WALCHAND COLLEGE OF ENGINEERING

SANGLI

(An Autonomous Institute)

A Mini-Project Report on

Driver Drowsiness Detection System

Submitted By

Miss. Shweta R. Sakhare	**2017BTEIT00028**
Miss. Arati R. Akoskar	**2017BTEIT00053**
Miss. Ankita T. Mankar	**2017BTEIT00065**

Miss A.D. Shinde	**Dr. S. P. Sonavane**
(Project Guide)	**(HOD)**

Department of Information Technology

Third Year B. Tech.

2019 - 2020

CERTIFICATE

This is to certify that project report entitled as Driver Drowsiness Detection.

Submitted by

Miss. Shweta R. Sakhare(2017BTEIT00028)

Miss. Arati R. Akoskar(2017BTEIT00053)

Miss. Ankita T. Mankar(2017BTEIT00065)

Has undergone a Mini Project work and successfully completed in the academic year 2019-2020(SEM-II).

Date : 28/05/2020 **Ms. A. D. Shinde**

Place : Sangli **(Project Guide)**

ACKNOWLEDGEMENT

We are rather infused by the kind guidance of Ms. A. D. Shinde who put in the cradle of our Engineering studies and evaluated us to this end and mean of our project. We could not have been able to complete the project without the valuable guidance of the Respected Panel. Class Teacher Mrs. B. S. Shetty and our hounarable HoD Dr. S.P.Sonavane madam gave her valuable guidance, experience and time to make the project a success. Without the guidance of all these respected members, we are sure to be orphans in the vast ocean of technologies. In the end, we would like to express a sincere thanks to all the people who helped us in the project completion directly or indirectly and feel lucky to have got their help.

ABSTRACT

Drowsiness and Fatigue of drivers are amongst the significant causes of road accidents. Every year, they increase the amounts of deaths and fatalities injuries globally. In this report, Driver Drowsiness System (DDS) is presented to reduce the number of accidents due to drivers fatigue and hence increase the transportation safety; this system deals with automatic driver drowsiness detection based on OpenCv python Technology. We used an algorithm to locate, track, and analyze both the drivers face and eyes to measure PERCLOS, a scientifically supported measure of drowsiness associated with slow eye closure.

TABLE OF CONTENT

INTRODUCTION

1.1 Introduction

A countless number of people drive on the highway day and night. Taxi drivers, bus drivers, truck drivers and people traveling long-distance suffer from lack of sleep. Due to which it becomes very dangerous to drive when feeling sleepy.The majority of accidents happen due to the drowsiness of the driver.

So, to prevent these accidents we will build a system using Python, OpenCV, and Keras which will alert the driver when he feels sleepy.Drowsiness detection is a safety technology that can prevent accidents that are caused by drivers who fell asleep while driving.

1.2 Title of project

Driver Drowsiness Detection System.

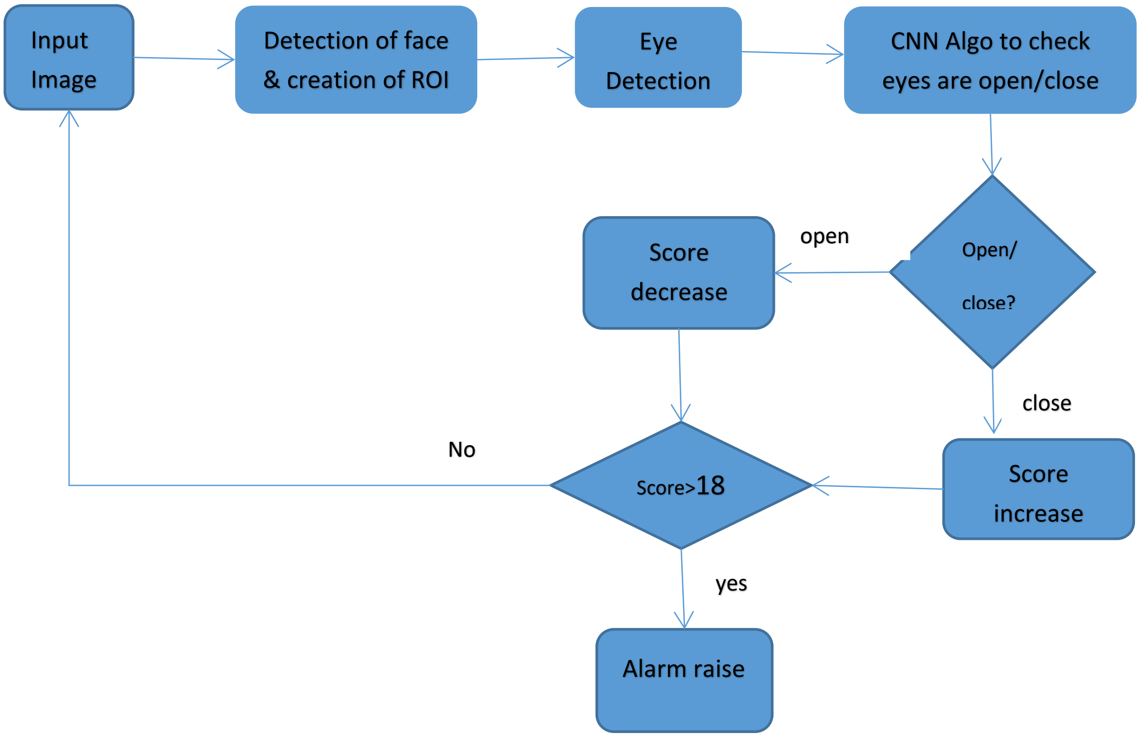
PROBLEM STATEMENT

Driver Drowsiness Detection System using OpenCv and Keras.

OBJECTIVES

- The main objective is to first design a system to detect driver’s drowsiness by continuously monitoring retina of the eye.
- To alert the driver on the detection of drowsiness by using buzzer or alarm.

IMPLEMENTATION



Model Architecture:

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.

The CNN model architecture consists of the following layers:

- Convolutional layer; 32 nodes, kernel size 3
- Convolutional layer; 32 nodes, kernel size 3
- Convolutional layer; 64 nodes, kernel size 3

- Fully connected layer; 128 nodes

The final layer is also a fully connected layer with 2 nodes.

Lets Understand how algorithm works step by step:

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.

```
for (x,y,w,h) in faces:
    cv2.rectangle(frame, (x,y), (x+w, y+h), (100,100,100), 1 )
```

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 = frame[ y : y+h, x : x+w ]
```

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

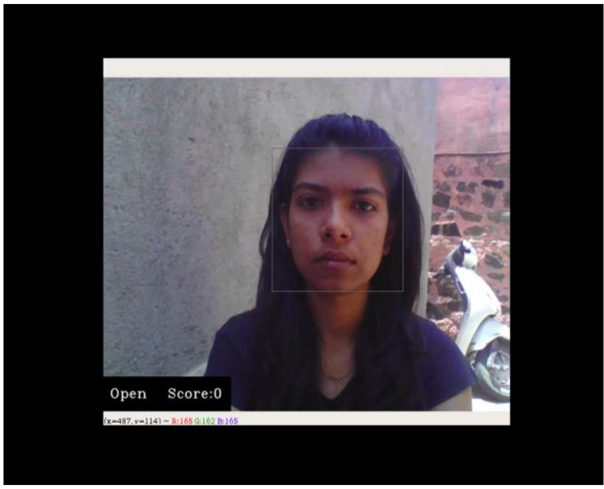
We are using CNN classifier for predicting the eye status. To feed our image into the model, we need to perform certain operations because the model needs the correct dimensions to start with. First, we convert the color image into grayscale using **r_eye = cv2.cvtColor(r_eye, cv2.COLOR_BGR2GRAY)**. 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.

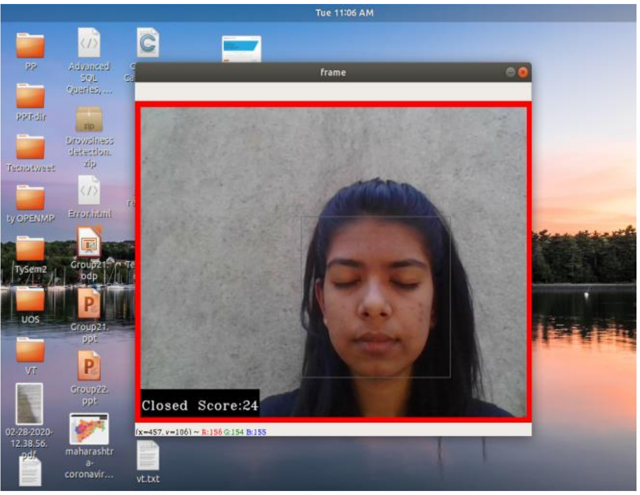
Step 5 – Calculate Score to Check whether Person is Drowsy

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.

```
cv2.putText(frame, “Open”, (10, height-20), font, 1, (255,255,255), 1, cv2.LINE_AA )
```

A threshold is defined for example if score becomes greater than 15 that means the person’s eyes are closed for a long period of time. This is when we beep the alarm using **sound.play()**





In the above, first image is when the eyes are open then score decrease,it detects the score and eyes are open.

In the second image,eyes are closed for some time then score will automatically increase and when the score is getting above 18 then it detect driver is drowsy and detect red frame and it raise the alarm.

SIGNIFICANCE OF PROJECT

As most of the accidents are caused due to drowsiness,so this project will help to decrease the crashes or accidents.

Many people use public transport facility for travelling.For their safety this system can be used in public and private vehicles.

Drowsiness detection is a safety technology that can prevent accidents that are caused by drivers who fell asleep while driving.

It can also be used for heavy vehicles for eg. Trucks.

FUTURE WORK

In future, this prototype can be extended to give alarm before sleeping by calculating the heart beat measure without physical disturbance i.e., non intrusive method using modified ECG methods. Usually in ECG method key points of body (For example chest, head, wrist etc.,) are sticked with wire. In the extended method, sticking wire may be avoided. This will lead us to a way to find out the optimum level of drowsiness. Further, this prototype will be extended to monitor the reflect ray from eye using nano camera. If the reflection ray is absent, then eye is closed otherwise eye is opened. We believe that this will create a better opportunity to detect drowsiness.

REFERENCE

1)Miss. Kanchan Sontakke, "Efficient Driver Fatigue Detection and Alerting System", International Journal of Scientific and Research Publications, Volume 5, Issue 7, July 2015

2)Puja Seemar, Anurag Chandna, "Drowsy Driver Detection using Image Processing", ISSN: 2277-9655, July 2017