Fatigue detection system for Driver's Safety

By Shreerang Mhatre

Introduction:

Countless mishaps regarding road accidents are caused by fatigue, tiredness, etc. About 50% of the accidents are road accidents. Road accidents can be due to inadequate way of driving, and these could arise if the driver is an alcoholic or drowsy. The drowsiness and the alcoholic condition of the driver have become a major cause of road accidents. This also has led to major challenges in developing a system for the prevention of this issue. IOT-based innovation tends to be much more practical to work with since it runs on a real-time system and can transfer all the data or information without any human interaction. Fatigue is a safety problem that has not yet been deeply tackled by any country, mainly because of its nature. Drowsiness, in general, is very difficult to measure or observe unlike alcohol and drugs, which have clear key indicators and tests that are available easily. An IOT–based system is designed to avoid countless mishaps due to drowsy drivers' behavioral and psychological changes by focusing on drivers' eye moments and health issues like heart attack, dizziness, and other health issues.

Fatigue Symptoms:

The fatigue detection algorithm is a camera-based computer vision technology that assesses driver alertness by monitoring the set of symptoms shown in the table below.

Fatigue Symptom	**Its Measurable Indicator**
Sleeping	Shut eyes duration
Frequent eyelid closure	Closed eyes ratio
Excessive blinking	Blinking frequency
Head tilting	Head tilt ratio
Yawning	Yawning ratio

Arduino:

Arduino is an open-source platform used for building electronics projects. Arduino consists of both a physical programmable circuit board (often referred to as a microcontroller) and a piece of software, or IDE (Integrated Development Environment) that runs on your computer, used to write and upload computer code to the physical board.

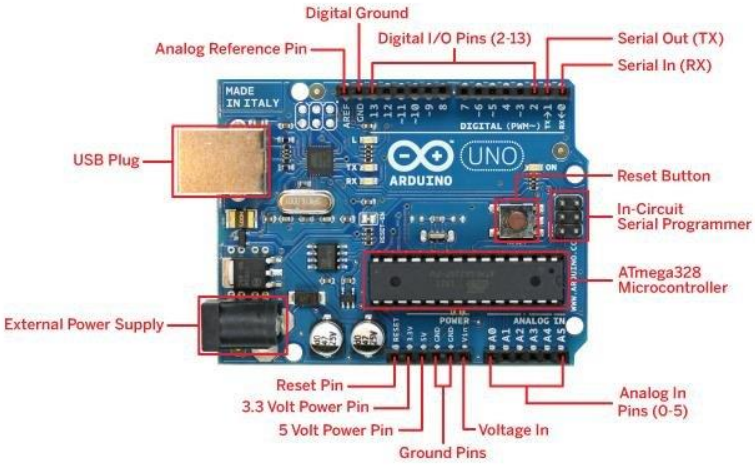
The Arduino platform has become quite popular with people just starting out with electronics, and for good reason. Unlike most previous programmable circuit boards, the Arduino does not need a separate piece of hardware (called a programmer) in order to load new code onto the board -- you can simply use a USB cable. Additionally, the Arduino IDE uses a simplified version of C++, making it easier to learn to program. Finally, Arduino provides a standard form factor that breaks out the functions of the micro-

controller into a more accessible package.

What Does it Do?

The Arduino hardware and software was designed for artists, designers, hobbyists, hackers, newbies, and anyone interested in creating interactive objects or environments. Arduino can interact with buttons, LEDs, motors, speakers, GPS units, cameras, the internet, and even your smart-phone or your TV! This flexibility combined with the fact that the Arduino software is free, the hardware boards are pretty cheap, and both the software and hardware are easy to learn has led to a large community of users who have contributed code and released instructions for a **huge** variety of Arduino-based projects.

Arduino UNO R3:



Arduino In Fatigue detection system for Driver's Safety:

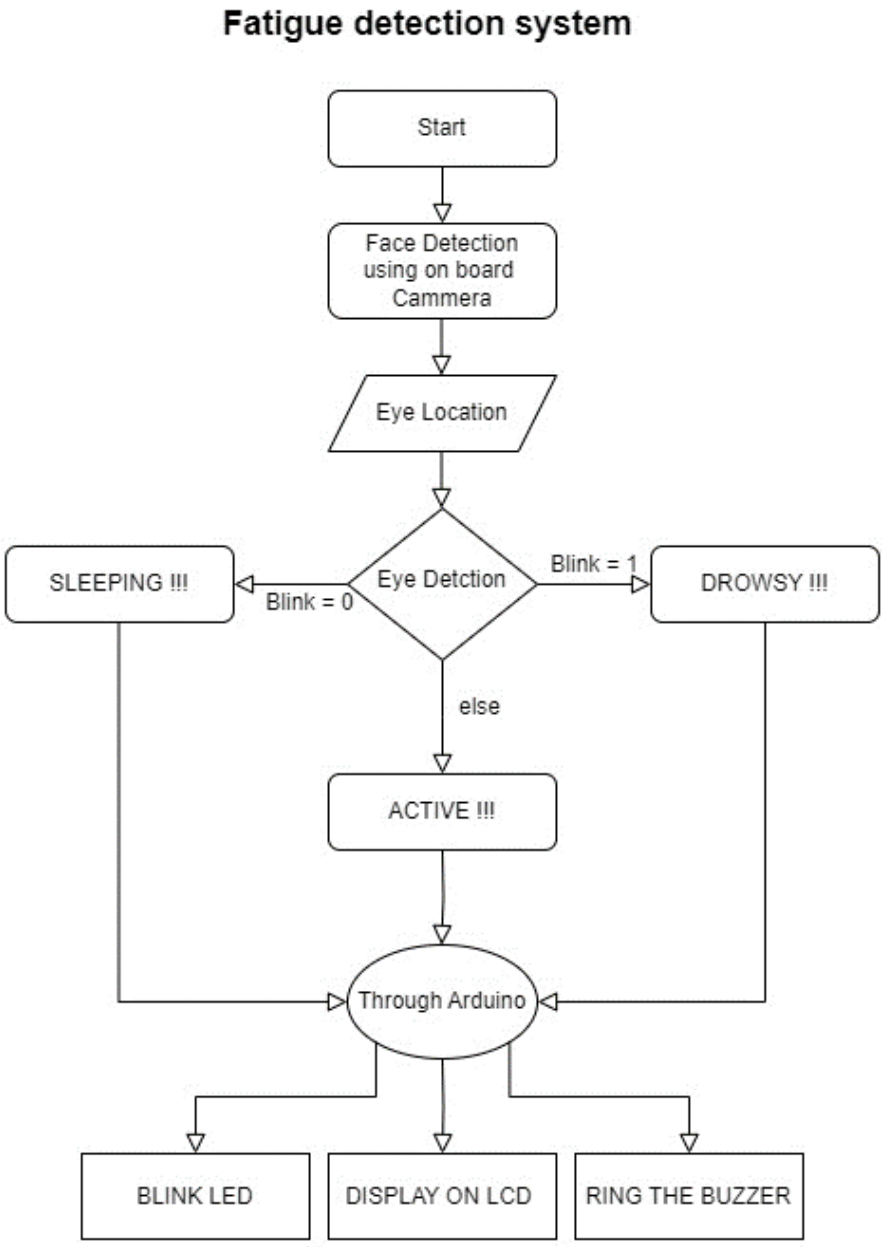
Arduino is a highly versatile platform for creating a Fatigue Detection System for Road Safety, offering easy integration with a variety of sensors, real-time data processing capabilities, and control over alert mechanisms such as buzzers and LEDs. Its compact and low-power design allows for portability within vehicles, and its user-friendly development environment simplifies the programming process. Additionally, Arduino's IoT capabilities enable data transmission for remote monitoring and analysis, all while being cost-effective and highly customizable. With a robust open-source community and educational support, Arduino is an excellent choice for both prototyping and developing practical solutions to prevent road accidents caused by driver fatigue.

EXISTING SYSTEM:

One of the challenges with current drowsiness detection systems is their reliance on devices that monitor physiological parameters like respiration rate, heart rate, and blood pressure. While these devices are effective in detecting fatigue, they can cause discomfort for drivers during prolonged use, potentially distracting them from focusing on the road. Moreover, it cannot be guaranteed that drivers will consistently wear these devices while driving, further limiting their efficacy.
Another concern is the potential for device loss or improper functioning. If drivers misplace or forget to wear these devices, the system's accuracy is compromised, as it heavily relies on real-time data. Even if the devices are worn, technical issues or incorrect usage may result in inaccurate readings, leading to false alarms or failures to detect actual instances of driver fatigue.

To address these limitations, there is a need for non-intrusive fatigue detection systems that can monitor driver alertness without requiring additional devices or causing discomfort. Such systems could utilize advanced computer vision techniques, analyzing driver behavior, eye movements, facial expressions, and other external cues to detect signs of drowsiness. By eliminating the need for wearable devices, these non-intrusive systems can improve both driver comfort and the overall accuracy of fatigue detection.

The architecture of Smart Parking System:



Project Description:

Road accidents, often attributed to factors like fatigue, drowsiness, and alcohol consumption, have been a persistent issue causing significant harm and loss. Fatigue, in particular, remains a challenging problem to address due to its elusive nature and the lack of clear indicators. To mitigate this issue, we present an IoT-based Fatigue Detection System, designed to enhance road safety by monitoring drivers' physical and behavioral conditions

Important Terms:

Face Detection:

This prototype takes input from the camera and tries to detect a face in the video input. The face is detected in a rectangle format and converted to a grayscale image and stored in the memory which can be used for training the model.

Eye Detection:

Since the model works on building a detection system for drowsiness, we need to focus on the eyes to detect drowsiness. The eyes are detected and tracked through the video input by implementing a grid pattern and measuring its distance.

Face Tracking:

Due to the real-time nature of the project, we need to track the faces continuously for any form of distraction. Hence the faces are continuously detected during the entire time.

Drowsiness:

Detection In the previous module the frequency is calculated and if it remains 0 for a longer period then the driver is alerted for the drowsiness through an alert from the system.

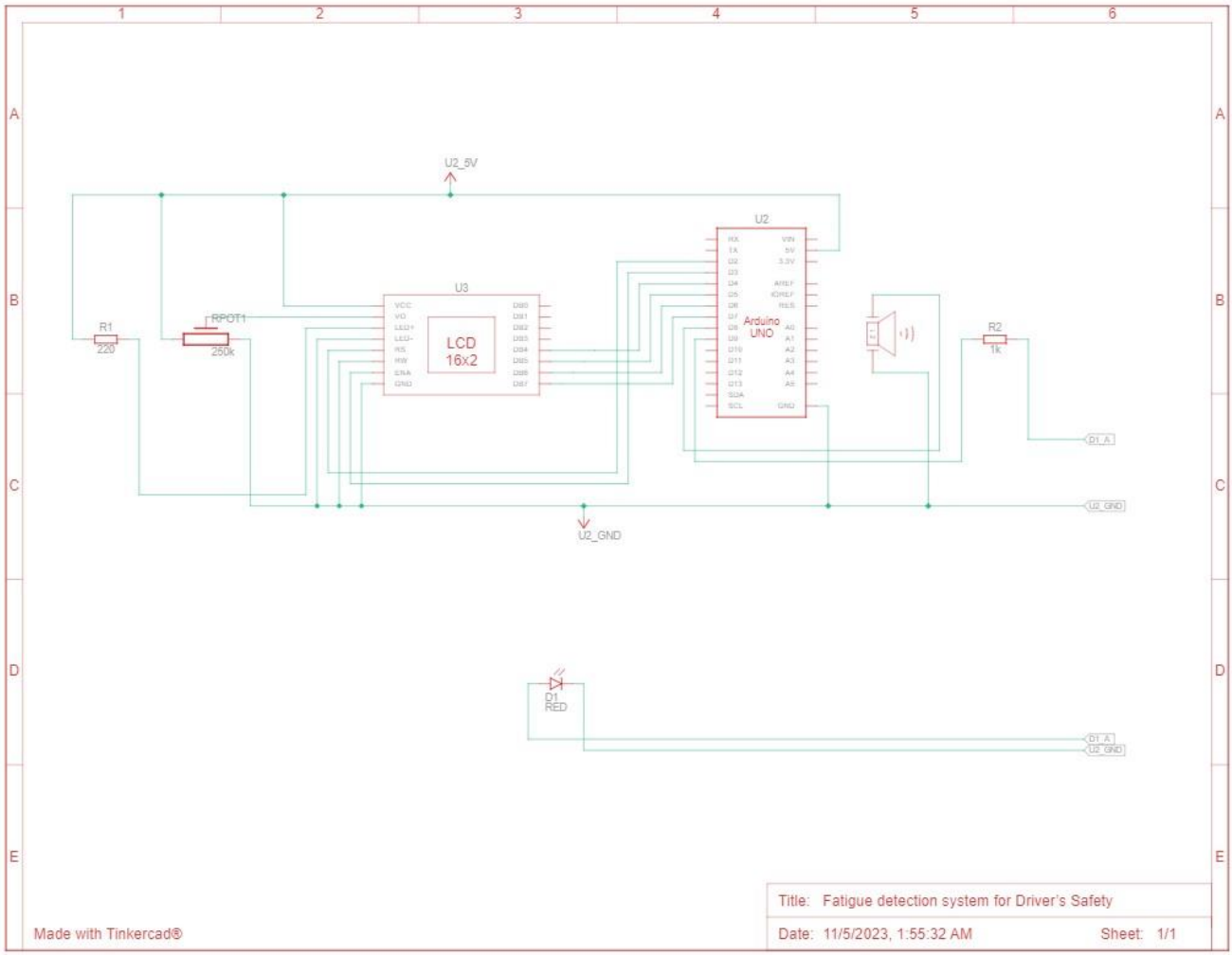
Hardware Used:

Component Name	Quantity
Board HD Camera (currently using Laptop webcam for testing)	**1**
Arduino UNO	**1**
Potentiometer (250 Kohm)	**1**
Resistor (220 Ohm)	**1**
Resistor (1 Kohm)	**1**
Powerful LED	**1**
Piezo Buzzer	**1**
LCD 16 x 2	**1**
Bread Board	**1**
Jumping wires	**N/A**

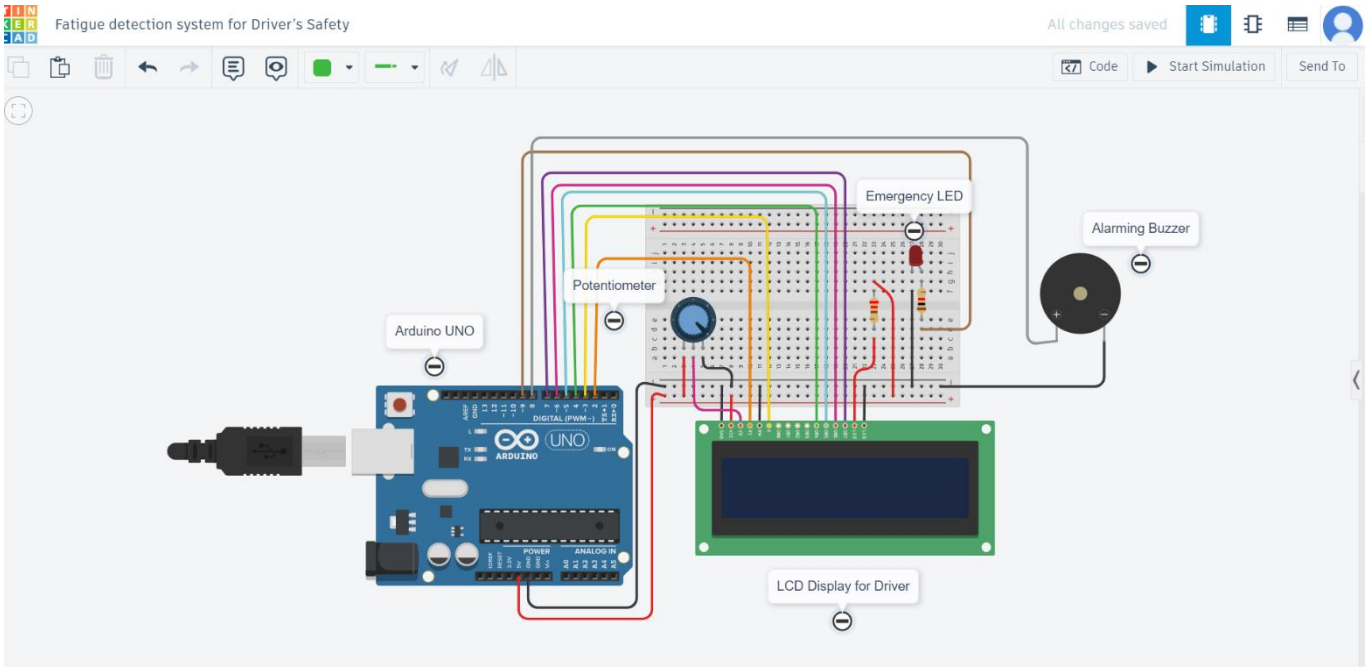
Libraries Used:

Libraries	Function
OpenCV Library	For basic image processing functions
NumPy	For array-related functions
Dlib	For deep learning-based Modules and face landmark detection
face_utils	From imutils for basic operations of conversion
Serial	For Serial Monitor and connection to Arduino
Time	For Delay time

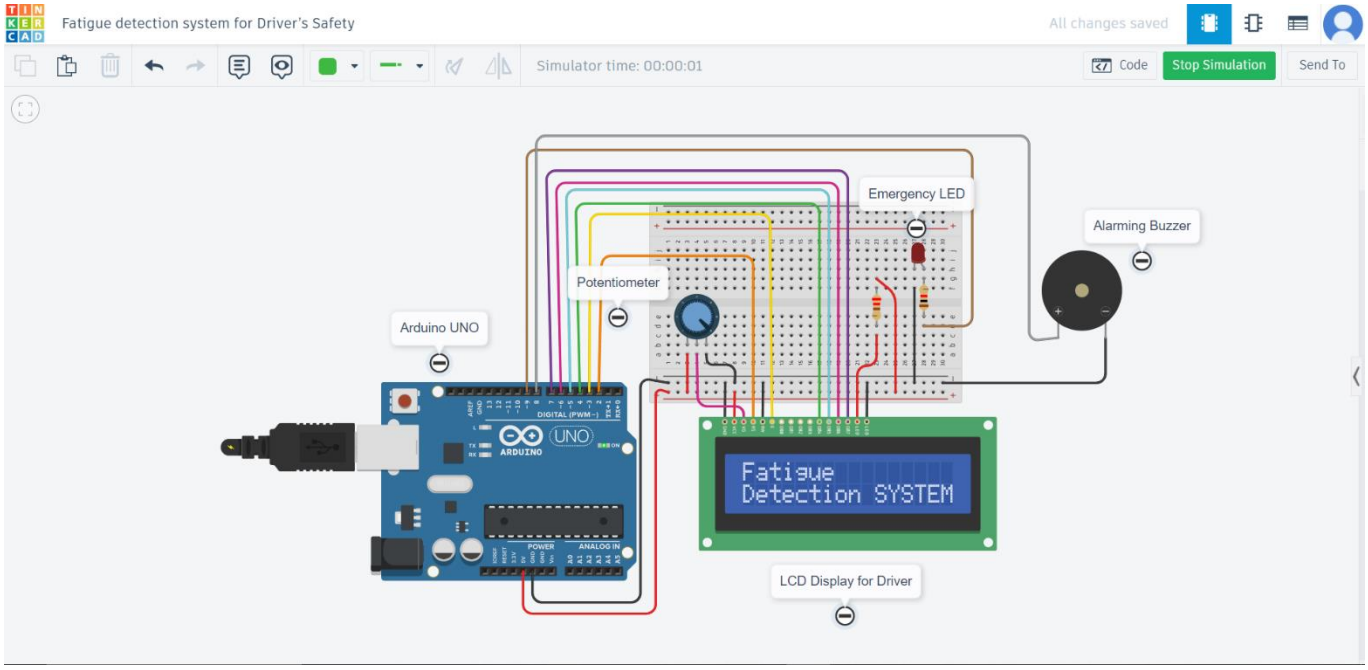
Circuit Diagram:



Complete Circuit:



Simulation:



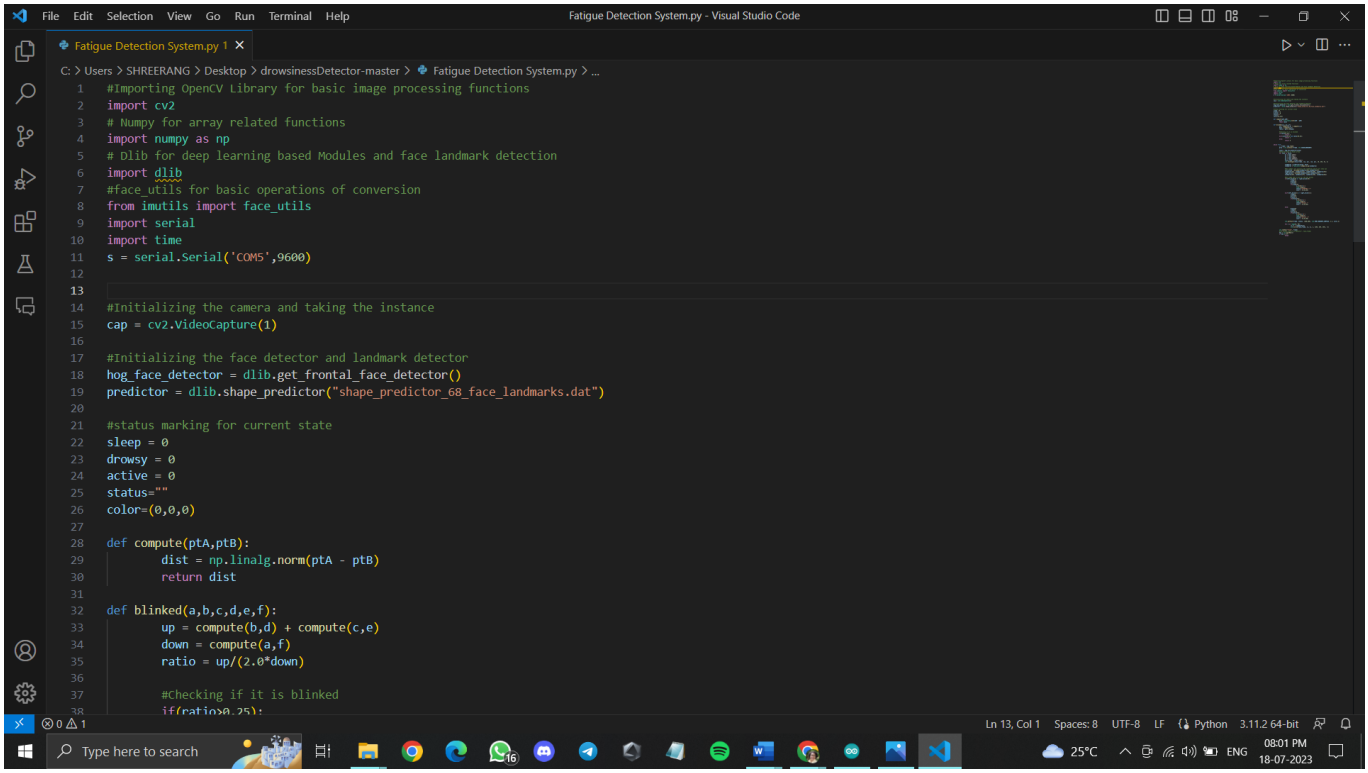
Tinkercad Link:

https://www.tinkercad.com/things/lSF7QlCoVxk?sharecode=xE1bOHucXAtwlHYiL61o93zvqxn4biqKFKvn-fZG85E

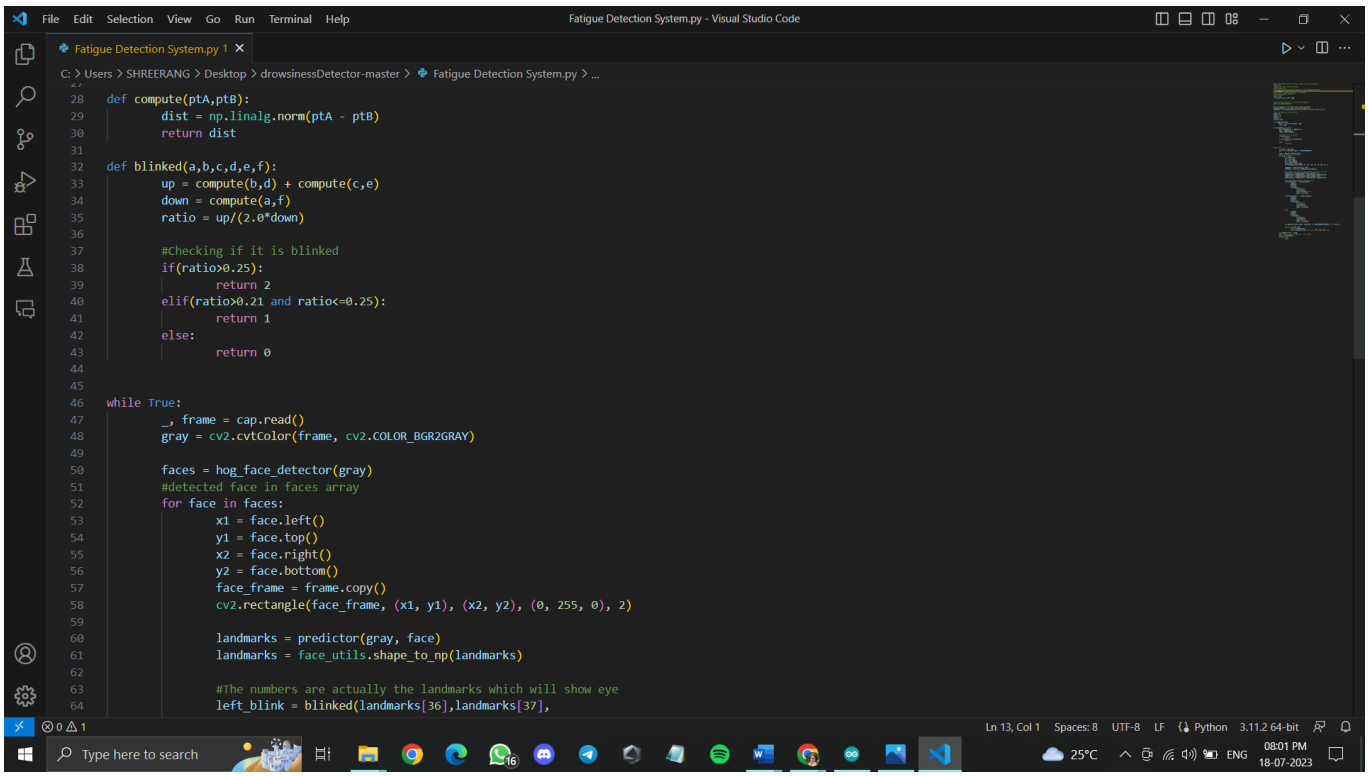
Arduino Code:

```
#include <LiquidCrystal.h>

LiquidCrystal lcd(2, 3, 4, 5, 6,7);
const int buzzer_Pin = 8;
const int led_Pin = 9;
char sleep_status = 0;

void setup() {
  Serial.begin(9600);
  pinMode(buzzer_Pin, OUTPUT);
  pinMode(led_Pin, OUTPUT);
  lcd.begin(16, 2);
  lcd.print("Fatigue ");
  lcd.setCursor(0,2);
  lcd.print("Detection SYSTEM");
  digitalWrite(buzzer_Pin, LOW);
  digitalWrite(led_Pin, LOW);
}
void loop()
{
    while (Serial.available() > 0)
  {
    sleep_status = Serial.read();
    if(sleep_status == 'a')
    {
        lcd.clear();
        lcd.print("Please wake up");
        digitalWrite(buzzer_Pin, HIGH);
        digitalWrite(led_Pin, HIGH);
        delay(2000);
        digitalWrite(buzzer_Pin, LOW);
        digitalWrite(led_Pin, LOW);

        delay(100);
    }
    else if(sleep_status == 'b')
    {
        lcd.clear();
        lcd.print("All Ok");
        lcd.setCursor(0,2);
        lcd.print("Drive Safe");
        digitalWrite(buzzer_Pin, LOW);
        digitalWrite(led_Pin, LOW);
        delay(2000);
    }
    else
    {
      /* Do Nothing */
    }
  }
}
```

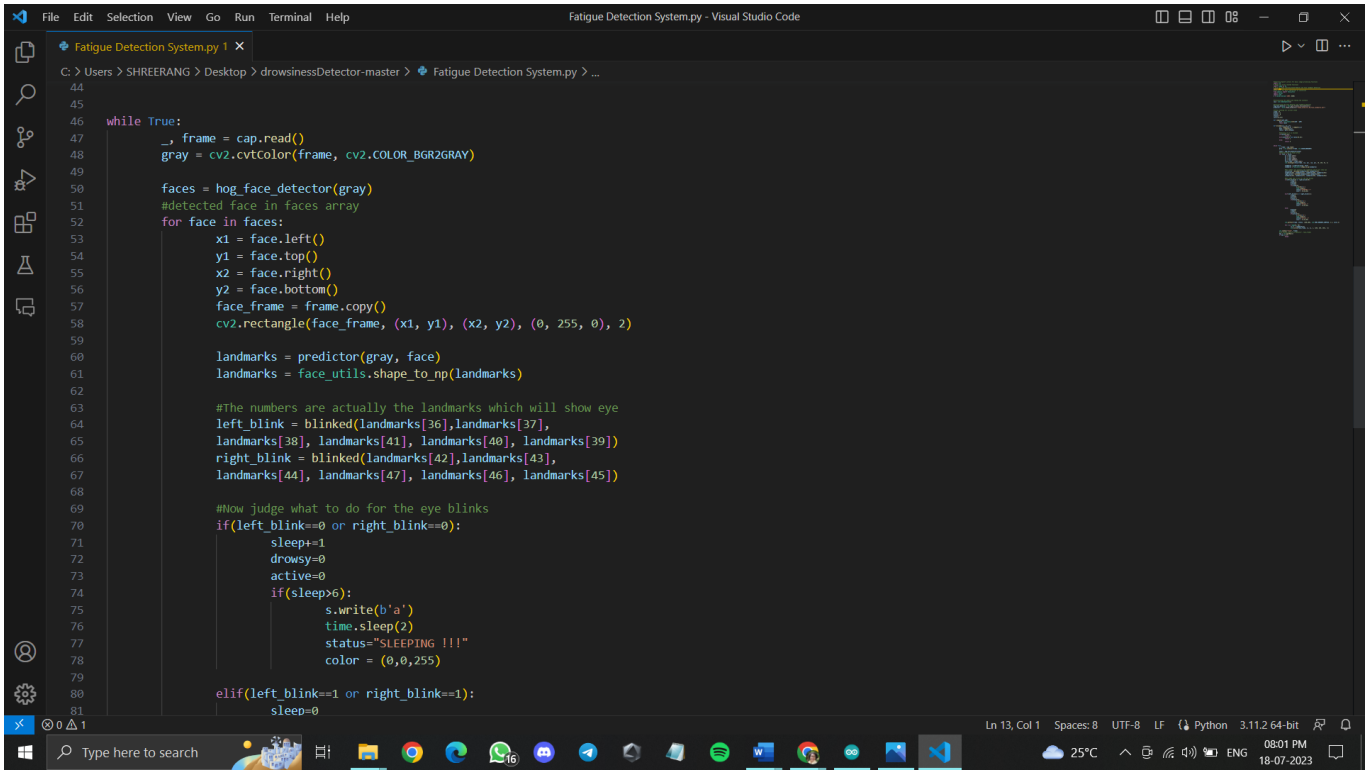
Python Code for Fatigue/Drowsiness Detection

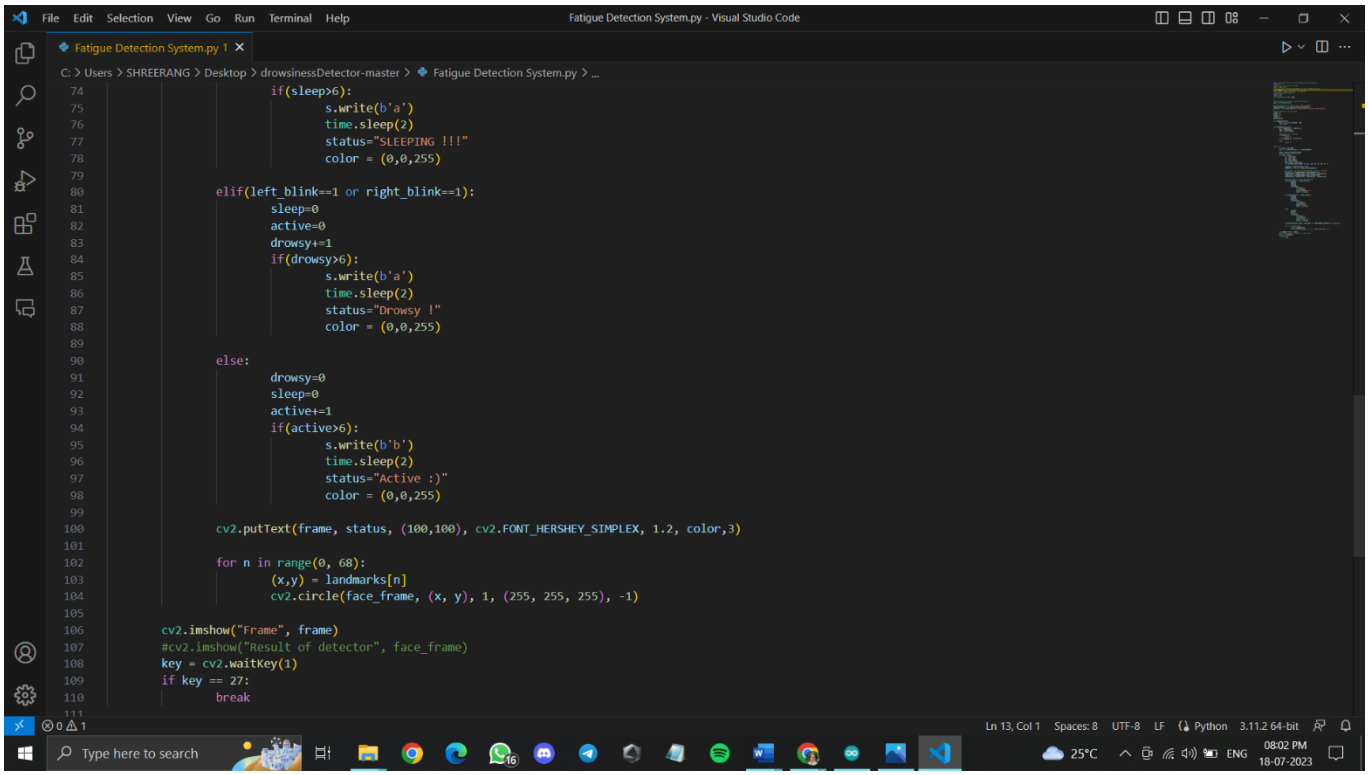


```python
#Importing OpenCV Library for basic image processing functions
import cv2
# Numpy for array related functions
import numpy as np
# Dlib for deep learning based Modules and face landmark detection
import dlib
#face_utils for basic operations of conversion
from imutils import face_utils
import serial
import time
s = serial.Serial('COM5',9600)


#Initializing the camera and taking the instance
cap = cv2.VideoCapture(1)

#Initializing the face detector and landmark detector
hog_face_detector = dlib.get_frontal_face_detector()
predictor = dlib.shape_predictor("shape_predictor_68_face_landmarks.dat")

#status marking for current state
sleep = 0
drowsy = 0
active = 0
status=""
color=(0,0,0)

def compute(ptA,ptB):
        dist = np.linalg.norm(ptA - ptB)
        return dist

def blinked(a,b,c,d,e,f):
        up = compute(b,d) + compute(c,e)
        down = compute(a,f)
        ratio = up/(2.0*down)

        #Checking if it is blinked
        if(ratio>0.25):
```



```python
def compute(ptA,ptB):
        dist = np.linalg.norm(ptA - ptB)
        return dist

def blinked(a,b,c,d,e,f):
        up = compute(b,d) + compute(c,e)
        down = compute(a,f)
        ratio = up/(2.0*down)

        #Checking if it is blinked
        if(ratio>0.25):
                return 2
        elif(ratio>0.21 and ratio<=0.25):
                return 1
        else:
                return 0


while True:
        _, frame = cap.read()
        gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)

        faces = hog_face_detector(gray)
        #detected face in faces array
        for face in faces:
                x1 = face.left()
                y1 = face.top()
                x2 = face.right()
                y2 = face.bottom()
                face_frame = frame.copy()
                cv2.rectangle(face_frame, (x1, y1), (x2, y2), (0, 255, 0), 2)

                landmarks = predictor(gray, face)
                landmarks = face_utils.shape_to_np(landmarks)

                #The numbers are actually the landmarks which will show eye
                left_blink = blinked(landmarks[36],landmarks[37],
```





Note- The above python code takes camera input from the webcam of my laptop for testing which can be further replaced by using an HD onboard camera in the car with sensors if needed.

Explanation:

The "Fatigue detection system for Driver's Safety" project is an innovative solution aimed at mitigating the risks associated with drowsy driving and enhancing road safety. By utilizing an Arduino microcontroller along with a range of sensors and cameras, this system provides continuous monitoring of the driver's vital signs and facial expressions.

The system incorporates advanced computer vision techniques. Cameras are strategically positioned to monitor the driver's facial expressions, eye blink patterns, and head movements. By analyzing these parameters, the system can accurately identify signs of drowsiness or fatigue.

In addition to facial data, the system can be employed with various sensors to gather real-time data on the driver's physiological parameters. This can include heart rate sensors, respiration rate sensors, and blood pressure monitors (added if required for certain drivers). These sensors work together to capture vital signs, enabling the system to assess the driver's level of fatigue accurately.

Once signs of drowsiness are detected, the system promptly alerts the driver to take necessary actions to prevent accidents. This can be done through a variety of methods, including audible alarms, visual cues, emergency LEDs displayed on a dashboard screen, or even haptic feedback such as vibrations in the steering wheel or seat.

The primary objective of this project is to improve road safety by addressing the issue of driver fatigue. By continuously monitoring vital signs and analyzing facial expressions and behavior, the system can provide timely warnings, allowing the driver to take appropriate measures to combat drowsiness. This could include taking a break, drinking coffee, opening a window for fresh air, or engaging in other stimulating activities.

Applications:

1. **Automobile Industry:** The system can be integrated into vehicles as a standard safety feature to monitor driver fatigue and alert the driver when needed, contributing to overall road safety.

2. **Fleet Management:** Companies that manage fleets of vehicles, such as trucking and logistics companies, can use this system to ensure the well-being of their drivers and reduce accidents caused by fatigue.

3. **Public Transportation:** Public transportation services, including buses and trains, can implement fatigue detection systems to ensure the safety of passengers and reduce the likelihood of accidents caused by drowsy drivers.

4. **Ride-Sharing and Taxi Services:** Ride-sharing and taxi companies can deploy this technology to enhance passenger safety and provide an added layer of security and reassurance.

5. **Long-Distance Travel:** Vehicles used for long-distance travel, like tour buses and intercity buses, can benefit from fatigue detection systems to protect both passengers and drivers during extended journeys.

6. **Fleet Insurance:** Insurance companies can offer discounts to businesses that equip their vehicles with fatigue detection systems, reducing the risk of accidents and insurance claims.

7. **Healthcare:** Healthcare providers can use this technology to monitor the alertness of patients who have medical conditions that affect their ability to drive safely, such as sleep disorders.

Conclusion:

In conclusion, the project is an innovative and comprehensive solution designed to address the critical issue of drowsy driving and enhance road safety. By integrating an Arduino microcontroller, cameras, and various sensors, the system continuously monitors the driver's vital signs, facial expressions, and behavior to accurately identify signs of fatigue.
Eye-state analysis-based methods have many benefits such as being non-intrusive and having low computation costs, high robustness, high accuracy, and so on. Through advanced computer vision techniques, the system can analyze facial data, eye blink patterns, and head movements, providing reliable indicators of drowsiness. Additionally, real-time data from physiological sensors, such as heart rate, respiration rate, and blood pressure, further contribute to assessing the driver's level of fatigue. Once signs of drowsiness are detected, the system promptly alerts the driver, enabling them to take appropriate actions to prevent accidents.
By promoting timely warnings and encouraging drivers to address their drowsiness, this system significantly reduces the risk of accidents caused by fatigue-related incidents. It acts as a vital tool in promoting safer journeys for drivers, passengers, and other road users, contributing to an overall safer road environment. Ultimately, this project's success lies in its potential to save lives and create a safer driving experience for everyone on the road.

Note: *Due to the absence of hardware I have performed the hardware part on tinker cad and the eye detection part on the VScode platform. Since the Tinkercad cannot take input from the laptop's webcam unlike the Arduino IDE; the circuit runs but not with the input of the camera. Hence the eye-detection code has been performed independently. With appropriate hardware and connection, the entire system can perform seamlessly.*

Reference:

Research papers:

https://www.ijraset.com/research-paper/iot-based-driver-drowsiness-detection-and-smart-alerting-system
https://www.sciencedirect.com/science/article/abs/pii/S2542660523000288
https://ijcrt.org/papers/IJCRT22A6749.pdf
http://www.ijrar.com/upload_issue/ijrar_issue_20543712.pdf
https://ieeexplore.ieee.org/document/8550026

YouTube:

https://www.youtube.com/watch?v=Ta7I0OB_RWU&ab_channel=RSAIreland
https://www.youtube.com/watch?v=0SZPKW5QdGY&ab_channel=MyCarDoesWhat
https://www.youtube.com/watch?v=qwUIFKi4V48&ab_channel=DeepLearning_by_PhDScholar
https://www.youtube.com/watch?v=tFNJGim3FXw&ab_channel=NicholasRenotte

Kaggle:

https://www.kaggle.com/code/adinishad/driver-drowsiness-using-keras
https://www.kaggle.com/datasets/prasadvpatil/mrl-dataset

Key reference for Python code:

https://www.youtube.com/watch?v=ksi42rwGyas&ab_channel=Aryanverma
https://github.com/infoaryan/Driver-Drowsiness-Detection