International Journal of All Research Education and Scientific Methods (IJARESM), ISSN: 2455-6211
Volume 10, Issue 6, June-2022, Impact Factor: 7.429, Available online at: www.ijaresm.com

Driver Drowsiness detection using AI

Aaftab Mohammad[1], Yagnik Pavagadhi[2], Shubhi Gupta[3], Abdul Mannan Khan[4], Harika Sirandass[5], Preksha S Patil[6]

---*******************---

ABSTRACT

Drowsiness and fatigue are one of the main causes leading to road accidents. They can be prevented by taking effort to get enough sleep before driving, drink coffee or energy drink, or have a rest when the signs of drowsiness occur. The popular drowsiness detection method uses complex methods, such as EEG and ECG. This method has high accuracy for its measurement but it need to use contact measurement and it has many limitations on driver fatigue and drowsiness monitor. Thus, it is not comfortable to be used in real time driving. This paper proposes a way to detect the drowsiness signs among drivers by measuring the eye closing rate and yawning. This project describes on how to detect the eyes and mouth in a video recorded from the experiment conducted by AIROS (American Institute of Road Safety). In the video, a participant will drive the driving simulation system and a webcam will be place in front of the driving simulator. The video will be recorded using the webcam to see the transition from awake to fatigue and finally, drowsy. The designed system deals with detecting the face area of the image captured from the video. The purpose of using the face area so it can narrow down to detect eyes and mouth within the face area. Once the face is found, the eyes and mouth are found by creating the eye for left and right eye detection and also mouth detection. A computer vision based thoughts have been used for the creation of a Drowsy Driver Detection System. The little camera has been utilized by framework that concentrates straight towards the essence of driver and checks the driver's eyes with a particular ultimate objective to perceive weakness. A notice sign is issued to alert the driver, in such circumstance when exhaustion is perceived. The framework oversees using information picked up for the picture to find the facial tourist spots, which gets the area where the eyes of an individual may exist. On the off chance that the eyes of driver are discovered close for a specific measure of casings, the proposed framework accept that the driver is falling asleep and an alarm of caution has been issued. The structure can work just when the eye are found, and works in encompassing lighting conditions too.

INTRODUCTION

1.1. BACKGROUND

Driver drowsiness or exhaustion has been a noteworthy factor in many countless road accidents and mishaps. According to the research around 1200 deaths and more than 76000 injuries were credited to drowsiness of driver's related crashes every year.

Driver drowsiness and fatigue is a major great factor which results into numerous road accidents. Developing and maintaining technologies which can effectively and efficiently detect or prevent drowsiness at the wheel and alert the driver before an accident is a major challenge in the field of accident prevention systems. Because of the dangerous that drowsiness can cause on the roads some methods need to be developed for preventing counteracting its effects.

1.2. PROBLEM STATEMENT

Driver's drowsiness is one of main and great causes of road accidents around the universe. Eventhough there are many road safety products designed to prevent road accidents due to driver's drowsiness and fatigue, most of them are very expensive in terms of cost, or producing false outputs like alerting drivers late.

Thus designing a prototype Drowsiness detection system for drivers which will focus on continuously and accurately monitoring the state of the driver's eyes in the real time to checkwhether they are open or closed for more than given period of time.

1.3. JUSTIFICATION

Driver drowsiness has become major reasons for the lives of many people. Thus main objective of this research is to come up with noninvasive, cost effective and efficient drowsiness detection system that can easily be implemented in driver monitoring system of any actual vehicle. Unlikemany other previous research works and commercially available behavioral measuring methods that focuses on eye closure only, this research will include other facial motions and behavioral changes that might be giving more reliability to the system.

Also this research will explore different ways of integration and input processing techniques and optimize the different system parameter to maximize the accuracy and the speed of detection.

By accepting the stream of images of the driver's facial movement and the car controller data including the steering wheel activity, the system can determine the status of the drivers as drowsyor non-drowsy.

1.4. PROPOSED SOLUTION

The main solution and objective of this research is to handle road accidents that results from driver's drowsiness. The system will collect stream of images from drivers face. This images willbe processed and categorized by the system in two categories after analyzing the region of interest i.e. left and right eye; if drivers eyes are closed for a given period of time the system willassume the driver is drowsy and thus the alarm will be ringing to alert the driver, otherwise if thedrivers eyes are open the driver is non-drowsy.

1.5. OBJECTIVES

1.5.1. GENERAL OBJECTIVES:

To design a system that detects drowsiness of the drivers and takes necessary steps to avoid occurrence of any accidents. The driver drowsiness detection system, being implemented in thisproject aims at being easily available and can be used with different types of vehicles.

1.5.2. SPECIFIC OBJECTIVES:

Driver drowsiness detection is a car safety technology which spares the life of the driver and passengers by avoiding accidents when the driver is getting fatigued.

1. The primary goal is to initially plan a framework to distinguish driver's sluggishness bypersistently checking their eyes.

2. The framework works disregarding driver wearing displays and in different lightningconditions.

3. To caution the driver on the identification of laziness by utilizing ringer or alarms to alert.

4. Speed of the vehicle can be reduced.

5. Traffic management can be maintained by reducing the accidents.

LITERATURE REVIEW

There are many previous researches regarding driver drowsiness detection system that can be used as a reference to develop a real-time system on detecting drowsiness for drivers. There is also several method which use different approaches to detect the drowsiness signs. According toAIROS (Americans Institute of Road Safety), from the year of 2007 until 2010, they were 439 cases of road accidents have been investigated by the AIROS crash team.

2.1. Drowsiness and Fatigue

Researchers stated that drowsiness is where a person is in the middle of awake and sleepy state.This situation leads the driver to not giving full attention to their driving. Therefore, the vehiclecan no longer be controlled due to the driver being in a semi-conscious state. According to Gianluca Borghini researchers' mental fatigue is a factor of drowsiness and it caused the person who experiences to not be able to perform because it decreases the efficiency of the brain to respond towards sudden events.

2.2. Electroencephalography (EEG) for Drowsiness Detection

Electroencephalography (EEG) is a method that measures the brain electrical activity. It can be used to measure the heartbeat, eye blink and even major physical movement such as head movement. It can be used on human or animal as subjects to get the brain activity. It uses a special hardware that place sensors around the top of the head area to sense any electrical brain activity. Examples of EEG Data Collecting Authors in mentioned that from the method that has been implemented by the previous researcher to detect drowsiness signs, the EEG method is bestto be applied for drowsiness and fatigue detection. In the method, EEG have four types of frequency components that can be analyzed, i.e. alpha (α), beta (β), theta (θ) and delta (δ). Whenthe power is increased in alpha (α) and delta (δ) frequency bands, it shows that the driver is facing fatigue and drowsiness.

The disadvantages of this method are, it is very sensitive to noise around the sensors. For example, when the person is doing the EEG experiment, the surrounding area must be completely silent. The noise will interfere with the sensors that detect the brain activity. Another disadvantage of this method is that even if the result might be accurate, it is not suitable to use for real driving application. Imagine when a person is driving and he is wearing something on his head with full of wires and when the driver moves their head, the wire may strip off from their place. Even though it is not convenient to be used for real-time driving but for experiment purposes and data collection, it is one of the best methods so far.

2.3. Drowsiness detection using face detection system

Drowsiness can be detected by using face area detection. The methods to detect drowsiness within face area are vary due to drowsiness sign are more visible and clear to be detected at face area. From the face area, we can detect the eyes location. From eyes detection, author in stated that there are four types of eyelid movement that can be used for drowsiness detection. They are complete open, complete close, and in the middle where the eyes are from open to close and viceversa.

The algorithm processes the images captured in grey-scale method; where the color from the images is then transformed into black and white. Working with black and white images is easierbecause only two parameters have to be measured. The author then performs the edge detectionto detect the edges of eyes so that the value of eyelid area can be calculated.

The problem occurring with this method is that the size area of eye might vary from one person to another. Someone may have small eyes and looks like it is sleepy but some are not. Other thanthat, if the person is wearing glasses, there is obstacle to detect eye region. The images that beingcaptured must be in certain range from the camera because when the distance is far from the camera, the images are blurred.

MATERIALS AND METHODS

3.1. METHODOLOGY

This paper will explain about the method that has been taken in order to reach the objectives ofthe project and a closer look on how the project is implemented. It is the analysis of each stage that will be faced in order to complete this project. Each selection and achievement of the method taken that has been implement in this project will be explained for each stage until the project is success. This project involves software usage which is MATLAB Computer Vision System. The methods used are existing method in MATLAB command to detect face, eyes, and mouth area.

3.1.1. Research Methodology

Usually, research methodology refers to a set of procedures that will be used to carry out a certain research. In order to complete this project systematically within the specified time, thereare some methodologies and activities that need to be planned and followed consistently.

3.1.2. Background of Study

Before starting any research or project, basic information of the related topic is required to ensure that the author understands what the project is all about. In this stage, the background of study helps the author understands the relation between drowsiness and fatigue. It also helps the author in understanding the seriousness of driving a motored vehicle in drowsiness condition. It is proven that driving the vehicle in fatigue and drowsiness condition is a lead factor to road accidents.

3.1.3. Literature Review

In this stage, it involved the study of the previous research done related to author project. This topic observes the relationship between drowsiness condition and handling a motored vehicle. Athorough observation was done on the existing method to detect the drowsiness. Different parameters have been used by previous researches. By focusing on the parameters which is detecting eyes and mouth, helps to narrow down the perspective of this project.

3.1.4. Previous Data Gathering and Analysis

In this stage, it was found that one of the best way to detect eyes and yawning is by algorithm. Some of current algorithms that are related to this project are reviewed to help developing the project. In the propose method measures the time for a person closed its eyes and if its eyes are closed longer than the normal eye blink time, it is possible that the person is falling asleep. Based on researches of human eye blinks, it has been identified that the average of human blink duration takes about 202.24ms while the blink duration of a drowsy individual takes about258.57ms.

After defining the method to be used in this project, authors obtain the video of the experiment conducted by MIROS where participants are driving in a simulated environment and being recorded for the whole session. The experiment takes time, about 60 to 90 minutes. The analysisof drowsiness detection is done manually by watching the full length of the video and mark down the drowsiness signs. The parameters of the data are: drowsiness, yawing and other signs occurs at starting and ending time. This is to calculate the duration of the signs occurred.

3.2. Proposed work

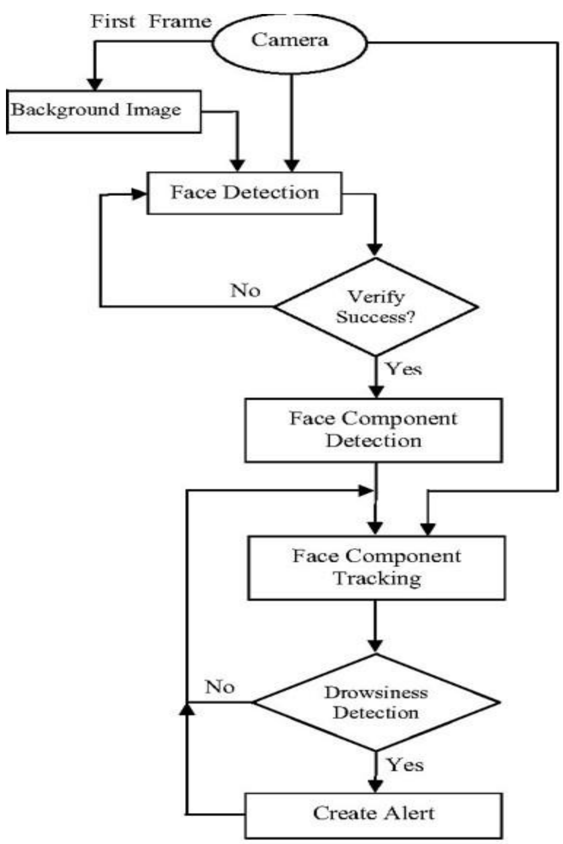


Fig 3. Flow Chart for Drivers Drowsiness Detection System

The flowchart of the proposed system has been shown in the above figure. The camera captures the image and sends to the processor of the laptop which consists of 32 bit memory card installedwith Open CV which helps in image processing.

If the signal crosses threshold of a set of continuous frames with EAR less than threshold value,it will automatically makes the alarm beep and the speed of the vehicle gets reduced. Otherwise that signal is rejected and next signal is processed.

3.3. WORKING:

Drivers face is monitored throughout using a video or web camera. In order to detect the drowsiness the first step is to detect the face using the set of frames taken by the camera. Then the location of the eyes is detected and retina of the eye is continuously monitored. The capturedimage is sent to the processor for image processing. It converts the received image to digital signal using Open CV.

The digital signal is transmitted from transmitter to the receiver. Both the transmitter and the receiver are paired up. The signal is then passed to the LPC2148, the microcontroller. If the signal crosses the threshold value of EAR for a given number of frames, then the alarm beepsand the speed of the vehicle is automatically reduced.

3.4. Requirements and specification

3.4.1. Software Requirements Specification

Python: Python **Libraries:**Numpy Spicy PlaysoundDlib Imutils OpenCv Etc.

Operating System

Windows or Ubuntu

3.4.2. Hardware Requirements Specification

1. Laptop with basic hardware.
2. Webcam.
3. Mouse.
4. Keyboard.

3.5. Requirement Analysis

Python:

Python is the basis of the program that we wrote. It utilizes many of the python libraries.

Libraries:

1. Numpy: Pre-requisite for Dlib
2. Scipy: Used for calculating Euclidean distance between the eyelids.
3. Playsound: Used for sounding the alarm
4. Dlib: This program is used to find the frontal human face and estimate its pose using 68 facelandmarks.
5. Imutils: Convenient functions written for Opencv.
6. Opencv: Used to get the video stream from the webcam, etc.

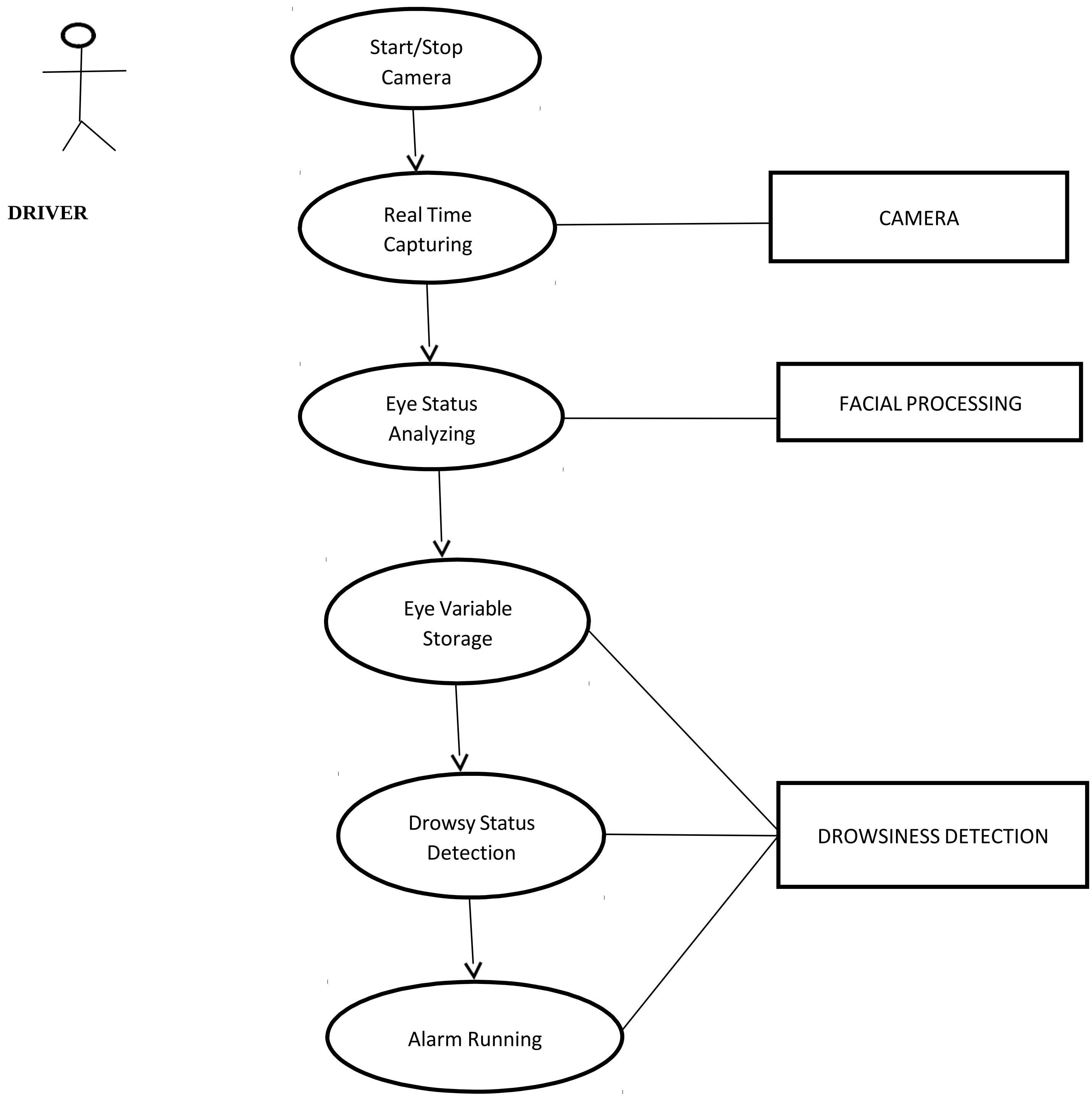
International Journal of All Research Education and Scientific Methods (IJARESM), ISSN: 2455-6211
Volume 10, Issue 6, June-2022, Impact Factor: 7.429, Available online at: www.ijaresm.com

7. OS: Program is tested on Windows 10 build 1903 and PopOS 19.04Laptop: Used to run our code.

Webcam: Used to get the video feed

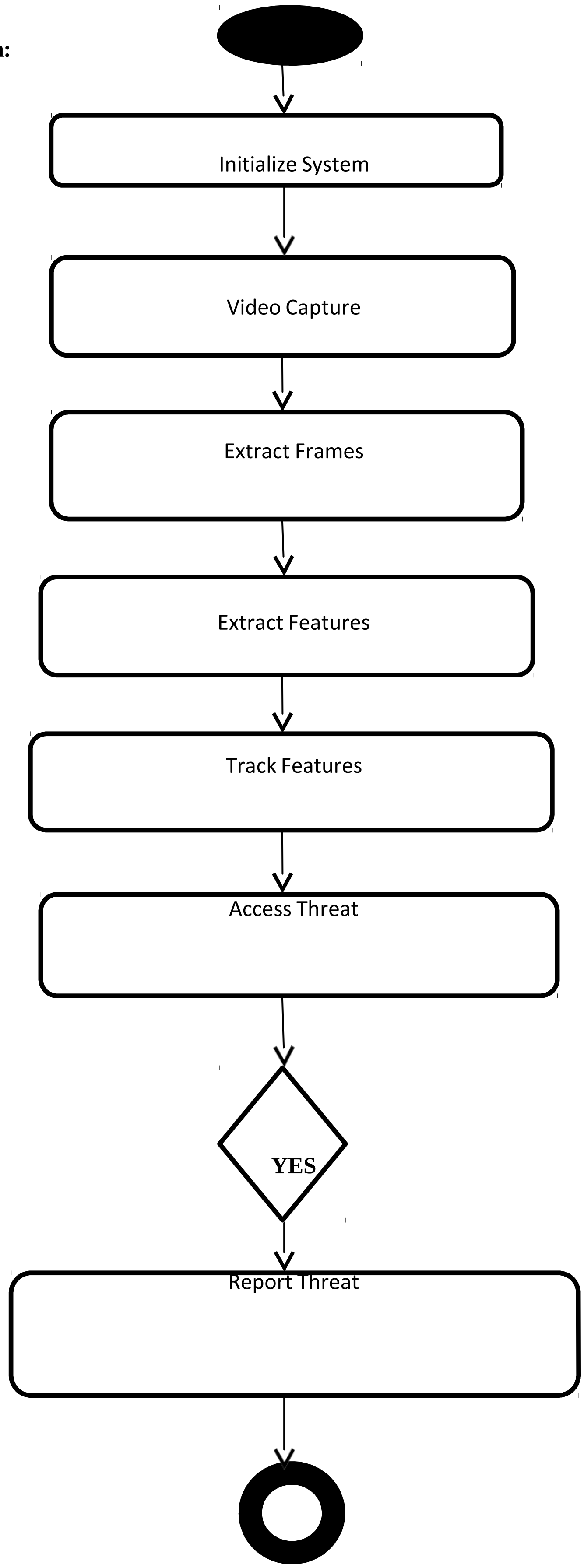
3.6 Conceptual Framework/Experimental/System Design

1. User Case Diagram:

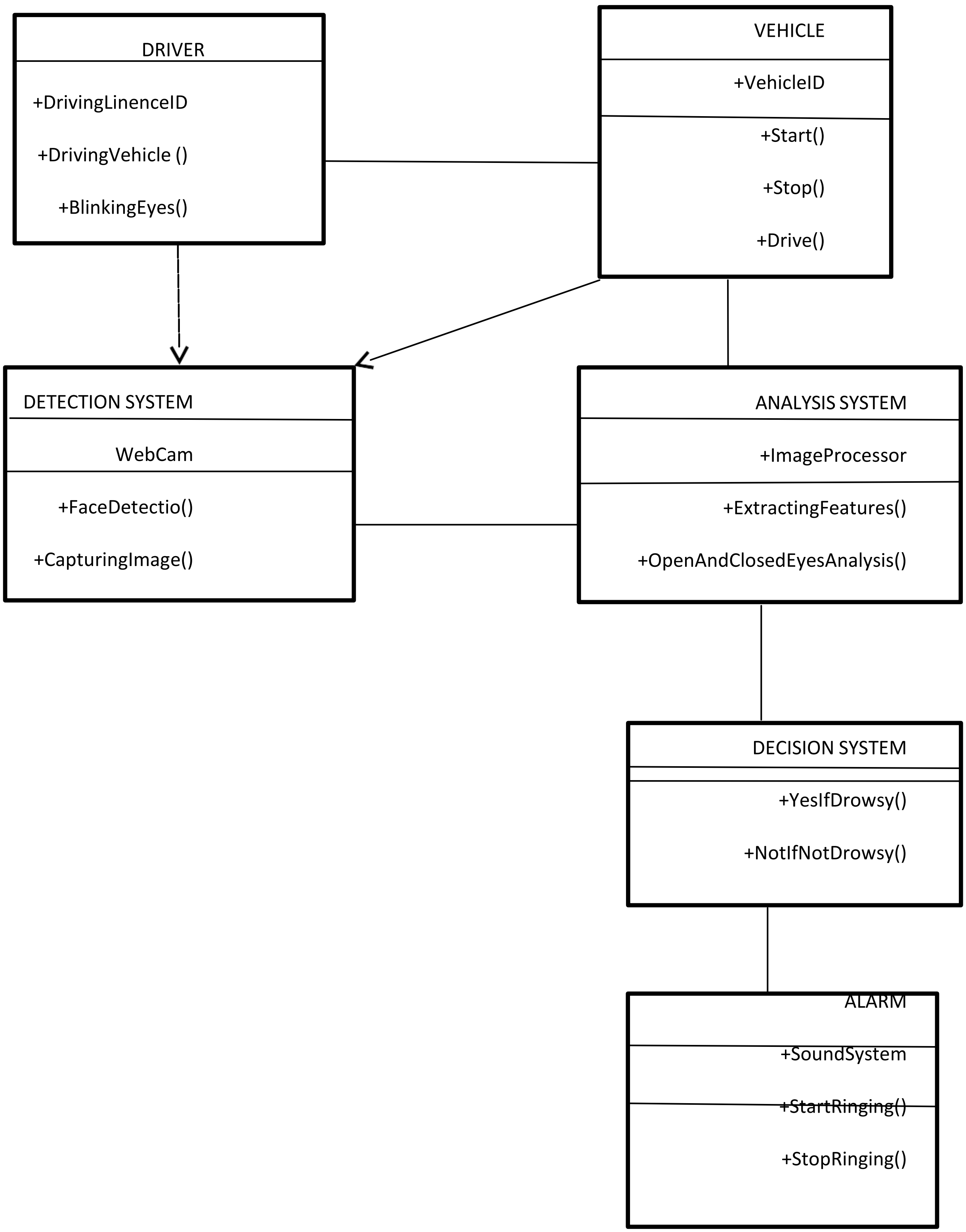


International Journal of All Research Education and Scientific Methods (IJARESM), ISSN: 2455-6211
Volume 10, Issue 6, June-2022, Impact Factor: 7.429, Available online at: www.ijaresm.com

2. Activity Diagram:



3. Class Diagram:



3.7. Computational Analysis:

Matching

Dashboard mounted camera is used to monitor the eyes of the driver in real time to detectdrowsiness.

Drowsiness Detection Design:

A camera is setup that looks for faces in the input video stream and monitors frames of faces. In the event that a face is identified, facial milestone identification is connected and the eye districtis removed from the edges of the video stream.

$$\text{EAR} = \frac{\|p_2 - p_6\| + \|p_3 - p_5\|}{2\|p_1 - p_4\|}$$

Fig. 4 Eye Aspect Calculation Formula

EAR algorithm:

Step 1: Use the Detected Eye region from the algorithm.

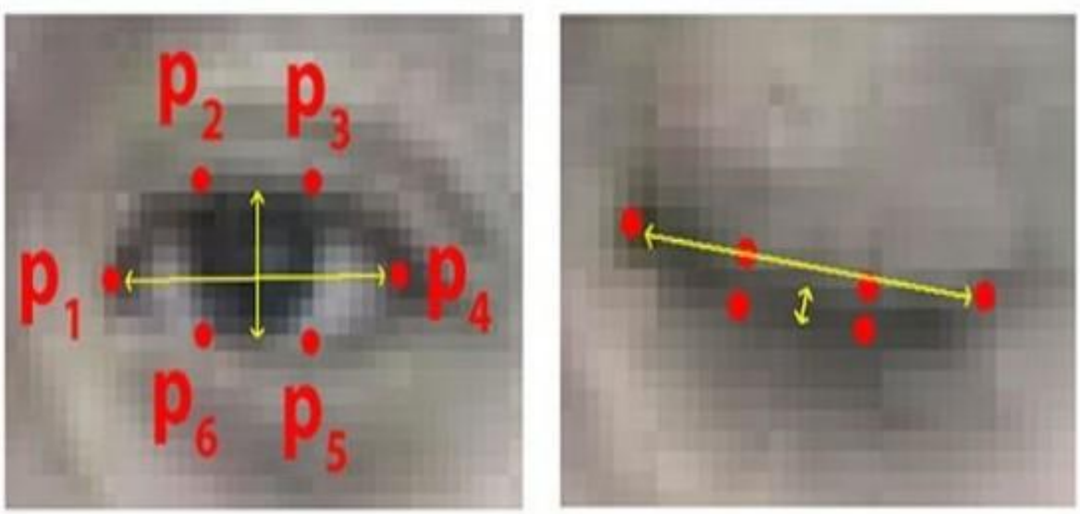
Step 2: Compute the Eye Aspect Ratio to determine if the eyes are closed.

Step 3: If EAR satisfies the drowsy condition then move to step 5.

Step 4: If EAR is normal then go to Step 1.

Step 5: Sound Alarm.

If the aspect ratio of the eye indicates that they have been closed for greater than a dedicatedfixed time, we will sound an alarm system so that the driver wakes up.



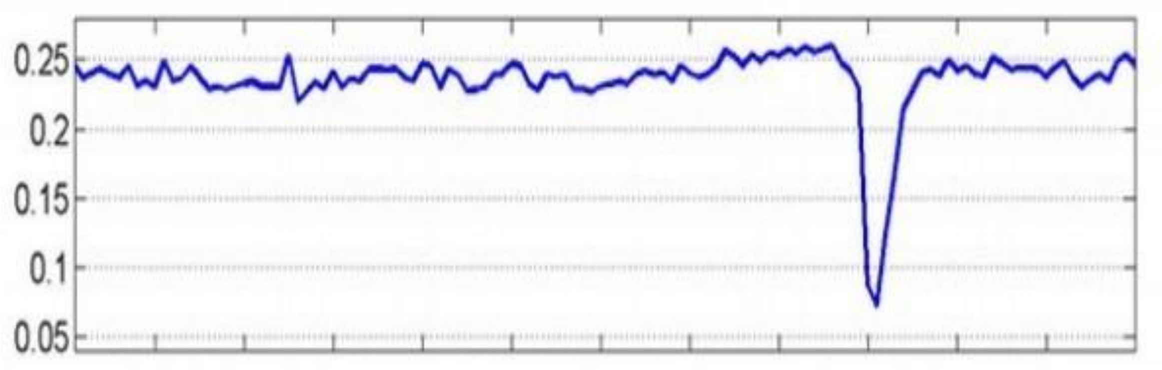


Fig. 5 Eye Aspect Ratio

We observe the aspect ratio of the eye remains constant for a period of time indicating that theeye was open, then it falls rapidly to zero and then increases again which indicates the person blinked.

We will be observing this eye aspect ratio in our drowsiness detector case to see if the value remains constant or falls to zero but not increases again implying that the driver has closed hiseyes for extended period.

Developing Image processing solutions using OpenCV & dlib

OpenCV was developed keeping image processing in mind. Every function and data struct of OpenCV concerns itself with an Image Processing library. Comparatively, Matlab is hugely ofgeneric use & slow.

Any usefulness can be accomplished by methods for tool kits in OpenCV, it might be moneyrelated tool compartments or explicit DNA tool stash

Also the dlib library comes with a oriented gradients based face detector histogram a facial landmark predictor comes bundled in the library.

Facial landmarks generated by dlib is an indexable list as described in below image

3.9.1 Algorithmic steps:-

The overall algorithm is pretty straightforward one. First we have used a camera which is setupat desirable position in a car that looks for faces stream.

If face gets detected, the facial landmark detection task is applied and region of eyes isextracted.

Once we get the eye region, we calculate the Eye Aspect Ratio to find out if the eye-lids aredown for a substantial amount of time.

On the off chance that the Eye Aspect Ratio demonstrates that the eyes are shut for a considerably long measure of time, the alert will sound noisy to wake the driver up. For the functionalities of the system and to make it work efficiently we have used OpenCv, dlib andPython.

The implementation of the drowsiness detector system includes machine learning algorithms which are in turn included in OpenCv ML algorithms. There are numerous ML algorithms but for our purpose we required only the **face detector algorithm.**

It is fundamentally an item discovery ground-breaking application. Additionally, prepared frontal face identifier is accessible with the OpenCv circulation. It works efficiently well overall.It can also be used to detect various different types of objects with the required software.

International Journal of All Research Education and Scientific Methods (IJARESM), ISSN: 2455-6211
Volume 10, Issue 6, June-2022, Impact Factor: 7.429, Available online at: www.ijaresm.com

Starting to build the detector system with OpenCv:

First, we create a new file drowsy_detect.py and write the following script in it: -

```python
# importing the necessary packages for the project
# this is project
import numpy as np
import pandas as pd
import playsound
import dlib
import cv2
import argparse
import imutils
import time
import serial
from imutils import face_utils
from scipy.spatial import distance as dist
from imutils.video import VideoStream
from threading import Thread
```

Now for calculating the eye-aspect ratio we need to compute the Euclidean distance between the facial landmarks points which in turn needs SciPy package in python. (It not a strict requirement but SciPy is needed if work related to computer vision or image processing is intended).

Also the package named imutils is needed for image processing and computer vision functions to assist the working with OpenCv.

The thread class is imported so that we can beep the alarm in a different thread from main threadso that it is ensured that our script doesn't stop/pause executing while the alarm beeps. In order to play a file of the wav or mp3 format, we need playsound library.

Facial landmarks and eye aspect ratio calculation:

For detecting and localizing facial landmarks we will require the dlib library hence we import it.

Eye_aspect_ratio function is defined to calculate the distance between the eye landmarks taken vertically and distances between the eye landmarks taken horizontally.

```python
Arduino = serial.Serial('/dev/ttyACM0', 1)
# time to connect arduino
time.sleep(1)


# function to calculate the EAR
def calculate_EAR(eye):
    # calculating the euclidian distance between vertical eye landmark
    A = dist.euclidean(eye[1], eye[5])
    B = dist.euclidean(eye[2], eye[4])

    # calculating the euclidean distance between horizontal eye landmark
    C = dist.euclidean(eye[0], eye[3])

    # formula for calculating EAR
    EAR = (A + B) / (2.0 * C)

    return EAR
```

International Journal of All Research Education and Scientific Methods (IJARESM), ISSN: 2455-6211
Volume 10, Issue 6, June-2022, Impact Factor: 7.429, Available online at: www.ijaresm.com

So, when the eye is open, the value returned for the eye aspect ratio will be a constant approximately. Then the value will rapidly decrease reaching zero in case of an eye blink.

When the eye is closed, eye aspect ratio again approaches to an approximate constant valuewhich is very smaller compared to that when the eye is open.

Therefore, the dip in the aspect ratio indicates blink of the eyes.

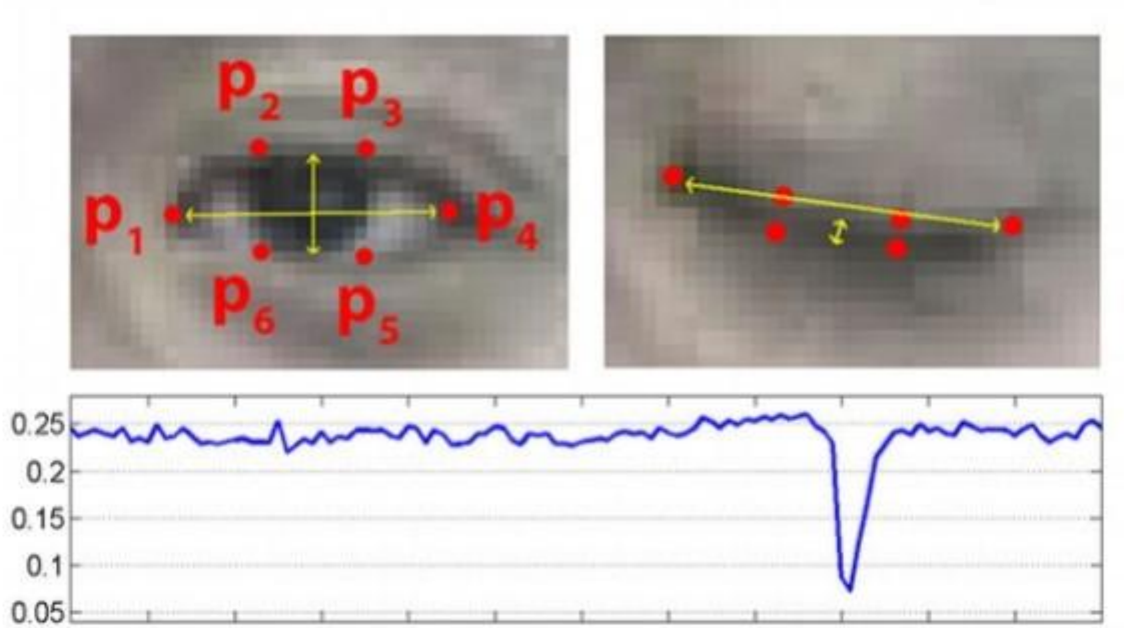


Fig. 7 Plot of Eye Aspect Ratio over Time

Important variables in the script:

EYE_AR_THRESH: it's a threshold value for the eye aspect ratio. If ratio becomes lower thanthis value, counter starts for the number of frames the eyes remain closed.

```
# arguments to be given at the time of running the code
ap = argparse.ArgumentParser()
ap.add_argument("-p", "--shape-predictor", required=True,
                help="path to facial landmark predictor")
ap.add_argument("-a", "--alarm", type=str, default="",
                help="path alarm .WAV file")
ap.add_argument("-w", "--webcam", type=int, default=0,
                help="index of webcam on system")
args = vars(ap.parse_args())

# these are two constants to describe about the threshold EAR and threshold value total number
EAR_threshold = 0.25
EAR_consec_frames = 50

# initializing counter for consecutive frames and boolean for alarm
COUNTER = 0
ALARM_ON = False
```

EYE_AR_CONSEC_FRAMES: If the value for the number of frames for which the eyes remainclosed exceeds this variable's value, the alarm is activated.

Dlib library for face detection:

The dlib library serves us with a facial landmark detector as well as facial landmark predictor.Below is the facial landmarks that are produced by the library.

Now from these landmarks, we just churn out the eye regions successfully.

```
# initialize dlib's face detector (HOG-based) and then create the facial landmark predictor
print("[INFO] loading facial landmark predictor...")
detector = dlib.get_frontal_face_detector()
predictor = dlib.shape_predictor(args["shape_predictor"])

# grab the indexes of the facial landmarks for the left and
# right eye, respectively
(left_Start, left_End) = face_utils.FACIAL_LANDMARKS_IDXS["left_eye"]
(right_Start, right_End) = face_utils.FACIAL_LANDMARKS_IDXS["right_eye"]

```

Test Cases to check the drowsiness:

Following is the table representing four test cases that are to encountered while doing this projectthat concerns with the drowsiness of the driver.

Test Cases	**Eyes Detected**	**Eyes Closure**	**Result**
Case 1	NO	NO	No Result
Case 2	NO	NO	No Result
Case 3	YES	NO	No Alarm
Case 4	YES	YES	Alarm Beep

Table 1 Test Cases

At the point when the eyes are shut for more than certain measure of edges then we find that thedriver is feeling tired. Henceforth these cases are distinguished is and a caution sounded.

3.8. Experimental and data collection procedure:

3.8.1. Data Collection procedure:

Capture the image of the driver from the camera.

Send the captured image to haarcascade file for face detection.

If the face is detected then crop the image consisting of the face only. If the driver is distractedthen a face might not be detected, so play the buzzer.

Send the face image to haarcascade file for eye detection.

If the eyes are detected then crop only the eyes and extract the left and right eye from that image. If both eyes are not found, then the driver is looking sideways, so sound the buzzer.

The cropped eye images are sent to the Hough transformations for detecting pupils, which will determine whether they are open or closed.

If they are found to be closed for five continuous frames, then the driver should be alerted by playing the buzzer.

Cut the 3/4th portion of the face image and send it to haarcascade file for mouth detection.

If the mouth is not detected, it indicates a yawn and so, the driver needs to be alerted by playing the buzzer.

Repeat the above procedure continuously in a rapid loop as long as the car is moving.

3.8.2. Experimental Procedure:

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 classifier CNN 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, ifvalue 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 decreasethe score.

We are drawing the result on the screen using cv2.putText() function which will display real timestatus 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 eyesare closed for a long period of time. This is when we beep the alarm using **sound.play()**

3.8.1 Experimental Results

The code has been implemented and the following results were recorded in a suitable environment for many people and fatigue detection was experimentally tested. According to the stepwise algorithm, the images of the driver have been captured and processed further.

Image has been captured by camera and saved to memory.

Face Detection is performed and it is checked whether face is detected or not. This is done by passing the image through Haar Cascade. If face is not detected, it means the driver is not looking in the right direction and hence it is considered as a condition of fatigue and needs to bealarmed about.

As the next step, if the face has been detected in the earlier step, then the image is processed to check for eye detection using Haar classfier. Thus, the pair of eyes is detected for being the inputfor the next step.

Once the face is detected, the mouth classifier is used to detect the presence of the mouth. If the mouth is detected as closed, that means the driver is not yawning. If for five consecutive frames,the face is not detected, it means that the yawning or other distraction of the driver is enough to create a potential danger and hence buzzer is set off, pointing fatigue detection.

Once the eyes are detected, the portion of the face containing the eyes is cropped in order thatonly this much of the image is further processed using a Hough transform to find whether thedetected eyes are open or close. For this purpose, each eye is tested independently.

Efficiency of the system can reach 90 percent in standard light conditions. However, the systemmay give erroneous results when the illumination level goes low. Thus, when the system was tested on various people as drivers in different lighting conditions and out of 100 samples, accuracy was higher in better lighting and when the person was without spectacles and other visible obstacles which came in line with the detection of face, eyes and mouth.

RESULTS

4.1. Summary:

To get the outcome a large no of pictures were taken and their accuracy in deciding eye flickersand drowsiness was tried.

For this venture we utilized a 5 megapixel webcam associated with the PC. The webcam had inbuilt white LEDs connected to it to show it is working. In real time scenario, infrared LEDs ought to be utilized rather than white LEDs with the goal that the framework is non-meddling. Inbuilt speaker is utilized to deliver sound output so as to awaken the driver when drowsiness isdetected.

The framework was tried for various individuals in various surrounding lighting conditions (daytime and evening time). At the point when the webcam backdrop illumination was turned ON and the face is kept at an ideal distance, at that point the framework can identify blinks anddrowsiness with over 95% accuracy.

This is a decent outcome and can be executed by real time systems as well. A sample output for various conditions in different pictures is given beneath. Three pictures were taken; one in whichjust the eyes were identified and the other in which they were not and another where drowsiness is detected.

Table and Analysis

Numerous examples with shifting exactnesses were assembled and consequently a table plottedfor them

I/P	Eyes Detection Accuracy	Drowsiness Accuracy
Sample 1	100%	87.5%
Sample 2	95%	100%
Sample 3	80%	62.5%
Sample 4	100%	87.5%
Sample 5	100%	100%
TOTAL	95%	87.5%

Table 2 Result Table

Every individual who volunteered for the test will be approached to squint multiple times and actlanguid multiple times amid the test procedure. The eye squinting exactness was determined by beneath referenced recipe

Eye Detection Accuracy = total number of times eyes detected / (total no. of eyes detected+ total no of times eyes not detected) **Drowsiness Detection Accuracy** = total no. of times alarm sounds / (total no. of times alarm sounds + total no of times alarm didn't sound)

Face or eyes sometimes might not be detected due to lack of ample ambient light. It will in general be seen from the above table that in case model 3 isn't mulled over, at that point the framework has an accuracy of about 100%. That said; the high proportion of disappointments intest 3 exhibits that the framework is slanted to botch and has certain obstacles. In test 3 we didn'tutilize ample backdrop lights for the webcam. The subsequent poor lighting conditions gave a very error prone result.

Limitations: -

1. Dependency on proper ambient light:-

With poor lighting conditions once in a while the framework is unfit to perceive the eyes. So itgives a wrong result which must be managed. Continuously circumstance infrared setting enlightenments should be used to repel from poor lighting conditions.

2. An optimum range is required: -

Exactly when the division among face and webcam isn't at perfect range then certain issues develop. Exactly when face is unreasonably close webcam (less than 25 cm), then the frameworkis unfit to perceive the face from the image. Right when face is a long way from the webcam (more than 80cm) by then the setting light is missing to edify the face fittingly. So eyes are not related to high precision which results in botch in recognizable proof of sluggishness. This issue isn't truly considered as progressively circumstance, the partition between driver's face and webcam is perfect so the issue never develops.

3. Orientation of face: -

At the point when the face is tilted to a specific degree it will in general be perceived, anywaypast this the framework can't identify the face. So when the face isn't recognized, eyes are alsonot distinguished.

4. Problem with multiple faces: -

In case more than one face is recognized by the webcam, at that point framework gives an incorrect result. This issue isn't huge as we have to recognize the tiredness of a solitary driver

5. Poor detection of a person's eyes with spectacles: -

At the point when the driver wears glasses the system may not detect eyes which is the most noteworthy disadvantage of these systems. This issue has not yet been settled and is a test forpractically all eye detection systems structured up until now.

5.1. RESULTS

After completing the algorithm modeling, the simulation of algorithm was performed by using the video from the experiment conduct by AIROS. Due to the video length cannot be run in MATLAB because the processor have not enough memory, the video were separated into 3 minutes each video. By doing that, each video contains not more than 400 images per video.

There a 10 participants who complete the driving experiment but as to show in the result, we only take a participant. Total video that have been separate is 31 videos.

In the image above, the drowsiness state occur from frame number 168 to 179, where the driver closed it eyes for 11 seconds. The result of the driver yawns just after the drowsiness state occurs.

This system is reliable because it detects the transition of the drowsiness state where the algorithms detects eacheyes and mouth changes each frame per second. Therefore, the Drowsiness Detection System by using Webcam has successfully done by using MATLAB Software. All the data and result from the simulation algorithm has been shown this paper.

CONCLUSIONS AND RECOMMENDATIONS

6.1. RECOMMENDATIONS

The main goal of this project is to develop a simulation system that can detect drowsiness using awebcam. The system must meet certain requirement which is detecting drowsiness throughout the video frames. Thus, the driver can avoid accident. Secondly, it has to detect only drowsiness signs so that the system will not misinterpret any random signs that it received from the driver.

Improvement on the algorithms to detect eyes and mouth need to be done for future implementation. Luminance changes have to be encounter to ensure the detection of the gradientof eyes is sufficient to improve the detection results. The quality of the video or images used in detecting drowsiness affects the result of the detection. Therefore, a good quality and high framerate of images (number of pixel) is one of the factors to get better detection. Better techniques can be used to compare which technique is more reliable in detecting drowsiness.

Implementing other method to detect drowsiness is also one of the improvements to the system so that it can ensure the system to be reliable to detect drowsiness. Besides that, a better internalspecification of laptop or device can be used to run the system in order to obtain a smooth execution of the algorithm and a reliable system.

Thus, by making this project successful, the numbers of road accident can be reduce when thisproject is implemented in the vehicle to detect the drowsiness of the driver.

6.2. CONCLUSIONS

Previously, the author focuses on developing the algorithm or command to detect drowsiness. The developments of the algorithm takes time due to the authors only have basic skill in using MATLAB. The author learns about the MATLAB commands by developing the algorithm onlywith the help from Computer Vision Toolbox System that is already built-in in

MATLAB software and also by trial and error of the shared file from the MathWorks. MathWorks is whereall the high skills of MATLAB users from all over the world share their works on algorithms.

Other than that, during the Paper , investigating the drowsiness signs and collecting the data from the video of the experiments have been the main job scope. It will be used as parameters to develop the simulation system in detecting drowsiness. Until now, one of this semester's project objectives has been achieved which is to study the video images of participants in the experiment of driving simulation conducted by MIROS. Several techniques to develop the simulation system have been discovered. There are also other objectives that this project needs to achieve.

Few techniques have been implemented in this project which has been found through the previous researches. Further adjustment of the algorithm and the techniques need to be done in order to meet the requirement of this project and to finish it within the given time frame.

Thus we would have successfully designed and developed partial implementation of the Driver Drowsiness Detector using Python and OpenCv along with the camera to detect the face. The system to be developed is to be tested and limitations are identified. The rest of the work will bedone according to what is planned already.

REFERENCES

[1] G. Borghini, L. Astolfi, G. Vecchiato, D. Mattia, and F. Babiloni, "Measuring neurophysiological signals in aircraft pilots and car drivers for the assessment of mental workload, fatigue and drowsiness," Neuroscience & Biobehavioral Reviews, 2012.

[2] B. T. Jap, S. Lal, P. Fischer, and E. Bekiaris, "Using EEG spectral components to assess algorithms for detecting fatigue," Expert Systems with Applications, vol. 36, pp. 2352- 2359,2009.

[3] D. Liu, P. Sun, Y. Xiao, and Y. Yin, "Drowsiness Detection Based on Eyelid Movement," in Education Technology and Computer Science (ETCS), 2010 Second International Workshop on,2010, pp. 49-52.

[4] H. Seifoory, D. Taherkhani, B. Arzhang, Z. Eftekhari, and H. Memari, "An Accurate Morphological Drowsy Detection," ed: IEEE, 2011.

[5] Ameratunga.S , Bailey.J, Connor.J, Civil.I, Dunn.R , Jackson.R , Norton.R, and Robinson.E, ―Driver sleepless and risk of serious injury to car occupants: Population control stud y.‖ British Medical Journal, vol. 324, 2002, pp. 1125–1129.

[6] Bronte.S, Bergasa.L, Delgado.B, Garcia.I, Hernandez.N and Sevillano.M, ― Vis ion- baseddrowsiness detector for a realistic driving simulator, ‖ in I EEE Intelligent Transportations Systems Conference (ITSC), 2010.

[7] Distant e .A, D'Orazio .T, Guaragnela .C and Leo. M , ― A visual approach for driverinattention detection, ‖ Pattern Recognition, vol. 40, no. 8, 2007, pp. 2341–2355.

[8] Bradski.G, Kaehler.A, -Learning OpenCV, O'Reilly, 2008