K. J. Somaiya Institute of Engineering and Information Technology, Sion, Mumbai

Accredited with 'A' Grade by NAAC (3.21 CGPA)

Three Programs Accredited by NBA

Best College Award by University of Mumbai (Urban Region), ISTE (MH), CSI (Mumbai)

DEPARTMENT OF INFORMATION TECHNOLOGY

Synopsis of Mini Project On

Driver Drowsiness Detection System

Prepared By:

Viraj Sanjay Gholap (Roll No. 16)
Burhanuddin Dilshad (Roll No. 11)

Under the guidance of:

Dr. Vaishali Gaikwad

Department of Information Technology

Academic Year: 2020-2021

(REV- 2019 'C' Scheme) - SEM IV (SE - IT)

K. J. SOMAIYA INSTITUTE OF ENGINEERING & INFORMATION TECHNOLOGY, SOMAIYA AYURVIHAR COMPLEX, EASTERN EXPRESS HIGHWAY, SION, MUMBAI - 400022.

K. J. SOMAIYA INSTITUTE OF ENGINEERING & INFORMATION TECHNOLOGY, MUMBAI

DEPARTMENT OF INFORMATION TECHNOLOGY

CERTIFICATE

This is to certify that following students:

	Roll No. / Seat No.
Viraj Sanjay Gholap	(Roll No. 16)
Burhanuddin Dilshad	(Roll No. 11)

have submitted Mini-Project Report on "Driver Drowsiness Detection System" as the partial fulfillment for the requirement of Second Year of Engineering (4th Semester) in S.E. - Information Technology under my guidance during the academic year 2020-2021.

Dr. Vaishali Gaikwad
Project Guide
Assistant Professor
Department of Information Technology

Dr. Radhika Kotecha
Head of Department
Associate Professor
Department of Information Technology

Date of Examination: 23/05/2021

Signature of Internal Examiner

Signature of External Examiner

Table of Contents

Acknowledgement

We would especially thank the Head of Department, Dr. Radhika Kotecha. We are thankful to our project supervisor Dr.Vaishali Gaikwad for her timely help and guidance in completing this project successfully and all teaching and non- teaching staff of Information Technology faculty who inspired us in completion of our project.

We would also like to express our deep regards and gratitude to our Mini-Project coordinator – Prof. Nasim B. Shah for his support and facilities provides to us for the same.

Lastly, I would like to thank all those who directly and indirectly helped in completion of this project.

Abstract

With the rise in population, the occurrence of road accidents has been increasing. A recent study showed that, around half million accidents occur in a year, in India itself. Out of which 60% of these accidents are caused due to Driver Drowsiness. 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. For that we will be using face tracking and video processing. Drowsiness is the state of feeling tired or sleepy. We all can be victim of drowsiness while driving, simply after too short night sleep, tired physical condition or during long journeys. Drowsiness is the state of feeling tired or sleepy. Driver fatigue affects the driving ability in the following 3 areas, It impairs coordination and It causes longer reaction times and also It impairs judgment Considering the hazards, drowsiness presents on the road, it is necessary to develop and efficient system which can work under low light environment and with better and faster speed. Nowadays the driver safety in the car is one of the most wanted system to avoid accidents.

Our objective of the project is to ensure the safety system. In this manner, a system which can keep a check of driver’s condition for drowsiness and alert the driver before it’s too late. For this we need a system which will focus on the open or closed state of driver’s eyes as by monitoring the state of the eyes detection of drowsiness is easy. Detection in real-time is the major challenge in the field of accident prevention system.

Chapter 1: Introduction

Drowsiness is simply defined as “a state of near sleep due to fatigue”. It is technically distinct from fatigue, which has been defined as a “disinclination to continue performing the task at hand”. The effects of sleepiness and fatigue are very much the same. Fatigue affects mental alertness, decreasing an individual’s ability to operate a vehicle safely and increasing the risk of human error that could lead to fatalities and injuries. Sleepiness slows reaction time, decreases awareness, and impairs judgment. Fatigue and sleep deprivation impact all transportation operators (for example: airline pilots, truck drivers, and railroad engineers).In both conditions, driver can’t focus on primary task of driving which may enhance the likelihood of crash occurrence. With the ever-growing traffic conditions, this problem will further deteriorate. For this reason, it is necessary to develop driver alertness system for accident prevention due to Driver Drowsiness

1.1 Motivation

Driver drowsiness is one of the main causes of road accidents around the world. Due to lack of sleep and tiredness drowsiness can occur while driving. The best way to avoid accidents caused by driver’s drowsiness is to detect drowsiness of the driver and warn him before fall into sleep. Driver drowsiness detection is a car safety technology which prevents accidents when the driver is getting drowsy. Various studies have suggested that around 20% of all road accidents are fatigue-related, up to 50% on certain roads, driver fatigue is a significant factor in a large number of vehicle accidents Situation in reality.

1.2 Objective

- Driver drowsiness detection is a car safety technology which helps to save the life of the driver by preventing accidents when the driver is getting drowsy
- The main objective is to first design a system to detect driver's drowsiness by continuously monitoring eyes.
- The system works in spite of driver wearing spectacles and in various lighting conditions
- To alert the driver on the detection of drowsiness by using buzzer or alarm.
- Traffic management can be maintained by reducing the accidents.

1.3 Scope

The future works may focus on the utilization of outer factors such as vehicle states, sleeping hours, weather conditions, mechanical data, etc, for fatigue measurement. Driver drowsiness pose a major threat to highway safety, and the problem is particularly severe for commercial motor vehicle operators. Twenty-four hour operations, high annual mileage, exposure to challenging environmental conditions, and demanding work schedules all contribute to this serious safety issue. Future work may be to automatically zoom in on the eyes once they are localized

Chapter 2: Literature Review

In 2008, Hong Suet. al. described ‘A Partial Least Squares Regression-Based Fusion Model for Predicting the Trend in Drowsiness’. They proposed a new technique of modeling driver drowsiness with multiple eyelid movement features based on an information fusion technique—partial least squares regression (PLSR), with to cope with the problem of strong collinear relations among eyelid movement features and, thus, predicting the tendency of the drowsiness. The predictive precision and robustness of the model thus established are validated, which show that it provides a novel way of fusing multi-features together for enhancing our capability of detecting and predicting the state of drowsiness.

In June, 2010, Bin Yang et. al. described ‘Camera- based Drowsiness Reference for Driver State Classification under Real Driving Conditions’. They proposed that measures of the driver’s eyes are capable to detect drowsiness under simulator or experiment conditions. Measures are assessed statistically and by a classification method based on a large dataset of 90 hours of real road drives. The results show that eye-tracking drowsiness detection works well for some drivers as long as the blinks detection works properly. Even with some proposed improvements, however, there are still problems with bad light conditions and for persons wearing glasses. As a summary, the camera based sleepiness measures provide a valuable contribution for a drows iness reference, but are not reliable enough to be the only reference. In 2011, M.J. Flores et. al. described ‘Driver drowsiness detection system under infrared illumination for an intelligent vehicle’. They proposed that to reduce the amount of such fatalities, a module for an advanced driver assistance system, which caters for automatic driver drowsiness detection and also driver distraction, is presented. In June, 2012, A. Cheng et. al. described 'Driver Drowsiness Recognition Based on Computer Vision Technology’. They presented a nonintrusive drowsiness recognition method using eye-tracking and image processing. A robust eye detection algorithm is introduced to address the problems caused by changes in illumination and driver posture. In 2013, G. Kong et. al. described 'Visual Analysis of Eye State and Head Pose for Driver Alertness Monitoring'. They presented visual analysis of eye state and head pose (HP) for continuous monitoring of alertness of a vehicle driver. Most existing approaches to visual detection of non-alert driving patterns rely either on eye closure or head nodding angles to determine the driver drowsiness or distraction level. In June, 2014, Eyosiyas et. al. described 'Driver Drowsiness Detection through HMM based Dynamic Modeling'. They proposed a new method of analyzing the facial expression of the driver through Hidden Markov Model (HMM) based dynamic modeling to detect drowsiness. August 2014, García et. al. described 'Driver Monitoring Based on Low-Cost 3-D Sensors'. They proposed a solution for driver monitoring and event detection based on 3-D information from a range camera is presented. The system combines 2-D and 3-D techniques to provide head pose estimation and regions-of-interest identification

The existing system of driver drowsiness detection system has following disadvantages. Mainly, using of two cameras in the system one for monitoring the head movement and the other one for facial expressions. The other disadvantage is aging of sensors and all these sensors are attached to the driver's body which may affect the driver. So to overcome all these disadvantages we designed a system in which a live camera is used for monitoring the driver drowsiness condition and alert the driver which reduces the road accidents. Because the level of drowsiness is measured approximately every 5 min, sudden variations cannot be detected using subjective measures. Another limitation to using subjective ratings is that the self-introspection alerts the driver, thereby reducing their drowsiness level.

Chapter 3: Functionalities of Proposed System

3.1 Proposed System

The “haar cascade files” folder consists of the xml files that are needed to detect objects from the image. In our case, we are detecting the face and eyes of the person.

The models folder contains our model file “cnnCat2.h5” which was trained on convolutional neural networks.We have an audio clip “alarm.wav” which is played when the person is feeling drowsy.“Model.py” file contains the program through which we built our classification model by training on our dataset. You could see the implementation of convolutional neural network in this file.“Drowsiness detection.py” is the main file of our project. To start the detection procedure, we have to run this file.

Step 1 – Take image as input from a camera.

Step 2 – Detect the face in the image and create a Region of Interest (ROI).

Step 3 – Detect the eyes from ROI and feed it to the classifier.

Step 4 – Classifier will categorize whether eyes are open or closed.

Step 5 – Calculate score to check whether the person is drowsy

3.2 The algorithm

Driver Drowsiness detection is a safety technology that can prevent accidents that are caused by drivers who fell asleep while driving. The objective of this intermediate Python project is to build a drowsiness detection system that will detect that a person's eyes are closed for a few seconds.

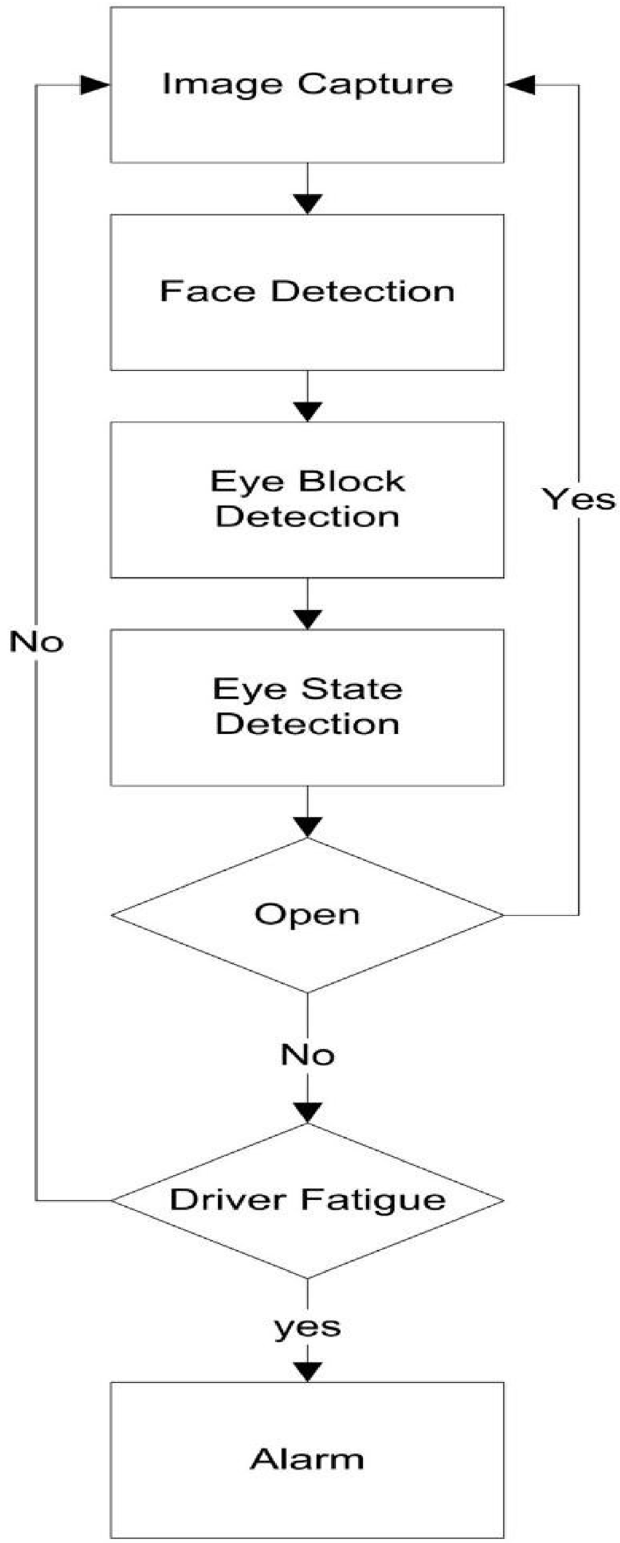

Fig 3.2.1 – Block Diagram

Step 1 – Take Image as Input from a Camera

With a webcam, we will take images as input. So to access the webcam, we made an infinite loop that will capture each frame. We use the method provided by OpenCV, cv2.VideoCapture(0) to access the camera and set the capture object (cap). cap.read**()** will read each frame and we store the image in a frame variable.

Step 2 – Detect Face in the Image and Create a Region of Interest (ROI)

To detect the face in the image, we need to first convert the image into grayscale as the OpenCV algorithm for object detection takes gray images in the input. We don't need color information to detect the objects. We will be using haar cascade classifier to detect faces. This line is used to set our classifier face = cv2.CascadeClassifier(' path to our haar cascade xml file'). Then we perform the detection using faces = face.detectMultiScale(gray). It returns an array of detections with x,y coordinates, and height, the width of the boundary box of the object. Now we can iterate over the faces and draw boundary boxes for each face.

Step 3 – Detect the eyes from ROI and feed it to the classifier

The same procedure to detect faces is used to detect eyes. First, we set the cascade classifier for eyes in leye and reye respectively then detect the eyes using left_eye = leye.detectMultiScale(gray). Now we need to extract only the eyes data from the full image. This can be achieved by extracting the boundary box of the eye and then we can pull out the eye image from the frame with this code.l_eye only contains the image data of the eye. This will be fed into our CNN classifier which will predict if eyes are open or closed. Similarly, we will be extracting the right eye into r_eye.

Step 4 – Classifier will Categorize whether Eyes are Open or Closed

We are using CNN classifier for predicting the eye status. To feed our image into the model, we need to perform certain operations because the model needs the correct dimensions to start with. First, we convert the color image into grayscale using r_eye = cv2.cvtColor(r_eye, cv2.COLOR_BGR2GRAY). Then, we resize the image to 24*24 pixels as our model was trained on 24*24 pixel images cv2.resize(r_eye, (24,24)). We normalize our data for better convergence r_eye = r_eye/255 (All values will be between 0-1). Expand the dimensions to feed into our classifier. We loaded our model using model = load_model('models/cnnCat2.h5') . Now we predict each eye with our model.lpred = model.predict_classes(l_eye). If the value of lpred[0] = 1, it states that eyes are open, if value of lpred[0] = 0 then, it states that eyes are closed.

Step 5 – Calculate Score to Check whether Person is Drowsy

The score is basically a value we will use to determine how long the person has closed his eyes. So if both eyes are closed, we will keep on increasing score and when eyes are open, we decrease the score. We are drawing the result on the screen using cv2.putText() function which will display real time status of the person.A threshold is defined for example if score becomes greater than 07 that means the person's eyes are closed for a long period of time. This is when we beep the alarm using sound.play().

3.3 Key functionalities of proposed system

The requirement for this Python project is a webcam through which we will capture images. You need to have Python (3.6 version recommended) installed on your system, then using pip, you can install the necessary packages.

1.OpenCV – pip install opencv-python (face and eye detection).

2.TensorFlow – pip install tensorflow (keras uses TensorFlow as backend).

3.Keras – pip install keras (to build our classification model).

4.Pygame – pip install pygame (to play alarm sound).

The model we used is built with Keras using Convolutional Neural Networks (CNN). A convolutional neural network is a special type of deep neural network which performs extremely well for image classification purposes. A CNN basically consists of an input layer, an output layer and a hidden layer which can have multiple numbers of layers. A convolution operation is performed on these layers using a filter that performs 2D matrix multiplication on the layer and filter.

Chapter 4: Implementation Details and Results

HARDWARE SPECIFICATION:

S.R. NO	Hardware type	Hardware Specification	Minimum requirement
01	PC	Intel i3, Ryzen3	4G RAM.
02	Webcam/Laptop camera	N.A.	N.A.

SOFTWARE SPECIFICATION:

S.R.NO	Software type	Software specification	Minimum version
1	Operating system	Windows 64 bit	Windows 7
2	Programming Language	Python 64-bit	3.6 Recommended

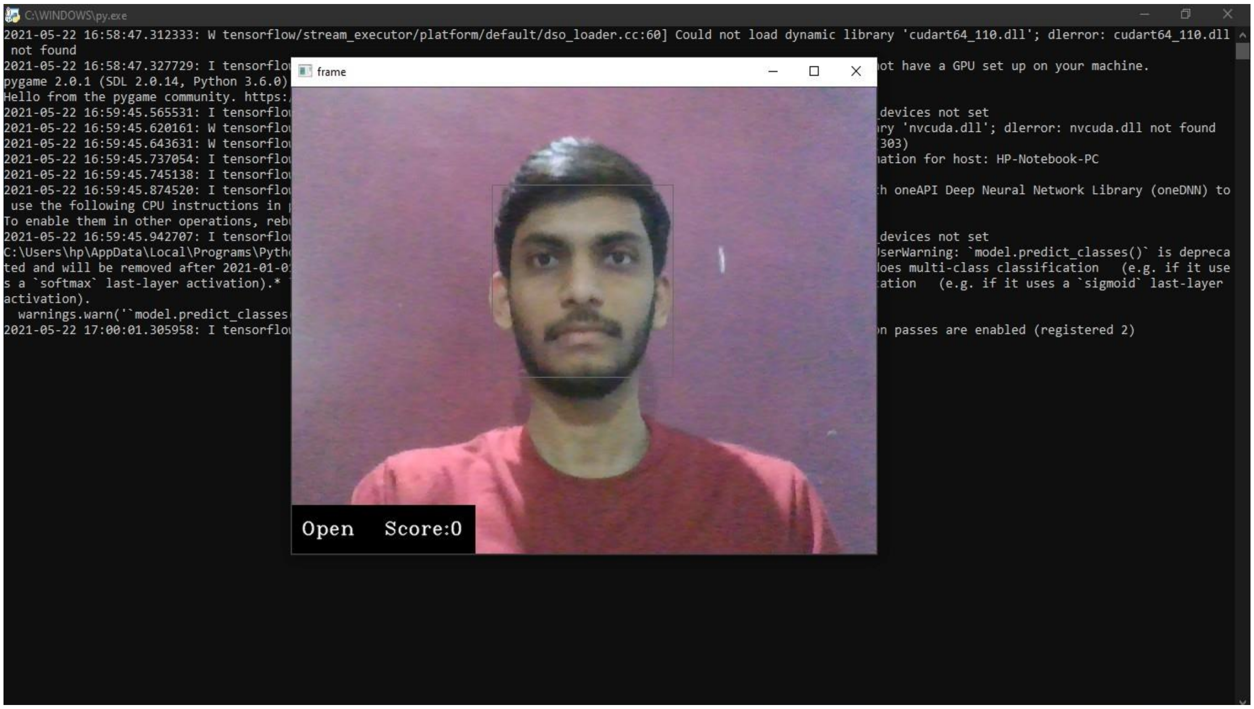

FIG 4.1 IMPLEMENTATION FRAME-1

In FIG 4.1 , the process of detecting the face had been started and as eyes are fully opened so the score is displaying zero can be seen.

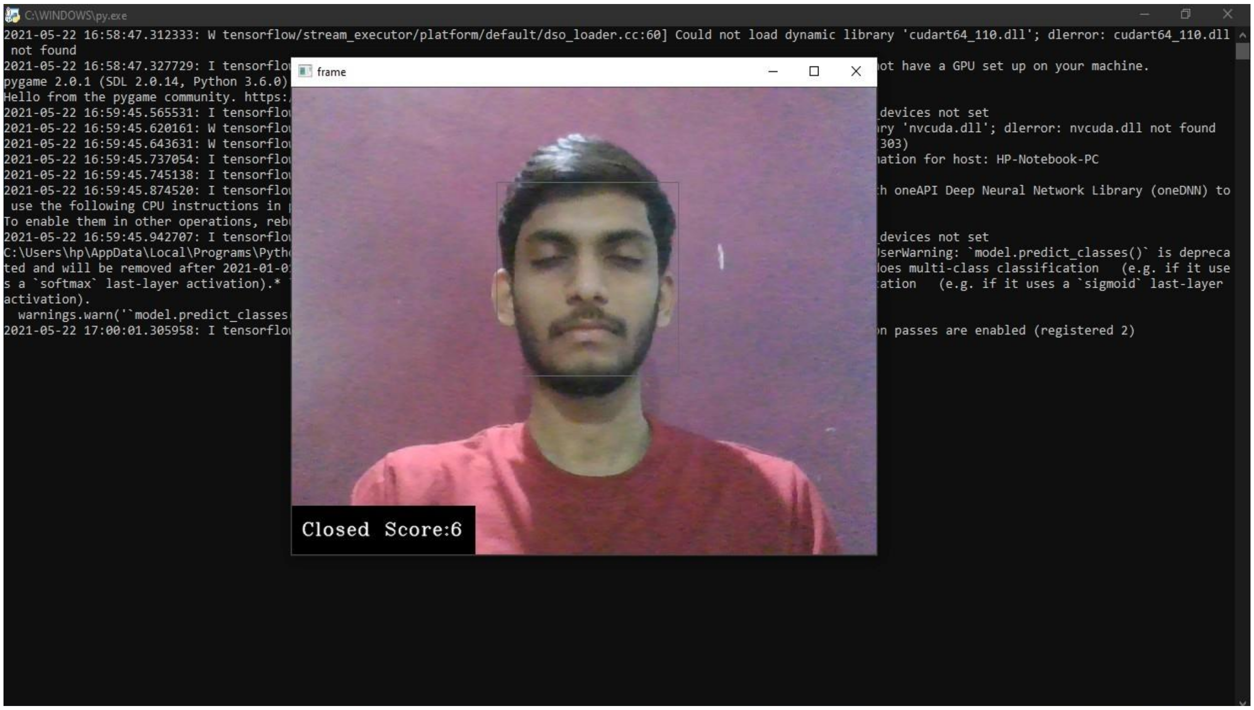

FIG 4.2 IMPLEMENTATION FRAME-2

In FIG 4.2 , the person eyes are closing slowly so the score will increase gradually can be seen.

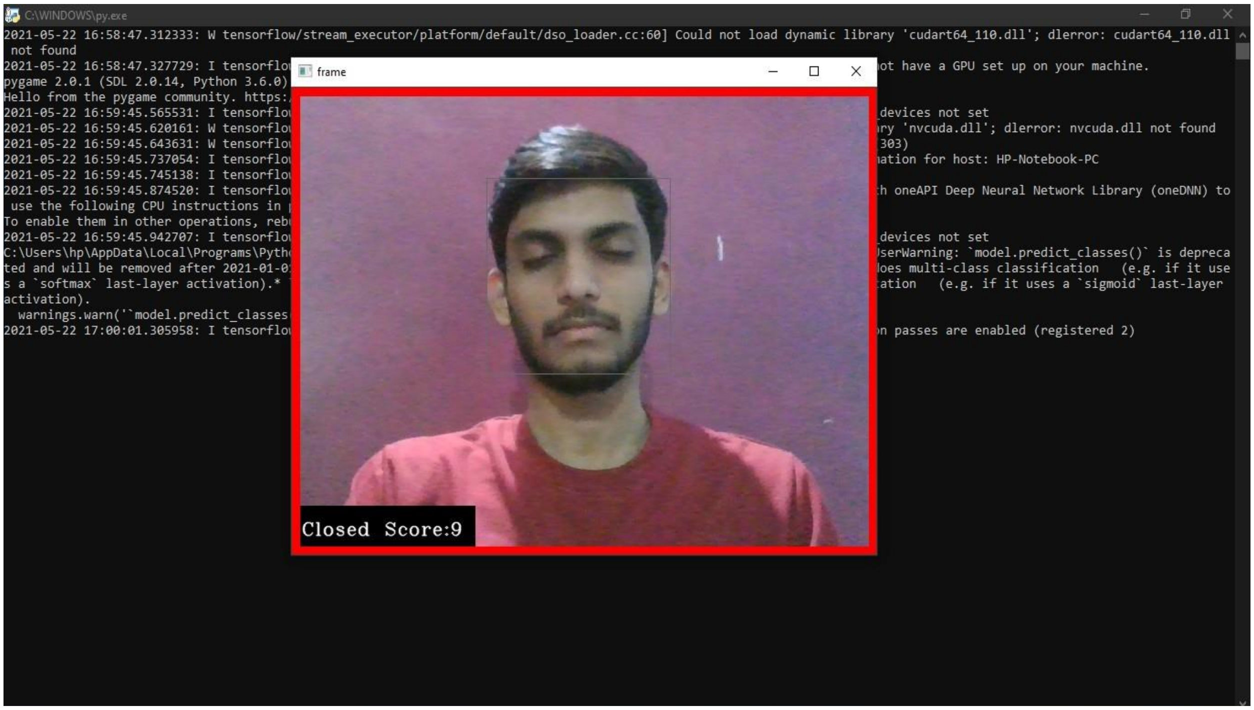

FIG 4.3 IMPLEMENTATION FRAME-3

In FIG 4.3 , the person eyes are completely closed and the score will increase to its extent and then the frame will turn red and alarm will ring simultaneously for alerting the person can be seen

Chapter 5: Conclusion & Future Scope

The drowsiness detection and correction system developed is capable of detecting drowsiness in a rapid manner. The system which can differentiate normal eye blink and drowsiness which can prevent the driver from entering the state of sleepiness while driving. The system works well even in case of drivers wearing spectacles and under low light conditions also. During the monitoring, the system is able to decide if the eyes are opened or closed. When the eyes have been closed for about two seconds, the alarm beeps to alert the driver and the speed of the vehicle is reduced. By doing this many accidents will reduced and provides safe life to the driver and vehicle safety. A system for driver safety and car security is presented only in the luxurious costly cars. Using drowsiness detection system, driver safety can be implemented in normal cars also.

The future works may focus on the utilization of outer factors such as vehicle states, sleeping hours, weather conditions, mechanical data, etc, for fatigue measurement. Driver drowsiness pose a major threat to highway safety, and the problem is particularly severe for commercial motor vehicle operators. Currently there is not adjustment in zoom or direction of the camera during operation. Future work may be to automatically zoom in on the eyes once they are localized.

Chapter 6: References

[1] Computer Vision Project | Driver drowsiness detection | Full code explanation OpenCV Python Dlib. https://youtu.be/ksi42rwGyas

[2] Vandna Saini, Rekha Saini "Driver Drowsiness Detection System and Techniques", IJCSIT, Vol. 5 (3), 2014.

[3] Make Your Own Haar Cascade Object Detector On Windows - https://youtu.be/ydSXgBZ1ybk

[4] Face Recognition with Python and OpenCV | Face Recognition https://www.mygreatlearning.com/blog/face-recognition/

[5] Real-time Drowsiness Detection Tutorial | Transfer Learning | TensorFlow | Python | OpenCV https://youtu.be/qwUIFKi4V48

[6] Presentation On Driver Drowsiness Detection System https://www.youtube.com/watch?v=TQwZ7E2B85k

[7] Drowsiness detection with OpenCV by Adrian Rosebrock https://www.pyimagesearch.com/2017/05/08/drowsiness-detection-opencv/

[8] https://www.researchgate.net/publication/319464008_Driver_Drowsiness_Detection_Systems

[9] The detection of drowsiness using a driver monitoring system https://www.tandfonline.com/doi/full/10.1080/15389588.2019.1622005

[10] Train Neural Network by loading your images |TensorFlow, CNN, Keras tutorial https://www.youtube.com/watch?v=uqomO_BZ44g

[11] Face Recognition using Deep Learning | Convolutional-Neural-Network | TensorFlow | TfLearn https://www.youtube.com/watch?v=-I-kSz0rblU

[12] Motivation For Drowsiness Detection Information Technology

https://www.ukessays.com/essays/information-technology/motivation-for-drowsiness-detection-information-technology-essay.php

[13] Driver drowsiness detection

https://www.slideshare.net/vigneshwarvs/driver-drowsiness-detection

[14] Neural Network - Investopedia

https://www.investopedia.com/terms/n/neuralnetwork.asp#:~:text=A neural network is a, organic or artificial in nature.

[15] How to build your own Neural Network

https://towardsdatascience.com/how-to-build-your-own-neural-network-from-scratch-in-python-68998a08e4f6

[16] What is Neural Network: Overview, Applications

https://www.simplilearn.com/tutorials/deep-learning-tutorial/what-is-neural-network

[17] Convolutional Neural Networks (CNNs) explained

https://www.youtube.com/watch?v=YRhxdVk_sIs

[18] Simple explanation of convolutional neural network | Deep Learning Tutorial 23 (Tensorflow & Python)

https://www.youtube.com/watch?v=zfiSAzpy9NM

[19] Face Recognition using Deep Learning | Convolutional-Neural-Network | TensorFlow | TfLearn

https://www.youtube.com/watch?v=-I-kSz0rblU