Electronic Integrated Computer Algorithm Journal
Vol. 3, No. 1, October 2025, pp 28~32

ISSN: 3031-0350, DOI: 10.62123/enigma.v3i1.109

Implementation of a Drowsiness Detection System in Four-Wheel Vehicle Drivers Using OpenCv

Farhan Riqi Ma'ajid [1], Al-Khowarizmi[1*]

[1] *Department of Information Technology, Universitas Muhammadiyah Sumatera Utara*, Indonesia.

DOI : 10.62123/enigma.v3i1.109

Received : September 07, 2025
Revised : October 05, 2025
Accepted : October 12, 2025

Keywords:
Drowsiness Detection, EAR , CNN, MediaPipe FaceMesh, OpenCV

ABSTRACT

Drowsiness while driving is one of the triggers of traffic accidents. This study proposes a non-invasive and economical computer vision-based real-time drowsiness detection system. The system combines Eye Aspect Ratio (EAR) to assess eye openness, Convolutional Neural Network (CNN) for open/closed eye classification, and MediaPipe FaceMesh for stable facial landmark extraction. The dataset is taken from Kaggle (Open and Closed classes, totaling 1,452 images) and processed through grayscale conversion, normalization, 64×64 pixel resizing, and augmentation. Drowsiness detection is triggered when EAR <0.25 and CNN classifies both eyes as closed for ±2 consecutive seconds; visual/audio alarms are automatically activated. Test results on 218 images show excellent performance with only 1 misclassification (≈99.5% accuracy), with no false alarms for the open eye class. The system is implemented as a Flask-based web application for easy cross-device access. These findings demonstrate an efficient visual approach that is feasible to be integrated as a driving safety feature.

1. INTRODUCTION

Driver drowsiness is a major contributing factor to traffic accidents [1][2][3], particularly in private vehicles and long-distance transportation. According to data from the WHO and the Indonesian National Police Traffic Corps (Korlantas Polri), the percentage of accidents caused by fatigue and drowsiness can reach more than 20% of total road accidents [4]. This situation highlights the urgency of a drowsiness detection system that is not only accurate but also practically applicable.

Various drowsiness detection methods have been developed [5][6][7][8]. They are broadly divided into three approaches: (1) physiological methods using EEG/ECG sensors, which offer high accuracy but are invasive and expensive; (2) behavioral-based methods such as monitoring head direction and blink duration, which are more practical but prone to error; and (3) computer vision-based methods, which are increasingly popular because they are non-invasive, inexpensive, and easy to integrate with standard cameras [9].

Several previous studies have used the Haar Cascade method to detect eyes [10][11], but this method is prone to failure in poor lighting conditions or changes in facial angle [12][13]. Another alternative is the use of CNNs purely for eye classification [14][15][16], but this method requires a large dataset to avoid overfitting. Therefore, this study combines two approaches: EAR to monitor changes in eye geometry and CNN to strengthen the classification results. The addition of MediaPipe FaceMesh as a replacement for Haar Cascade provides superior accuracy in eye landmark detection [17][18], resulting in more stable EAR. By combining these methods, this study aims to produce a drowsiness detection system that: (1) can operate in real time, (2) remains computationally lightweight so it can run on ordinary devices, and (3) has high accuracy with a low false alarm rate.

2. METHODOLOGY

2.1 Dataset and Pre-Management

The data used comes from the Drowsiness Dataset (Kaggle), which consists of two classes: open eyes and closed eyes, with 726 images each. A total of 1,452 images were separated into training data (80%) and test data (20%). Examples of sample images from the dataset are shown in the following figure.

*Corresponding Author Email: alkhowarizmi@umsu.ac.id

Figure 1. Visualization of Sample Images

Each image underwent a preprocessing process, including:

1. Converting color images to grayscale to reduce complexity.
2. Normalizing pixel values to the [0–1] range for a more uniform distribution.
3. Resizing to 64x64 pixels to fit the CNN input.
4. Augmentation in the form of ±15° rotation, horizontal flip, and lighting variations to expand the dataset's diversity and prevent overfitting.

2.2 CNN Architecture

The CNN used consists of several main blocks:

1. Convolutional Layer (32 and 64 filters, 3x3 kernel, ReLU activation).
2. Pooling Layer (MaxPooling 2x2).
3. Dropout (0.25 and 0.5) to reduce overfitting.
4. Fully Connected Layer with 128 neurons.
5. Output Layer with 2 neurons (Softmax) for open/closed eye classification.

Training used the Adam optimizer with a learning rate of 0.001, a Categorical Crossentropy loss function, a batch size of 32, and 25 epochs.

2.3 Eye Landmark Detection with MediaPipe Facemesh

To detect facial landmarks, MediaPipe FaceMesh is used, capable of stably generating 468 facial landmark points. Of these points, six points in the eye area are taken to calculate the EAR (Early Reach). EAR is calculated using the ratio of the vertical and horizontal distances between the eyes. A consistently decreasing EAR indicates closed eyes.

2.4 Ear and CNN Fusion Rules

The EAR is used as the first indicator. If the EAR is <0.25, the system verifies with the CNN. If the CNN predicts both eyes are closed for approximately 2 seconds (approximately 60 frames at 30 fps), the system declares the driver drowsy.

2.5 System Implementation

The system was built using Python with OpenCV for image processing [19][20], TensorFlow/Keras for CNN training, MediaPipe for landmark extraction, and Flask as a web-based application server. The system is capable of displaying real-time video streaming with "AWAKE" or "DROWSY" status indicators and automatic audio alarms.

3. RESULTS AND DISCUSSION

3.1 CNN Training Results

The trained CNN model demonstrated rapid convergence, with training accuracy reaching >99% at the 20th epoch. Evaluation on the test data (218 images) yielded 99.5% accuracy with only 1 misclassification. Precision, recall, and F1-score each reached 0.99–1.00 for both classes, indicating a balanced model with no class-specific bias.

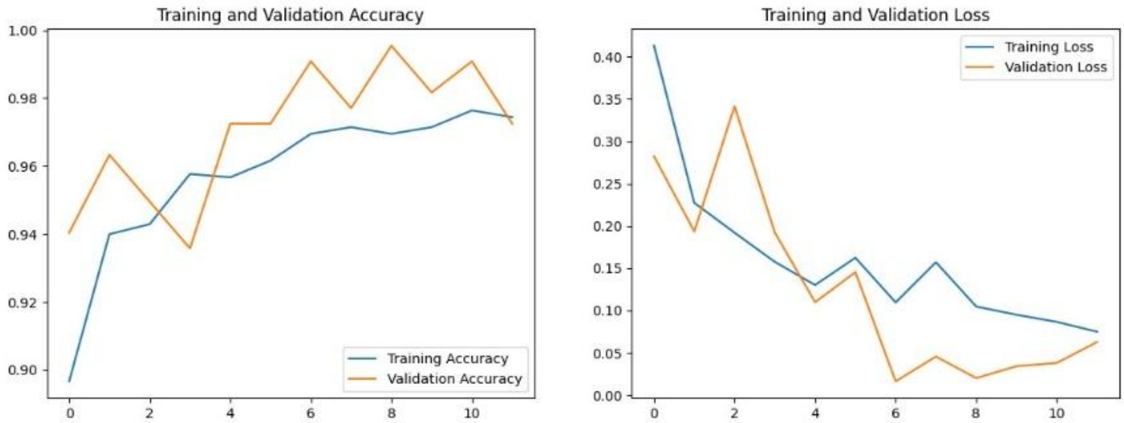

Figure 2. Graph of CNN Model Training Accuracy and Loss

The graph shows that the CNN model consistently improved in accuracy, reaching over 98% at the 10th epoch. Meanwhile, the training and validation loss values continued to decrease, indicating the model successfully generalized without significant overfitting.

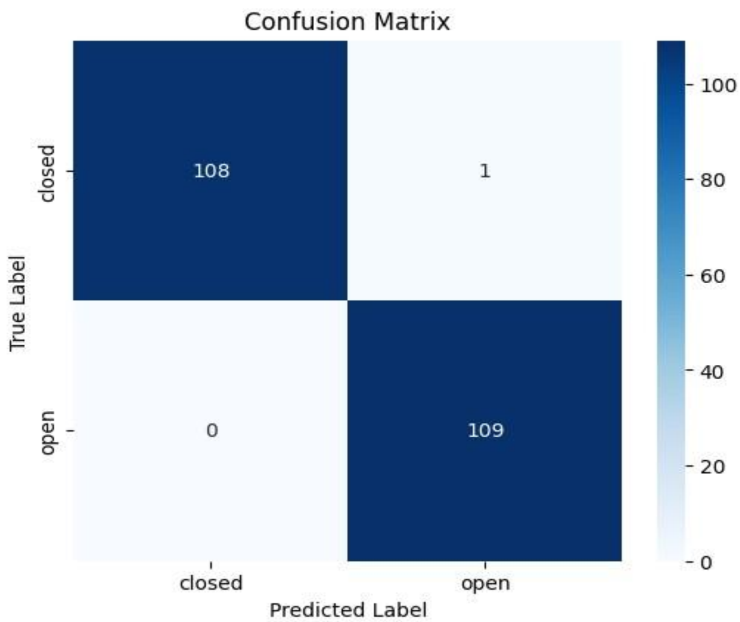

Figure 3. Confusion Matrix Results of Open and Closed Eye Classification

	precision	recall	f1-score	support
closed	1.00	0.99	1.00	109
open	0.99	1.00	1.00	109
accuracy			1.00	218
macro avg	1.00	1.00	1.00	218
weighted avg	1.00	1.00	1.00	218

Figure 4. CNN Classification Report

3.2 Ear and Facemesh Performance

FaceMesh-based EAR calculations have proven more stable than Haar Cascade. EAR is capable of detecting short-term eye closures (0.5–2 seconds) and adapting to variations in head position. However, using EAR alone is susceptible to light interference, so integration with a CNN enhances the validation of the results.

3.3 Ear and CNN Integration

This fusion approach successfully reduced false alarms. Using only CNN, the system sometimes incorrectly classified open eyes in shadows as closed. Using only EAR, the system could incorrectly detect when the driver was looking down. By combining the two, the error rate decreased significantly.

3.4 Realtime Test

In real-time testing, the system was able to process video at approximately 25–30 fps using a mid-range laptop. A Flask-based web interface displayed an EAR overlay, driver status, and visual/audio alarms. Alarm responses occurred an average of 2 seconds after fully closed eyes were detected, according to the set threshold parameters.

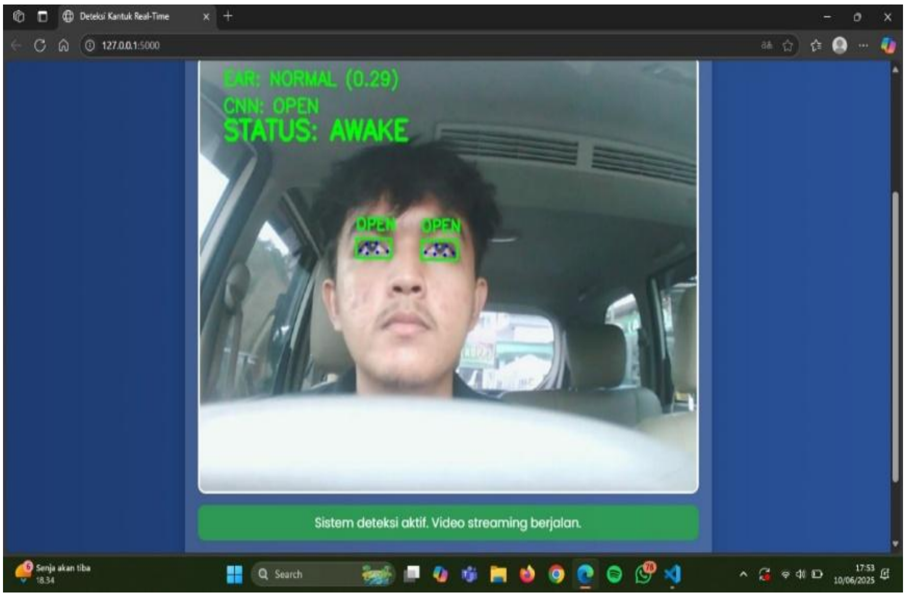

Figure 5. Real-Time Drowsiness Detection System View with Flask

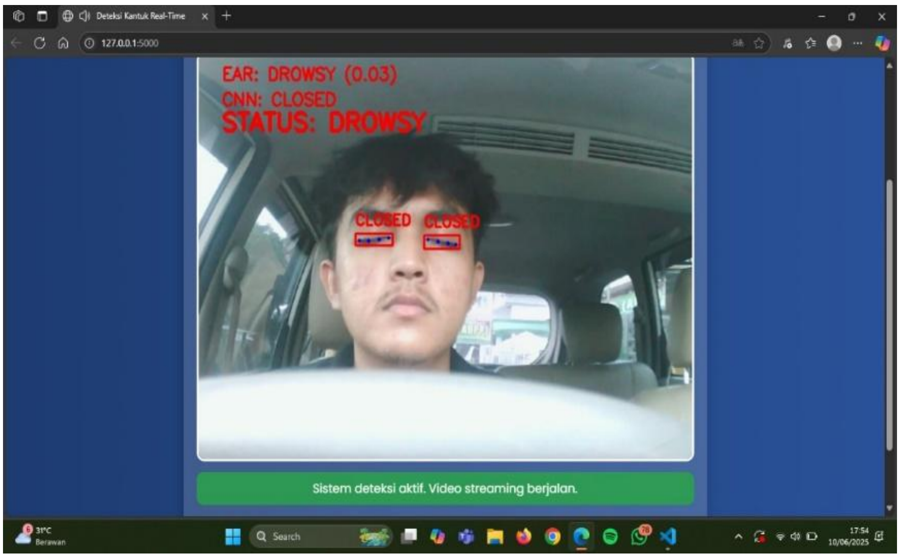

Figure 6. Real-Time Drowsiness Detection System View with Flask

Real-time implementation results demonstrate the system's ability to detect the driver's condition in real-time. When the eyes are open, the status is displayed as AWAKE with EAR normal and CNN open. Conversely, when the eyes are detected closed for more than two seconds, the status changes to DROWSY and an alarm is activated. This demonstrates that the system is operating as designed, providing rapid early warnings, and operating stably in a real-world environment inside the vehicle.

4. RESULT

1. Advantages: The system is lightweight, accurate, and easy to implement because it is based on a regular camera. The combination of EAR + CNN provides a balance between speed and precision.
2. Limitations: The dataset is limited to only two classes (eyes open/closed) without real-world conditions (e.g., glasses, low light, extreme head movements).
3. Development potential: The system can be extended with yawn detection (mouth aspect ratio), integration of other sensors (e.g., accelerometer), and model optimization using TensorFlow Lite to run on edge devices like the Raspberry Pi or ESP32-CAM.

5. CONCLUSSION

This research successfully designed a computer vision-based drowsiness detection system by combining Eye Aspect Ratio (EAR) and Convolutional Neural Network (CNN), and using MediaPipe FaceMesh as an eye landmark detector. Test results demonstrated very high accuracy (99.5%), with a very low misclassification rate and virtually no false alarms. The system was also proven to run in real time on common devices with a fast alarm response. These results conclude that the developed system meets the criteria for being effective, efficient, and applicable for detecting driver drowsiness. However, further research with a more varied dataset and direct testing in real-life driving conditions is needed to ensure the system's robustness to lighting, eyewear use, and variations in driver behaviour.

REFERENCES

[1] G. Zhang, K. K. W. Yau, X. Zhang, and Y. Li, "Traffic accidents involving fatigue driving and their extent of casualties," *Accid Anal Prev*, vol. 87, pp. 34–42, 2016.

[2] A. Moradi, S. S. H. Nazari, and K. Rahmani, "Sleepiness and the risk of road traffic accidents: A systematic review and meta-analysis of previous studies," *Transp Res Part F Traffic Psychol Behav*, vol. 65, pp. 620–629, 2019.

[3] A. T. Kashani, M. R. Moghadam, and S. Amirifar, "Factors affecting driver injury severity in fatigue and drowsiness accidents: a data mining framework," *J Inj Violence Res*, vol. 14, no. 1, p. 75, 2022.

[4] S. Saleem, "Risk assessment of road traffic accidents related to sleepiness during driving: a systematic review," *Eastern Mediterranean Health Journal*, vol. 28, no. 9, pp. 695–700, Sep. 2022, doi: 10.26719/emhj.22.055.

[5] Y. Albadawi, M. Takruri, and M. Awad, "A review of recent developments in driver drowsiness detection systems," *Sensors*, vol. 22, no. 5, p. 2069, 2022.

[6] I. Stancin, M. Cifrek, and A. Jovic, "A review of EEG signal features and their application in driver drowsiness detection systems," *Sensors*, vol. 21, no. 11, p. 3786, 2021.

[7] B. Fu, F. Boutros, C.-T. Lin, and N. Damer, "A survey on drowsiness detection–modern applications and methods," *IEEE Transactions on Intelligent Vehicles*, 2024.

[8] E. Magán, M. P. Sesmero, J. M. Alonso-Weber, and A. Sanchis, "Driver drowsiness detection by applying deep learning techniques to sequences of images," *Applied Sciences*, vol. 12, no. 3, p. 1145, 2022.

[9] M. I. B. Ahmed *et al.*, "A deep-learning approach to driver drowsiness detection," *Safety*, vol. 9, no. 3, p. 65, 2023.

[10] S. Davanathan, N. S. A. M. Taujuddin, and S. Sari, "Driving fatigue detection system using Haar cascade technique," *Evolution in Electrical and Electronic Engineering*, vol. 5, no. 1, pp. 436–442, 2024.

[11] J. More, D. Sutar, R. Sequeira, and V. Chavan, "Eye detection using haar cascade classifier," *International Research Journal of Engineering and Technology (IRJET)*, vol. 8, no. 05, pp. 2356–2395, 2021.

[12] V. Viswanatha, A. C. Ramachandra, G. L. Reddy, A. V. S. T. Reddy, B. P. K. Reddy, and G. B. Kiran, "An Intelligent Camera Based Eye Controlled Wheelchair System: Haar Cascade and Gaze Estimation Algorithms," in *2023 International Conference on Applied Intelligence and Sustainable Computing (ICAISC)*, IEEE, 2023, pp. 1–5.

[13] M. N. Andrean *et al.*, "Comparing haar cascade and yoloface for region of interest classification in drowsiness detection," *Jurnal Media Informatika Budidarma*, vol. 8, no. 1, pp. 272–281, 2024.

[14] A. P. Sunija, S. Kar, S. Gayathri, V. P. Gopi, and P. Palanisamy, "Octnet: A lightweight cnn for retinal disease classification from optical coherence tomography images," *Comput Methods Programs Biomed*, vol. 200, p. 105877, 2021.

[15] I. Topaloglu, "Deep learning based convolutional neural network structured new image classification approach for eye disease identification," *Scientia Iranica*, vol. 30, no. 5, pp. 1731–1742, 2023.

[16] T. J. Jun *et al.*, "TRk-CNN: transferable ranking-CNN for image classification of glaucoma, glaucoma suspect, and normal eyes," *Expert Syst Appl*, vol. 182, p. 115211, 2021.

[17] Aman and A. L. Sangal, "Drowsy alarm system based on face landmarks detection using mediapipe facemesh," in *Proceedings of First International Conference on Computational Electronics for Wireless Communications: ICCWC 2021*, Springer, 2022, pp. 363–375.

[18] S. A. Jakhete and N. Kulkarni, "A comprehensive survey and evaluation of MediaPipe Face Mesh for human emotion recognition," in *2024 8th International Conference on Computing, Communication, Control and Automation (ICCUBEA)*, IEEE, 2024, pp. 1–8.

[19] S. Singh, R. Verma, and A. K. Singh, "Image filtration in Python using openCV," *Turkish Journal of Computer and Mathematics Education*, vol. 12, no. 6, pp. 5136–5143, 2021.

[20] U. Sharma, T. Goel, and J. Singh, "Real-time image processing using deep learning with opencv and python," *J Pharm Negat Results*, pp. 1905–1908, 2023.