**MINOR PROJECT REPORT**

on

**DRIVER DROWSINESS DETECTION SYSTEM USING CNN**

*Submitted in partial fulfillment for the award of the degree*

of

## **BACHELOR OF TECHNOLOGY**

in

## **ELECTRICAL AND COMPUTER ENGINEERING**

by

BORRA UMESH CHANDRA (RA2011044010014)



## FACULTY OF ENGINEERING AND TECHNOLOGY

SRM Nagar, Kattankulathur 603203

Kancheepuram Dist

**NOVEMBER 2023**

**BONAFIDE CERTIFICATE**

Certified that this project report titled “DRIVER DROWSINESS DETECTION SYSTEM USING CNN” is the bonafide work of Borra Umesh Chandra(Reg.No.RA20110440100014), Tandle Sampath Kumar(Reg.No. RA2011044010011), Shaik Riyaz Ahmed (Reg.No. RA2011044010012),

T Rakshann RA2011044010021) who carried out the project work under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

Signature of the Faculty Incharge Signature of the Course Coordinator

**Dr. D. Karthikeyan Dr. UTHRA.R**

Assistant Professor Assistant Professor

Department of EEE Department of EEE

SRM IST SRM IST

Internal Examiner External Examiner

DATE:

# ABSTRACT

# The sleepiness of the driver and rash driving are the significant reasons for street mishaps, which bring about loss of significant life, and weaken the security in the street traffic. Solid and exact driver sluggishness frameworks are needed to forestall street mishaps and to further develop street traffic wellbeing. Different driver drowsiness detection frameworks have been planned with various innovations which have a partiality towards the novel boundary of recognizing the sluggishness of the driver. These driving examples were examined dependent on the speed increase framework, speed of the vehicle, Revolution per minute (RPM), facial acknowledgment of the driver. The facial example of the driver is treated with 2D Convolution Neural Network (CNN) to recognize the conduct and driver's feeling.

# The proposed model is executed utilizing Transfer Learning, OpenCV and the test results demonstrate that the proposed model distinguishes the driver's feeling and tiredness more successfully than the current innovations. The destruction of one's life because of street mishaps has been expanding because of human deception. Different mindfulness programs have been celebrated to destroy sign prospects over covering yet the diagram stays dramatic.For investigating and surrendering Driver's Drowsiness state utilizing eyes as our subject, with Transfer Learning approach. Here, we have subsumed the Inception model over MRL Eye Dataset, separating it into 02 classes as Open and Closed eyes. In our review, we accomplished a precision of over 97% and make sure our viability for making. A pitiful just as an excellent model for Driver Drowsiness Detection.

# LIST OF FIGURES

Fig 1: System Design for Driver drowsiness Detection using Convolution Neural

# Network

# Fig 2: Depiction of facial features using haar cascade

# Fig 3: Pre-Processing and Data Augmentation

# Fig4: Depicts about Basic principle of transfer learning

Fig 5: Basic Architecture of Inception V3 Model

Fig 6: Algorithm of Driver drowsiness Detection System

Fig 7: Drowsy eyes Detection

Fig 8: Training, Validation accuracy vs number of epoch

Fig 9: Training, Validation loss vs number of epoch

# 

# 

# LIST OF TABLES

Table 1: Various accuracy of model against test data

**CONTENTS**

[ABSTRACT iii](#_TOC_250002)

[LIST OF FIGURES iv](#_TOC_250001)

[LIST OF TABLES v](#_TOC_250000)

1. INTRODUCTION 1
2. LITERATURE SURVEY 3
3. METHODOLOGY
   1. MOTIVATION 9
   2. SYSTEM DESIGN 9
   3. ALGORITHMS 15
4. RESULTS AND DISCUSSION 21
5. REALISTIC CONSTRAINTS 23
6. CONCLUSION 24
7. REFERENCES 25

**CHAPTER 1**

**INTRODUCTION**

Driver drowsiness and weariness are some of the significant reasons for engine vehicle mishaps. Improvement of a tiredness recognition framework is considered both a modern and educational test. In the vehicle area, a driver-ready control framework has been created by the Open CV that cautions the driver associated with being languid. It utilizes a vehicle-mounted camera that is associated with the path takeoff cautioning framework.

Following the comparable endeavor, a sluggishness discovery framework had been presented by Bosch, which settles on choices utilizing the information got from the sensor that is situated at the directing wheel, vehicle's driving speed, blinker, and the camera utilized for path help mounted at the front of the vehicle. Be that as it may, such wellbeing frameworks utilized for distinguishing laziness aren't far and wide and is whimsical among drivers since they are accessible just in extravagance vehicles.

As of late, the field of AI has made spearheading progress, particularly in the space of profound neural learning. Hence, the utilization of new advancements and plans can be considered as a compelling way to deal with increment the effectiveness of continuous laziness discovery just as to give a component that can be generally utilized by drivers. In this unique situation, we propose profound learning-based way to deal with recognizing the sluggishness of drivers continuously utilizing the OpenCV YOLO model, which is a profound learning-based constant article discovery calculation and inception v3 learned model, that plays out the undertaking of drowsy characterization.

Humans have always invented machines and devised techniques to ease and protect their lives, for mundane activities like traveling to work, or for more interesting purposes like aircraft travel. With the advancement in technology, modes of transportation kept on advancing and our dependency on it started increasing exponentially. It has greatly affected our lives as we know it. Now, we can travel to places at a pace that even our grandparents wouldn’t have thought possible. In modern times, almost everyone in this world uses some sort of transportation every day.

Some people are rich enough to have their own vehicles while others use public transportation. However, there are some rules and codes of conduct for those who drive irrespective of their social status. One of them is staying alert and active while driving. Neglecting our duties towards safer travel has enabled hundreds of thousands of tragedies to get associated with this wonderful invention every year. It may seem like a trivial thing to most folks but following rules and regulations on the road is of utmost importance.

While on road, an automobile wields the most power and in irresponsible hands, it can be destructive and sometimes, that carelessness can harm lives even of the people on the road. One kind of carelessness is not admitting when we are too tired to drive. In order to monitor and prevent a destructive outcome from such negligence, many researchers have written research papers on driver drowsiness detection systems. But at times, some of the points and observations made by the system are not accurate enough. Hence, to provide data and another perspective on the problem at hand, in order to improve their implementations and to further optimize the solution, this project has been done

**CHAPTER II**

**LITERATURE SURVEY**

1. Zuzana Képešiová; Ján Cigánek; Štefan Kozák, “Driver Drowsiness Detection Using Convolutional Neural Networks” in IEEE access 2020 Cybernetics & Informatics (K&I), DOI:10.1109/KI48306.2020.9039851.

The paper manages programmed identification of driver sleepiness. Distinguishing the driver's sleepiness behind the guiding haggle alarming him might lessen street mishaps. Sleepiness for this situation is caught utilizing an auto camera, by which, in light of the caught picture, the neural organization perceives regardless of whether the driver is conscious or tired. The convolutional neural network (CNN) innovation has been utilized as a part of a neural organization, where each frame is assessed independently and the normal of the last 20 frames is assessed, which relates to roughly one second in the preparation and test dataset.

To begin with, we dissect strategies for picture division, and foster a model in light of convolutional neural organizations. Utilizing an explained dataset of in excess of 2000 picture cuts we train and test the division organization to separate the driver passionate status from the pictures.

VGG Face was prepared over 2.6 million of visual sources of info and can sort them into 2622 unique classifications. A required input is a 224x244 large picture in RGB channel. The last model uses pretrained VGG Face as a base, yet to meet our requirements, the last layer of VGG Face is eliminated and on top of that little head model is added. This head model comprises of completely associated layers, trailed by clump standardization, tanh initiation capacity and dropout. Developed model was prepared with custom dataset isolated into clumps measured of 32 during 50 epochs in neural network training process and with Adam classifier and clear-cut cross entropy.

VGG Face layers were not retrained, however freezed and a head model was tuned as it were. The entire preparing process on referenced equipment required around 15 hours of preparing, while the consequences of the model were 90.77% preparing information exactness, 98.02% approval information accuracy. Despite great outcomes, when the model is tried on a totally new individual, that was excluded from dataset, a neural network battles to recognize sleepiness in an individual's fascial articulation and recommending an individual knows more often than not. Productive motion that is perceived in others is slight head

falling. Then again, every one of the recordings of caught subjects, indeed, even ones recorded later, were perceived effectively.

Despite good outcomes, when the model is tried on a totally new individual, that was excluded from dataset, a neural network battles to recognize sluggishness in an individual's fascial articulation and recommending an individual knows more often than not.

Productive motion that is perceived in others is slight head falling. Then again, every one of the recordings of caught subjects, indeed, even ones recorded later, were perceived effectively.

2.Maryam Hashemi, Alireza Mirrashid & Aliasghar Beheshti Shirazi. “Driver Safety Development: Real Time Driver Drowsiness Detection System Based on Convolutional Neural Network” in springer, 2020 / Published online: 31 August 2020 ©.

This paper centers around the test of driver wellbeing out and about and presents an original framework for driver tiredness discovery. In this framework, to recognize the falling rest condition of the driver as the indication of sleepiness, Convolutional Neural Networks (CNN) are utilized with in regards to the two objectives of continuous application, including high precision and speed. Three organizations introduced as a likely organization for eye status classification wherein one of them is a Fully Designed Neural Network (FD-NN) also, others use Transfer Learning in VGG16 and VGG19 with extra planned layers (TL-VGG).

Absence of an accessible and precise eye dataset firmly feels in the space of eye conclusion discovery. In this manner, another extensive dataset proposed. The exploratory outcomes show the high precision and low computational intricacy of the eye conclusion assessment and the capacity of the proposed structure on tiredness discovery. To find the eyes, first need to gauge the headbox, Viola and Jones calculation utilized for head location and facial milestone approach is an execution of Kazemi and Sullivan work.

This is a relapse tree AI strategy and was prepared on the ibug 300-W face milestone dataset. The benefit of this way to deal with others is the high precision of location in different head positions. In the wake of arriving at the milestone point of the eye, ROI is trimmed. The face is imbalance shape, henceforth for laziness recognition, just one eye is satisfactory to be thought of. This methodology diminishes the computational time for location perceptibly. The matchless quality of FD-NN contrast with other proposed networks is non-intricacy and speed for continuous undertakings.

This network applied to ZJU and broadened dataset. In the second and third organizations, the idea of move learning and pre-prepared CNN used to separate elements. For this reason, VGG16 and VGG19 chose as the proposed pre-prepared organizations. VGG16 is a profound convolutional neural network proposed by Simonian and Zisserman. VGG16 contains 16 layers and prepared on the ImageNet dataset, which comprises of an enormous number of pictures and having 1000 classes. VGG19 is a more top to bottom variant of VGG16 with 19 layers.

It additionally prepared on the ImageNet data set. The organization intended for pictures with the size of 224×224 pixels, in any case, can suggest different sizes, by the same token. These organizations learn low-level elements with the heaviness of the ImageNet dataset, and significant level provisions extricate with three last added completely associated layers. The first layer is an input layer, second is enactment with Relu work, and the last layer is a Sigmoid capacity as a yield layer. We call these organizations Transfer Learning VGG16 (TL-VGG16) and Move Learning VGG19 (TL-VGG19).

The objective of utilizing transfer learning is to have a more profound organization with higher exactness, particularly when the preparation data set is little, and we can't prepare a profound neural organization with that dataset. Transfer learning is likewise ready to diminish preparing time.

1. K. Satish; A. Lalitesh; K. Bhargavi; M. Sishir Prem; T Anjali. “Driver Drowsiness Detection” in IEEE access International Conference on Communication and Signal Processing, 978-1-7281-4988-2/20/$31.00 ©2020 IEEE.

This paper shows a new test model is intended for distinguishing sleepiness of driver is introduced to lessen mishaps brought about by this issue which builds transport wellbeing. In this work, two different ways are utilized to distinguish the laziness of an individual successfully. First Driver face is caught and eye retina recognition and facial component extraction are done and squinting qualities are determined then limit esteems are set. Also, the Arduino module is utilized which is coordinated with elastomeric sensors for constant computation of driver hand strain on the vehicle guiding haggle limit esteem is set. The outcome from the two techniques is taken as contribution for taking the official conclusion and alarming the driver.

Driver Drowsiness Detection requires equipment and programming parts which incorporate sensors to identify hand pressure and send qualities to Arduino and camera to distinguish face furthermore, eyes and cycle eye squinting rate. In this venture, a few techniques are applied which are clarified in this paper. Hoard calculation is utilized to pre-process the picture which remembers picture resize and shading standardization for this undertaking.

Hoard is utilized to identify productive provisions from for eye recognition also, extricate HOG highlights from the picture examples and gives the specific area of eyes from the caught picture of the driver. After catching the driver's picture and pre-processing the next interaction to appraise the laziness of a driver dependent on the squinting eye flickering rate. Qualities are determined for each edge and changes in the flicker rate are confirmed with the limit esteem. For adequately recognizing eye flickering rate Hoard is utilized which is helpful for face identification and gives precise eye identification rate.

1. Fouzia, Roopa Lakshmi R, Jayant Kumar A Rathod, Ashwitha S Shetty, Supriya k. “**Driver Drowsiness Detection System Based on Visual Features”** in IEEE access, 978-1-5386-1974-2/18/$31.00 ©2018 IEEE.

The driver drowsiness detection system is proposed in this paper, which makes use of eyeblink counts for detecting drowsiness. Specifically, the proposed framework continuously analyses the eye movement of the driver and alerts the driver by activating the vibrator when he/she is drowsy. When the eyes are detected closed for a too long time, a vibrator signal is generated to warn the driver.

The experimental results of the proposed system, which is implemented on Open CV and Raspberry Pi environment with a single camera view, illustrate the good performance of the system in terms of accurate drowsiness detection results and thereby reducing road accidents.

The proposed system is to detect closed eyes to observe drivers' fatigue and alert the driver with a buzzer and vibration on positive detection. During monitoring, the system is able to decide whether the eyes are open, closed, or drowsy. When the eyes were detected closed for too long, a warning signal is issued. This was done by mounting a camera in front of the driver and continuously capturing its real-time video using Open CV in Raspberry Pi.

Driver drowsiness can be determined from several symptoms that manifest in drowsy drivers' faces. Through the analysis of the eye status, the system will be able to tell whether the driver is drowsy or not. Initially, when the camera is in on state, video streams are continuously captured from the driver's face. To detect the eye blink, the current state of the eye is needed which is either open or closed. If the state of eyes changes from closed to open, it indicates an eye blinking.

This paper presents a drowsiness detection system in light of shape indicator calculation, that identifies the eyes, and additionally counts the eye flicker rate followed by tiredness recognition at ongoing. In the proposed framework, the subtleties

concerning the eye status is gotten through picture handling calculations, which offer a non-intrusive way to deal with distinguish sleepiness with practically no inconvenience and impedance.

1. WANGHUA DENG1, RUOXUE WU1“**Real-Time Driver Drowsiness Detection system using Facial Features**” in IEEE Access.

The face, a significant piece of the body, passes on a great deal of data. At the point when a driver is in a condition of weakness, the looks, e.g., the recurrence of squinting and yawning, are not quite the same as those in the ordinary state. In this paper, we propose a framework called DriCare, which recognizes the drivers' weariness status, like yawning, flickering, and span of eye conclusion, utilizing video pictures, without preparing their bodies with gadgets. Attributable to the inadequacies of past calculations, we present another face-following calculation to further develop the following exactness. Further, we planned another identification technique for facial areas in view of 68 central issues. Then, at that point, we utilize these facial areas to assess the drivers' state. By joining the provisions of the eyes and mouth, DriCare can caution the driver utilizing a weakness cautioning. The test results showed that DriCare accomplished around 92% accuracy.

The proposed framework, DriCare, is fabricated utilizing a business camera auto gadget, a cloud server that cycles video information, and a business cell phone that stores the outcome. While driving, the auto's camera catches the driver's representation what's more, transfers the video transfer to the cloud server in real time. Then, at that point, the cloud server examines the video and recognizes the driver's level of languor. In this stage, three principal parts are examined: the driver's face following, facial keyregion acknowledgment, and driver's exhaustion state. To meet the continuous execution of the framework, we utilize the MC-KCF calculation to follow the driver's face and perceive the facial key districts dependent on central issue recognition. Then, at that point, the cloud server assesses the driver's state when the conditions of the eyes and mouth change. At long last, the cloud server sends the outcome to the driver's cell phone and other applications, through which a notice tone is communicated if the driver is seen to be sluggish

**CHAPTER III**

**METHODOLOGY**

**MOTIVATION**

The annihilation of one’s life due to road accidents has been increasing due to human fallacy. Various awareness programs have been commemorated to eradicate portent possibilities over crust but the graph stays exponential. In this paper, we’ve tried to incorporate a lightweight system for scrutinizing and conceding Driver’s Drowsiness state using eyes as our subject, with Transfer Learning approach. Here, we have subsumed the Inception V3 model over MRL Eye Dataset, stratifying it into 02 classes as Open and Closed eyes. In our study, we achieved an accuracy of more than 92% and cocksure our efficacy for creating an insubstantial as well as an exemplary model for Driver Drowsiness Detection.

# SYSTEM DESIGN

# 

Fig 1: System Design for Driver drowsiness Detection using Convolution Neural

Network

# BLOCK DESCRIPTION:

# Input From Data Set / Camera

# Image Pre-processing and Data Augmentation.

# Image Segmentation

# Transfer Learning (Inception V3)

# Feature Extraction

# Classification

# Output

Convolution Neural Network: It is a Deep Learning algorithm which can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one from the other.

The architecture performs a better fitting to the image dataset due to the reduction in the number of parameters involved and reusability of weights.

The architecture performs a better fitting to the image dataset due to the reduction in the number of parameters involved and reusability of weights. It is widely used in Pattern Recognition, Image Processing and Visual Computing.

1.Video Processing:

\*Reading Video Frames

* For this step, OpenCV-python, a python version of OpenCV library, is used. Either can be done from a dataset or a camera in a real-time manner.

\*Detecting Faces and Eyes

* Detecting faces with haar cascade frontal face detector
* It is an Object Detection Algorithm used to identify faces in an image or a real time video.

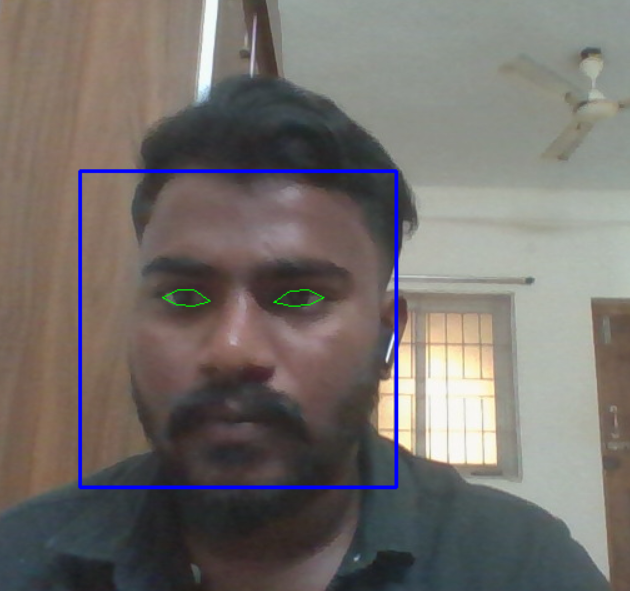
****

Fig 2: Depiction of facial features using haar cascade

2.Image Preprocessing And Data Augumentation:

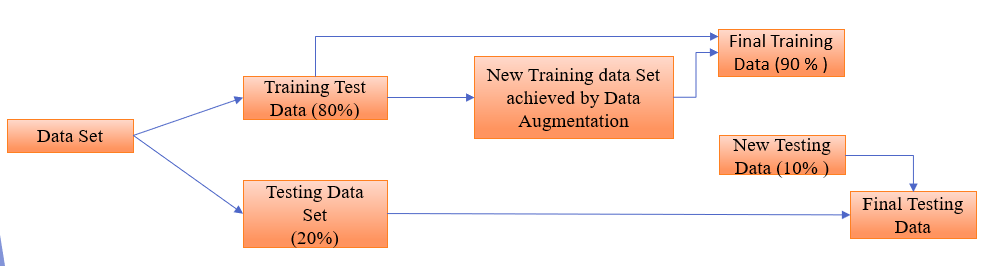


Fig 3: Pre-Processing and Data Augmentation

\*MRL Eye data set is used for the following project.

\*Eye State is the most important attribute for the model.

\*Available data is divided among the testing and training.

\*Data Augmentation: To generate new data using existed data

\*The data is divided for training and testing in the ratio of 80:20, then with the help of data augmentation new data will be generated and the further ratio will be divided as fig-2.

3.Training a Convolution Neural Network:

There are three processes included in this section:

1. Image Segmentation
2. Transfer Learning
3. Training the model (Inception V3)
4. Image Segmentation:

* Partitioning image into different sections by using contour.
* Contour is an continues line or curve that cover or bound the region of object in an image.
* OpenCV read images in BGR format.
* Image turns into feature vector, then it is given as input to the Convolution neural network.

b) Transfer Learning:

* + It is a research problem in machine learning that focuses on storing knowledge gained while solving one problem and applying it to a different but related problem.
  + It only imports the knowledge but not the result of it as it is trained in more than 1000 scenario by industry.
  + Inception v3 is a widely-used image recognition model that has been shown to attain
  + greater than 78.1% accuracy on the ImageNet dataset
  + The Knowledge of solving will be transferred but not past output, this way the model
  + goes for best of its accuracy.

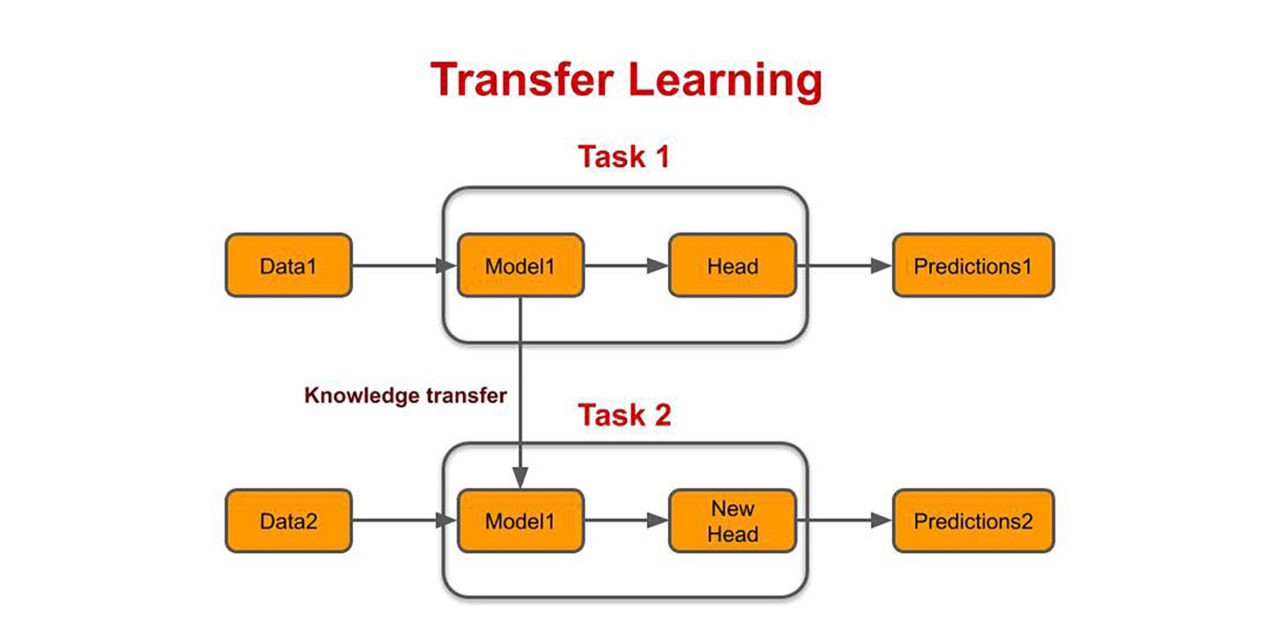


Fig4: Depicts about Basic principle of transfer learning

c)Inception V3:

* Inception Layer is a combination of all those layers namely, 1×1 Convolutional layer, 3×3 Convolutional layer, 5×5 Convolutional layer with their output filter banks concatenated into a single output vector forming the input of the next stage.
* Max pooling is a pooling operation that selects the maximum element from the region of the feature map covered by the filter. Thus, the output after max-pooling layer would be a feature map containing the most prominent features of the previous feature
* Dropout layers help in controlling for overfitting as it drops a faction of parameters.
* Dense layer is the regular fully-connected layer with a specific activation function
* 1×1 Convolutional layer before applying another layer, which is mainly used for dimensionality reduction
* All Layers must be train to get higher accuracy.

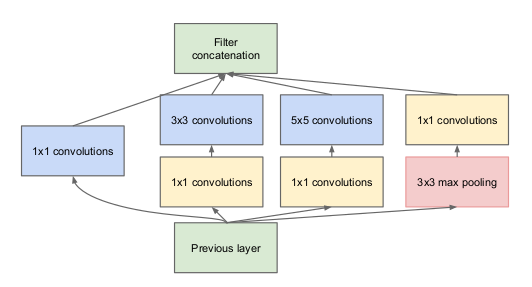


Fig 5: Basic Architecture of Inception V3 Model

Training:

Import the layers Input, Flatten, Dense, max poolan 2D layers to the inception v3 and

train the model with training data.

1. It is sparsely connected architecture, smaller the model less the overfitting.
2. \*Threshold Value: No of epoch =10 (If it takes more epoch, decrease the learning Rate-Hyper Tuning).

4. Feature Extraction, Classification and Output

* By using Haar Features, we extract the region of eye and face.
* It detects frame in grey scale format for better accuracy and send to model to detect drowsiness.
* Threshold Time for drowsiness in project is 0.4 for model training and score to recognize eye drowsiness is 15.
* If the features from eyes, face indicates that he’s drowsy it gives an immediate alarm sound.

**ALGORITHM:**

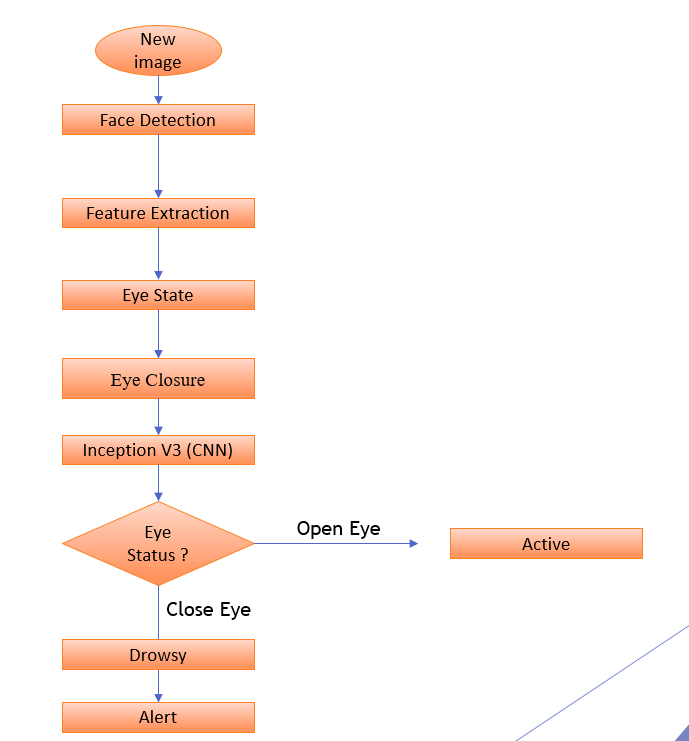
****

Fig 6 : Algorithm of Driver drowsiness Detection System

\*After training of model using transfer learning, it will be linked to OpenCV for execution

Importation of Haar Cascade:

1) ace\_cascade = cv2.CascadeClassifier(cv2.data.haarcascades + 'haarcascade\_frontalface\_default.xml')

eye\_cascade = cv2.CascadeClassifier(cv2.data.haarcascades + 'haarcascade\_eye.xml')

OpenCV Face Detection :

''This script uses OpenCV's haarcascade (face and eye cascade) to detect face

and eyes in a given input image.'''

#Import necessary libraries

import cv2 as cv

import numpy as np

#Load face cascade and hair cascade from haarcascades folder

face\_cascade = cv.CascadeClassifier("haarcascades/haarcascade\_frontalface\_default.xml")

eye\_cascade = cv.CascadeClassifier("haarcascades/haarcascade\_eye.xml")

#Read image in img and convert it to grayscale and store in gray.

#Image is converted to grayscale, as face cascade doesn't require to operate on coloured images.

img = cv.imread('images/test.jpeg')

gray = cv.cvtColor(img, cv.COLOR\_BGR2GRAY)

#Detect all faces in image.

faces = face\_cascade.detectMultiScale(gray, 1.3, 5)

#Draw a rectangle over the face, and detect eyes in faces

for (x,y,w,h) in faces:

cv.rectangle(img,(x,y),(x+w,y+h),(255,0,0),2)

#ROI is region of interest with area having face inside it.

roi\_gray = gray[y:y+h, x:x+w]

roi\_color = img[y:y+h, x:x+w]

#Detect eyes in face

eyes = eye\_cascade.detectMultiScale(roi\_gray)

for (ex,ey,ew,eh) in eyes:

cv.rectangle(roi\_color,(ex,ey),(ex+ew,ey+eh),(0,255,0),2)

cv.imshow('Image', img)

cv.waitKey(0)

cv.destroyAllWindows()

MAIN CODE

'''This script detects if a person is drowsy or not,using dlib and eye aspect ratio

calculations. Uses webcam video feed as input.'''

#Import necessary libraries

from scipy.spatial import distance

from imutils import face\_utils

import numpy as np

import pygame #For playing sound

import time

import dlib

import cv2

#Initialize Pygame and load music

pygame.mixer.init()

pygame.mixer.music.load('audio/alert.wav')

#Minimum threshold of eye aspect ratio below which alarm is triggerd

EYE\_ASPECT\_RATIO\_THRESHOLD = 0.3

#Minimum consecutive frames for which eye ratio is below threshold for alarm to be triggered

EYE\_ASPECT\_RATIO\_CONSEC\_FRAMES = 50

#COunts no. of consecutuve frames below threshold value

COUNTER = 0

#Load face cascade which will be used to draw a rectangle around detected faces.

face\_cascade = cv2.CascadeClassifier("haarcascades/haarcascade\_frontalface\_default.xml")

#This function calculates and return eye aspect ratio

def eye\_aspect\_ratio(eye):

A = distance.euclidean(eye[1], eye[5])

B = distance.euclidean(eye[2], eye[4])

C = distance.euclidean(eye[0], eye[3])

ear = (A+B) / (2\*C)

return ear

#Load face detector and predictor, uses dlib shape predictor file

detector = dlib.get\_frontal\_face\_detector()

predictor = dlib.shape\_predictor('shape\_predictor\_68\_face\_landmarks.dat')

#Extract indexes of facial landmarks for the left and right eye

(lStart, lEnd) = face\_utils.FACIAL\_LANDMARKS\_IDXS['left\_eye']

(rStart, rEnd) = face\_utils.FACIAL\_LANDMARKS\_IDXS['right\_eye']

#Start webcam video capture

video\_capture = cv2.VideoCapture(0)

#Give some time for camera to initialize(not required)

time.sleep(2)

while(True):

#Read each frame and flip it, and convert to grayscale

ret, frame = video\_capture.read()

frame = cv2.flip(frame,1)

gray = cv2.cvtColor(frame, cv2.COLOR\_BGR2GRAY)

#Detect facial points through detector function

faces = detector(gray, 0)

#Detect faces through haarcascade\_frontalface\_default.xml

face\_rectangle = face\_cascade.detectMultiScale(gray, 1.3, 5)

#Draw rectangle around each face detected

for (x,y,w,h) in face\_rectangle:

cv2.rectangle(frame,(x,y),(x+w,y+h),(255,0,0),2)

#Detect facial points

for face in faces:

shape = predictor(gray, face)

shape = face\_utils.shape\_to\_np(shape)

#Get array of coordinates of leftEye and rightEye

leftEye = shape[lStart:lEnd]

rightEye = shape[rStart:rEnd]

#Calculate aspect ratio of both eyes

leftEyeAspectRatio = eye\_aspect\_ratio(leftEye)

rightEyeAspectRatio = eye\_aspect\_ratio(rightEye)

eyeAspectRatio = (leftEyeAspectRatio + rightEyeAspectRatio) / 2

#Use hull to remove convex contour discrepencies and draw eye shape around eyes

leftEyeHull = cv2.convexHull(leftEye)

rightEyeHull = cv2.convexHull(rightEye)

cv2.drawContours(frame, [leftEyeHull], -1, (0, 255, 0), 1)

cv2.drawContours(frame, [rightEyeHull], -1, (0, 255, 0), 1)

#Detect if eye aspect ratio is less than threshold

if(eyeAspectRatio < EYE\_ASPECT\_RATIO\_THRESHOLD):

COUNTER += 1

#If no. of frames is greater than threshold frames,

if COUNTER >= EYE\_ASPECT\_RATIO\_CONSEC\_FRAMES:

pygame.mixer.music.play(-1)

cv2.putText(frame, "You are Drowsy", (150,200), cv2.FONT\_HERSHEY\_SIMPLEX, 1.5, (0,0,255), 2)

else:

pygame.mixer.music.stop()

COUNTER = 0

#Show video feed

cv2.imshow('Video', frame)

if(cv2.waitKey(1) & 0xFF == ord('q')):

break

#Finally when video capture is over, release the video capture and destroyAllWindows

video\_capture.release()

cv2.destroyAllWindows()

**CHAPTER IV**

**RESULTS AND DISCUSSION**

The whole model has been facilitated and the eyes state has been detected with glasses and without glasses to ensure the model's efficacy. The model got an accuracy of 92% over 84,000 instances, giving a liable and effective stratagem for inhibiting accidents due to drowsiness. The real-time demonstration has shown in fig.7

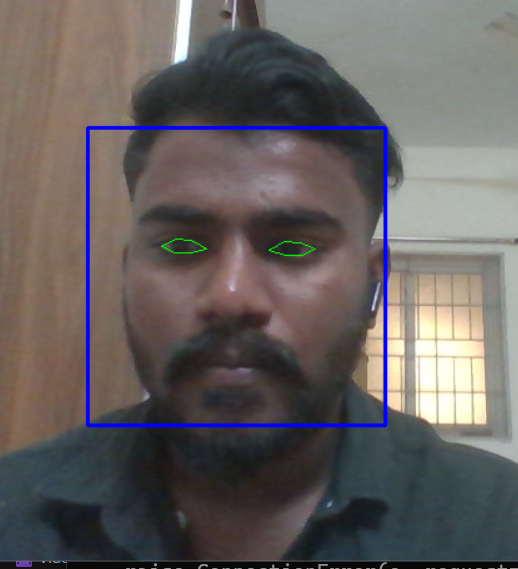


Fig.7: drowsy eye detection

The Buzzer beams when the score has been increased over 15, the eye closure movement has been detected

After training the model, The model will be verified throough test data upto certain number of epoch as follows

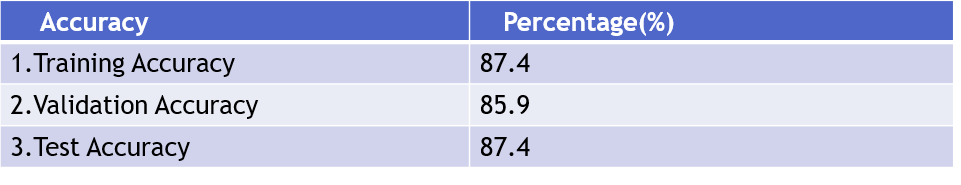


Table 1 : Various accuracy of model against test data

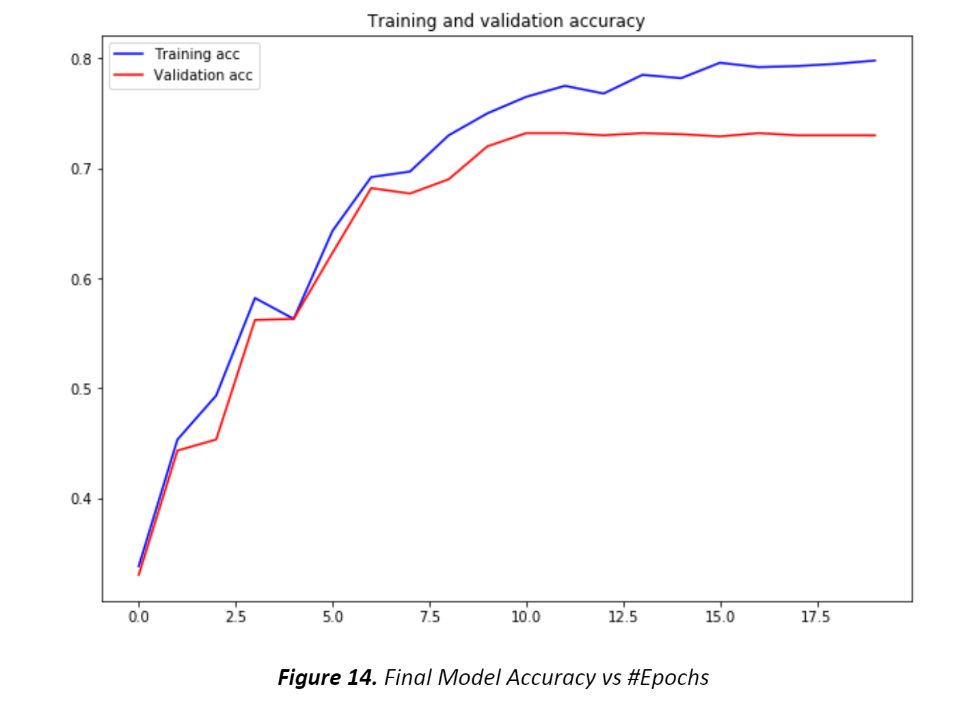


Fig 8: Training, Validation accuracy vs number of epoch

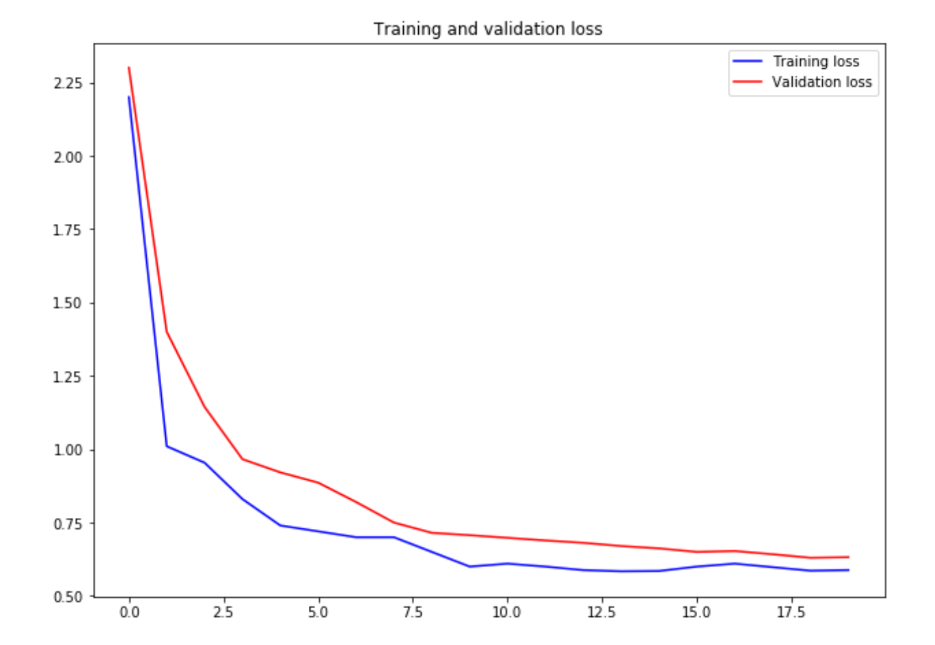


Fig 9: Training, Validation loss vs number of epoch

**CHAPTER V**

**REALISTIC CONSTRAINTS**

The datasets which we utilized were exceptionally enormous in size so that while preparing it will require some investment as a result of less GPU and processor of the PC.

For the Driver’s Drowsiness Detection, the camera is to be placed in such a way that the image is not affected by the light exposure, and the head is in the right position, and provided with uninterrupted power supply.

To pick correct algorithm for our undertaking where it's going to supply a higher overall performance, we tried many distinct algorithms which took a number of time and stress whether we could complete our project in time or not.

**CHAPTER VI**

**CONCLUSION**

The model proposed offers a viable solution for Driver’s Drowsiness Detection and thereby

prohibiting accidents. The other methods are effective however whilst applied in real-time can affect output for large datasets which includes HRV, EOG, PPG, and so forth. Thus, a light-weight embedded machine may be idea of as a genuine pathway for successful integration of this technique i.e., Driver’s Drowsiness

Detection with an alarm over motors. Increased rate of fatality can be avoided and a veritable system can be accessible by means of refuting pointless approaches. Different Deep Learning models can be used and the variety of Classes can be expanded to make the system greater correct and specific the use of Transfer Learning.

**CHAPTER VII**

**REFERENCES**

* [1] 1-Hartman, K. and J. Strasser, Saving Lives Through Advanced Vehicle Safety Technology: Intelligent Vehicle Initiative Final Report. 2005, Department of Transportation: Washington, DC.
* [2] A. Liu, Z. Li, L. Wang and Y. Zhao, "A practical driver fatigue detection algorithm based on eye state," *2010 Asia Pacific Conference on Postgraduate Research in Microelectronics and Electronics (Prime Asia)*, Shanghai, 2010, pp. 235-238.
* [3] Zhong, G., Ying, R., Wang, H., Siddiqui, A., & Choudhary, G., Drowsiness Detection with Machine Learning. [Online]. Available: <https://towardsdatascience.com/drowsiness-detection-with-machine-learning-765a16ca208a> (Date of Access 02 / 06 /2020)
* [4] Computer Vision Lab, National Tsuing Hua University, Driver Drowsiness Detection Dataset. [Online]. Available: <http://cv.cs.nthu.edu.tw/php/callforpaper/datasets/DDD/> (Date of Access 20 / 04 /2020)
* [5] V. Kazumi and J. Sullivan, “One millisecond face alignment with an ensemble of regression trees”, *2014 IEEE Conference on Computer Vision and Pattern Recognition* 1867-1874, 2014.