GALGOTIAS UNIVERSITY

(Established under Galgotias University Uttar Pradesh Act No. 14 of 2011)

DRIVER DROWSINESS DETECTION

A Report for the Evaluation 3 of Project 2

Submitted by

SAURAV SINGH

(1613112041)

in partial fulfilment for the award of the degree

of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING WITH SPECIALIZATION OF DATA ANALYTICS

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING

Under the Supervision of

Mr. SANCHIT SAPRA , M.Tech.,

Assistant Professor

APRIL / MAY- 2020

SCHOOL OF COMPUTING AND SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

Certified that this project report **"DRIVER DROWSINESS DETECTION SYSTEM "** is the bonafide work of "**SAURAV SINGH (1613112041)"** who carried out the project work under my supervision.

SIGNATURE OF HEAD

Dr. MUNISH SHABARWAL,
PhD (Management), PhD (CS)
Professor & Dean,
School of Computing Science & Engineering

SIGNATURE OF SUPERVISOR

Mr. SANCHIT SAPRA , M.Tech.
Assistant Professor
School of Computing Science & Engineering

TABLE OF CONTENTS

Abstract

Drowsy Driver Detection System has been developed using a non-intrusive machine vision based concepts. The system uses a small monochrome security camera that points directly towards the driver's face and monitors the driver's eyes in order to detect fatigue. In such a case when fatigue is detected, a warning signal is issued to alert the driver. This report describes how to find the eyes, and also how to determine if the eyes are open or closed. The algorithm developed is unique to any currently published papers, which was a primary objective of the project. The system deals with using information obtained for the binary version of the image to find the edges of the face, which narrows the area of where the eyes may exist. Once the face area is found, the eyes are found by computing the horizontal averages in the area. Taking into account the knowledge that eye regions in the face present great intensity changes, the eyes are located by finding the significant intensity changes in the face. Once the eyes are located, measuring the distances between the intensity changes in the eye area determine whether the eyes are open or closed. A large distance corresponds to eye closure. If the eyes are found closed for 5 consecutive frames, the system draws the conclusion that the driver is falling asleep and issues a warning signal. The system is also able to detect when the eyes cannot be found, and works under reasonable lighting conditions.

Introduction

(I) Overall Description

Automotive population is increasing exponentially in our country. The Biggest problem

regarding the increased use of vehicles is the rising number of road accidents. Road accidents are undoubtedly a global menace in our country. The frequency of road accidents in India is among the highest in the world. According to the reports of the National Crime Records Bureau (NCRB) about 135,000 road accidents-related deaths occur every year in India. The Global Status Report on Road Safety published by the World Health Organization (WHO) identified the major causes of road accidents are due errors and carelessness of the driver. Driver sleepiness, alcoholism and carelessness are the key contributions in the accident scenario.

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.

(II) Purpose

The drowsiness detection system can be used for different applications. One of them is heavy vehicles for example trucks, since the drivers of trucks have long driving periods. It can also be used for commercial vehicles. Many people use public transport facility for travelling. For their safety this system can be used in public vehicles. Heavy things are lifted by using cranes and transporting them to other places. So for overloaded cranes and mobile cranes this system can be used to avoid accidents related to drowsiness.

(III) Scope

In this Python project, we will be using OpenCV for gathering the images from webcam

and feed them into a Deep Learning model which will classify whether the person's eyes are 'Open' or 'Closed'. 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 various advantages of the implemented system are mentioned below

1. Detection of drowsiness

2. Decreasing road accidents

3. No need of monitoring cameras or other devicesare attached or aimed at the driver.

4. This method is practically applicable.

PROPOSED SYSTEM

The model we used is built with Keras using **Convolutional Neural Networks (CNN)**. 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

Let’s now understand how our 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.

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

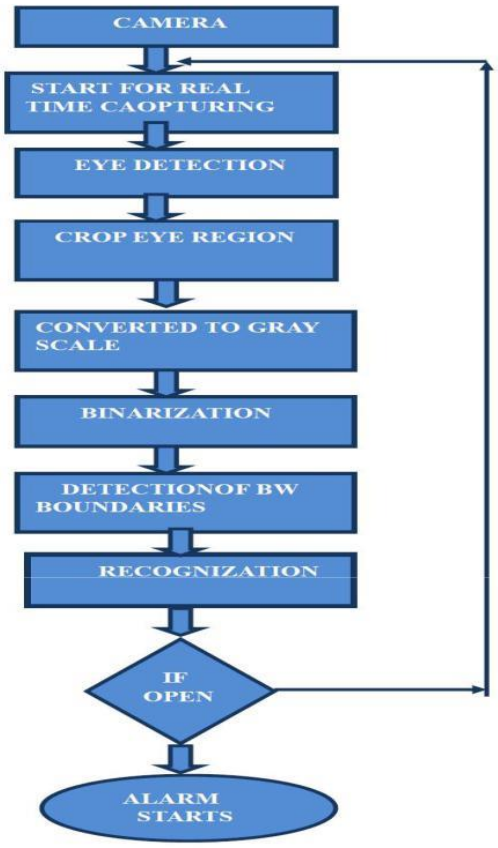
image. This can be achieved by extracting the boundary box of the eye and then we can pull out the eye image from the frame with this code.

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.



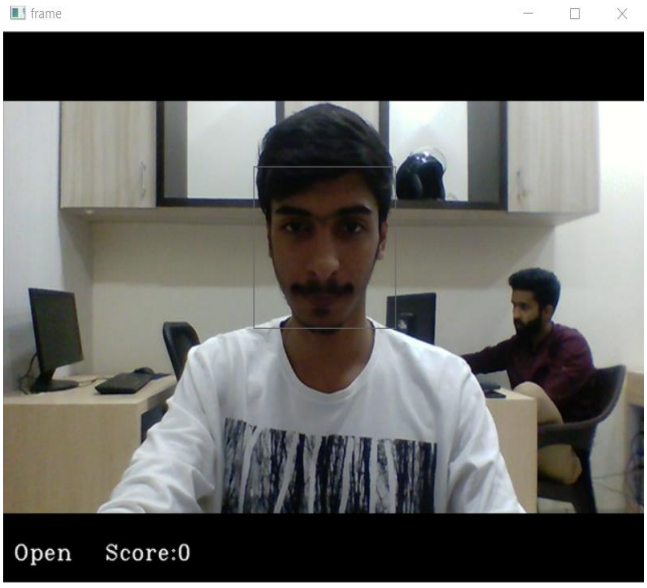
RESULT/ OUTPUT

A. Drowsiness Detection Function

The state of the eyes (whether it is open or closed) is determined by distance between the first two intensity changes found in the above step. When the eyes are closed, the distance between the y – coordinates of the intensity changes is larger if compared to when the eyes are open.

B. Judging Drowsiness

When there are 5 consecutive frames find the eye closed, then the alarm is activated, and a driver is alerted to wake up. Consecutive number of closed frames is needed to avoid including instances of eye closure due to blinking.



CONCLUSION

The driver abnormality monitoring system developed is capable of detecting drowsiness, drunken and reckless behaviours of driver in a short time. The Drowsiness Detection System developed based on eye closure of the driver can differentiate normal eye blink and drowsiness and detect the drowsiness while driving. The proposed system can prevent the accidents due to the sleepiness while driving. The system works well even in case of drivers wearing spectacles and even under low light conditions if the camera delivers better output.

Information about the head and eyes position is obtained through various self-developed image a warning signal is issued. processing judges the driver's alertness level on the basis of continuous eye closures. processing algorithms. During the monitoring, the system is able to decide if the eyes are opened or closed. When the eyes have been closed for too long,

REFERENCES

- Bill Fleming, “New Automotive Electronics Technologies”, International Conference on Pattern Recognition, pp. 484- 488,December 2012.
- Ann Williamson and Tim Chamberlain,“Review of on-road driver fatigue monitoring devices”, NSW Injury Risk Management Research Centre, University of New South Wales, , July 2013.
- E. Rogado, J.L. García, R. Barea, L.M. Bergasa, Member IEEE and E. López, February, 2013, “Driver Fatigue Detection System”, Proceedings of the IEEE International Conference on Robotics and Biometics, Bangkok, Thailand.
- Bill Fleming, “New Automotive Electronics Technologies”, International Conference on Pattern Recognition, pp. 484- 488,December 2012.
- Miaou, “Study of Vehicle Scrap page Rates,” Oak Ridge National Laboratory, Oak Ridge, TN,, S.P.,April 2012.