Article

Research on the Analysis and Recognition System for Dangerous Driving Behaviors Based on Convolutional Neural Networks

Yunpeng Liu [1,*] and Joan Lazaro [1]

[1] University of the East, Manila, Philippines
* Correspondence: Yunpeng Liu, University of the East, Manila, Philippines

Abstract: To address the critical need for real-time monitoring of hazardous behaviors that compromise road safety, this study presents an intelligent detection system centered on the lightweight YOLOv11n model. The system integrates multiple functional modules, including object detection, multi-object tracking using ByteTrack and BoTSORT, and behavior recognition logic based on temporal windows, thereby forming a comprehensive technical workflow encompassing data acquisition, preprocessing, inference, tracking, behavior analysis, and early warning. It is capable of accurately detecting six common dangerous behaviors-yawning, eye closure, smoking, drinking, mobile phone usage, and in-vehicle conversations-while providing timely alerts to mitigate potential risks. Extensive experiments demonstrate that the system achieves a validation accuracy of 65.55%, a recall rate of 79.71%, and an mAP50 of 69.26%. The inference speed for individual images is maintained within 30-60 milliseconds, and video processing can consistently reach 20-30 frames per second, ensuring real-time performance suitable for practical deployment. The proposed framework not only enhances detection precision but also supports continuous monitoring and risk prevention in real-world driving environments, providing a technically feasible and operationally efficient solution for improving driver safety.

Keywords: convolutional neural network; detection of dangerous driving behaviors; YOLO11n; behavior discrimination; real-time warning

Received: 06 December 2025
Revised: 26 January 2026
Accepted: 06 February 2026
Published: 11 February 2026

1. Introduction

With the continuous increase in global car ownership, road traffic safety has emerged as a pressing public safety concern worldwide. Each year, approximately 1.3 million people lose their lives in road traffic accidents, and nearly 50 million sustain serious injuries, both fatal and non-fatal, leading to substantial economic and social losses for drivers, passengers, and their families [1]. The overall economic burden of road traffic accidents in a given country is estimated to account for roughly 3% of its gross domestic product. In the United States alone, a significant portion of traffic-related fatalities and injuries is linked to distracted and drowsy driving, with thousands of deaths and hundreds of thousands of injuries occurring annually due to these preventable behaviors [2]. Similarly, in China, national traffic statistics indicate that a substantial share of road accidents is caused by distracted driving, representing the leading single factor contributing to traffic casualties. These observations underscore that unsafe driving behaviors, particularly driver fatigue and distraction, remain the critical bottleneck limiting improvements in traffic safety [3].

Studies on traffic accident causation further reveal that human factors dominate as the primary contributors, with improper driving operations accounting for a large majority of incidents [4]. Evidence suggests that timely detection and warning of risky driving behaviors can significantly reduce accident occurrence, with preventive measures potentially avoiding a vast majority of traffic collisions. Vehicles equipped with driver monitoring systems capable of detecting fatigue or distraction demonstrate substantially lower accident rates compared to those lacking such technologies. These findings highlight the urgent need for intelligent systems that can monitor driver behavior in real time, accurately identify potentially dangerous actions, and provide timely alerts to mitigate risk. Developing such a system requires the integration of advanced data analysis techniques, machine learning algorithms, and real-time sensing technologies to ensure reliability and efficiency under varied driving conditions [5].

Addressing this challenge not only contributes to enhancing overall traffic safety but also supports the broader objectives of sustainable mobility and public welfare [6]. By focusing on precise behavior detection and early warning mechanisms, research in this field aims to reduce accident rates, protect human life, and minimize the social and economic impact of road traffic incidents. Consequently, the design and implementation of comprehensive dangerous driving behavior detection systems have become a central topic in contemporary vehicle safety research, with significant implications for both policy development and practical applications [6].

2. Research Status

Existing approaches for detecting dangerous driving behaviors can generally be categorized into three main types: methods based on physiological features, methods based on vehicle behavioral characteristics, and methods relying on computer vision. Techniques utilizing physiological features are capable of directly reflecting the driver's internal state, providing valuable insights into fatigue, distraction, or stress levels [7]. However, these approaches are limited by the need for invasive sensors, susceptibility to signal interference, and substantial individual variability, which pose significant challenges for large-scale and practical deployment. In contrast, detection methods based on vehicle behavioral characteristics are non-invasive and relatively easy to implement, offering a practical avenue for monitoring driver safety. Nevertheless, these methods are heavily influenced by external conditions such as road surface, traffic environment, and individual driving habits, which reduces the reliability of detection when used in isolation. As a result, vehicle behavior-based approaches are typically employed as auxiliary tools and are most effective when integrated with visual or physiological signals to improve overall detection performance [8].

In recent years, rapid advances in artificial intelligence, particularly in deep learning and computer vision, have enabled significant progress in the detection of dangerous driving behaviors. Modern computer vision-based systems can extract and analyze subtle facial expressions, eye movements, and hand gestures, facilitating non-invasive, high-accuracy monitoring in real-time [9]. Deep convolutional neural networks (CNNs) have been widely adopted to model complex patterns of driver behavior, often combined with multi-scale feature extraction and ensemble learning techniques to enhance detection robustness. Despite these advancements, current methods still face critical limitations in practical deployment. Challenges include insufficient real-time performance, large and computationally intensive models, lack of efficient multi-object tracking, and underdeveloped early warning mechanisms. Achieving lightweight and accelerated models without compromising detection accuracy remains a key technical hurdle. Furthermore, designing adaptive early warning strategies that can filter transient noise and provide timely alerts with minimal cognitive load and millisecond-level responsiveness is essential for real-world application.

Addressing these challenges, this study focuses on the development of an integrated system capable of accurate and real-time detection of dangerous driving behaviors. Leveraging CNN-based object detection, multi-object tracking, and temporal behavior analysis, the system is designed to identify six common risky behaviors-yawning, eye closure, smoking, drinking, mobile phone usage, and in-vehicle conversation-while providing immediate alerts to the driver. By combining high-precision visual analysis with temporal discrimination logic, the proposed framework effectively balances accuracy and real-time performance, creating a practical solution suitable for deployment in modern vehicles. This integrated approach not only enhances detection reliability but also provides a foundation for continuous monitoring and proactive risk prevention, contributing to improved traffic safety and reduced accident rates in complex driving scenarios.

3. System Architecture Design

In the design of the system architecture, the whole system follows the IPO (Input-Processing-Output) design paradigm to construct an efficient architecture. The input layer collects real-time video streams or static images via on-vehicle cameras, which are uniformly processed into standard data of 640×640 pixels in the RGB color space with pixel values normalized to the range of [0,1] through OpenCV. As the core part, the processing layer adopts the YOLO11n model for object detection, integrates the ByteTrack and BoTSORT algorithms to achieve multi-object tracking, and counts the duration of abnormal behaviors through temporal analysis logic to trigger early warnings. The output layer visually displays the detection results through the PySide6 interface, and simultaneously provides functions such as audio alarm, local data recording and real-time performance statistics.

In the process of system engineering implementation, a custom dataset containing more than 8,000 images was constructed in this study, covering six core dangerous behaviors including yawning, eye closure, smoking, drinking, mobile phone use and in-car conversation. For development tool selection, the model was trained based on the Python language with the PyTorch framework, and the desktop GUI was developed using PySide6, which greatly ensured the responsiveness of the system. The overall system adopts a five-layer hierarchical architecture design, as shown in figure 1, which consists of the user interface layer, application logic layer, deep learning model layer, computer vision framework layer and hardware acceleration layer from top to bottom. Data interaction between each layer is realized through standardized interfaces, achieving favorable module decoupling and system scalability.

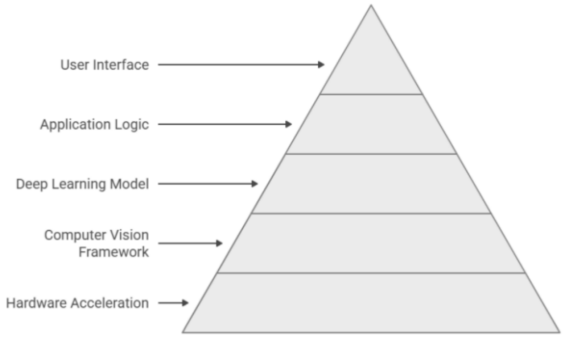


Figure 1. System Layered Architecture.

3.1. User Interface Layer

Located at the topmost layer of the system architecture, this layer is responsible for providing a diverse range of user interaction interfaces, mainly including the PySide6 desktop graphical user interface (GUI) and the command line interface (CLI). The PySide6 desktop GUI is built on the Qt framework to form a multi-tab (TabWidget) interface, which integrates functional modules such as image detection, video detection, real-time camera detection and parameter configuration, and supports real-time parameter adjustment (confidence threshold, IoU threshold, tracking algorithm selection, etc.) as well as visual display of detection results. The command line interface provides a batch script interface to support automated tasks such as batch image inference and video file processing.

3.2. Application Logic Layer

As the core of the system's business logic, this layer is responsible for coordinating the execution processes of each functional module. It consists of four main functional modules: the training module, inference module, tracking module and alarm management module. The training module encapsulates the training process of the YOLO11 model and supports custom dataset training, model fine-tuning, hyperparameter optimization and other functions. The inference module enables single image detection, batch image detection, frame-by-frame video stream detection and related functions. The tracking module integrates two multi-object tracking algorithms, ByteTrack and BoTSORT, to achieve cross-frame association and trajectory smoothing of detection results, and supports the dynamic configuration of tracking parameters (tracking threshold, trajectory length, etc.). The alarm management module implements the real-time monitoring of dangerous behaviors and an alarm trigger mechanism, including functions such as behavior duration statistics, threshold judgment and audio alarm playback. The application logic layer realizes the asynchronous execution of video stream processing through multi-threading programming technology (the threading module), thus avoiding blocking the response of the user interface.

3.3. Deep Learning Model Layer

This layer provides object detection capabilities. Based on YOLO11n, a lightweight CNN architecture (GhostConv+CBAM+SPPFN/PAN-FPN) is constructed herein, with refined inter-layer optimization tailored to the feature requirements of dangerous driving detection. This enables the model to accurately capture small-target features and fine-grained behavioral features, suppress background interference, and improve detection accuracy from the source of feature extraction. The layer supports the dynamic loading of model weights and a variety of optimization technologies including inference acceleration (TensorRT, ONNX Runtime) and model quantization (INT8 quantization).

3.4. Computer Vision Framework Layer

This layer provides underlying image processing and numerical computing capabilities, mainly including the OpenCV library, NumPy library and PyTorch deep learning framework. It is responsible for preprocessing operations such as image reading and format conversion, as well as visual drawing of detection results. Through standardized data formats (NumPy arrays, PyTorch tensors), this layer achieves seamless integration with the upper model layer.

3.5. Hardware Acceleration Layer

Located at the bottommost layer of the system architecture, this layer is responsible for accelerating the computing process by utilizing hardware resources. It supports a variety of computing devices including NVIDIA GPU (CUDA), multi-core CPU parallelism and Apple Silicon (MPS). Through the device abstraction interface of PyTorch, it enables automatic selection and switching of computing devices, ensuring the system's compatibility across different hardware environments.

4. Implementation of Key Technologies

This study adopts the lightweight YOLO11n as the base detection model, achieves a balance between detection accuracy and real-time video stream processing through multi-dimensional optimization, and effectively improves the generalization ability of the model.

4.1. Construction of the YOLO11n Detection Model

On the basis of the original YOLO11n, the system is improved in a targeted manner, as illustrated in Figure 2. The CBAM attention mechanism is introduced into the backbone network, traditional convolutions are replaced with GhostConv lightweight convolutions, and the classic SPPFN+PAN-FPN structure is retained in the neck network. The system adopts C2f+GhostConv of YOLO11n as the core module of the backbone network, with the input being a standardized 640×640 driver image. Through multiple rounds of "convolution-activation-pooling-residual connection", three types of hierarchical features are extracted from shallow to deep layers, each corresponding to different requirements of dangerous driving detection, and redundant computations are reduced through lightweight design throughout the process. The CBAM attention mechanism is added after each C2f module in the backbone network, which automatically enhances features related to dangerous driving and suppresses background interference features through the dual branches of channel attention and spatial attention. SPPFN+PAN-FPN is adopted as the neck network, whose core function is multi-scale feature fusion in the "top-down + bottom-up" manner. It fuses the three layers of features (80×80, 40×40, 20×20) from the backbone network into three groups of fused feature maps and outputs them to the detection head, ensuring that dangerous driving behavior features of different scales can be effectively extracted and utilized. All feature extraction operations are completed based on this architecture without additional feature extraction networks, realizing end-to-end detection and adapting to the real-time requirements of on-vehicle edge devices.

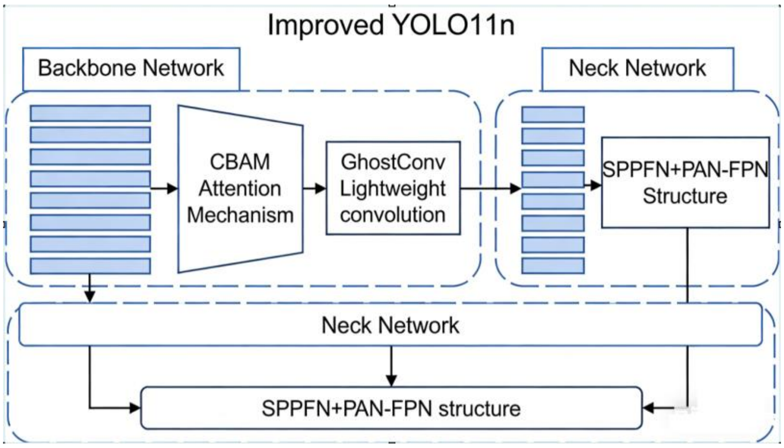


Figure 2. Improved Model Diagram of YOLO11n.

Based on the one-stage architecture of YOLO11n, this system integrates behavior classification and target position regression (locating behavioral carriers such as the driver's face, hands, cigarettes and mobile phones) into a single detection head, which shares all results of feature extraction and multi-scale fusion. Custom anchor boxes are generated via K-means clustering to match exclusive anchor boxes for dangerous driving behaviors of different scales and types. This enables the classification branch to perform feature analysis only on behavior-related regions, avoiding interference from background regions and improving classification accuracy.

4.2. Customized Loss Function

Aiming at two major classification pain points in dangerous driving detection, namely category imbalance and the difficulty in distinguishing fine-grained behaviors, this system abandons the default cross-entropy loss of YOLO and adopts a joint customized loss function combining classification loss and regression loss. This enables the model to focus on learning the feature differences of minority-class behaviors and similar fine-grained behaviors during training, while improving the positioning accuracy of bounding boxes, thus enhancing the detection accuracy from the perspective of optimization objectives.

(1) Improvement of Focal Loss for Addressing Class Imbalance and Fine-Grained Behavior Distinction

To tackle the two major issues of class imbalance and the difficulty in distinguishing fine-grained behaviors in dangerous driving detection, a fine-grained behavior penalty term is added to the traditional Focal Loss.

$$\mathrm{FL}(\mathrm{p_t}) = -\mathrm{a_t} \times (1 - \mathrm{p_t})^{\gamma} \times \log(\mathrm{p_t}) + \beta \times |\mathrm{p_t} - \mathrm{p}_{t-sim}|$$

Where:

P_t: The model's predicted probability for the true category;

α_t: Class balance factor: For class imbalance, a larger value is set for minority dangerous behaviors and a smaller value for majority ones;

γ: Hard-easy sample modulation factor: Optimized through experiments, it is set to 2 in this system, which exponentially amplifies the loss of hard-to-distinguish samples and makes the model focus on the training of hard samples;

$\beta\times|P_t-P_{t-sim}|$: Fine-grained behavior penalty term, where β is the penalty coefficient (0.1) and Pt−sim is the predicted probability of similar behaviors. This term enlarges the difference in the model's predicted probabilities for similar behaviors to achieve accurate distinction.

(2) Adoption of EIOULoss for Improving Localization Accuracy of Behavioral Regions

Abandoning IOU/CIOULoss, the system adopts EIOULoss, which decomposes the bounding box regression loss into overlap loss, center distance loss and aspect ratio loss. This makes the model's bounding box regression more accurate and reduces behavioral misjudgment caused by localization deviations.

$Loss_{EIOU}=Loss_{IOU}+Loss_{dist}+Loss_{wh}$

Among them, $Loss_{dist}$ denotes the center distance loss between the predicted box and the ground-truth box, and $Loss_{wh}$ denotes the aspect ratio loss. Both directly optimize the position and size of the bounding box, leading to a faster convergence speed and higher localization accuracy.

(3) Construction of a Joint Loss Function to Balance the Optimization Weights of Classification and Regression

The improved Focal Loss is combined with EIOU Loss to construct a joint loss function as the total optimization objective of the model, and reasonable weight coefficients are set to ensure the synergistic improvement of classification accuracy and localization accuracy.

$Loss_{total}=Loss_{Focal}+\lambda\times Loss_{EIOU}$

Where λ is the weight coefficient, set to 5 through experimental optimization to balance the classification and regression losses and enable the model to optimize classification accuracy and localization accuracy simultaneously.

5. Experimental Analysis

Transfer learning was adopted for model training with the pre-trained weights of YOLO11n on the COCO dataset. The training process was configured with a maximum of 100 epochs, a batch size of 4, and an initial learning rate of 0.005. A cosine annealing learning rate scheduling strategy was employed, and an early stopping mechanism (with a patience parameter of 20 epochs) was configured.

5.1. Model Training Performance Evaluation

A total of 100 epochs were completed in the training process, with the total training time being approximately 3.3 hours. As shown in Figure 3, the loss curves of both the training and validation sets showed a continuous overall decline with obvious convergence characteristics. The loss dropped rapidly at the initial stage of training and then the fluctuations slowed down. The loss trend of the validation set was generally consistent with that of the training set, indicating no obvious overfitting.

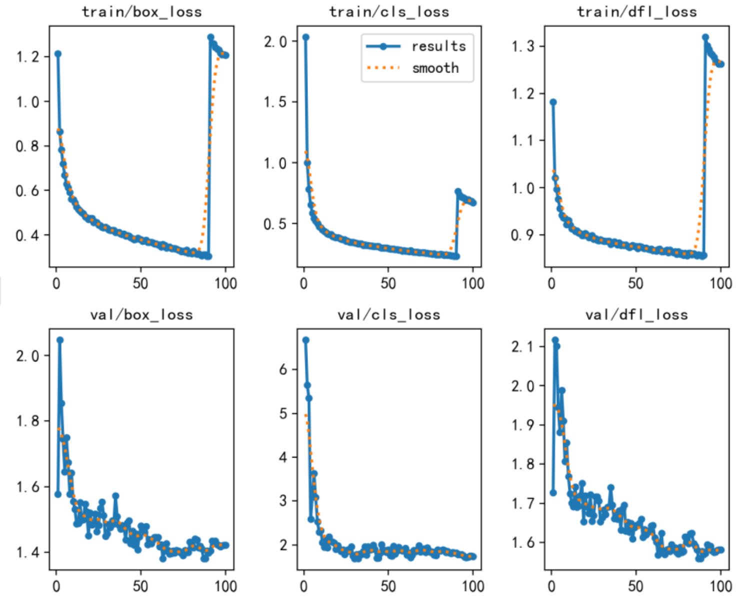


Figure 3. Train/Val loss.

Performance evaluation results on the validation set demonstrate that the model achieved favorable detection performance after 100 training epochs, as shown in Figure 4. The precision reached 65.55%, indicating that 65.55% of the samples predicted as positive by the model were actually positive cases. The recall stood at 79.71%, meaning the model could detect 79.71% of the actual dangerous driving behavior targets. The mAP50 metric was 69.26%, which comprehensively evaluates the model's average detection accuracy at an IoU threshold of 0.5. The mAP50-95 metric reached 42.29%, which is evaluated using a more stringent IoU threshold range. The F1-score calculated based on precision and recall was approximately 71.9%.

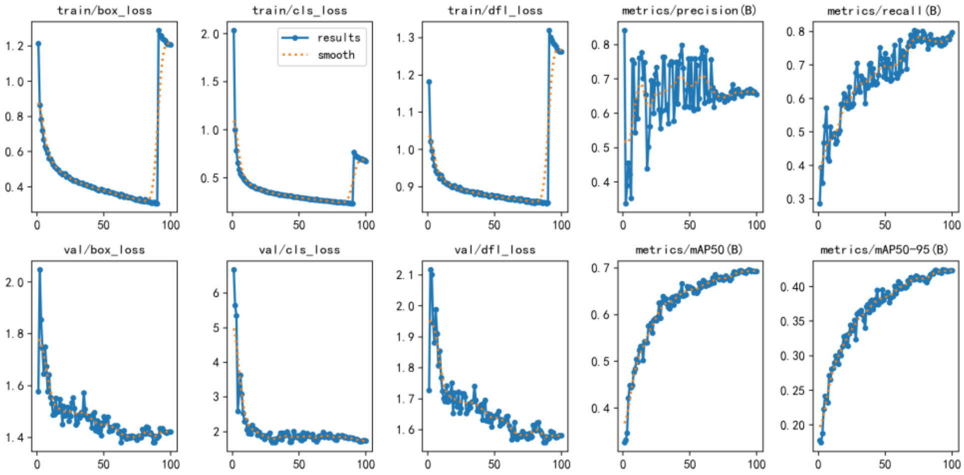
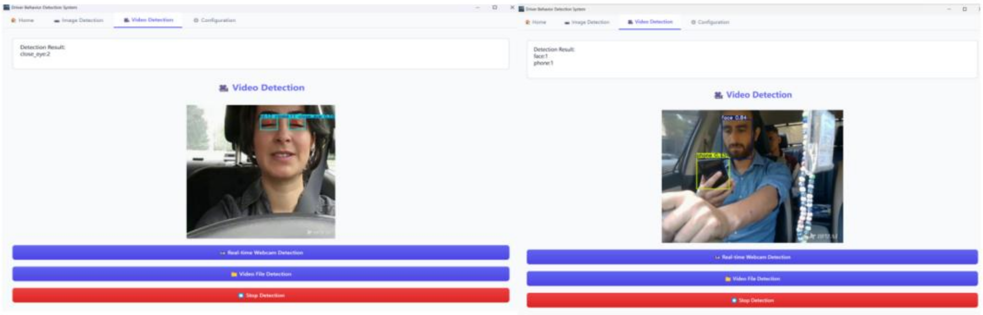


Figure 4. Results.

5.2. System Performance Test

System performance tests show that the system can correctly process common image formats (JPG, PNG, BMP, etc.) with accurate visualization of detection results, as shown in Figure 5.. It supports detection on local video files, real-time detection via USB cameras, and multi-object tracking. When a dangerous behavior is detected to last beyond the set threshold, the system can trigger audio alarms accurately, achieving the detection of six types of dangerous driving behaviors: eye closure, yawning, smoking, drinking, phone use, and in-car conversation.



Detection Result:
close_eye:2

Detection Result:
face:1
phone:1

Figure 5. System Testing Interface.

In a GPU-accelerated environment, the inference process for a single image consists of three main stages: preprocessing, model inference and postprocessing, with a total inference time of approximately 30-60 ms, corresponding to an inference speed of about 16-33 FPS, which meets the performance requirements for real-time detection. For video file detection and real-time camera detection, the system can achieve a processing frame rate of 20-30 FPS in a GPU environment.

Meanwhile, long-term operation tests demonstrate that the system maintains stable memory usage during continuous operation with no obvious memory leaks, stable GPU video memory occupancy without out-of-memory issues, normal system response, and no freezes or crashes in the user interface, indicating excellent system stability.

At the system integration level, a graphical user interface with functions including image detection, video detection and real-time camera monitoring was developed based on the PySide6 framework. The ByteTrack/BoTSORT multi-object tracking algorithms and the time window-based dangerous behavior discrimination logic were integrated into the system, forming a complete technical chain of data collection-preprocessing-model inference-tracking association-behavior discrimination-alarm triggering.

6. Research Conclusions

Based on the YOLO11n architecture, this study constructed an end-to-end detection system based on convolutional neural network technology by introducing the CBAM attention mechanism to enhance the focus on fine-grained features, integrating GhostConv to enable efficient model operation, and incorporating effective methods such as object detection, multi-object tracking and temporal behavior discrimination logic. The system achieves accurate identification and real-time early warning of six types of dangerous driving behaviors including yawning, eye closure, smoking, drinking, mobile phone use and in-car conversation, meeting the requirements of on-vehicle real-time detection.

References

1. J. S. Bajaj, N. Kumar, R. K. Kaushal, H. L. Gururaj, F. Flammini, and R. Natarajan, "System and method for driver drowsiness detection using behavioral and sensor-based physiological measures," *Sensors*, vol. 23, no. 3, p. 1292, 2023. doi: 10.3390/s23031292
2. X. Tong, and M. J. C. Samonte, "Research on dangerous driving behavior recognition method based on convolutional neural network," In *Third International Conference on Image Processing, Object Detection, and Tracking (IPODT 2024)*, October, 2024, pp. 189-195.

3. Q. Xiong, J. Lin, W. Yue, S. Liu, Y. Liu, and C. Ding, "A deep learning approach to driver distraction detection of using mobile phone," In *2019 IEEE Vehicle Power and Propulsion Conference (VPPC)*, October, 2019, pp. 1-5. doi: 10.1109/vppc46532.2019.8952474
4. R. C. Coetzer, and G. P. Hancke, "Eye detection for a real-time vehicle driver fatigue monitoring system," In *2011 IEEE Intelligent Vehicles Symposium (IV)*, June, 2011, pp. 66-71.
5. R. Kapuscinski, "The other," *Verso Books*, 2018.
6. B. T. Dong, H. Y. Lin, and C. C. Chang, "Driver fatigue and distracted driving detection using random forest and convolutional neural network," *Applied Sciences*, vol. 12, no. 17, p. 8674, 2022.
7. S. A. El-Nabi, W. El-Shafai, E. S. M. El-Rabaie, K. F. Ramadan, F. E. Abd El-Samie, and S. Mohsen, "Machine learning and deep learning techniques for driver fatigue and drowsiness detection: a review," *Multimedia Tools and Applications*, vol. 83, no. 3, pp. 9441-9477, 2024.
8. T. Khan, G. Choi, and S. Lee, "EFFNet-CA: an efficient driver distraction detection based on multiscale features extractions and channel attention mechanism," *Sensors*, vol. 23, no. 8, p. 3835, 2023. doi: 10.3390/s23083835
9. M. A. Uddin, N. Hossain, A. Ahamed, M. M. Islam, A. Khraisat, A. Alazab, and M. A. Talukder, "Abnormal driving behavior detection: A machine and deep learning based hybrid model," *International Journal of Intelligent Transportation Systems Research*, vol. 23, no. 1, pp. 568-591, 2025. doi: 10.1007/s13177-025-00471-2

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of GBP and/or the editor(s). GBP and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.