International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-11 Issue-3, September 2022

An Approach for Detecting Drowsy Drivers in Vehicle using CNN Techniques

Ch. Prathyusha, Mrutyunjaya S Yalawar

Abstract: ***On average, 1200 accidents happen on Indian roadways every day, resulting in 400 fatalities and numerous injuries. Accidents brought on by sleepiness can be the result of fatigue or intoxication. Long periods of driving or drinking can make drivers drowsy, which is their main source of distraction. can make drivers drowsy, which is their main source of distraction. There's a chance that this diversion will lead to. The driver, additional passengers, and pedestrians were also murdered in addition to those in the other vehicles. along a highway A driver's negligence on the road could lead to their own demise, the deaths of others, and a challenging scenario for those people's families. To avoid such accidents, I proposed a system that alerts the driver if she or he begins to feel sleepy. We put the approach into practice by employing a machine learning model based on computer vision. The driver's face is fed into a classification algorithm that has been trained on images of non-drowsy and drowsy faces. This algorithm uses landmark detection to determine whether the face is sleepy or awake. The system generates an alarm if the driver's face is sleepy. The alarm can alert the driver that he or she is drowsy and allow the driver to take the necessary actions. So, in order to avoid these accidents, we will create a system using Python, OpenCV, and Keras that will alert the driver if he feels unsafe. sleepy. Drowsiness detection is a safety technology that can help prevent accidents caused by drowsiness.by drivers who nodded off while driving.***

Keywords: ***Convolutional Neural Networks, Drowsy driver, Drowsiness detection, Machine learning.***

I. INTRODUCTION

Every day and night, a large number of people use the highway. Taxi drivers, bus drivers, two truck drivers, and people travelling long distances are all sleep deprived. As a result, driving while sleepy is extremely dangerous. Drowsiness detection is a safety technology that helps to avoid accidents caused by distracted drivers. Driver drowsiness causes a large number of accidents each year. According to a recent survey, around 1200 deaths are caused by tiredness-related issues. This becomes a major issue not only for drivers but also for other people who use the road. The majority of accidents are caused by the driver's drowsiness. To avoid these accidents, we will create a system using Python, OpenCV, and Keras that will alert the driver if he becomes sleepy. The primary goal of this project is to create a system that detects fatigue using Opencv, Keras, and Tensorflow. This program's goal is to create a system that can instantly determine whether a driver's eyes are open or closed when they are within a certain distance. The idea behind this system is that a person's eyes can indicate the symptoms of their fatigue and prevent accidents by raising an alarm that is, ideally, loud enough to alert the driver to the impending accident and bring the driver back to sense. This project works by first identifying the area containing the eyes. To accomplish this, the entire face must first be searched for and identified before selecting a Region of Interest, or region containing the eyes. The system then determines whether the eyes are open or closed.

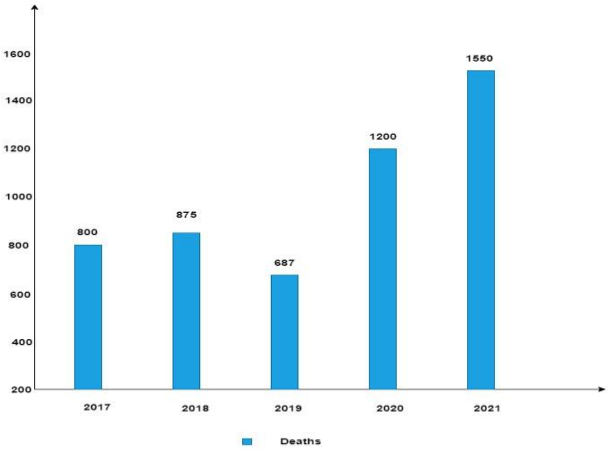

Fig1. Deaths occurred due to drowsiness

In this fig1, it shows that each year how many members deaths are caused due the drowsiness while travelling. Each year, drowsy driving accounts for about 100,000 crashes, 71,000 injuries, and 1,550 fatalities. Drowsiness driving contributes to an estimated 9.5% of all crashes.

II. LITERATURE REVIEW

The eye coordinates used in this paper [1] are used to track a person's eye movement, assess whether eye is closed or opened, and create an alarm if the driver seems sleepy. This paper [2] examines the condition of closed eyes using an algorithm based on Haar detectors and convolutional neural networks, taking into use existing databases of photos of faces and eyes. Use of neural network-based approaches to identify such microsleep and sleepiness is the main topic of this article [3].

Manuscript received on 31 July 2022 | Revised Manuscript received on 04 August 2022 | Manuscript Accepted on 15 September 2022 | Manuscript published on 30 September 2022.

* Correspondence Author

Ch Prathyusha, Department of Computer Science and Engineering, CMR Engineering, (CMREC) College, Hyderabad (Telangana), India. Email: prathyushach06@gmail.com

Mrutyunjaya S Yalawar, Department of Computer Science and Engineering, CMR Engineering (CMREC) College, Hyderabad (Telangana), India. Email:mrutyunjaya.cmrec20@cmrec.ac.in

The accuracy of this study was increased by classifying tiredness using facial landmarks that the camera picked up and sent to the convolutional neural network (CNN) [4]. The technology informs the driver with alarms or reduces acceleration when anomalies are found, lowering the risk of accidents. This paper [5] discusses the development of a drowsy driver alert system utilizing such a technique, , in which the Video Stream Processing (VSP) is examined using the Eye Aspect Ratio and Euclidean distance of the eye. For accurate eye detection, the face landmark method is also employed. The IoT module delivers an alert messages with collision impact and position information when driver weariness is identified, alerting the driver via Raspberry pi monitoring system. In this paper [6], Driver fatigue and drowsiness are thought to be indicated by a variety of physical and facial signs, such as tired eyes and yawning. These traits are an indication of a troublesome driver's state. By using a heartbeat sensor and night vision camera to continuously monitor the driver's level of sleepiness, this project [7] addresses the problem. A heartbeat sensor and a night vision camera, respectively, are used to continually monitor the driver's heartbeat count and the amount of eye blinks for a period of two minutes. This developed system mathematically manipulates the driver's level of sleepiness and weariness based on the measured data. An exhaustive assessment of recent efforts on driver sleepiness detection and alarm systems is presented in this paper [8]. We also discuss the many machine learning methods that are utilised to assess the driver's condition, including the PERCLOS algorithm, the HAAR-based cascade classifier, and OpenCV. Finally, we list the difficulties that the present systems are facing and discuss the corresponding research potential. This paper [9] suggests monitoring the eye closure rate and yawning rate as a means to identify sleepiness symptoms in drivers. This study explains how to recognise the mouth and eyes in a video taken during an experiment by MIROS (Malaysian Institute of Road Safety). In simulated driving circumstances, this paper [10] investigates tired drivers' spontaneous yawning behavior as a symptom of intoxication. We examine a tagged dataset of recordings of awake versus sleep-deprived drivers and show the relationship between yawning, handover-facial touches, and face occlusions. Along with yawning and eye movement, we suggest that face touching can be employed as an unique cue in automated drowsiness detection.

III. METHODOLOGY

A. Module Description

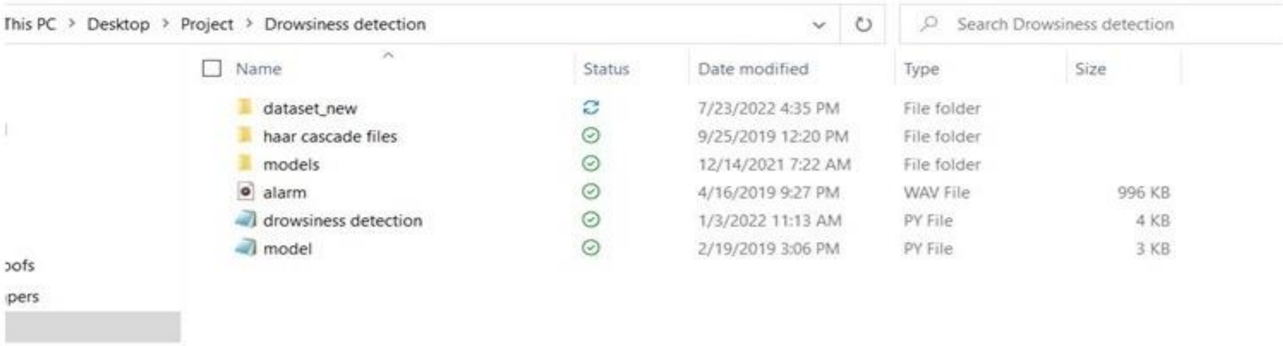

Fig 2: Module Description

- The "haar cascade files" folder contains the xml files required to detect objects in the image. In our case, we're looking for the person's face and eyes.
- Our model file "cnnCat2.h5", which was trained using convolutional neural networks, is located in the models folder.
- When someone is feeling sleepy, an audio clip called "alarm.wav" is played.
- The "Model.py" file contains the programme that we used to train our classification model on our dataset. This file contains an implementation of a convolutional neural network.
- Our project's main file is "Drowsiness detection.py" We must run this file to begin the detection procedure.

Let's go over our algorithm step by step.

Step 1 - As input, use an image from a camera.

We will use a webcam to capture images as input. So, in order to access the webcam, we created an infinite loop that captures each frame. We employ the cv2 method provided by OpenCV. For the camera's access and to set the capture object, use Video Capture (0) (cap). Each frame is read using cap.read(), and the image is saved in a frame variable.

Step 2: Detecting Faces in Images and Creating Regions of Interest (ROI).

To detect the face in the image, we must first convert it to grayscale because the OpenCV algorithm for object recognition only accepts grayscale images as input. Color information is not required to detect the objects. To detect faces, we will employ the haar cascade classifier.Face = cv2.CascadeClassifier('path to our haar cascade xml file') is used to set our classifier face. The detection is then carried out using faces = face. detectMultiScale(gray). It gives back a list of detections with the height, width, and x, y coordinates of the object's boundary box. We can now iterate over the faces, drawing boundary boxes for each one.

Step 3: Extract the eyes from the ROI and feed them to the classifier.

The method for detecting eyes is the same as that for detecting faces. We first set the cascade classifier for eyes in leye and reye, then detect the eyes with left eye = leye. Detect MultiScale(gray).

Retrieval Number: 100.1/ijrte.C72460911322
DOI: 10.35940/ijrte.C7246.0911322
Journal Website: www.ijrte.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

We must now extract only the data from the eyes from the entire image. This can be accomplished by extracting the eye's boundary box and then using this code to extract the eye image from the frame. **l_eye** only contains the eye's image data. This information will be fed into our CNN classifier, which will predict whether the eyes are open or closed. Similarly, the right eye will be extracted into **r_eye.**

Step 4: The classifier will determine whether the eyes are open or closed.

The CNN classifier is being used to predict the eye status. To feed our image into the model, we must first perform some operations because the model requires the proper **dimensions to** begin with. First, we use r eye = cv2.cvtColor(r eye, cv2.COLOR BGR2GRAY) to convert the colour image to grayscale. The image is then resized to 24*24 pixels because our model was trained on 24*24 pixel images using cv2.resize(r eye, (24,24)). We normalise our data to improve convergence. r eye/255 = r eye (All values will range from 0 to 1). To feed our classifier, increase the dimensions. Model = load model('models/cnnCat2.h5') was used to load our model. Now we use our model to predict each eye, lpred = model.predict classes(l eye). If the value of lpred[0] is 1, the eyes are open; if the value of lpred[0] is 0, the eyes are closed.

Step 5 - Determine whether the person is drowsy by calculating a score.

The score is essentially a value that will be used to calculate how long the individual has closed his eyes. So, if both eyes are closed, the score increases; if both eyes are open, the score decreases. The cv2.putText() function is being used to draw the outcome on the screen, which will show the person's status in real time. A threshold is defined, for example, if the score exceeds 15, it indicates that the person's eyes have been closed for an extended period of time. This occurs when we use sound to sound the alarm. play().

B. Model Architecture

We built the model with Keras and Convolutional Neural Networks. A convolutional neural network is a type of deep neural network that performs exceptionally well for image classification. A CNN is comprised of three layers: an output layer ,the input layer, and a hidden layer with multiple layers. These layers are subjected to a convolution operation through the use of a filter that multiplies the layer's and the filter's 2D matrices together.The layers of the CNN model architecture are as follows:

- Convolutional layer with 32 nodes and a kernel size of 3
- Convolutional layer with 32 nodes and a kernel size of 3
- Convolutional layer with 64 nodes and a kernel size of 3
- Fully connected layer with 128 nodes

C. Prerequisites

The only thing we need for this Python project is a webcam to capture images. Python (3.7 version recommended) must be installed on your system before installing the required packages with pip.

- **OpenCV** – pip install OpenCV -python
  OpenCV is used for face and eye detection
- **TensorFlow –** pip install TensorFlow
  Keras uses backend as TensorFlow..
- **Keras –** pip install Keras
  Keras is used to construct classification models.
- **Pygame** – pip install Pygame
  Pygame is used to generate alarm sounds.

IV. RESULTS

Before you begin to format your paper, first write and save the content as a separate text file. Complete all content and organizational editing before formatting.

When eyes are opened

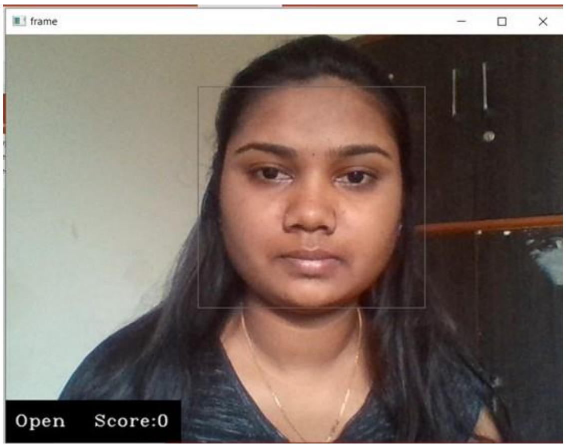

Fig 3: When Eyes are Opening

When eyes are closed

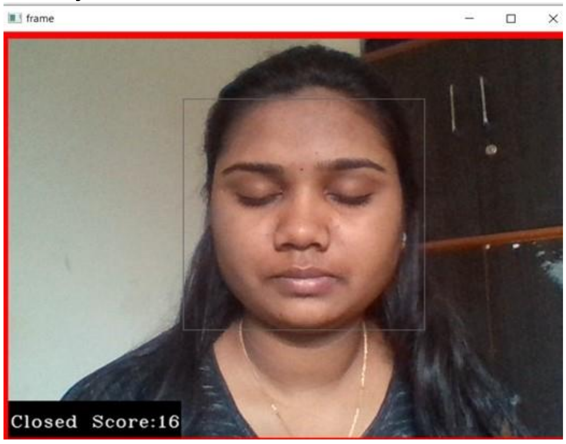

Fig 4: When Eyes are closing

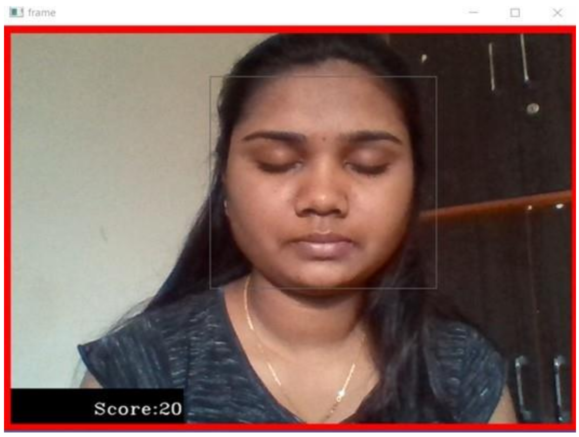

Fig 5: When Eyes are closing

Retrieval Number: 100.1/ijrte.C72460911322
DOI: 10.35940/ijrte.C7246.0911322
Journal Website: www.ijrte.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

When eyes are opened using spectacles

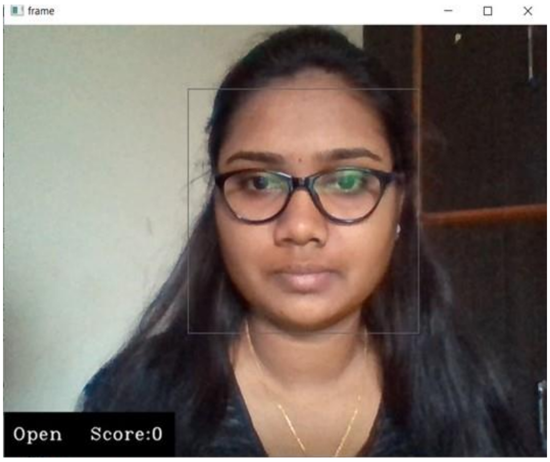

Fig 6: When Eyes are Opening using spectacles

When eyes are closed using spectacles

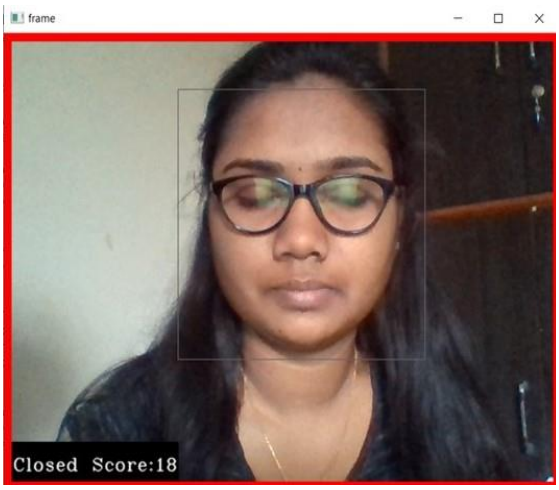

Fig 7: When Eyes are closing using spectacles

V. CONCLUSION

The developed driver abnormality monitoring system has the ability to detect drowsiness, drunkenness, and reckless driving behavior in a short period of time. The Drowsiness Detection System, which was developed based on the driver's eye closure, can distinguish between normal eye blink and drowsiness and detect drowsiness while driving. The proposed system can help to avoid accidents caused by sleepiness while driving. The system works well even when drivers wear spectacles and in low-light situations if the camera produces a higher quality image. When information about the position of the head and eyes is obtained using various self-developed image processing techniques, a warning signal is generated. The processing evaluates the driver's alertness level based on continuous eye closures. Algorithms for processing During monitoring, the system can determine whether the eyes are open or closed. When the eyes have been closed for an extended period of time.

REFERENCES

1. Alert Maganti Manasa, Vikas B,K. Subhadra.,"Drowsiness detection using Eye-Blink frequency and Yawn count for Driver": International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075, Volume-9 Issue-2S3, December 2019 [CrossRef]
2. N.A. Andriyanov1* , A.A. Lutfullina1: "Eye recognition system to prevent accidents on the road":The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XLIV-2/W1-2021 4th Int. Worksh. on "Photogrammetric & computer vision techniques for video surveillance, biometrics and biomedicine", 26–28 April 2021, Moscow,Russia. [CrossRef]
3. Rateb Jabbar∗†, Mohammed Shinoy∗ , Mohamed Kharbeche∗ , Khalifa Al-Khalifa§ , Moez Krichen‡ , Kamel Barkaoui†: "Driver Drowsiness Detection Model Using Convolutional Neural Networks Techniques for Android Application": arXiv:2002.03728v1 [cs.CV] 17 Jan 2020 [CrossRef]
4. Venkatramaphanikumar Sistla[1*] ,Venkata K.K. Kolli[1] Navya B. Kukkapalli[2] , Sri S. Katuri , Sasikala Vallabhajosyula[1] : "Stacked Ensemble Classification Based Real-Time Driver Drowsiness Detection: International Journal of Safety and Security Engineering": Vol. 10, No. 3, June, 2020, pp. 365-371 Journal homepage: http://iieta.org/journals/ijsse. [CrossRef]
5. Anil Kumar Biswal,1 Debabrata Singh,2 Binod Kumar Pattanayak,1 Debabrata Samanta , 3 and Ming-Hour Yang 4 : "IoT-Based Smart Alert System for Drowsy Driver Detection: Hindawi Wireless Communications and Mobile Computing": Volume 2021, Article ID 6627217, 13 pages https://doi.org/10.1155/2021/6627217. [CrossRef]
6. Mahek Jain, Bhavya Bhagerathi, Sowmyarani C N: Real-Time Driver Drowsiness Detection using Computer Vision: International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249-8958 (Online), Volume-11 Issue-1, October 2021. [CrossRef]
7. B.Lakshmi Priya, M.Prithviraj, C.Baraniraj, P.Duraikannu : "Accident Prevention System using Driver Drowsiness Detection: Volume 3, Issue 7, July – 2018 International Journal of Innovative Science and Research Technology":ISSN No:-2456-2165. [CrossRef]
8. V B Navya Kiran, Anisoor Rahman, Raksha R, Varsha K N, Dr. Nagamani N P:"Driver Drowsiness Detection": International Journal of Engineering Research & Technology (IJERT) ISSN: 2278-0181 Published by, www.ijert.org NCAIT - 2020 Conference Proceedings.
9. Badiuzaman Bin Baharu: "Driver drowsiness detection by using webcam":Universiti Teknologi PETRONAS Bandar Seri Iskandar 31750 Tronoh Perak Darul Ridzuan.
10. Zhuoni Jie∗, Marwa Mahmoud†, Quentin Stafford-Fraser‡, Peter Robinson§, Eduardo Dias¶ and Lee Skrypchuk44 : Analysis of yawning behaviour in spontaneous expressions of drowsy drivers: 2018 13th IEEE International Conference on Automatic Face & Gesture Recognition.

AUTHORS PROFILE

Ch. Prathyusha, Received B.Tech degree in Computer science and engineering from Geetanjali college, Hyderabad and pursuing M.Tech in Computer science and engineering from CMR Engineering College, Hyderabad. Her area of research interest is machine learning, Artificial intelligence, Deep Learning.

Mr. Mrutyunjaya S Yalawar working as Assistant Professor at CMR Engineering College, Medchal, Hyderabad, India and having more than 3 years of experience in IT-Industries as Software Developer and more than 6 years of Experience in Teaching Field. His Research area includes Artificial Intelligence, Machine Learning, NLP, Cyber Security, Blockchain. He published about more than 15 Papers in various National and International Journals.