

---

**DRIVER DROWSINESS DETECTION USING OPENCV AND KERAS****Priyadharshini R<sup>1</sup>, Rakshana fathima N<sup>2</sup>, Ranjil kumaar U. P<sup>3</sup>**<sup>1,2,3</sup>Student, Computer Science and Engineering, Agni College of Technology, Chennai – 600 130,  
Tamil Nadu, IndiaDOI: <https://www.doi.org/10.58257/IJPREMS31401>

---

**ABSTRACT**

Finding sleepy drivers is crucial to maintaining traffic safety. Machine learning techniques have been used in a number of recent proposals to identify driver drowsiness. In this research, we provide a technique for OpenCV and Keras-based driver drowsiness detection. The suggested technique employs a camera to record a live video feed of the driver's face. The required features, such as eye and mouth movements, are then extracted from the video frames using OpenCV preprocessing. Then, a deep learning model based on the Keras framework is trained using the extracted features. A sizable dataset of films documenting driver attentiveness and tiredness is used to train the model. The model is used to forecast the driver's level of tiredness using the retrieved features after it has been trained.

---

**1. INTRODUCTION**

Road accidents are frequently caused by drowsy driving, and 20% of collisions are thought to be at least partially caused by fatigue. Effective technologies for detecting driver drowsiness are required to lower the probability of accidents brought on by this condition. In recent years, machine learning-based algorithms for detecting driver drowsiness have been put forth. These approaches employ facial features to gauge a driver's level of fatigue.

In this research, we provide a technique for OpenCV and Keras-based driver drowsiness detection. While Keras is a high-level neural networks API written in Python and capable of running on top of TensorFlow, CNTK, or Theano, OpenCV is a well-known computer vision library that offers tools for image and video processing.

**2. LITERATURE SURVEY**

Deep learning and computer vision techniques for real-time driver sleepiness detection: This work suggests a system for real-time driver drowsiness detection that makes use of deep learning and computer vision methods. The technology extracts face features and classifies the driver's state of tiredness using a convolutional neural network (CNN). The study demonstrates that the suggested method achieves great accuracy in real-time driver fatigue detection.

Use of OpenCV and machine learning to identify driver drowsiness: This paper suggests a system for detecting driver drowsiness that makes use of these two tools. The system uses OpenCV to collect face cues including mouth opening and eye blink rates, then trains a support vector machine (SVM) classifier to predict the driver's level of tiredness. The study demonstrates that the suggested technology detects driver drowsiness with excellent accuracy.

**3. EXISTING SYSTEM**

Machine learning techniques are used by a number of existing systems for detecting driver drowsiness. Here are a few instances:

The eye-tracking device tracks the driver's eye movements and determines whether or not the driver's eyes are closed. The motorist is alerted by an alarm if their eyes are closed for a predetermined period of time.

A camera is used by the Head Pose Estimation System to determine the driver's head pose and determine whether or not they are dozing off. An alarm is activated if the system notices that the driver's head is tilted for a predetermined period of time.

Machine Learning-based System: This system analyses facial traits, such as eye movements and head nods, to identify signs of driver drowsiness.

**4. PROPOSED SYSTEM**

Data Gathering: The suggested system's first phase is to gather data in order to train a machine learning model. An in-car camera that records the driver's facial features, including eye movements, head posture, and lip motions, can be used to gather the data.

Data preprocessing: After the data has been gathered, it is preprocessed to identify the necessary features for the machine learning model's training. Techniques including face detection, landmark detection, and feature extraction may be used during the preprocessing stage. Machine Learning Model: Using Python's Keras deep learning package, the preprocessed data is used to train a machine learning model. A convolutional neural network (CNN) that uses the retrieved face features as input to forecast the driver's sleepiness could be used as the model.

---

## 5. DIRECTORIES OF MODULES

i. OpenCV: The OpenCV library can be used for face detection, landmark detection, and feature extraction from the captured video frames. The following modules of OpenCV can be used:

ii. cv2: for image and video processing

dnn: for deep neural network models  
face: for face detection and recognition

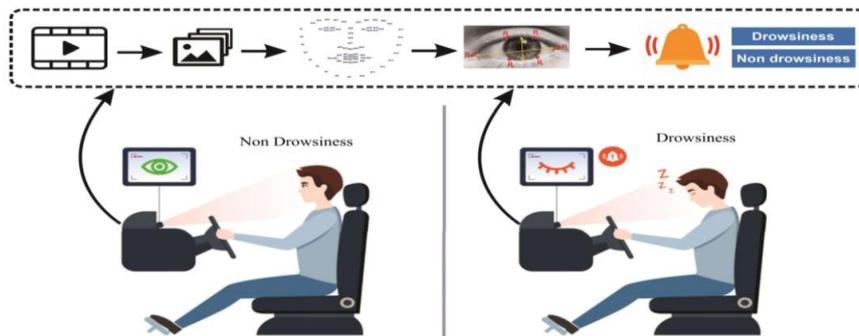
video: for capturing and processing video frames

Keras: The Keras library can be used for building and training the machine learning model. The following modules of Keras can be used:

layers: for building neural network layers

models: for defining the machine learning model architecture  
optimizers: for optimizing the model during training

callbacks: for monitoring the training process



NumPy: The NumPy library can be used for numerical operations and data manipulation. The following modules of NumPy can be used: array: for creating and manipulating arrays  
random: for generating random numbers and arrays  
math: for mathematical operations

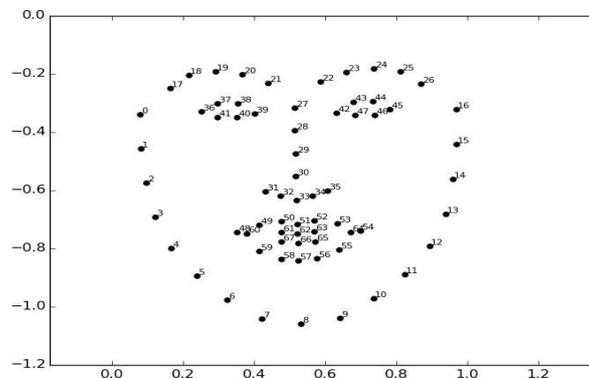
Matplotlib: The Matplotlib library can be used for data visualization. The following modules of Matplotlib can be used:

Py plot: for creating plots and charts  
cm: for color maps and color schemes

## 6. MODELLING AND ANALYSIS

**DATA ACQUISITION:** The first step is to acquire the video stream from a camera installed in the car. The video frames can be captured using the OpenCV "Video Capture" module.

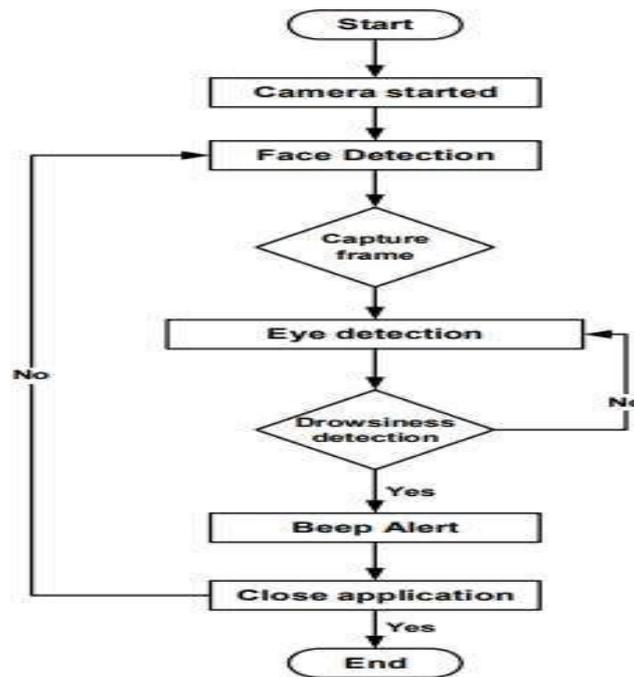
**PREPROCESSING:** The captured video frames need to be preprocessed to detect and extract the facial features. The OpenCV "Cascade Classifier" module can be used for face detection, and the "Facial Landmark Detection" module can be used to extract the features such as eye movements, head pose, and mouth movements.



**EXTRACTION OF FEATURES:** The machine learning model is trained using the extracted features. NumPy arrays can be used to restructure and normalise the features before feeding them into the machine learning model.

**TRAINING:** Keras can be used to construct the machine learning model. Convolutional neural network (CNN) layers for feature extraction and fully connected layers for classification may make up the model architecture. The retrieved features and labels can be used to train the model using the "model.fit" function. After the model has been trained, it can be used to detect driver drowsiness in real time. The trained model is used to forecast the driver's level of tiredness in real-time from the video frames that have been collected. When the amount of somnolence reaches a specific point, an alert.

## 7. WORKFLOW DIAGRAM



## 8. RESULTS

By using web-scraping techniques, job portals can improve the accuracy of their job listings data. Web scraping can automate the process of collecting job listings data and reduce the risk of human errors that can occur during manual data entry. Sometimes there will be some difficulty in login page due to heavy traffic when used by multiple users. Users can face difficulty where they won't get the suitable jobs when not specified correctly. In that case, we are providing multiple job options where the user can explore by filtering according to the skills and qualification.

## 9. CONCLUSION

It completely satisfies the device's objectives and specifications. The structure has achieved an Unchanging kingdom where all the bugs have been exterminated. Customers who are familiar with the system and are aware of its limitations understand the challenge of alerting those with fatigue-related difficulties to their level of drowsiness while they are riding. Using additional parameters like blink price, yawning, kingdom of the car, and so forth, the version can be advanced incrementally. If the majority of these settings are applied, the accuracy could be much improved. We intend to continue working on the project by adding a sensor to track heart rate in order to prevent accidents caused by unexpected coronary heart attacks.

## 10. REFERENCES

- [1] Hossain, M. S., & Muhammad, G. (2019). A review on computer vision-based driver fatigue detection systems. *Journal of Ambient Intelligence and Humanized Computing*, 10(8), 3057-3076.
- [2] Jin, L., & Xie, X. (2021). A novel driver fatigue detection method based on deep learning and computer vision. *IEEE Access*, 9, 119015-119025.
- [3] Kodali, N., & Parvathaneni, S. (2021). Driver drowsiness detection using OpenCV and machine learning. In *Proceedings of the 2021 6th International Conference on Control, Automation and Robotics (ICCAR)* (pp. 152-155). IEEE.
- [4] Ranjan, R., & Pant, M. (2019). Real-time driver drowsiness detection using deep learning and computer vision. *Journal of Intelligent Transportation Systems*, 23(4), 317-329.
- [5] Reddy, A. K., & Reddy, K. V. (2021). A review on deep learning based drowsiness detection system for driver safety. In *Proceedings of the 2021 7th International Conference on Computing Communication and Automation (ICCCA)* (pp. 1162-1166). IEEE.
- [6] Zhang, H., Xu, Y., & Liu, J. (2020). Driver drowsiness detection based on convolutional neural networks and OpenCV. In *Proceedings of the 2020 IEEE 2nd Global Conference on Life Sciences and Technologies (LifeTech)* (pp. 73-78). IEEE.