

Date	06 May 2023
Team ID	NM2023TMID21996
Project Name	Drowsiness detection and Alerting system

1 INTRODUCTION

Driver drowsiness is a significant cause of road accidents, leading to loss of lives and property. To address this critical issue, we present a project that leverages computer vision, OpenCV, IBM IoT service, and Node-RED to develop a drowsiness detection system. The primary goal of our project is to monitor the driver's level of drowsiness in real-time and provide timely alerts to ensure their safety and the safety of other road users.

1.1 Project Overview

The drowsiness detection system utilizes computer vision techniques to analyze the driver's facial features and eye movements, detecting signs of drowsiness such as drooping eyelids, prolonged eye closure, and changes in facial expressions. By employing OpenCV, we can accurately and efficiently process the video feed from a camera mounted inside the vehicle.

To facilitate communication and reporting, we integrate the IBM IoT service into our system. This enables us to send the drowsiness status data, including timestamps and severity levels, to a centralized platform. Node-RED, a powerful flow-based programming tool, is used to create an intuitive dashboard for viewing the drowsiness status. Family members or other authorized users can access the dashboard to receive real-time information about the driver's condition.

1.2 Purpose

The purpose of our project is to improve road safety by actively monitoring and addressing driver drowsiness. By detecting signs of drowsiness in real-time, our system provides timely alerts, prompting the driver to take necessary actions such as resting or stopping for a break. The integration of computer vision, IoT technologies, and visualization tools empowers our drowsiness detection system to not only monitor the driver but also create a collaborative environment for ensuring road safety.

By implementing this project, we aim to achieve the following objectives:

Reduce the number of accidents caused by drowsy driving.

Enhance driver awareness and prompt them to take appropriate actions.

Provide real-time drowsiness status updates to family members or other concerned individuals.

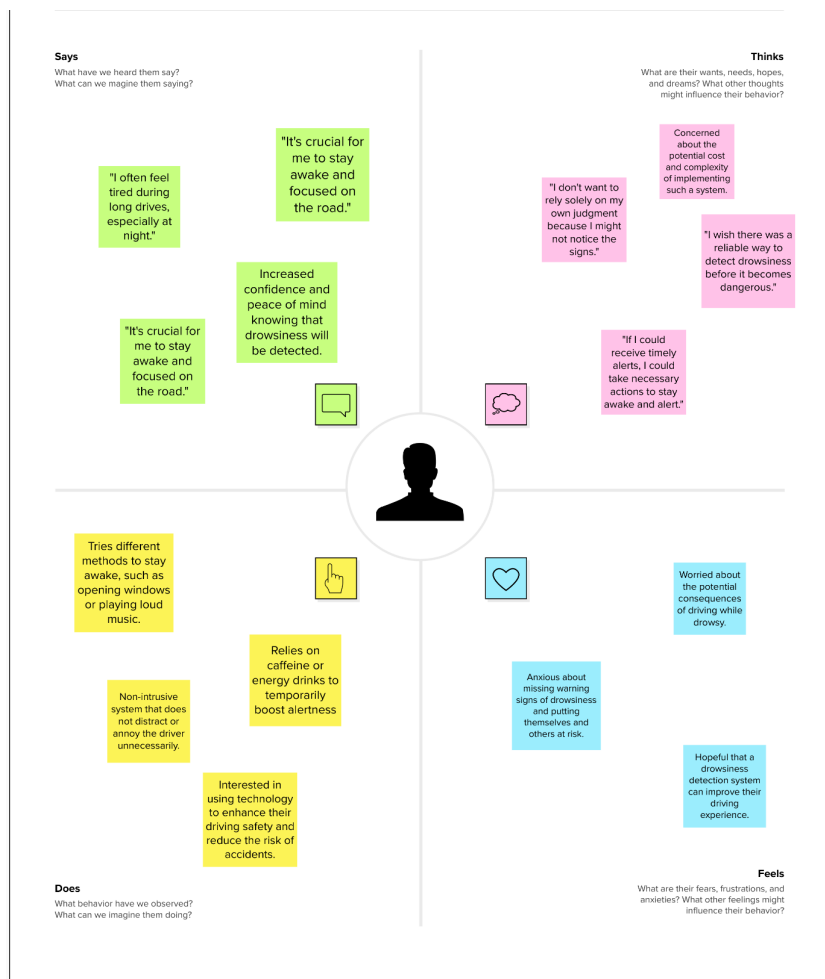
Create a scalable and adaptable system that can be integrated into various vehicle types and configurations.

Through our project, we strive to make a meaningful contribution towards improving road safety, saving lives, and preventing injuries caused by driver drowsiness.

2 IDEATION & PROPOSED SOLUTION

2.1 Problem Statement Definition

Real-time Driver Drowsiness Detection for Enhanced Road Safety



2.3 Ideation & Brainstorming

Step-1: Team Gathering, Collaboration and Select the Problem Statement

Step-2: Brainstorm, Idea Listing and Grouping

Step-3: Idea Prioritization

 **Brainstorming.pdf**

2.4 Proposed Solution

Our proposed solution is to develop a real-time driver drowsiness detection system that utilizes computer vision techniques, OpenCV, and IoT technologies to enhance road safety and mitigate the risks associated with drowsy driving. The system will continuously monitor the driver's facial expressions and eye movements, accurately detecting signs of drowsiness and providing timely alerts.

Computer Vision and OpenCV: We will leverage computer vision algorithms and techniques, utilizing OpenCV, to analyze the driver's facial features, eye movements, and other relevant parameters. By employing image processing and pattern recognition, we will accurately identify drowsiness indicators such as drooping eyelids, prolonged eye closure, and changes in facial expressions.

Real-time Monitoring: The system will operate in real-time, capturing and processing video feed from a camera mounted inside the vehicle. By implementing efficient algorithms, we will ensure low latency and real-time responsiveness, enabling prompt detection of drowsiness signs.

Alert System: Once drowsiness is detected, the system will generate timely alerts to notify the driver and prompt them to take necessary actions. These alerts can be in the form of visual notifications on a display screen or auditory warnings, ensuring the driver's attention is drawn to their drowsy state.

Integration with IoT and IBM IoT Service: To facilitate seamless communication and data transmission, we will integrate the system with IoT technologies. Specifically, we will utilize the

IBM IoT service to establish a connection between the system and a centralized platform. This will enable the transmission of drowsiness status data, including timestamps and severity levels, to be securely and efficiently transmitted.

Node-RED Dashboard for Reporting: To provide real-time status updates and reporting capabilities, we will utilize Node-RED to create an intuitive dashboard. Family members or authorized users can access this dashboard to monitor the driver's drowsiness status remotely. The dashboard will present relevant metrics and visualizations, ensuring easy interpretation of the data.

3. Requirement Analysis

3.1. Functional Requirements:

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Device Registration	Registration through IBM Cloud
FR-2	Node-Red Hosting	Hosting via heroku / firebase / IBM cloud

3.2. Non-functional Requirements:

Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	Usability	Clear Feedback: Users should receive clear and easily understandable feedback regarding their drowsiness state and any required actions
NFR-2	Security	Secure Communication: All communication between the system and external

		devices/platforms should be encrypted to prevent unauthorized access.
NFR-3	Reliability	Continuous Monitoring: The system should consistently monitor drowsiness levels without interruptions to ensure reliable performance.
NFR-4	Performance	Real-time Detection: The system should detect drowsiness in real-time, providing immediate alerts and responses. Low Latency: The system should have minimal delay in processing sensor data and delivering timely feedback to users.
NFR-5	Availability	High Uptime: The system should have a high uptime, ensuring it is available for use whenever needed.
NFR-6	Scalability	High Uptime: The system should have a high uptime, ensuring it is available for use whenever needed.

4. Project Design

4.1. Solution Architecture

The system uses raspberry pi and a camera to detect drowsiness of the driver

It uses OpenCV and Imutils to detect the drowsiness and sends the status to IBM Cloud

Node-Red is used to Show the status of the driver that could be used for Live Status Update

Solution Architecture Diagram:

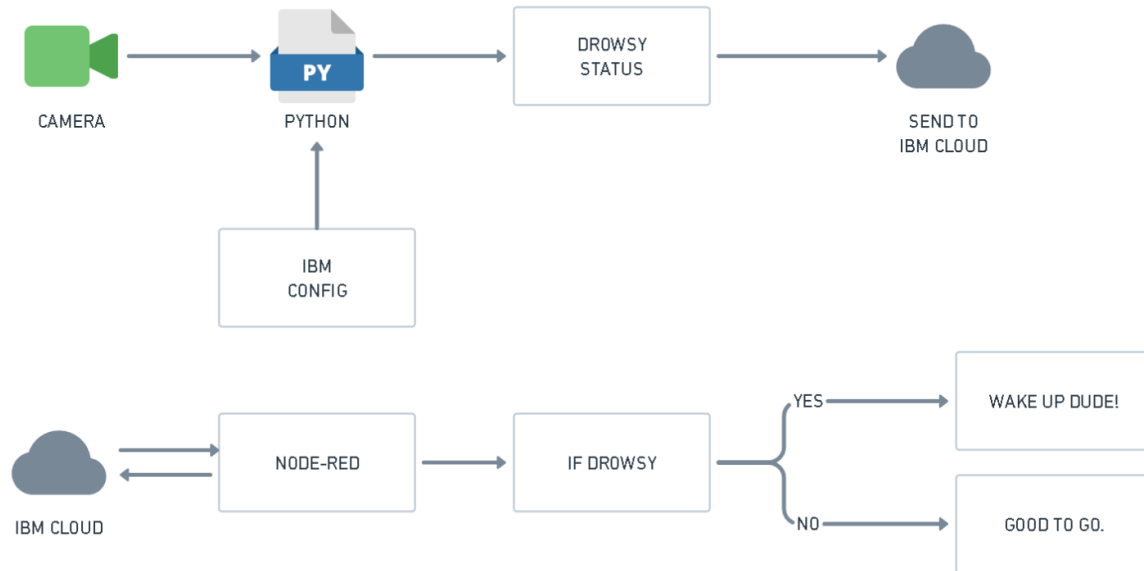


Figure 1: Architecture and data flow of the Drowsiness detection and Alerting system

5. Coding And Solutioning

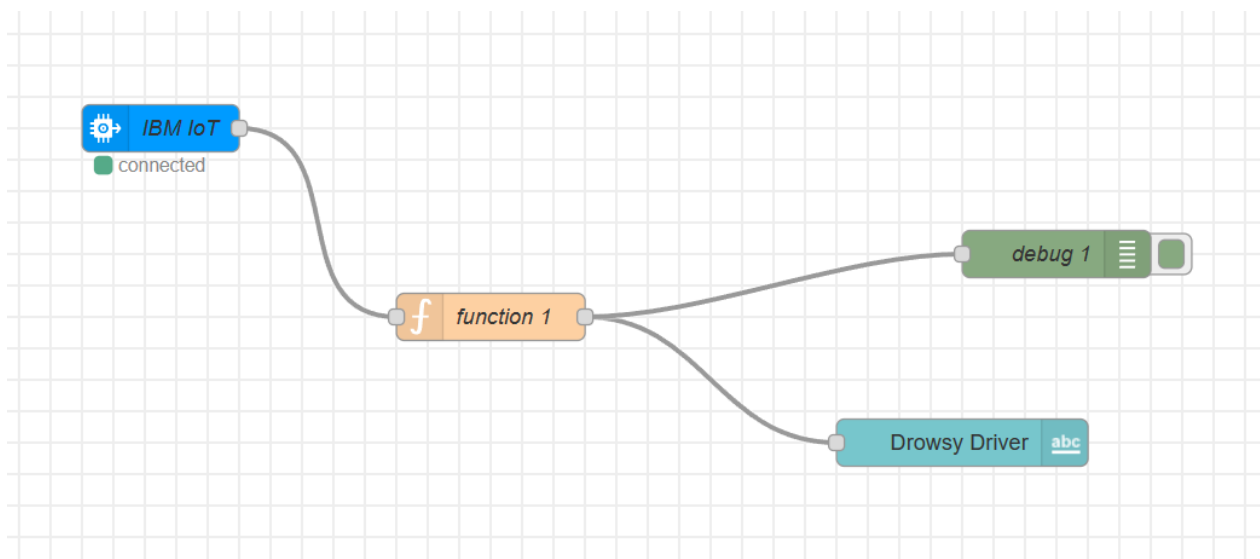


Figure 2: Node-Red Connections

Raspberrypi Code

```
import cv2
from imutils import face_utils
import dlib
from scipy.spatial import distance

import ibmiotf.application
import ibmiotf.device
import random

# Provide your IBM Watson Device Credentials
organization = "51y8x8"
deviceType = "abcd"
deviceId = "wokwi"
authMethod = "token"
authToken = "12345678"

import sys

def ibmstart(x):

    def myCommandCallback(cmd):
        print("Command received: %s" % cmd.data['command'])
        print(cmd)
```

```
try:
    deviceOptions = {"org": organization, "type": deviceType,
                     "id": deviceId, "auth-method": authMethod, "auth-token": authToken}
    deviceCli = ibmiotf.device.Client(deviceOptions)
    # .....
```

```
except Exception as e:
    print("Caught exception connecting device: %s" % str(e))
    sys.exit()
```

```
deviceCli.connect()
data = {'Status': x}
# print data
```

```
def myOnPublishCallback():
    print("Published Status = %s" % x, "to IBM Watson")
```

```
success = deviceCli.publishEvent(
    "DD", "json", data, qos=0, on_publish=myOnPublishCallback)
if not success:
    print("Not connected to IoT")
```

```
deviceCli.commandCallback = myCommandCallback
deviceCli.disconnect()
```

```
# Function to calculate the eye aspect ratio (EAR)
def eye_aspect_ratio(eye):
```



```
# Calculate the euclidean distances between the two sets of
```

```
# vertical eye landmarks (x, y)-coordinates
```

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

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

```
# Calculate the euclidean distance between the horizontal
```

```
# eye landmark (x, y)-coordinates
```

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

```
# Compute the eye aspect ratio
```

```
ear = (A + B) / (2.0 * C)
```

```
return ear
```

```
# Constants for eye aspect ratio (EAR) thresholds
```

```
EAR_THRESHOLD = 0.25
```

```
EAR_CONSEC_FRAMES = 20
```

```
# Load the facial landmark predictor from dlib
```

```
predictor_path = 'shape_predictor_68_face_landmarks.dat' # Path to the shape predictor model  
file
```

```
detector = dlib.get_frontal_face_detector()
```

```
predictor = dlib.shape_predictor(predictor_path)
```

```
# Initialize counters and drowsy status
```

```
frame_counter = 0
```

```
drowsy = False
```

```
lm = False
```

```
# Start the video capture
```

```
video_capture = cv2.VideoCapture(0)
```

```
while True:
```

```
    # Read a frame from the video capture
```

```
    ret, frame = video_capture.read()
```

```
    if not ret:
```

```
        break
```

```
    # Resize the frame for faster processing
```

```
    frame = cv2.resize(frame, (640, 480))
```

```
    # Convert the frame to grayscale
```

```
    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
```

```
    # Detect faces in the grayscale frame
```

```
    faces = detector(gray, 0)
```

```
    if len(faces) != 0:
```

```
        face = faces[0]
```

```
        # Predict the facial landmarks
```

```
        shape = predictor(gray, face)
```

```
        shape = face_utils.shape_to_np(shape)
```

```
        # Extract the left and right eye coordinates
```

```
        left_eye = shape[42:48]
```

```

right_eye = shape[36:42]

# Calculate the eye aspect ratios (EARs)
left_eye = eye_aspect_ratio(left_eye)
right_eye = eye_aspect_ratio(right_eye)

# Average the EARs of both eyes
ear = (left_eye + right_eye) / 2.0

# Check if the EAR is below the threshold
if ear < EAR_THRESHOLD:
    frame_counter += 1
else:
    # Reset the frame counter
    frame_counter = 0
    drowsy = False

# Check if drowsiness is detected for consecutive frames
if frame_counter >= EAR_CONSEC_FRAMES:
    drowsy = True

    cv2.putText(frame, "Drowsy!", (10, 30),
                cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 2)

if lm != drowsy:
    lm = drowsy
    ibmstart(drowsy)

# Display the eye aspect ratio on the frame

```

```

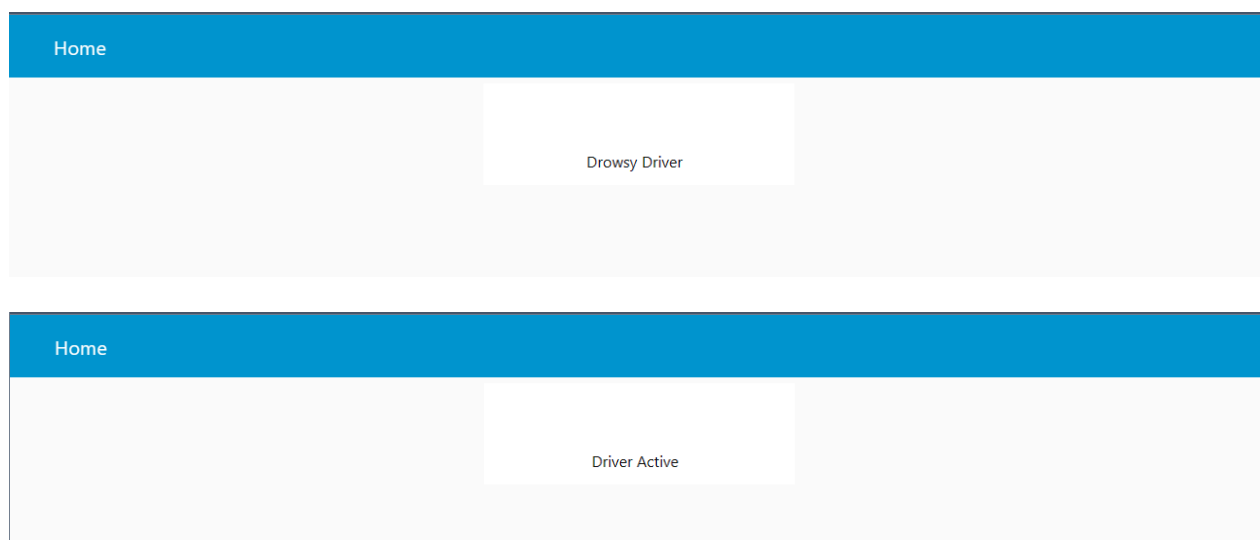
cv2.putText(frame, "EAR: {:.2f}".format(ear), (300, 30),
            cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 255, 0), 2)
# Display the resulting frame
cv2.imshow("Drowsiness Detection", frame)

# Quit the program if 'q' is pressed
if cv2.waitKey(1) & 0xFF == ord('q'):
    break

# Release the video capture and close the windows
video_capture.release()
cv2.destroyAllWindows()

```

5.1 Remote Status Visibility



6. RESULTS

The system is properly able to detect drowsy drivers in non obstructed conditions and is able to alert the driver and update the dashboard correctly.

7. ADVANTAGES AND DISADVANTAGES

Advantages:

1. **Enhanced Road Safety:** The system contributes to improved road safety by actively detecting and alerting drivers about drowsiness, reducing the risk of accidents caused by fatigue.
2. **Real-time Monitoring:** The system provides real-time monitoring of the driver's drowsiness status, allowing for prompt intervention and necessary actions to be taken.
3. **Objective Detection:** By leveraging computer vision techniques, the system provides objective and consistent detection of drowsiness indicators, minimizing reliance on subjective assessments.
4. **Customizable Alerts:** The system can be configured to generate alerts in various forms, such as visual notifications or auditory warnings, tailored to the driver's preferences and needs.
5. **IoT Integration:** The integration with IoT technologies enables seamless data transmission and communication, facilitating centralized monitoring and reporting of drowsiness status.
6. **Scalability:** The system can be implemented in different vehicle types and configurations, allowing for scalability and wider adoption across various transportation industries.

Disadvantages:

1. **False Alarms:** The system may generate false alarms if certain conditions, such as temporary occlusion of the face or sudden head movements, are detected as drowsiness indicators. This can potentially lead to driver annoyance or reduced trust in the system.
2. **Dependence on Camera Placement:** The accuracy of the system heavily relies on the proper placement and alignment of the camera inside the vehicle. Any obstruction or improper positioning may affect the system's effectiveness.
3. **Limited Accuracy in Certain Scenarios:** The system's accuracy may be reduced in challenging lighting conditions or when the driver's facial expressions are less pronounced, potentially leading to lower detection accuracy.
4. **Hardware and Technical Requirements:** Implementing the system may require specific

hardware components, such as a camera and processing unit, which can add to the overall cost and technical complexity of the solution.

5. **Privacy Concerns:** The use of a camera inside the vehicle raises privacy concerns, as it captures and processes personal data. Appropriate measures should be taken to ensure data security and comply with privacy regulations.
6. **Driver Reliance and Overconfidence:** There is a risk that drivers may become overly reliant on the system, potentially leading to complacency or a false sense of security. It is important to educate drivers about the system's limitations and encourage responsible driving practices.

8. CONCLUSION

In conclusion, the development of a real-time driver drowsiness detection system using computer vision techniques, such as OpenCV, combined with IoT integration and a user-friendly dashboard, holds immense potential for enhancing road safety and mitigating the risks associated with drowsy driving. By addressing the significant problem of driver drowsiness, this project aims to save lives, reduce accidents, and promote safer driving practices.

The proposed solution offers several advantages, including enhanced road safety through objective detection, real-time monitoring, and customizable alerts. The integration with IoT technologies enables seamless data transmission, facilitating centralized monitoring and reporting of drowsiness status. Furthermore, the scalability of the system allows for its implementation across various vehicle types and configurations, making it applicable in different transportation industries.

However, it is important to acknowledge some limitations and considerations, such as the potential for false alarms, dependence on camera placement, and privacy concerns associated with capturing and processing personal data. These aspects should be carefully addressed to ensure the system's effectiveness, user acceptance, and compliance with privacy regulations.

In conclusion, the drowsiness detection system offers a valuable solution to a critical problem, paving the way for safer roads and reducing the risks associated with drowsy driving. Through ongoing improvements, addressing limitations, and promoting awareness among drivers, the system can continue to evolve and make a significant impact in enhancing road safety.

By implementing this project, we strive to contribute to the collective efforts in reducing

accidents caused by driver fatigue, protecting lives, and creating a safer environment for all road users.

9. FUTURE SCOPE

The future scope for this project includes several potential areas for further development and improvement:

1. **Advanced Drowsiness Detection Algorithms:** Explore and implement advanced computer vision algorithms and machine learning techniques to enhance the accuracy and robustness of drowsiness detection. This may involve leveraging deep learning models or incorporating additional biometric signals, such as heart rate variability or EEG data, for more precise detection.
2. **Multi-modal Sensor Integration:** Consider integrating additional sensors, such as infrared or depth sensors, to capture more comprehensive data about the driver's behavior and physiological state. Combining multiple modalities can enhance the accuracy and reliability of drowsiness detection, particularly in challenging lighting conditions or with drivers wearing accessories like sunglasses.
3. **Personalization and Driver Profiling:** Develop methods to personalize the drowsiness detection system based on individual drivers' characteristics and behavior. By creating driver profiles, the system can adapt its thresholds and alert mechanisms to suit each driver's specific needs, improving the overall effectiveness and reducing false alarms.
4. **Real-time Fatigue Assessment:** Move beyond binary drowsiness detection and develop algorithms that can assess the level of driver fatigue in real-time. This can provide more granular insights into the driver's fatigue state, allowing for tailored interventions and proactive measures to prevent fatigue-related accidents.
5. **Integration with Vehicle Systems:** Explore integration possibilities with the vehicle's onboard systems, such as advanced driver assistance systems (ADAS) or lane departure warning systems. By integrating with these systems, the drowsiness detection system can trigger additional safety measures or interventions, such as adaptive cruise control or vibrating seat alerts, to further assist the driver in staying alert and maintaining safe driving behavior.
6. **Long-term Driver Monitoring and Analysis:** Implement a long-term driver monitoring and analysis component that can collect and analyze data over extended periods. This can enable identifying patterns of drowsiness, fatigue, and driving behavior, leading to a better understanding of underlying factors and potential interventions for individual drivers or across a larger population.
7. **Collaboration with Automotive Industry:** Collaborate with automotive manufacturers or

fleet management companies to integrate the drowsiness detection system into vehicles as a standard safety feature. This would involve ensuring compatibility, meeting industry regulations and standards, and conducting rigorous testing and validation to ensure reliable performance across different vehicle models and environments.

By exploring these future avenues, the drowsiness detection system can continue to evolve, offering enhanced capabilities, improved accuracy, and increased adoption in various transportation sectors. The ultimate goal is to further reduce accidents caused by driver drowsiness, saving lives, and making roads safer for everyone.