

Safety Enhancement In Vehicles: Drowsiness Detection Using Convolutional Neural Networks

Mohammed Abdul Bari¹, Ahmed Sayeed Bahattab², Madiha Khulood³, Summayya Begum⁴

^{1,2,3} B.E. Student, Department of IT, Lords Institute of Engineering and Technology, Hyderabad

⁴ Assistant Professor, Department of IT, Lords Institute of Engineering and Technology, Hyderabad

sumayyabegum@lords.ac.in

ABSTRACT

Advancements in computer vision have significantly benefited drivers, particularly through technologies like self-driving cars. Approximately 20% of accidents result from driver fatigue and drowsiness, posing a serious issue. While numerous solutions have been proposed to address this, many are unsuitable for real-time application. These methods struggle with challenges such as variations in facial features and changes in lighting conditions. Our goal is to implement an intelligent system that significantly reduces road accidents by monitoring key facial features of the driver, such as eye closure, blinking rate, yawning, and head movements. This system continuously observes the driver using a webcam. Facial features, including eye movements, are tracked using a cascade classifier. Images of the eyes are extracted and processed through a custom-designed Convolutional Neural Network (CNN) to determine whether both eyes are closed. Based on this classification, an eye closure score is generated, and if the system detects signs of drowsiness, an alarm is triggered to alert the driver.

I. INTRODUCTION

A variety of safety mechanisms have been developed to minimize the risk of accidents on the road, particularly in vehicles, with fatigue being recognized

as one of the most significant contributors to collisions. Research has indicated that driver drowsiness is responsible for a considerable percentage of road accidents. A leading automobile manufacturer reported that up to 17% of fatal accidents are attributed to drowsy drivers. Furthermore, studies conducted by Volkswagen AG revealed that between 5% to 25% of all fatal accidents occur due to drivers falling asleep at the wheel. Drowsiness not only impairs attention and decision-making abilities but also reduces a driver's reaction time, increasing the likelihood of accidents.

Recognizing this growing issue, there is a critical need for an intelligent system that can actively monitor and detect driver fatigue to help reduce the occurrence of road accidents. Our project focuses on developing a sophisticated driver drowsiness detection system that can continuously assess signs of fatigue and alert the driver in real-time, potentially preventing fatal outcomes. The system leverages Convolutional Neural Networks (CNN) to analyze key indicators of drowsiness, such as eye movements, blinking rates, and facial cues, ensuring the driver remains alert.

Previous research has identified three major factors for detecting driver drowsiness: vehicle-based, behavioral, and physiological measures. However, many of these studies have struggled to implement their findings in real-time due to



limitations in data processing and adaptability. Our approach addresses these shortcomings by proposing a reliable, real-time system that accurately classifies whether a driver is sleepy or not, helping to ensure safer driving environments and significantly reducing the risk of accidents.

II. LITERATURE SURVEY

A basic approach to detecting eyes [5] focuses on measuring variations in concentration within the eye area. However, it neglects other key movements, such as eye blinking, yawning, and head rotation, making it unsuitable for real-time scenarios. The method also utilizes a two-dimensional Convolutional Neural Network (CNN) [2] to detect movement, but it overlooks the importance of three-dimensional sequential associations. It relies on replicated datasets and disregards head movement. Three-dimensional temporal data [8] has been integrated into the detection of movement directions, and a deep residual CNN model [3] has been applied to classify eye movements. However, this CNN is only trained on a single image-based approach. GoogleNet [9] focuses primarily on eye opening, using frame-by-frame predictions, which results in high memory consumption.

Another approach analyzes head rotation, eye conditions, and mouth movements [4], though it ignores sequential dependencies. Instead of using raw pixel values as input [1], specific facial features are reduced, but the process of feature extraction is not clearly explained. Sequential dependencies [10] are crucial for learning across multiple electrodes and incorporating three Dimensional input movements. Electroencephalography (EEG) signals have

struggled to develop effective algorithms for detecting fatigue, and CNN with data augmentation [7] and combined datasets deliver slightly lower accuracy.

Traditional methods for detecting drowsiness focus on gestures related to muscular, cognitive, and cardiovascular states, while other techniques monitor overall driver performance based on vehicle data. A third approach uses vision-based methods as a non-invasive way to detect driver fatigue. These methods often use datasets trained on sleep-deprived drivers, with frameworks designed to capture eye movements and other facial features. However, typical frameworks rely heavily on blink detection, often overlooking other informative facial cues. Additionally, modern facial detection techniques, such as those using the dlib library, suffer from standardization issues, impacting their ability to reliably detect sleepiness.

A basic approach to detecting eyes [5] focuses on measuring variations in concentration within the eye area. However, it neglects other key movements, such as eye blinking, yawning, and head rotation, making it unsuitable for real-time scenarios. The method also utilizes a two-dimensional Convolutional Neural Network (CNN) [2] to detect movement, but it overlooks the importance of three-dimensional sequential associations. It relies on replicated datasets and disregards head movement. Three-dimensional temporal data [8] has been integrated into the detection of movement directions, and a deep residual CNN model [3] has been applied to classify eye movements. However, this CNN is only trained on a single image-based approach. GoogleNet [9] focuses primarily on eye opening, using frame-by-frame predictions, which results in high memory

consumption.

Another approach analyzes head rotation, eye conditions, and mouth movements [4], though it ignores sequential dependencies. Instead of using raw pixel values as input [1], specific facial features are reduced, but the process of feature extraction is not clearly explained. Sequential dependencies are crucial for learning across multiple electrodes and incorporating three-

III.PROPOSED SYSTEM

The proposed system aims to provide an effective solution for detecting driver drowsiness in real-time by overcoming the limitations of traditional methods that relied on manually selected, hand-crafted features. In contrast to older approaches, this system uses custom-designed Convolutional Neural Network (CNN) to process live images of the driver, enabling continuous monitoring through a webcam. The webcam captures video footage of the driver, which is then broken down into individual frames for analysis. Each frame undergoes a detection process using pre-trained classifiers, such as the Haar Cascade Classifiers available in OpenCV, to identify the face and eyes. Once the facial regions, specifically the eyes, are detected, the system extracts eye images from the frames. These eye images are then fed into a CNN, which is comprised of several 2D convolutional layers with kernel sizes of 5x5 and 3x3, and valid padding.

The CNN also includes max-pooling layers (2x2) to reduce the spatial dimensions of the data, followed by fully connected layers that classify whether the eyes are open or close. The system continuously tracks eye movements and calculates an eye closure score.

When the CNN detects that the driver's eyes have been closed for 15 consecutive frames, it raises an alarm to notify the driver of potential drowsiness. This early warning system helps prevent accidents by alerting the driver before a dangerous situation arises. One of the key advantages of this system is its ability to automatically extract and learn features from the data without the need for manual intervention, making the detection of drowsiness more accurate and efficient. The custom CNN is designed to address the standardization problems found in existing systems, ensuring reliable and consistent detection of driver fatigue. By incorporating advanced feature extraction and classification, the system delivers a robust, real-time solution that significantly enhances road safety by reducing the risk of accidents caused by driver fatigue.

System Architecture

Figure 1 illustrates the complete flow of the system designed for detecting driver drowsiness.

In the initial phase, the driver is continuously monitored via a webcam that captures live video footage of the driver's face. This video input is then broken down into individual frames, allowing for further analysis. For each frame, the driver's face and eyes are detected using Haar Cascade classifiers. Which are widely used for object detection, particularly for recognizing human faces and features. Once the eyes are successfully identified in each frame, they are saved as individual images, forming the basis of the dataset required for training the Convolutional Neural Network (CNN).

The next step in the process involves preparing the eye dataset for training the CNN model. To enhance the size and diversity of the dataset, data



augmentation is applied, which involves techniques such as flipping, rotating, and zooming in on the images. This step ensures that the CNN is well-equipped to handle various lighting conditions, head orientations, and other variations in real-world scenarios.

Once the dataset is prepared, the system performs a series of image preprocessing steps on the captured images of the left and right eyes. These steps include converting the images to grayscale, resizing them to a uniform resolution, and normalizing the pixel values. This preprocessing step is crucial for reducing computational complexity and improving the CNN's performance, as the model will have to process a more consistent and simplified dataset.

Following preprocessing, the images are fed into a pre-trained CNN model, which is composed of multiple layers, including convolutional layers, max-pooling layers, and fully connected (dense) layers. The convolutional layers extract key features from the eye images, such as patterns related to eye closure. The max-pooling layers reduce the dimensionality of the data, preserving the most important features while reducing computational load. The dense layers then perform the final classification task, predicting whether the driver's eyes are open or closed.

Throughout the process, the system calculates a score based on the driver's eye closure rate.

If the system detects that the driver's eyes remain closed for a certain number of consecutive frames (typically 15), it concludes that the driver is drowsy. In such a case, the system triggers an alarm to alert the driver, helping to prevent potential accidents caused by fatigue.

This system architecture ensures that the driver is

monitored continuously in real-time, with the CNN-based model providing accurate and efficient detection of drowsiness. By addressing key challenges such as lighting conditions, image

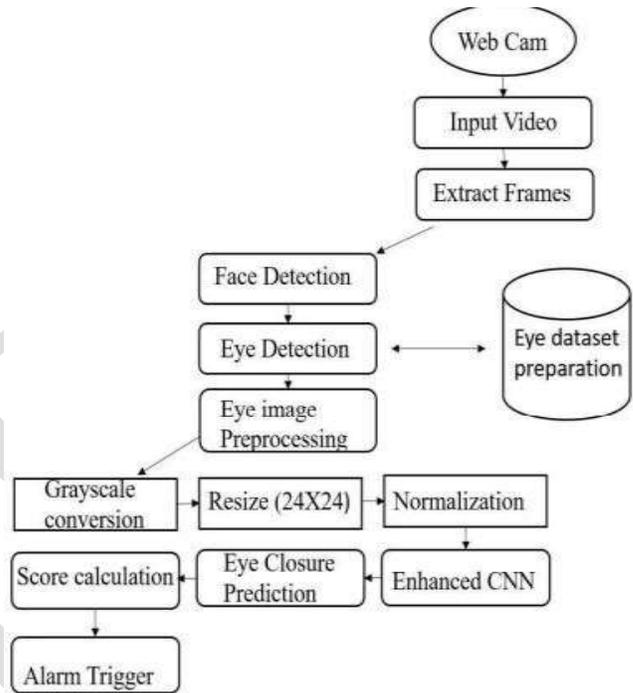


Fig. 1 Proposed System Architecture of CNN

variations, and driver movement, the system offers a robust solution to enhance road safety and reduce the risk of accidents due to driver fatigue.

Implementation description

The system designed for detecting driver sleepiness utilizes a combination of several advanced technologies and methods. It involves components such as eye detection and face recognition, data augmentation, an enhanced Convolutional Neural Network (CNN), and a triggering alarm mechanism.

Eye and Face Detection:

For the eye and face detection component, the system leverages pre-trained models provided by OpenCV.



These models are accessed through OpenCV's loading method, specifically using Haar cascade classifiers. These classifiers are pre-trained on a set of images and stored in XML files within the OpenCV directory. The system utilizes these Haar cascade models to identify and detect eyes in images. The detection process involves using a cascade classifier, which applies a series of increasingly complex classifiers to the input image to detect the presence of eyes. The multiscale method is then employed to draw bounding rectangles around detected eyes, allowing the system to effectively locate and isolate the eyes within the image.

Dataset Preparation:

The dataset for the system is prepared through a series of preprocessing and labeling steps. Eye images are captured using a webcam and are categorized based on their state: open or closed. The captured images are converted to grayscale to reduce computational complexity and focus on the relevant features. These images are then resized to a fixed dimension of 24x24 pixels to standardize the input size. Normalization is applied to ensure that pixel values are scaled to a consistent range, which aids in improving the performance of the CNN.

Data Augmentation:

To enhance the robustness and diversity of the dataset, data augmentation techniques are employed. Using Keras, the system applies a range of transformations to the existing data, including rotations, brightness adjustments, shearing, and zooming. These augmentations help in simulating various real-world conditions and improve the generalization capability of the model by creating multiple variations of the original images.

Convolutional Neural Network (CNN) Model:

The CNN model used in this system consists of multiple layers with configurable parameters designed to process the input eye images. The convolutional layers utilize the ReLU (Rectified Linear Unit) activation function with a kernel size of 3x3, which helps in extracting features from the images. Following the convolutional layers, the fully connected layers use the Softmax activation function to classify the eye states as either open or closed.

The system also integrates dropout, dense, and activation layers to enhance the learning efficiency and prevent overfitting. Dropout layers are used to randomly disable neurons during training, which helps the model generalize better. Dense layers are fully connected layers that learn high-level features and make predictions based on the extracted features.

Triggering and Feedback Mechanism:

To alert the driver when signs of drowsiness are detected, the system utilizes the Pygame library. This library is used to provide real-time feedback to the driver through visual or auditory signals when the system detects that the driver is starting to fall asleep.

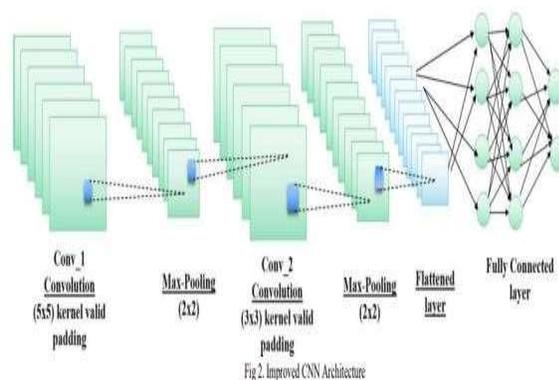


Fig 2. Improved CNN Architecture

The system calculates a score based on the duration for which the driver’s eyes are closed. The score increases when both eyes are closed, indicating higher drowsiness, and decreases when the eyes are open. This score is used to display the driver’s current alertness level.

Training and Evaluation:

The CNN model is trained over 10 epochs, with performance metrics such as training accuracy and loss being monitored throughout the training process. The training process is aimed at fine-tuning the model’s parameters to achieve optimal performance. The results of the training, including accuracy and loss, are illustrated in figures that provide insight into the model’s learning progress and effectiveness

IV.RESULT AND DISCUSSION

The examination of the appearance finding methods is shown in Table I.

The precision swiftness of Figure 5 and loss of Figure 6 is faster than expectable training. This paper uses amplified database in order to get train and for open and close eye images it uses original database that is available for testing

TABLE I. FACE RECOGNITION SYSTEMS INVESTIGATION

Techniques	Features count	Dataset	Accuracy
Geometric	38400	47	90
Mixture-Distance	23800	685	94
Eigen faces	26400	860	95
CV_DNN	22500	880	97.05
Enhanced DNN	30800	1000	99.10

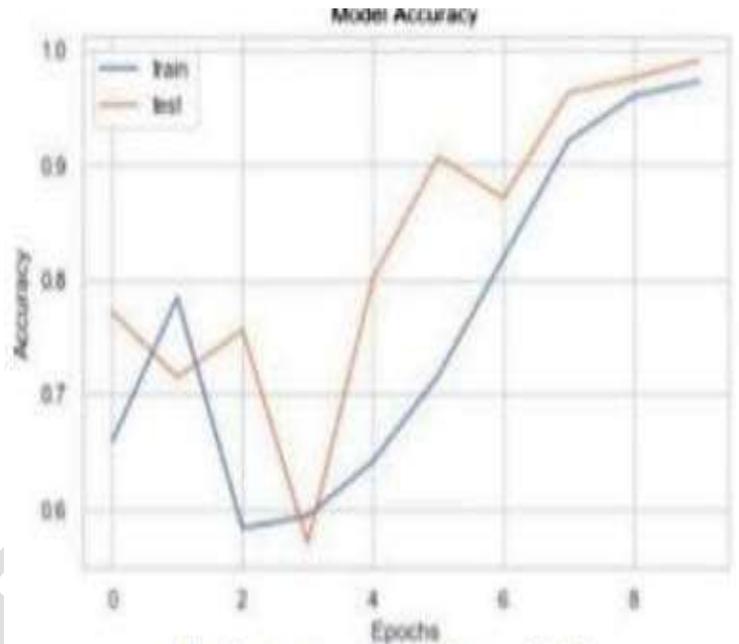


Fig. 5. Accuracy of model Improved CNN

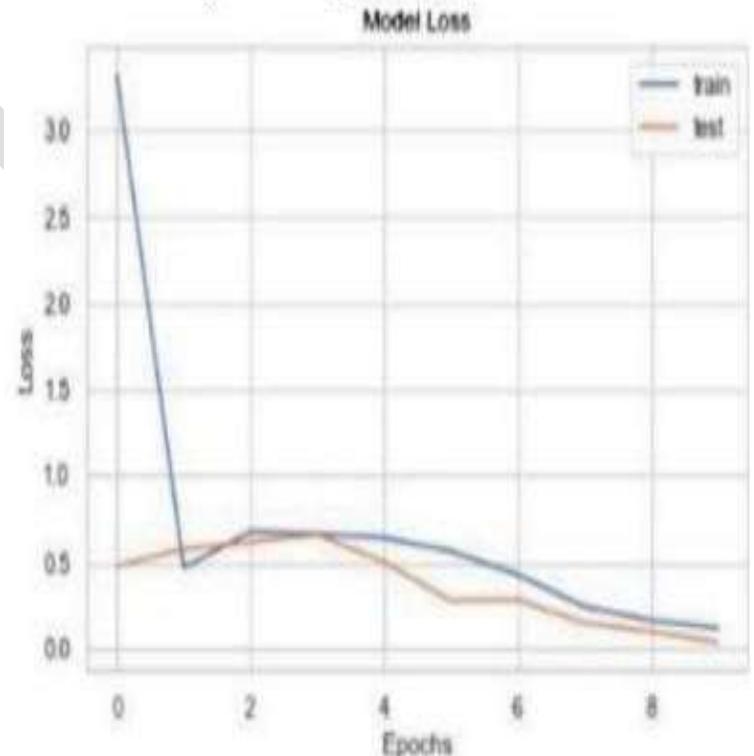


Fig. 6. Loss of model Improved CNN

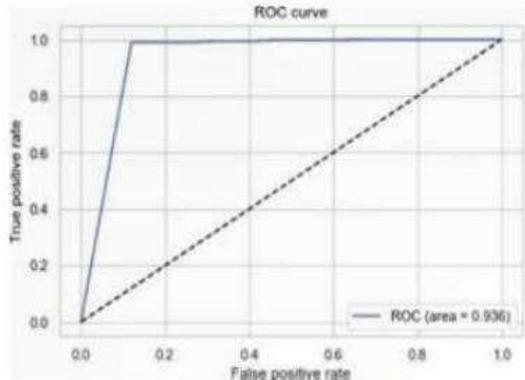


Fig. 7. ROC curve of Improved CNN

Table II shows the count of pictures in closing the eye is less than the count of opening the eye.

TABLE IV: EPOCH-10 PRECISION

Techniques	Epoch-10
ResNet	20.42
AlexNet	23.17
VGGNet	20.85
ProposedNet	11.39

TABLE II. DATABASES - ORIGINAL AND AUGMENTED IMAGES

Image Types	Original Database	Augmented Database
Eye opened	4563	24680
Eye closed	3867	20430

TESTING

After training the CNN, we proceeded to test the model using a dataset of 1,000 images, which

includes both open and closed eyes. The results of this testing are summarized in Table III. The paper evaluates the proposed sleepiness detection technique using various parameters. Specifically, precision is defined as the ratio of true positive detections to the total number of detections made by the model.

TABLE III: ORGANIZATION STATEMENT

Average	Precision	Recall	f1-score	Support
0.0	1.0	0.34	0.51	500
1.0	0.60	1.00	0.75	500
Accuracy	-	-	0.67	1000
Macro avg	0.80	0.67	0.63	1000
Weighted avg	0.80	0.67	0.63	1000

Table IV displays the investigation completed on ResNet, AlexNet, VGGNet and ProposedNet of images for Epoch-10.

The confusion matrix of real and untrue-positive ratings: Array([[169,331],[0,500]], dtype=int64) The Receiver Operating Characteristic curve is plotted as shown in Figure 7 denote testing information with the suggested technique built on the quantity of training periods.

Figure 8 Shows the face and eye identifying using haar classifier.



Fig. 8. Detection of face and eye



Figure 9 shows that both right and left eyes are fully open. Hence, it is concluded that the person is in alert state



Fig. 9. Alert state

Figure 10 shows that both right and left eyes are partially closed. Hence we can say that the person is in drowsy state.



Fig. 10. Drowsy state

V. CONCLUSION

The proposed model for detecting driver sleepiness relies on an effective Convolutional Neural Network (CNN) architecture specifically designed to identify signs of drowsiness based on eye closure. The development process involved creating a dataset comprising images of both open and closed eyes. To train and evaluate the CNN, 75% of the dataset was used for training, while the remaining 25% was reserved for testing.

Initially, video data was converted into individual frames, and face and eye detection was performed on

each frame. The enhanced CNN was then trained to recognize and classify eye movements accurately. The system is programmed to trigger an alarm if the driver's eyes remain closed for 15 consecutive frames, providing timely alerts to prevent potential accidents due to drowsiness.

The CNN model achieved a high accuracy rate of approximately 97% in detecting eye states. However, the testing precision was 67%, indicating some variability in the model's performance during evaluation. This demonstrates the model's strong capability in recognizing drowsiness while also highlighting areas where further refinement may be needed to improve precision.

VI. REFERENCES

- [1] Gupta, P., Saxena, N., Sharma, M., & Tripathi, J. (2018). *Deep Neural Network for Human Face Recognition*. International Journal of Engineering and Manufacturing, 8(1), 63-71.
- [2] Jeyasekar, A., & Iyengar, V. R. (2019). *Driver Drowsiness Detection Based on Behavioral Changes Using ResNet*. International Journal of Recent Technology and Engineering (IJRTE), 8(3), 25-30.
- [3] Devi, M. S., & Bajaj, P. R. (2008). *Driver Fatigue Detection Based on Eye Tracking*. Proceedings of the First International Conference on Emerging Trends in Engineering and Technology, 1, 649-652.
- [4] Park, S., Pan, F., Kang, S., & Yoo, C. D. (2016). *Driver Drowsiness Detection System Based on Feature Representation Learning Using Various Deep Networks*. In Computer Vision ACCV 2016 Workshops. Springer International Publishing.



- [5] Ahmed, T. U., Hossain, S., Hossain, M. S., Islam, R. U., & Andersson, K. (2019). *Facial Expression Recognition Using Convolutional Neural Networks with Data Augmentation*. Proceedings of the International Conference on Informatics, Electronics & Vision (ICIEV), Washington, USA.
- [6] Sinha, U., Mehta, K. K., & Shrivastava, A. K. (2016). *Real-Time Implementation for Monitoring Drowsiness Condition of a Train Driver Using Brain Wave Sensors*. International Journal of Computer Applications, 139(9), 25-30.
- [7] Ma, X., Chau, L. P., & Kim, J. (2017). *Driver Fatigue Detection Using Two-Stream Convolutional Neural Networks*. IEEE International Conference on Orange Technologies (ICOT), 155-158.
- [8] Zhang, W., & Su, J. (2017). *Driver Yawning Detection Based on Long Short-Term Memory Networks*. IEEE Symposium Series on Computational Intelligence (SSCI), Honolulu, HI, USA.
- [9] Gao, Z., Wang, X., Yang, Y., Mu, C., Cai, Q., Dang, W., & Zuo, S. (2019). *EEG-Based Spatial-Temporal Analysis for Driver Drowsiness Detection*. IEEE Transactions on Neural Networks and Learning Systems, 30(9)