Driver Drowsiness Detection Using Facial Features

Pragathi M.
Assistant Professor,
Dept. of CS Engg.,
Sir M.   Visvesvaraya
Institute ofTechnology,
Bengaluru, India.

Prakhar
UG Student, Dept. of CS Engg.,
Sir M. Visvesvaraya Institute of
Technology, Bengaluru, India.

Siivei KS
UG Student, Dept. of CS Engg.,
Sir M. Visvesvaraya Institute of
Technology, Bengaluru, India.

Sunder Singh
UG Student, Dept. of CS Engg.,
Sir M.Visvesvaraya Institute of
Technology, Bengaluru, India.

Toshit Mall
UG Student, Dept. of CS Engg.,
Sir M. Visvesvaraya Institute of
Technology, Bengaluru, India.

ABSTRACT:

One of the main causes of car accidents is drowsy driving. Using a combination of techniques for detecting eye closure and yawning, we provide a robust and intelligent strategy for detecting driver tiredness in this work to address this growing problem. This method involves installing a camera inside the car to record the driver's facial look. The first phase involves using computer vision algorithms to identify and track the face region in the recorded video sequence. The driver's state is finally assessed, and if drowsiness is found, a warning message and an alert are given to the driver. Our tests provide strong support for the proposed theory. The major goal of this research is to create a non-intrusive system that can recognize driver weariness and deliver a prompt warning.

I. INTRODUCTION:

A safety feature in automobiles called driver drowsiness detection prevents accidents by detecting when the driver is about to doze off. Most people close their eyes and sleep for 5–6 seconds. In this instance, is referring to complete slumber. On the other hand, micro sleep is seen when a motorist does a little period of sleeping (2-3 seconds). Driver sleepiness, in contrast to driver distraction, lacks a cause and is instead characterised by a gradual loss of focus on the road and traffic demands. However, both driver indolence and distraction may result in a reduction in attention, response time, psychomotor coordination, and information processing.

According to several studies, tiredness may be a factor for up to 48% of some specific types of streets and up to 22% of all vehicle related accidents. According to data from the Society of Automotive Engineers in the USA, there is one driver fatigue-related mortality for every eight people killed in traffic accidents.

The block diagram of the entire system is displayed in the above figure. Based on the video captured by the front-facing camera, real-time processing of an incoming video stream is done to determine the driver's degree of exhaustion. If the level of fatigue is determined to be high, the output is transmitted to the alarm, and the alarm system is triggered.

II. LITERATURE SURVEY:

In [5], an embedded system for detecting driver intoxication is presented in this paper. The authors suggest an effective and low-cost approach for measuring a driver's state of tiredness utilizing accelerometers, infrared sensors, and other gear connected to a microcontroller. The technology is made to be installed in cars as a preventative measure against collisions brought on by tired or intoxicated drivers.

In [6], the study introduces a new technique for detecting lane compliance and driver fatigue. The

authors offer a thorough overview of the available solutions to this issue as well as a thorough analysis of relevant works in the subject. Then, based on the idea of lane discipline detection and analysis, they suggest a novel strategy utilizing computer vision. Experiments in actual driving situations are used to test this method, and encouraging findings are given.

The study in [7] introduces a new technique for detecting lane compliance and driver fatigue. The authors offer a thorough overview of the available solutions to this issue as well as a thorough analysis of relevant works in the subject. Then, based on the idea of lane discipline detection and analysis, they suggest a novel strategy utilizing computer vision. Experiments in actual driving situations are used to test this method, and encouraging findings are given.

[8] In this paper, the possibility of detecting driver tiredness by combining yawning and eye closure detection is explored. The authors provide a cutting-edge technique that makes use of frequency-based features that are retrieved from video pictures captured by a webcam installed on a car's dashboard. They then use fuzzy logic techniques to accurately categorize each frame as either awake or asleep by fusing these traits with additional indicators like mouth movement and head position.

In order to assess a driver's level of sleepiness, the research in [9] introduces a revolutionary driver drowsiness detection method that integrates facial characteristics with eye closure time. The authors outline their suggested approach as well as the drawbacks of earlier approaches and how they were enhanced. Then they assess their strategy using multiple datasets and contrast it with other methodologies. They conclude by summarizing their research and making recommendations for further investigation.

This study in [10] describes an eye tracking-based driver tiredness monitoring and warning system. The system's design as well as its elements, such as an image capture device, an infrared illuminator source, an imaging lens, and a display unit, are discussed by the authors. They go over various methods for gathering images from the driver's eyes in order to look for indicators of weariness. Finally, they offer experimental findings that show how their system can detect driver sleepiness indications with high levels of accuracy.

Sl.No	Year	Author	Title	Method Used	Accuracy
1	2019	Naoto Nojiri, Xiangbo Kong, Lin Meng	Machine Learning and Deep Learning based Eye Status recognition for Driver Drowsiness	Deep learning methods like AlexNet and GoogleNet for classifying eye status.	85%
2	2019	Md. Yousuf Hossain, Fabian Parsia George	IOT Based Real-Time Drowsy Driving Detection System for the Prevention of Road Accidents.	The proposed system uses eye closure ratio as input parameter to detect the drowsiness of the driver.	NA
3	2021	M.S. Satyanarayana, T.M. Aruna, Y.K. Guruprasad	Continuous monitoring and identification of driver drowsiness alert system	EAR(Eye Aspect Ratio) and SMS System	90%
4	2022	Salah Eddine Bekhouche, Yassine Ruichek, Fadi Dornaika	Driver drowsiness detection in video sequences using hybrid selection of deep features.	A pretrained CNN model is used to extract more abstract and discriminative facial features.	Upto 82 %
5	2023	Hajime Kumagai, Kengo Kawaguchi, Hiroyuki Sawatari, Yuka Kiyohara, Mitsuo Hayashi, Toshiaki Shiomi.	Dashcam video footage-based analysis of microsleep-related behaviors in truck collisions attributed to falling asleep at the wheel.	PERCLOS (percentage of eyelid closure over the pupil over time) to measure driver drowsiness.	NA

II. IMPLEMENTATION:

We will be using a combined model which consists of Image acquisition, Image pre-processing and classification and Convolutional Neural Network (CNN).

In this work, step by step, we use the data strategybelow

1. Import the necessary libraries: cv2, numpy, dlib, and os.
2. Initialize the video capture instance and the face detector and landmark detector.
3. Create a function named compute that takes two points as input and returns the Euclidean distance between them.
4. Create another function named blinked that takes six points as input and returns 2 if the eyes are closed, 1 if the eyes are partially closed, and 0 if the eyes are open.
5. Inside a while loop, read the video frames and convert them to grayscale.
6. Detect faces in the grayscale image and for each face, draw a rectangle around it.
7. Detect the facial landmarks for the current face.
8. Compute the distance between the upper and lower lips and check if the person is yawning.
9. Determine the state of the person's eyes by calling the blinked function.
10. Update the status and color variables based on the state of the person's eyes and the distance between their lips.
11. Display the status and the facial landmarks on the video frames.
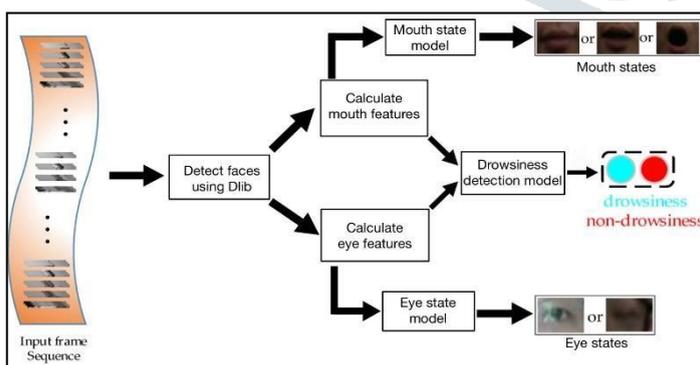12. Exit the loop when the user presses the "Esc" key.

Fig: System Architecture

To evaluate and analyze the performance of the proposed method, we performed many experiments on the NTHU dataset. Following are the detailed explanation of the implementation steps:

Importing Libraries
The implementation starts with importing necessary libraries. cv2 is imported for basic image processing functions, numpy for array related functions, dlib for deep learning-based Modules and face landmark detection, and face_utils for basic operations of conversion.

Initializing Camera and detectors
The VideoCapture() function from the cv2 library is used to initialize the camera instance. Then, dlib's get_frontal_face_detector() function and shape_predictor() function are used to initialize the face detector and landmark detector, respectively. The shape_predictor() function requires the path to the shape_predictor_68_face_landmarks.dat file, which is a pre-trained file for detecting 68 facial landmarks.

Defining Helper Functions
Two helper functions are defined to compute the distance between two points and to detect a blink. The compute() function calculates the Euclidean distance between two points in a 2D space. The blinked() function takes six facial landmarks as input and computes the ratio of the sum of the distances between the upper eyelid and the eyebrow and the lower eyelid and the cheek, to the distance between the horizontal midpoint of the eyes and the vertical midpoint of the face. Based on this ratio, the function returns 0 if not blinked, 1 if partially blinked, and 2 if fully blinked.

Processing Video Feed
The implementation then enters a while loop that continuously captures frames from the camera and processes them. cap.read() function reads a single frame from the camera. cv2.cvtColor() function is used to convert the frame to grayscale, which is required by the face detector.

Detecting Faces and Facial Landmarks
The detector() function from dlib detects faces in the grayscale frame. for loop is used to loop over all the detected faces. The top-left and bottom-right coordinates of each detected face are used to draw a rectangle around the face using cv2.rectangle() function. The predictor() function

from dlib is used to predict facial landmarks for each face. The facial landmark detection algorithm returns a set of (x, y) coordinates for 68 specific points on the face.

Detecting Mouth and Eyes

The shape_to_np() function from face_utils is used to convert the detected facial landmark points to a NumPy array. The coordinates of the upper and lower lips are extracted using the coordinates of the 62nd and 66th landmarks, respectively. The blinked() function is called twice to detect blinks in both eyes using the landmarks around the eyes.

Determining Status of Driver

Based on the number of blinks and the distance between the upper and lower lips, the program determines whether the driver is active, drowsy, or sleeping. If the distance between the lips is greater than 25, it is considered a yawn. The program keeps a count of the number of consecutive blinks and yawns and sets the status variable accordingly. A color is also assigned to the status, which is used to display the status text on the video feed.

Displaying Results

The putText() function from cv2 is used to display the status text on the video feed. The circle() function from cv2 is used to draw a circle around each facial landmark point. Finally, the original video feed and the processed video feed with the detected facial landmarks are displayed using imshow() function from cv2.

Exiting the Program

The program continues to loop until the user presses the escape key (ESC). Once the user presses the escape key(ASCII -27), the program comes to a halt.

Fig: Dataflow model
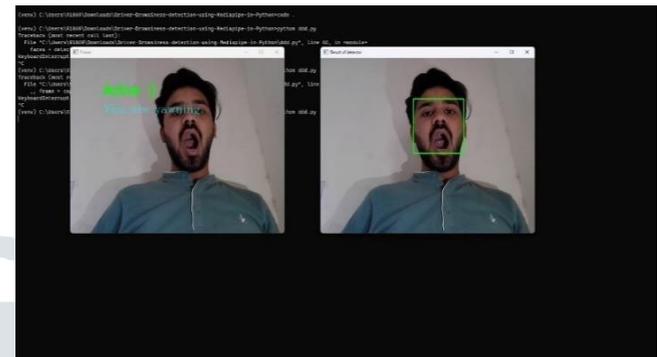

Fig: Active State

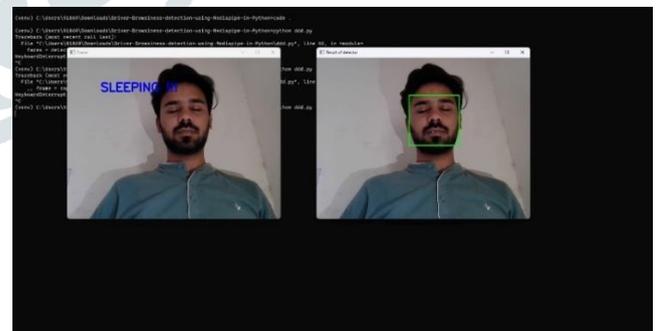Fig: Yawning

Fig: Drowsy State

Fig: Sleeping State

III. EXPERIMENTAL RESULT:

To evaluate and analyze the performance of the proposed method, we performed many experiments on the dataset [14]. This dataset contains 20 subjects from different ethnicities,wearing or not wearing accessories and shakiness from the camera. The video sequences
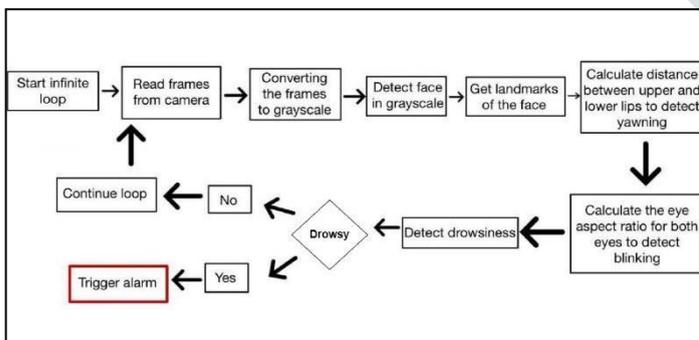
have been recorded under day and night conditions. The test provided an accuracy of 98%

IV. CONCLUSION:

Potential tiredness concerns can be averted by monitoring the driver's behavioural features and assessing their level of consciousness. To make the system more robust, both yawning and closed-eye recognition approaches are implemented, while the algorithms are basic and suited to commercial applications.

V. REFERENCES:

[1]   Machine Learning and Deep Learning based Makeup Considered Eye Status Recognition for Driver Drowsiness,Procedia Computer Science,Volume 147,2019,Pages 264-270,ISSN 1877-0509,Machine Learning and Deep Learning based Makeup Considered Eye Status Recognition for Driver Drowsiness,Procedia Computer Science,Volume 147,2019,Pages 264-270,ISSN 1877-0509,

[2]   M. Hossain and F. George, "IOT Based Real-Time Drowsy Driving Detection System for the Prevention of Road Accidents," in 2018 International Conference on Intelligent Informatics and Biomedical Sciences (ICIIBMS), Bangkok, Thailand, 2019 pp. 190-195. doi: 10.1109/ICIIBMS.2018.8550026

[3]   M.S. Satyanarayana, T.M. Aruna, Y.K. Guruprasad, Continuous monitoring and identification of driver drowsiness alert system, Global Transitions Proceedings,Volume 2, Issue 1,2021,Pages 123-127,ISSN 2666-285X,

[4]   Salah Eddine Bekhouche, Yassine Ruichek, Fadi Dornaika, Driver drowsiness detection in video sequences using hybrid selection of deep features, Knowledge-Based Systems,Volume 252, 2022, 109436, ISSN 0950-7051,

[5]   Hajime Kumagai, Kengo Kawaguchi, Hiroyuki Sawatari, Yuka Kiyohara, Mitsuo Hayashi, Toshiaki Shiomi,Dashcam video footage-based analysis of microsleep-related behaviors in truck collisions attributed to falling asleep at the wheel,Accident Analysis & Prevention,Volume 187,2023,107070,ISSN 0001-4575

[6]   Tianyi Hong and Huabiao Qin, "Driver's drowsiness detection in embedded system," 2007 IEEE International Conference on Vehicular Electronics and Safety, 2007, pp. 1-5, doi: 10.1109/ICVES.2007.4456381.

[7]   Y. Katyal, S. Alur and S. Dwivedi, "Safe driving by detecting lane discipline and driver drowsiness," 2014 IEEE International Conference on Advanced Communications, Control and Computing Technologies, 2014, pp. doi:10.1109/ICACCCT.2014.7019248.

[8]   J. W. Baek, B. -G. Han, K. -J. Kim, Y. -S. Chung and S. -I. Lee, "Real-Time Drowsiness Detection Algorithm for Driver State Monitoring Systems," 2018 Tenth International Conference on Ubiquitous and Future Networks (ICUFN), 2018, pp. 73-75, doi: 10.1109/ICUFN.2018.8436988.

[9]   M. Omidyeganeh, A. Javadtalab and S. Shirmohammadi, "Intelligent driver drowsiness detection through fusion of yawning and eye closure," 2011 IEEE International Conference on Virtual Environments, Human-Computer Interfaces and Measurement Systems Proceedings, 2011, pp. 1-6, doi: 10.1109/VECIMS.2011.6053857.

[10]   B. Alshaqaqi, A. S. Baquhaizel, M. E. Amine Ouis, M. Boumehed, A. Ouamri and M. Keche, "Driver drowsiness detection system," 2013 8th International Workshop on Systems, Signal Processing and their Applications (WoSSPA), 2013, pp. 151-155, doi: 10.1109/WoSSPA.2013.6602353.

[11]   H. Singh, J. S. Bhatia and J. Kaur, "Eye tracking based driver fatigue monitoring and warning system," India International Conference on Power Electronics 2010 (IICPE2010), 2011, pp. 1-6, doi: 10.1109/IICPE.2011.5728062