine if the Lips are closed or open

Step 3: If LAR satisfies the drowsy condition then move to step 5

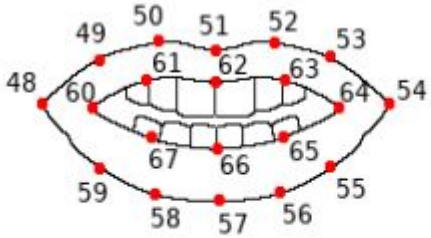
Step 4: If LAR is normal then go to Step 1

Step 5: Sound Alarm

Lips

To find out if there is a yawn or not we will work exactly as before on the eyes section. First let’s define a list with numbers corresponding to the lips part of a face as shown in the facial landmarks image.

```
lips = [60, 61, 62, 63, 64, 65, 66, 67] # Landmark indexes
```



Yawn Detection

We will define a function similar to ***eye_aspect_ratio*** function, in order to calculate **Lips Aspect Ratio** (L.A.R.). Let's enumerate the facial landmarks corresponding to the "interior" lips with points L_1 to L_8.

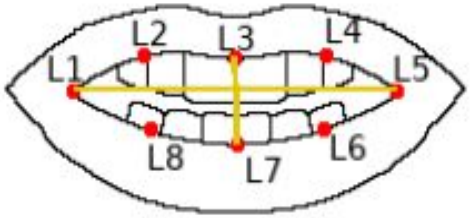


Fig. 9: Vertical and Horizontal distance

LAR threshold value to 0.5

Now we can calculate the vertical distance between points L_3 and L_7 ($dist1$) and the horizontal distance between points L_1 and L_5 ($dist2$). So the L.A.R. (from now on **lar**) is defined as :

$$lar = \frac{dist1}{dist2}$$

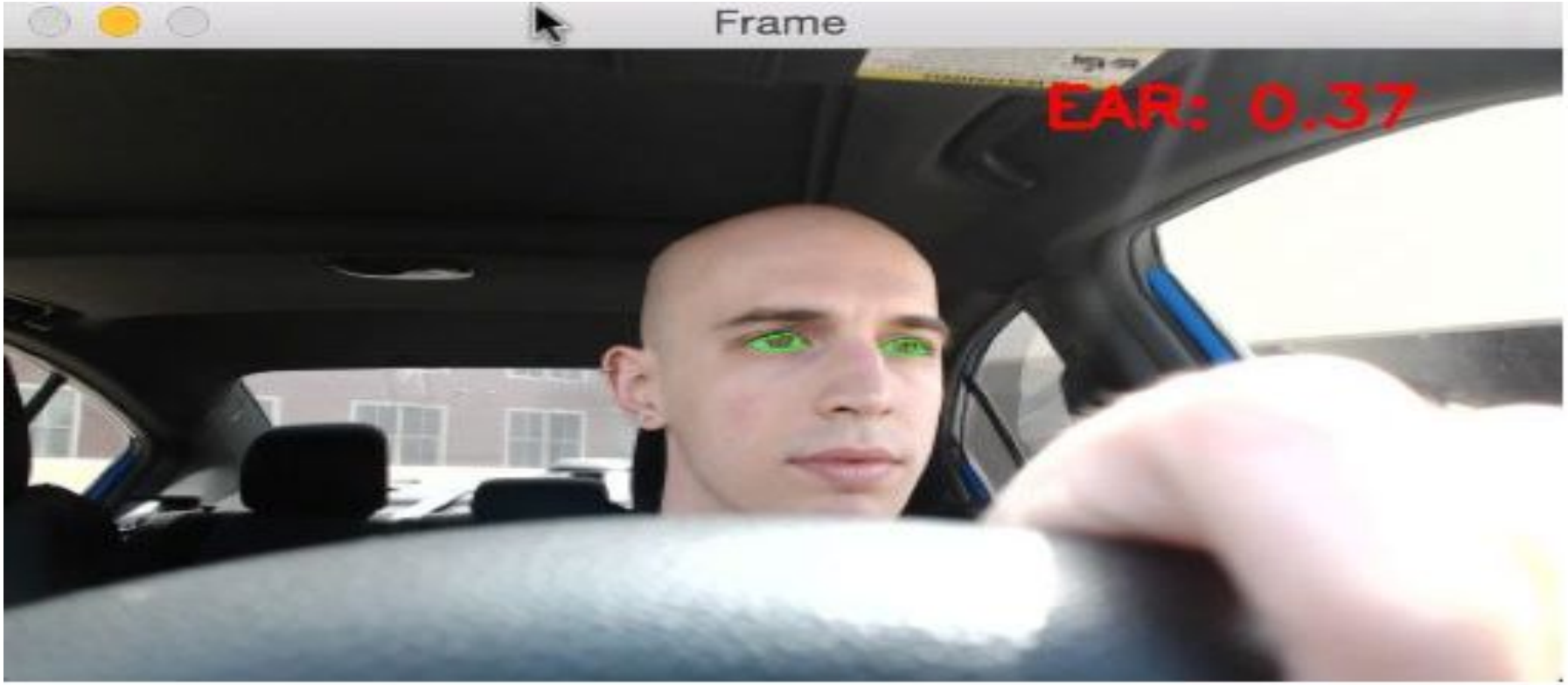
Drowsiness detection

By combining eye aspect ration (*ear*) and lips aspect ratio (*lar*) values we can deside if there is drowsiness or not.

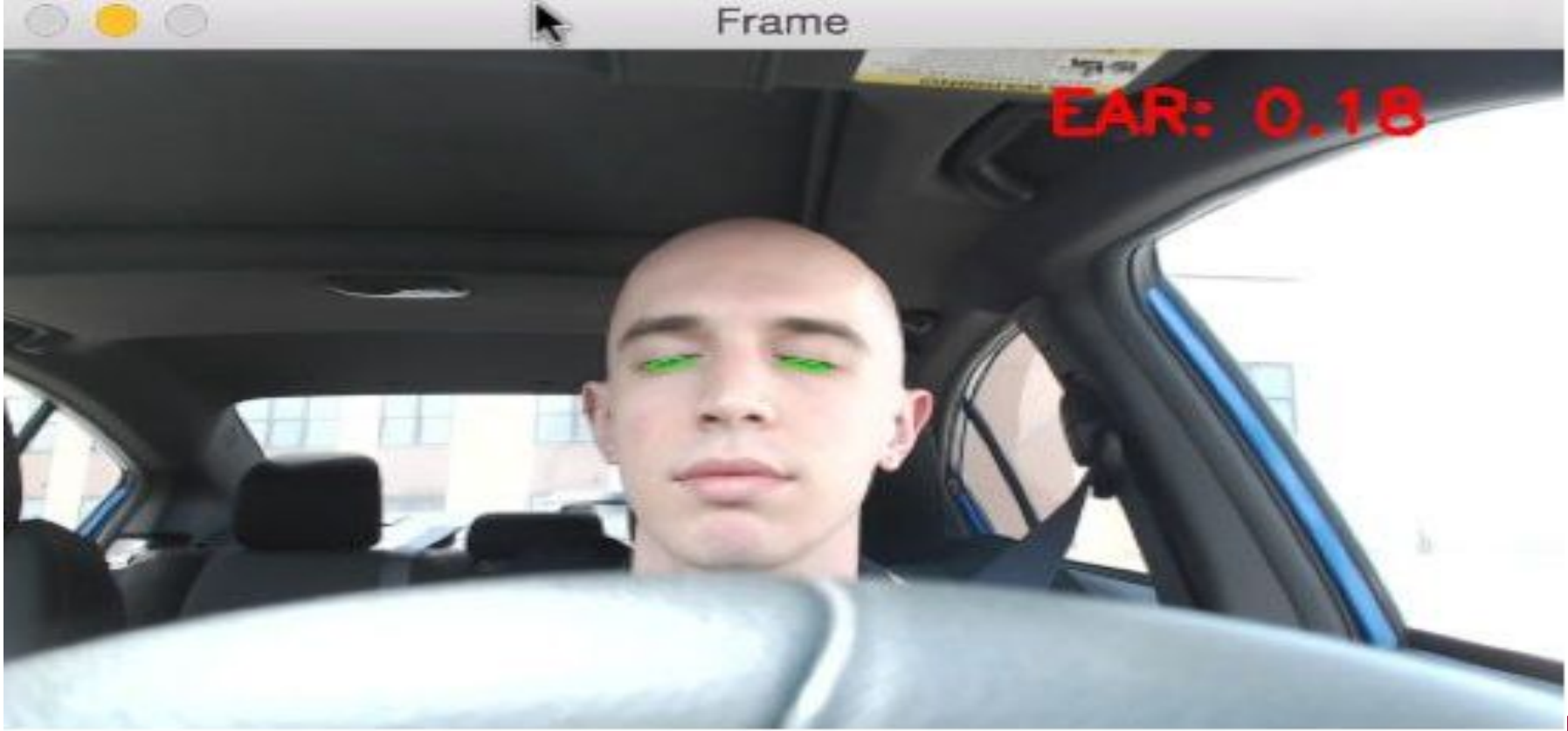
```
if total_yawns > 2 or total_blinks > 3:
    cv2.putText(frame, "ALERT", (w-120, 160), font, 1.2, (0, 0, 255), 4)
```

EAR =Eye Aspect ratio EAR threshold=0.3

LAR= Lips Aspect Ratio LAR threshold =0.5

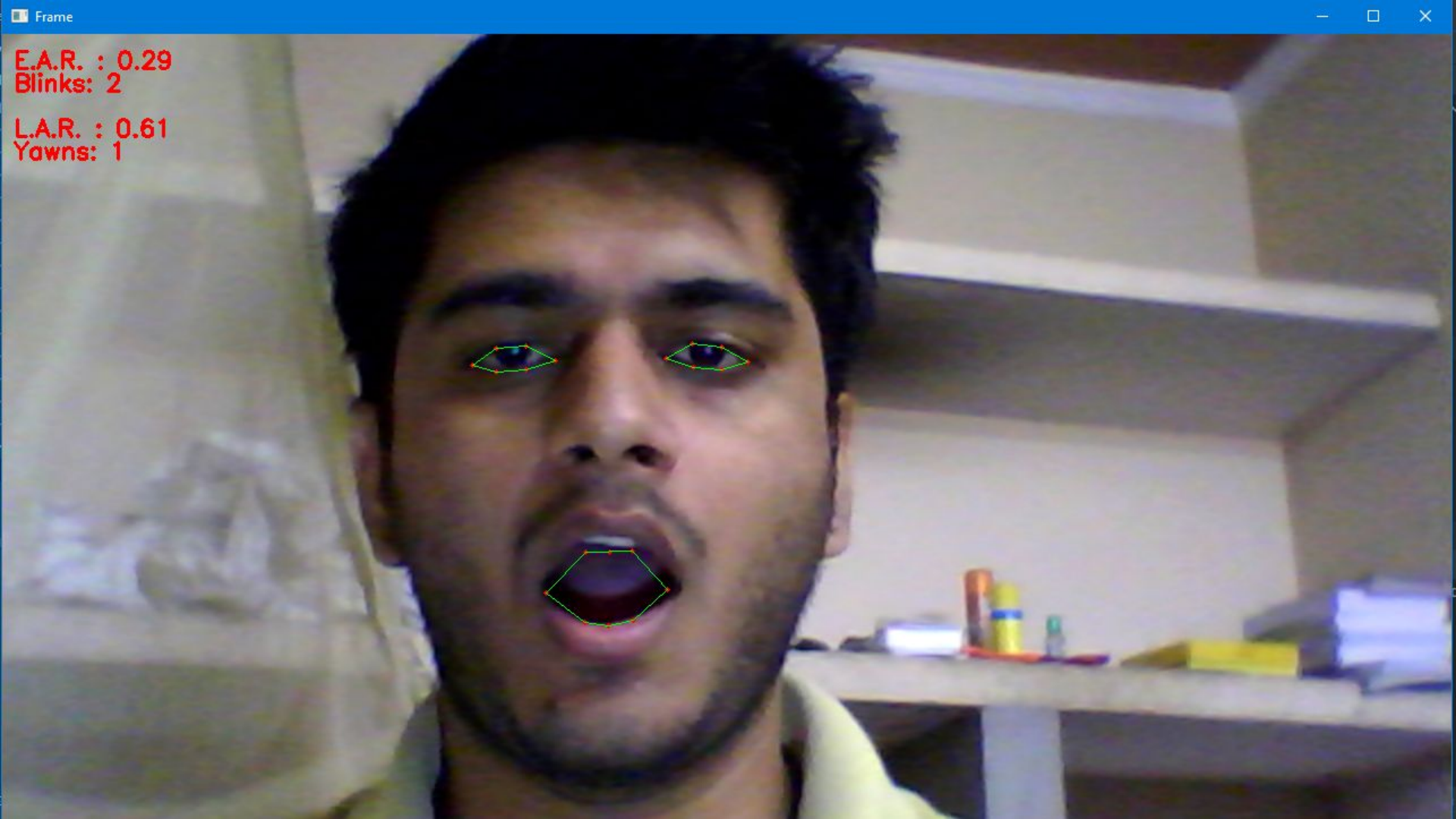


Calculating EAR (to check whether Eye is open or not)



Decrease in Eye Aspect Ratio when Eye is closed


Frame
E.A.R. : 0.28
Blinks: 6
L.A.R. : 0.02
Yawns: 0
ALERT


Frame
E.A.R. : 0.29
Blinks: 2
L.A.R. : 0.61
Yawns: 1

Conclusion

- It has a wide scope in the future and can be improved to detect that in car driver is wearing a seat belt or mask.
- The vehicle manufacturers can make this system inbuilt by using the dashboard screen and speakers.
- The system can be effectively used in locomotives and flights for detecting driver drowsiness.
- System can be improved to work with if driver is wearing spectacles.
- In near future it can be associated with traffic police if person violate so many rules he/she can be punished.

Reference:

- Aryan verma Youtube (https://www.youtube.com/watch?v=ksi42rwGyas&t=647s)
- Drowsiness Detection with OpenCV (https://www.pyimagesearch.com/2017/05/08/drowsiness-detection-opencv/)
- V B Navya Kiran, Raksha R, Anisoor Rahman, Varsha K N, Dr. Nagamani N P, 2020, Driver Drowsiness Detection, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) NCAIT – 2020 (Volume 8 – Issue 15)

Thank you