International Journal of Modern Developments in Engineering and Science
Volume 4, Issue 4, April 2025
https://www.ijmdes.com | ISSN (Online): 2583-3138

Driver Drowsiness Detection Using Artificial Intelligence and Machine Learning

Dhaval Rane[1*], Rohit Danavale[2], Abhijit Rathiya[3], Shwet Pawar[4]

[1]*Professor, Department of Electronics and Telecommunication Engineering, K.C. College of Engineering and Management Studies and Research, Thane, India*

[2,3,4]*Student, Department of Electronics and Telecommunication Engineering, K.C. College of Engineering and Management Studies and Research, Thane, India*

***Abstract*: Drowsiness greatly contributes to the prevalence of road accidents where drivers can get injured or lose their lives. This paper provides a solution to the driving drowsiness issue with the Driver Drowsiness Detection System which tracks a driver's alertness levels to provide warnings when needed. The detection system captures the driver's eye movements, blinking frequency, and head tilting angles through a camera placed on the front dashboard and analyzes them using computer vision techniques. Drowsiness levels are detected using drowsiness detection algorithms or existing frameworks through input signals from the camera. Once the driver exhibits signs of fatigue, the system alerts the driver using visual or audio signals to allow the driver to act upon it. The system is developed with a preventative measure against accidents to improve safety on roads and reduce the number of accidents related to driver exhaustion.**

***Keywords*: Artificial Intelligence, Machine Learning.**

1. Introduction

Global road safety experts have pointed out that driver fatigue is one of the major reasons for traffic accidents. Drowsiness or fatigue affects the driver's alertness which might lead to him losing control over the vehicle. Sleep-deprived drivers suffer from lack of attention, slower reflexes, increased forgetfulness, and poor judgment which when undetected can have devastating outcomes. Unlike alcohol intoxication which is easier to track, driver drowsiness isn't easily noticed as there are no singular indicators which can mark it.

Traffic driven accidents have increased and to curb them, various advanced technologies have emerged like the intelligent transport systems. DDD System or Driver Drowsiness Detection System is one of the examples which is directed to mitigate risk of fatigue on driver and enhance road safety. Action from the driver's side can be monitored to determine the different levels of alertness. The help of technologies can also forecast exhausted driving alert and prevent unwanted accidents from happening.

2. Literature Review

Monitoring driver drowsiness has become one of the driving concerns considering the safety challenges when it comes to controlling the drowsy driver vehicle dynamics. One of the key aspects related to the safety of driving is ensuring that the monitoring system does not interfere with the vehicle control hence drawing the driver attention away from the road.

Most people have focused their attentions on capturing rather subtle changeable features like the head or eye movements or facial expressions while others troubleshoot eye level camera systems to achieve the best vehicle monitoring from every single frame in the stream. Although a non-inazole approach, infrared cameras aid in head and gaze tracking, these offer significantly less discomfort to the users rather than traditional monitoring systems that require a high degree of user involvement.

Many works focused on using non-invasive techniques to tackle the problem, targeting morphology of surrounding tissues and the activity signal linked directly to the subject. Blinking, eye closing, head nodding and any directions, as well as changes in facial expression are equally treated at behavior-based range detector.

3. Methodology

The method of operation of the Driver Drowsiness Detection System is with a focus on automating facial feature analysis capturing the driver's face and using computer vision and machine learning techniques to monitor their facial features in real-time. The system has multiple stages which include data acquisition, face and eye detection, feature extraction, drowsiness classification, alert generation, and others. Below each of them is described in more detail:

Data Acquisition: A camera is mounted on the vehicle dashboard focusing attention on the driver's face. The system captures video frames in real-time. For development and testing purposes, there are many datasets that are available to the public, for instance, YawDD or NTHU-DDD, which contain videos of drivers annotated as either drowsy or alert.

Face and Eye Detection: In the following step the driver's face and the eyes that appear in every frame of the video need to be detected and cropped. For this purpose Haar Cascade Classifier and Dlib's facial landmark for eye and face detection can be used, where identification of facial components like the eyes, nose, and mouth is done to track movement of eyes and

*Corresponding author: dhavalrane2912@gmail.com

blinking.

Feature Extraction: From the captured face area, features connected to dowsiness are extracted. One of the indicators that is widely used is the Eye Aspect Ratio (EAR), which gives the relation of the distance of certain marks in the eye.

4. Program Algorithm

Step 1: Set Up the System

Install relevant packages such as OpenCV, Dlib, and Scipy.Configure preliminary face and eye detectors (Haar cascades or Dlib's shape predictor, for example). Set drowsiness EAR parameter and drowsiness frame cap.

Step 2: Open the Video Capture

Connect to the webcam or video streaming service. Continuously fetch frames from the camera.

Step 3: Face and Eye Recognition:

Change frame to grayscale for easier processing. Face Detection with the selected detector. Detect important facial aspects using facial geometry algorithms. Get eye coordinates from the detected facial geometrical model.

Step 4: Determine EAR

Use the eye geometrical model to obtain eye landmarks for both ears.Calculate mean EAR of both eyes.

Step 5: Drowsiness Detection Threshold Check

If EAR < set threshold: Adjust drowsiness frame count

Else:- Set count back to 0

Step 6: Trigger Drowsiness Detection Alert.

If drowsiness frame count is greater than the limit allowed:

- Show warning message (for example, "Drowsiness Detected") -Activate sound or visual notification

Step 7: Controlling Output Display

Mark the face being detected and show EAR on the video frame. If no alert is needed, display the live video stream.

Step 8: End Condition

This loop will run until the designated exit key is pressed. for example 'q' text goes here. Stop the camera and clear all visual interfaces.

5. Program code

```
In import face_recognition
import cv2
import time
import cv2
import RPi.GPIO as GPIO
BUZZER= 23
GPIO.setup(BUZZER, GPIO.OUT)
previous ="Unknown"
count=0
video_capture = cv2.VideoCapture()
file = 'image_data/image.jpg'
img_image = face_recognition.load_image_file("img.jpg")
img_face_encoding
= face_recognition.face_encodings(img_image)[0]
face_locations = []
face_encodings = []
face_names = []
process_this_frame = True
while True:
    ret, frame = video_capture.read()
    small_frame = cv2.resize(frame, (0, 0), fx=0.25, fy=0.25)
    rgb_small_frame = small_frame[:, :, ::-1]
    if process_this_frame:
    face_locations =
ace_recognition.face_locations(rgb_small_frame)
     face_encodings =
face_recognition.face_encodings(rgb_small_frame,
face_locations)
        cv2.imwrite(file, small_frame)
        face_names = []
        for face_encoding in face_encodings:
     matches =
face_recognition.compare_faces(known_face_encodings,
face_encoding)
         name = "Unknown"
    face_distances =
face_recognition.face_distance(known_face_encodings,
face_encoding)
         best_match_index = np.argmin(face_distances)
          if matches[best_match_index]:
          name = known_face_names[best_match_index]
          direction= eye_game.get_eyeball_direction(file)
          print(direction)
        #eye_game.api.get_eyeball_direction(cv_image_array)
           if previous != direction:
           previous=direction
            else:
           print("old same")
           count=1+count
           print(count)
           if (count>=10):
           GPIO.output(BUZZER, GPIO.HIGH)
           time.sleep(2)
           GPIO.output(BUZZER, GPIO.LOW)
           print("Alert!! Alert!! Driver Drowsiness Detected")
           cv2.putText(frame, "DROWSINESS ALERT!", (10,
30),
   2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 2)
   face_names.append(name)
        process_this_frame = not process_this_frame
   for (top, right, bottom, left), name in
zip(face_locations,face_names):
     # Scale back up face locations since the frame we detected
in was scaled to 1/4 size
       top *= 4
       right *= 4
       bottom *= 4
       left *= 4
       # Draw a box around the face
         cv2.rectangle(frame, (left, top), (right, bottom), (0,
255, 0), 2)
         # Draw a label with a name below the face
        cv2.rectangle(frame, (left, bottom - 35), (right, bottom),
```

```
(0, 0,   255), cv2.FILLED)
            font = cv2.FONT_HERSHEY_DUPLEX
            cv2.putText(frame, name, (left + 6, bottom - 6), font, 1.0, (0, 0, 255), 1)
        cv2.imshow('Video', frame)
        if cv2.waitKey(1) & 0xFF == ord('x'):
            break
    cv2.destroyAllWindows()
```

6. Prominent Features

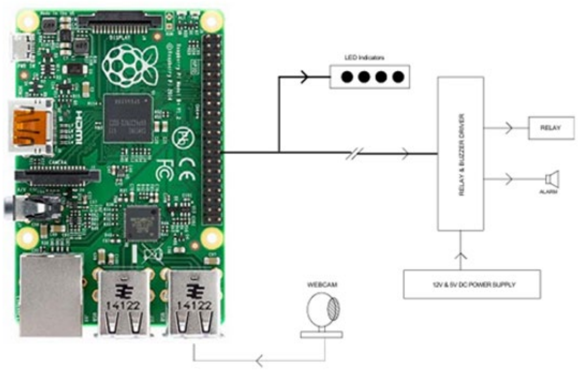

Fig. 1. Block diagram

A. Real-Time Monitoring

Continuously monitors the driver's facial features and behavior in real time using a camera.

B. Non-Intrusive Detection

Uses computer vision instead of wearable sensors, making it comfortable and practical for the driver.

C. Eye Aspect Ratio (EAR) Based Analysis

Tracks eye blink rate and duration of eye closure using the EAR method to detect signs of drowsiness.

D. Facial Landmark Detection

Identifies and analyzes facial features such as eyes, mouth, and head position for more accurate detection.

E. Drowsiness Alert System

Triggers alerts (e.g., audio, visual, or vibration) when drowsiness symptoms are detected to warn the driver.

F. Machine Learning Integration

Employs machine learning or deep learning models to classify driver states (drowsy or alert) more effectively.

G. Customizable Sensitivity

Thresholds for EAR and detection sensitivity can be adjusted based on driver behavior or vehicle type.

H. Lightweight and Efficient

Optimized to work on low-power systems like Raspberry Pi or embedded car systems.

I. Failsafe Mechanism

Provides fallback alerts or records data for later analysis if real-time detection fails.

J. Scalability and Integration

Can be integrated into Advanced Driver Assistance Systems (ADAS) in modern vehicles for added safety.

7. Hardware

- Raspberry Pi
- 8 Mega Pixel Camera
- Buzzer
- SPDT Relay
- Motor – Demonstration
- 12V DC Power Supply

8. Software

- Raspberry Latest OS
- OpenCV
- Python
- Tensflow

9. Conclusion

The Driver Drowsiness Detection System constructs an model revealing how computer vision can be used to monitor a driver’s facial features such as eyes and blinks to detect drowsiness, issue alerts, and prevent potential accidents. It's an effective solution that practical helps reduce road fatalities by sensing real-time fatigue signs.

Head movement, yawning, and other body signals could be added to enhance system accuracy. It could also be applied to commercial fleets to enable shared monitoring and controlling of a vehicle’s various systems for better management which could unlock different forms of additional advancements as technology continues to grow.

The camera-based technology displays hands free advanced monitoring could offer improved driving safety while not interfering with the driver’s performance making systems like these perfect for real world application. Integrating machine learning and algorithms with image processing makes drivers more relaxed and drowsy while ensuring responsiveness and effectiveness is maintained at all times.

References

[1] Jiao, Y., Deng, Y., Luo, Y., & Lu, B. (2020). Driver sleepiness detection from EEG and EOG signals using GAN and LSTM networks. Neurocomputing, 408, 100-111.

[2] Ramzan, M., Khan, H.U., Awan, S.M., Ismail, A., Ilyas, M., & Mahmood, A. (2019). A Survey on State-of-the-Art Drowsiness Detection Techniques. IEEE Access, 7, 61904-61919.

[3] V. U. Maheswari, R. Aluvalu, M. P. Kantipudi, K. K. Chennam, K. Kotecha and J. R. Saini, "Driver Drowsiness Prediction Based on Multiple Aspects Using Image Processing Techniques," in IEEE Access, vol. 10, pp. 54980-54990, 2022.

[4] Albadawi, Y., Takruri, M., & Awad, M. (2022). A Review of Recent Developments in Driver Drowsiness Detection Systems. Sensors (Basel, Switzerland), 22(5), 2069.

[5] J. S. Bajaj, N. Kumar and R. K. Kaushal, "Comparative Study to Detect Driver Drowsiness," 2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), Greater Noida, India, 2021, pp. 678-683.

[6] N. K. Gupta, A. K. Bari, S. Kumar, D. Garg and K. Gupta, "Review Paper on Yawning Detection Prediction System for Driver Drowsiness," 2021

5th International Conference on Trends in Electronics and Informatics (ICOEI), Tirunelveli, India, 2021, pp. 1-6.
[7] M. Ngxande, J. -R. Tapamo and M. Burke, "Bias Remediation in Driver Drowsiness Detection Systems Using Generative Adversarial Networks," in IEEE Access, vol. 8, pp. 55592-55601, 2020.
[8] National Highway Traffic Safety Administration Drowsy Driving. https://www.nhtsa.gov/risky-driving/drowsy-driving
[9] Tefft B. C. (2012). Prevalence of motor vehicle crashes involving drowsy drivers, United States, 1999-2008. Accident; analysis and prevention, 45, 180–186.
[10] National Institutes of Health Drowsiness. https://www.nhlbi.nih.gov/health-topics/education-and-awareness/sleep-health