DOI: https://doi.org/10.62985/j.huit_ojs.vol26.no2E.421

A DEEP LEARNING APPROACH FOR DRIVER FATIGUE DETECTION

Li Zhenyang[1], Nguyen Dinh Hoa[2]

[1]*Graduate School of Mathematics, Kyushu University, Fukuoka, Japan*

[2]*Department of Automation Engineering, School of Electrical and Electronic Engineering, Hanoi University of Science and Technology, Vietnam*

Email: [1]*lzyddzyx990406@outlook.com*, [2]*hoa.nguyendinh2@hust.edu.vn*

Received: 18 December 2025; Revised: 9 March 2026; Accepted: 3 April 2026

ABSTRACT

Driver fatigue is a major contributing factor to traffic accidents worldwide, motivating the development of reliable, non-invasive, and real-time monitoring systems. This paper presents a vision-based driver fatigue detection system built upon the YOLO (You Only Look Once) deep learning framework. The proposed system integrates face detection and eye-state classification (open/closed) to infer driver fatigue from monocular video streams captured by a standard camera. Compared with traditional fatigue detection approaches based on physiological signals or vehicle dynamics, the proposed method requires no wearable sensors and exhibits low deployment cost and high practical applicability. YOLO is adopted as the core detection model due to its anchor-free design, efficient feature extraction, and fast inference speed. A customized dataset is constructed by combining public datasets with self-collected images, augmented through random background fusion, scaling, and rotation to enhance robustness and generalization. Experimental results demonstrate that the proposed system achieves high detection accuracy and stable real-time performance under typical driving conditions. The results indicate that the proposed approach is suitable for real-world driver monitoring applications and can serve as a foundation for more advanced fatigue detection systems.

Keywords: Driver fatigue detection, deep learning, eye state recognition, computer vision, YOLO.

1. INTRODUCTION

Driver fatigue has been widely recognized as one of the major contributing factors to traffic accidents worldwide. Long-duration driving, monotonous road environments, and insufficient rest can significantly reduce a driver's attention and reaction capability, thereby increasing the risk of severe accidents. According to multiple transportation safety reports, fatigue-related accidents often result in higher fatality rates compared with other types of traffic incidents, which highlights the importance of effective fatigue detection and early warning systems.

Existing driver monitoring systems can be broadly categorized into contact-based and non-contact-based approaches. Contact-based methods typically rely on physiological sensors such as electroencephalogram (EEG) [1], electrocardiogram (ECG), or steering wheel grip sensors [2]. Although these methods can provide relatively accurate measurements, they often suffer from high cost, intrusive installation, and poor user acceptance, which limit their practical deployment in real-world vehicles.

In contrast, vision-based fatigue detection methods have gained increasing attention due to their non-intrusive nature and low deployment cost. By analyzing facial features such as eye closure, blinking frequency, and head pose, these systems can infer driver fatigue states without requiring any physical contact with the driver [3]. With the rapid development of deep learning and computer vision, convolutional neural network (CNN)-based approaches have demonstrated superior performance over traditional handcrafted feature-based methods.

Among various vision-based techniques, eye-state analysis remains one of the most reliable indicators of driver fatigue. Prolonged eye closure or frequent blinking is closely correlated with reduced alertness. Therefore, accurate and robust eye-state detection under varying illumination conditions and complex backgrounds is a key challenge for practical fatigue detection systems.

In recent years, the YOLO (You Only Look Once) [4] series of object detection algorithms, which are developed based on CNN structures, has shown remarkable performance in terms of detection accuracy and real-time efficiency. Compared with two-stage detectors, YOLO-based methods offer faster inference speed while maintaining competitive accuracy, making them particularly suitable for real-time driver monitoring applications. Motivated by these advantages, this study adopts a YOLOv8-based framework to perform face localization and eye-state classification simultaneously.

This paper focuses on the design and implementation of a real-time driver fatigue detection system based on eye-state analysis. The main contributions of this work can be summarized as follows:

(1) A real-time, vision-based driver fatigue detection system is proposed, which integrates face detection and eye-state (open/closed) classification within a unified YOLO framework.

(2) A customized dataset construction and data augmentation strategy is designed by combining facial images, background-only images, and random image fusion, effectively improving robustness and reducing false detections in non-face regions.

(3) A lightweight temporal fatigue decision logic based on consecutive eye-closure duration is introduced, enabling reliable fatigue inference while maintaining low computational cost.

The organization of this paper is as follows. Section 2 reviews related work on driver fatigue detection, including physiological-signal-based, vehicle-behavior-based, and vision-based approaches. Section 3 presents the overall system architecture and details the proposed methodology, including the YOLOv8 framework and loss function design. Section 4 describes the dataset construction, data augmentation strategy, and experimental setup. Section 5 reports the fatigue decision logic, the experimental results, real-time performance evaluation, and ablation studies. Section 6 discusses the limitations of the proposed system, concludes the paper and outlines directions for future work.

2. RELATED WORK

Research on driver fatigue detection has attracted significant attention over the past decades. Existing methods can be divided into three main categories: physiological-signal-based methods, vehicle-behavior-based methods, and vision-based methods.

2.1. Fatigue Detection Based on Physiological Signals

Physiological-signal-based methods directly measure the driver's biological signals to infer fatigue. EEG-based approaches are among the most representative, as EEG signals can reflect brain activity associated with alertness and drowsiness. Mardi et al. [1] used EEG features combined with statistical tests to distinguish between awake and drowsy states,

achieving promising accuracy. However, EEG acquisition requires electrodes to be attached to the driver's scalp, which limits practicality in real driving environments.

Other physiological indicators such as skin conductance and heart rate [5] variability have also been explored. Skin conductance reflects changes in the autonomic nervous system and has been used in combination with machine learning techniques to detect driver drowsiness. Heart rate variability-based methods analyze fluctuations in heartbeat intervals to estimate fatigue levels. Despite their effectiveness, these approaches generally rely on wearable sensors, which may cause discomfort and reduce user acceptance.

2.2. Fatigue Detection Based on Vehicle Behavior

Vehicle-behavior-based methods infer driver fatigue by analyzing vehicle operation parameters. Commonly used indicators include steering wheel angle, steering correction frequency, vehicle speed variation, and lane-keeping performance [6]. For instance, changes in steering behavior have been shown to correlate with driver fatigue, especially during long-duration driving tasks. Alternatively, the detection of the driver's grip force on the steering wheel can be used as a criterion for judgment [7].

While these methods are non-invasive and do not require direct monitoring of the driver, their performance is strongly influenced by external factors such as road geometry, traffic conditions, and vehicle characteristics. As a result, it is difficult to design a universally reliable fatigue detection system based solely on vehicle behavior.

2.3. Vision-Based Fatigue Detection

Vision-based approaches utilize cameras to capture images or videos of the driver and analyze facial cues associated with fatigue. Common visual features include eye closure, blink rate, yawning, head pose, and gaze direction. Soukupová and Čech [3] introduced the Eye Aspect Ratio (EAR), a geometric measure derived from facial landmarks that can effectively distinguish between open and closed eye states.

With the emergence of deep learning, convolutional neural networks (CNNs) and object detection frameworks such as YOLO have been widely applied to driver monitoring tasks. Sikander and Anwar [8] provided a comprehensive review of driver fatigue detection systems based on artificial intelligence, highlighting the advantages of deep learning methods in terms of accuracy and robustness. Building on these advances, the present study adopts YOLOv8 to achieve efficient and real-time eye-state recognition.

3. SYSTEM OVERVIEW AND METHODOLOGY

3.1. Overview of the Proposed System

The proposed driver fatigue detection system consists of three main components: (i) image acquisition; (ii) deep learning-based detection; and (iii) fatigue state inference. A standard RGB camera is used to capture real-time video of the driver's face. Each video frame is processed by the YOLOv8 model to detect the face region and classify the eye state as open or closed. Yawning is not considered in the current research. Based on temporal information derived from consecutive frames, the system determines whether the driver exhibits signs of fatigue.

3.2. YOLO Framework

YOLO is a one-stage object detection algorithm that formulates detection as a single regression problem. Unlike two-stage detectors, YOLO directly predicts bounding box coordinates and class probabilities from the entire image in an end-to-end manner. This design enables high inference speed, making YOLO particularly suitable for real-time applications.

YOLOv8 introduces several improvements over earlier versions, including an anchor-free architecture, optimized network design, and enhanced data augmentation strategies. These improvements contribute to better detection accuracy and faster convergence during training. Compared to newer versions of YOLO, YOLOv8 is more lightweight and has a faster processing time. Hence, it is chosen in the current research.

3.3. Loss Function

The YOLO model is trained by minimizing a multi-task loss function that combines multiple objectives. In this study, three main components of the loss function are used: localization loss, classification loss, and distribution focal loss.

Localization Loss: Localization loss measures the discrepancy between the predicted bounding box and the ground truth. YOLOv8 adopts the Complete Intersection over Union (CIoU) loss, which considers overlap area, center distance, and aspect ratio to improve bounding box regression accuracy.

For each required prediction box, YOLO needs to regress its center coordinates and dimensions. Let the real bounding box be (x, y, w, h), and the predicted bounding box be $(\hat{x}, \hat{y}, \hat{w}, \hat{h})$. In the original YOLO formulation, the localization loss is defined as:

$$L_{box} = \sum_{i=1}^{N} \left[(x_i - \hat{x_i})^2 + (y_i - \hat{y_i})^2 + \left(\sqrt{w_i} - \sqrt{\hat{w_i}}\right)^2 + \left(\sqrt{h_i} - \sqrt{\hat{h_i}}\right)^2 \right] \quad (1)$$

In the newer versions of YOLO, this formula has been replaced by the CIoU (Complete Intersection over Union) function:

$$L_{box} = 1 - CIoU\left(box, \widehat{box}\right) \quad (2)$$

Classification Loss: Classification loss evaluates the difference between the predicted class probabilities and the ground truth labels. Cross-entropy loss is commonly used to quantify this difference and guide the network to correctly classify eye states.

For each predicted box containing the target, YOLO simultaneously predicts its category probability. Let the real category be the vector *p(c)* and the predicted category probability is $\hat{p}(c)$, the Classification Loss is:

$$L_{cls} = \sum_{i=1}^{N} \sum_{c=1}^{C} p_i\,(c) log\left(\hat{p_i}(c)\right) \quad (3)$$

Distribution Focal Loss: Distribution Focal Loss is employed to improve the precision of bounding box coordinate regression by modeling coordinate predictions as discrete probability distributions. This approach enhances localization accuracy, especially for small objects such as eyes.

Let the p_k is the predicted coordinate distribution, t_k is the discrete distribution corresponding to real coordinates, Distribution Focal Loss is:

$$L_{dfl} = \sum_{k} CE(p_k, t_k) \quad (4)$$

where *CE*(•,•) denotes the cross-entropy loss, which measures the discrepancy between the predicted probability distribution and the target distribution.

The total loss function is a weighted sum of three components described above, with adjustable coefficients to balance their relative contributions during training, as follows. Each part has control parameters, namely λ_{box}, λ_{cls}, λ_{dfl}, to adjust their respective influence.

$$L = \lambda_{box} L_{box} + \lambda_{cls} L_{cls} + \lambda_{dfl} L_{dfl} \quad (5)$$

4. TRAINING AND EXPERIMENTAL SETUP

4.1. Dataset Preparation

The dataset used in this study is constructed from three main sources: (i) facial images from the CEW dataset [9]; (ii) background images from ImageNet that do not contain human faces [10]; and (iii) self-collected images captured under conditions similar to real driving environments. Facial images are labeled according to eye state (open or closed), while background images serve as negative samples to reduce false detections, as demonstrated in Fig. 1.

Fig. 1. Three types of images: (left) facial images from CEW dataset; (center) self-collected images; and (iii) background images from ImageNet.

Among them, facial images are divided into two categories based on open and closed eyes. A blank background image refers to other images that do not contain a face, used to reduce the probability of program misjudgment and as a material for image fusion. The environment of the self-captured images is consistent with the environment during real-time verification, because in practical applications, we can train for specific scenarios such as the driver sitting in the cab, and these images can simulate targeted training situations.

4.2. Data Augmentation and Image Fusion

To enhance dataset diversity and improve model generalization, a random image fusion strategy is employed. Facial images are randomly combined with background images, and transformations such as scaling, rotation, translation, and slight color perturbations are applied. This process simulates variations in camera position, driver posture, and environmental conditions. An illustration of this method is depicted in Fig. 2 and Fig. 3.

From a statistical perspective, data augmentation effectively increases the support of the training data distribution, reducing the risk of overfitting. From a practical perspective, it allows the model to better handle unseen backgrounds and lighting conditions commonly encountered in real driving scenarios.

Fig. 2. Random image fusion.

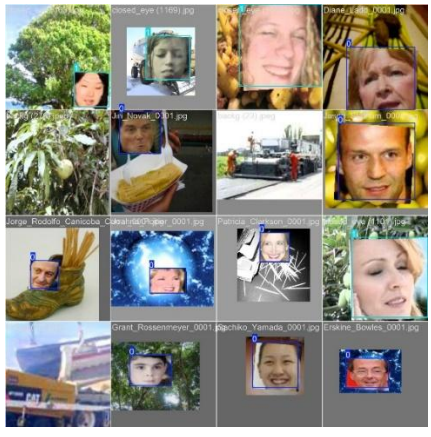

Fig. 3. The dataset that has been fused and annotated.

After preprocessing and augmentation, the final dataset consists of thousands of labeled images covering open-eye, closed-eye, and background-only cases. The dataset is split into training, validation, and test subsets according to a fixed ratio to ensure fair evaluation. The inclusion of background-only images significantly reduces false detections in non-face regions.

4.3. Training and Experimental Setup

The model is trained using the PyTorch-based implementation of YOLOv8. Training is conducted with 30 epochs, an input image size of 640×640 pixels, and a batch size of 16. Default loss weights are used, as preliminary experiments show that satisfactory performance can be achieved without extensive hyperparameter tuning.

Training and validation are performed on a standard personal computer equipped with a camera. After training, the model is evaluated using both offline validation metrics and real-time video testing.

5. EXPERIMENTAL RESULTS

5.1. Experimental Environment

All experiments are conducted on a standard personal computer equipped with an integrated RGB camera. The operating system is Windows, and the implementation is based on the PyTorch deep learning framework. During offline training, GPU acceleration can be utilized if available, while real-time testing is performed on CPU only to reflect realistic deployment conditions. The camera is positioned approximately 50–70 cm in front of the driver's face, simulating typical in-vehicle installation scenarios.

Lighting conditions during experiments include normal indoor illumination and slightly dim environments, which are representative of common driving situations. No infrared illumination is used in the current system.

5.2. Fatigue Decision Logic

In the proposed system, driver fatigue is inferred based on temporal analysis of eye-state classification results. Let $s_t \in \{0,1\}$ denote the predicted eye state at time t, where 0 represents open eyes and 1 represents closed eyes. A temporal window of length T is defined to accumulate consecutive eye-closure events. Then the accumulated eye-closure duration D within the window is calculated as:

$$D = \sum s_t , t = 1, \cdots, T$$

If the accumulated duration exceeds a predefined threshold θ, the driver is classified as fatigue. In the current implementation, T and θ are fixed parameters determined empirically. In our simulation, $T = 2$ seconds. This simple yet effective logic allows real-time fatigue detection while maintaining low computational overhead.

5.3. Evaluation Metrics

To quantitatively evaluate the performance of the proposed driver fatigue detection system, several commonly used metrics in object detection and classification tasks are adopted. Precision and recall are used to measure the correctness and completeness of eye-state detection results, respectively.

In addition, the mean Average Precision (mAP) is employed as the primary metric to evaluate overall detection performance across classes. In this study, the mean Average Precision at an Intersection over Union (IoU) threshold of 0.5, denoted as mAP@0.5 or mAP50, is

adopted. A detection is considered correct if the IoU between the predicted bounding box and the corresponding ground-truth bounding box exceeds 0.5.

The IoU is defined as the ratio between the area of overlap and the area of union of the predicted and ground-truth bounding boxes. Compared with stricter evaluation criteria such as mAP@0.5:0.95, mAP50 provides a more practical assessment for real-time, application-oriented detection tasks, where moderate localization accuracy is sufficient for reliable fatigue inference.

By jointly considering precision, recall, and mAP50, the evaluation metrics comprehensively reflect the detection accuracy, robustness, and practical usability of the proposed system.

5.4. Quantitative Results

Quantitative results obtained during the training and validation process indicate that the proposed YOLOv8-based model achieves high detection accuracy for both face detection and eye-state classification. The model demonstrates stable convergence behavior and maintains consistent detection performance across different validation samples.

Fig. 4. Training results and validation

In particular, the model trained with the proposed dataset construction strategy achieves an mAP50 of 0.99, indicating reliable detection capability under typical experimental conditions, as indicated in Fig. 4. Moreover, the performance of mAP50 is better than that of mAP50:95, which confirms the aforementioned discussion.

The corresponding training and validation curves are shown in Fig. 5, where smooth convergence and no obvious overfitting behavior can be observed. The inclusion of background-only images in the training dataset effectively reduces false detections in non-face regions, contributing to improved robustness and generalization performance.

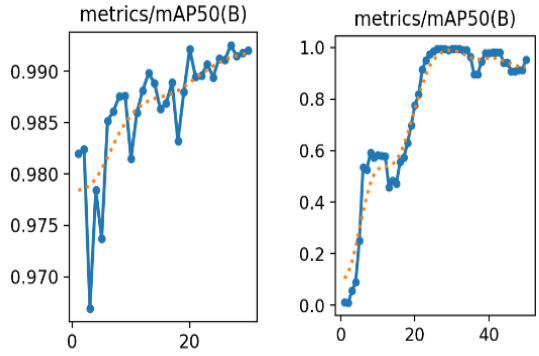

Fig. 5. Comparison of training strategies with (left) and without background fusion (right).

5.5. Real-Time Performance Evaluation

As the research objective is real-time detection, it is also necessary to verify whether the model can perform correct real-time detection in close proximity to real situations, and to use video implementation for verification.

Real-time performance is evaluated using a laptop-integrated camera under typical indoor lighting conditions. The laptop GPU is GTX1060, while its camera resolution is 1280×720. The system processes video frames continuously and displays accurate detection results with minimal latency, as shown in Fig. 6. Experimental results show that the model can run at real-time speed on a standard personal computer without the need for specialized hardware acceleration, confirming its suitability for deployment in real driving scenarios.

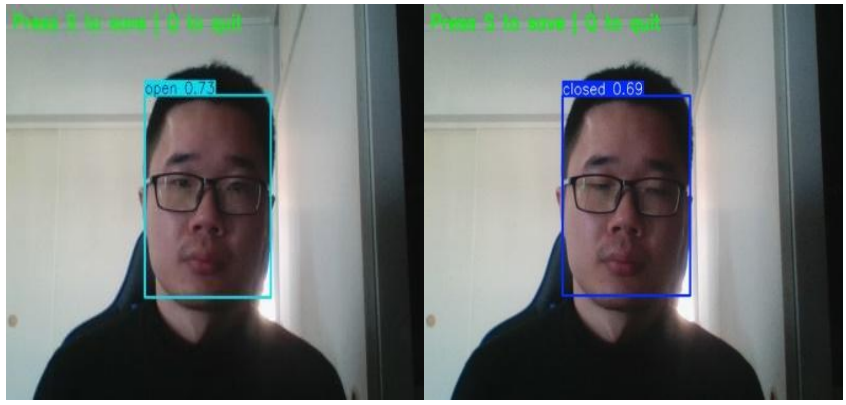

Fig. 6. Real time test results.

5.6. Ablation Study

To analyze the contribution of individual components in the proposed system, an ablation study is conducted focusing on dataset construction strategies. Two models are trained under identical settings: one using the proposed background image fusion strategy, and the other using only facial images without background augmentation.

Experimental results show that the model trained with background fusion exhibits higher robustness in cluttered scenes and achieves improved precision by reducing false positive detections. This confirms that background-only images and random fusion play an important role in enhancing generalization ability.

6. RESEARCH CONCLUSION AND FUTURE WORK

This study proposes and implements a deep learning based driver fatigue detection method, which mainly judges the driver's fatigue state through face detection and eyes' opening and closing state recognition. The system adopts the YOLO series object detection model to perform real-time detection of the driver's facial area and further classify the eye state, thereby achieving effective extraction of fatigue related behavioral features. The experimental results show that this method can stably detect the driver's face and eye status under normal lighting conditions, and has good real-time performance, meeting the basic application needs in practical driving scenarios.

Compared with traditional methods based on artificial features or rules, the deep learning model used in this paper has strong generalization performance, reduces dependence on human experience, and improves the robustness of the system in complex environments.

Although the method proposed in this article has achieved certain results, there is still room for further improvement. Firstly, the current system uses a fixed threshold to determine fatigue status, and in the future, a dynamic threshold mechanism based on vehicle speed, road environment, or driving behavior can be introduced to improve the accuracy and adaptability of fatigue determination. Secondly, more fatigue related features can be included in the detection range, such as blink frequency, duration of eye closure, changes in head posture, and gaze direction, to achieve a multi feature fusion fatigue recognition model. Last but not least, extreme lighting conditions, occlusions caused by sunglasses, or large head movements may degrade detection accuracy. These challenges suggest that incorporating additional visual cues and adaptive strategies could further enhance system robustness.

Future work will focus on incorporating dynamic fatigue thresholds that adapt to vehicle speed, road conditions, and individual driving behavior. Additional fatigue-related features such as blink frequency, eye-closure duration, head pose, and gaze direction will also be explored to build a multi-feature fusion fatigue detection system. Furthermore, testing under more diverse real-world driving conditions will be conducted to further validate system robustness and generalization ability.

REFERENCES

[1] Z. Mardi, S. N. Ashtiani, and M. Mikaili, "EEG-based Drowsiness Detection for Safe Driving Using Chaotic Features and Statistical Tests," *Journal of Medical Signals and Sensors*, vol. 1, no. 2, pp. 130–137, 2011. Available: https://pmc.ncbi.nlm.nih.gov/articles/PMC3342623/

[2] R. Li, Y. V. Chen, and L. Zhang, "A Method for Fatigue Detection Based on Driver's Steering Wheel Grip," *International Journal of Industrial Ergonomics*, vol. 82, Art. no. 103083, 2021. doi: https://doi.org/10.1016/j.ergon.2021.103083

[3] T. Soukupová and J. Čech, "Real-Time Eye Blink Detection Using Facial Landmarks," in *Proc. Computer Vision Winter Workshop (CVWW)*, 2016, pp. 1–8. Available: https://vision.fe.uni-lj.si/cvww2016/proceedings/papers/05.pdf

[4] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2016, pp. 779–788. doi: https://doi.org/10.1109/CVPR.2016.91

[5] A. Amidei, S. Spinsante, G. Iadarola, S. Benatti, F. Tramarin, P. Pavan, and L. Rovati, "Driver Drowsiness Detection: A Machine Learning Approach on Skin Conductance," *Sensors*, vol. 23, no. 8, Art. no. 4004, 2023. doi: https://doi.org/10.3390/s23084004

[6] K. Lu, A. S. Dahlman, J. Karlsson, and S. Candefjord, "Detecting Driver Fatigue Using Heart Rate Variability: A Systematic Review," *Accident Analysis & Prevention*, vol. 178, Art. no. 106830, 2022. doi: https://doi.org/10.1016/j.aap.2022.106830

[7] J. Xi, S. Wang, T. Ding, J. Tian, H. Shao, and X. Miao, "Detection Model on Fatigue Driving Behaviors Based on the Operating Parameters of Freight Vehicles," *Applied Sciences*, vol. 11, no. 15, Art. no. 7132, 2021. doi: https://doi.org/10.3390/app11157132

[8] G. Sikander and S. Anwar, "Driver Fatigue Detection Systems: A Review," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 6, pp. 2339–2352, 2019. doi: https://doi.org/10.1109/TITS.2018.2868499.

[9] F. Song, X. Tan, X. Liu, and S. Chen, "Eyes Closeness Detection From Still Images With Multi-Scale Histograms of Principal Oriented Gradients," *Pattern Recognition*, vol. 47, no. 9, pp. 2825–2837, 2014. doi: https://doi.org/10.1016/j.patcog.2014.03.024

[10] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A Large-Scale Hierarchical Image Database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2009, pp. 248–255. doi: https://doi.org/10.1109/CVPR.2009.5206848