Dean&Francis
ISSN 2959-6157

Real-Time Pilot Fatigue Monitoring: A YOLO-Based Deep Learning Approach

Jiayi Peng[1, *]

[1]College of Astronautics, Nanjing University of Aeronautics and Astronautics, Nanjing, 210000, China
Corresponding author: 182310703@nuaa.edu.cn

Abstract:
Pilot fatigue constitutes one of the most critical factors compromising flight safety. Currently, the majority of deep learning algorithms focus primarily on improving detection accuracy, while neglecting the critical requirements of real-time performance and lightweight design essential for fatigue driving detection systems. Therefore, this paper proposes an enhanced approach based on the YOLOv8 object detection network. The improved network architecture, integrated with a fatigue driving assessment mechanism, significantly accelerates detection speed while maintaining robust performance. First, a lightweight VanillaNet architecture is adopted to replace the traditional backbone network of YOLOv8, effectively reducing both the computational complexity and parameter count of the model. Subsequently, the original bounding box regression loss function is replaced with a Wise-IoU loss function incorporating a dynamic non-monotonic focusing mechanism, which enhanced the model's precision. Experimental results demonstrate that the VanillaNet architecture reduces the model's parameter count by 2 million, while the enhanced Wise-IoU loss function improves detection accuracy by 6.1%, collectively achieving real-time performance and lightweight requirements for fatigue driving detection.

Keywords: Fatigue driving detection; YOLO; VanillaNet; Wise-IoU; real-time

1. Introduction

In recent years, pilot fatigue has emerged as a growing concern within the aviation community. Pilot fatigue constitutes a critical factor compromising flight safety. According to NASA statistics, 60-80% of aviation accidents are attributed to human error, with fatigue contributing to 20.1% of reported flight incidents. Relevant studies indicate that pilot fatigue significantly impairs fundamental cognitive and performance capacities. Furthermore, fatigue contributes to emotional alterations, diminished motivation, and an elevated risk of operational errors among flight crew members. Therefore, the real-time and accurate detection of pilot fatigue, coupled with immediate feedback to the pilot or supervisory management sys-

tems, has become an urgent issue to be addressed in the field of aviation safety.

Numerous scholars worldwide have engaged in research on fatigue detection for motor vehicle drivers and have achieved remarkably significant results. PERCLOS is a vital physiological metric for assessing driver fatigue levels, which determines fatigue states by calculating the percentage of time eyes remain closed over a specific period. Research indicates that the PERCLOS algorithm defines fatigue driving when a driver's eyelid closure time exceeds 80% within a one-minute monitoring window [1,2]. In 2019, CNN was initially adopted for fatigue driving detection, followed by the introduction of three prominent object detection architectures—SSD, Faster R-CNN, and YOLO—to enhance the precision of fatigue state recognition. Joiner's team implemented an enhanced YOLOv3 model to detect driver's facial features for fatigue assessment, utilizing the Pascal dataset annotated with Labelimg for model training and evaluation. The final overall detection accuracy reached 98% [3]. In 2023, Li Hao's team employed a MobileNetV3-CloUs enhanced YOLOv5 model to detect driver fatigue by monitoring eye closure states and yawning behaviour, utilizing multi-criteria evaluation to achieve efficient fatigue state identification[4]. However, these models exhibit relatively slow detection speeds, substantial parameter volumes, and high computational complexity. Consequently, achieving real-time performance and lightweight deployment has become a critical challenge in pilot fatigue detection technology.

To address these challenges, this paper proposes a fatigue detection method that simultaneously achieves real-time performance and lightweight design: By integrating the PERCLOS criterion with an optimized YOLOv8 architecture, we develop an integrated real-time pilot fatigue monitoring system. The enhanced YOLOv8 network accelerates detection speed while reducing both parameter count and computational complexity, ultimately realizing lightweight and real-time fatigue detection for pilots.

2. Theoretical and Technical Foundations of Fatigue Driving Detection

2.1 PERCLOS: Principles and Criteria for Detection

Currently, the PERCLOS algorithm is widely adopted for detecting driver fatigue states. This algorithm quantifies fatigue levels by calculating the proportion of eye closure duration over a specified time window. Studies have confirmed a positive correlation between driver fatigue levels and the PERCLOS metric, indicating that increased fatigue leads to a higher percentage of eye closure time over a defined monitoring period[5].

The PERCLOS-based fatigue detection pipeline primarily involves two stages: initial facial feature extraction to determine eye closure states, followed by quantitative computation of eye closure duration ratio through the PERCLOS algorithm. The PERCLOS algorithm incorporates three distinct evaluation criteria: EM, P70, and P80[6]. The EM criterion, defined as the proportion of time during which the pupil area is obscured by eyelids by over 50%, represents a relatively lenient threshold for fatigue assessment. The P70 criterion is defined as the proportion of time during which the pupil area is obscured by eyelids exceeding 70%, while P80 requires a more stringent 80% occlusion threshold. Notably, the P80 standard represents the most rigorous evaluation benchmark among the three metrics. Three distinct criteria can be adapted to accommodate more diverse and complex scenarios, thereby enabling safer and more reliable detection capabilities across different contexts.

The mathematical formulation for calculating the PERCLOS value is defined as follows:

$$PERCLOS = \frac{frame_{wink}}{frame_{sum}} \times 100\% \quad (1)$$

In the above equation, the variable *frame sum* denotes the total number of video frames captured per unit time, whereas *frame wink* indicates the number of frames identified as eye closure events during the same temporal interval.

Research finding indicates a consistent mapping relationship between PERCLOS values and driver fatigue levels, where the metric demonstrates a monotonic increase with escalating fatigue severity while maintaining strong linear correlation properties[7]. Experimental result indicates that PERCLOS values below 0.03 correspond to a normal state, values between 0.03 and 0.21 represent moderate fatigue, and values exceeding 0.21 indicate severe fatigue. Statistical data from the experiments are presented in Table 1[7].

Table 1. Statistical Analysis of Fatigue Driving Dat

	Mean Value	Standard Deviation	Between-Group Standard Deviation	Count
Blink Count	22.6	14.4	5.002	120

ISSN 2959-6157

Steering Wheel Rotation Count	4.6	1.4	2.383	120
Average Steering Wheel Rotation Amplitude	24.8	6.2	2.530	120
PERCLOS(P80)	0.14	0.023	0.018	120

2.2 Object Detection Algorithm

2.2.1 Overview of object detection algorithm

Object detection algorithm represents a critical technology in the field of computer vision, designed to identify target objects within images or videos while simultaneously determining their spatial locations and categorical labels. The workflow of object detection algorithm primarily comprises training and testing phases, with a detailed flowchart illustrated in Figure1.Object detection algorithm is commonly categorized into two primary classes: machine learning-based approaches and deep learning-based methodologies. Machine learning-based detection algorithms represent traditional approaches in object detection, typically involving the extraction of image features followed by regional classification and regression using specialized classifiers, ultimately employing object detectors for target verification. In contrast, deep learning-based detection algorithms employ deep neural networks for both feature learning and object detection. Currently, deep learning-based detection algorithms can be broadly categorized into two major paradigms: two-stage detectors and single-stage detectors. The YOLOv8 model adopted in this study is a one-stage algorithm, which performs object localization and classification directly from full-image features, enabling a substantial increase in detection speed.

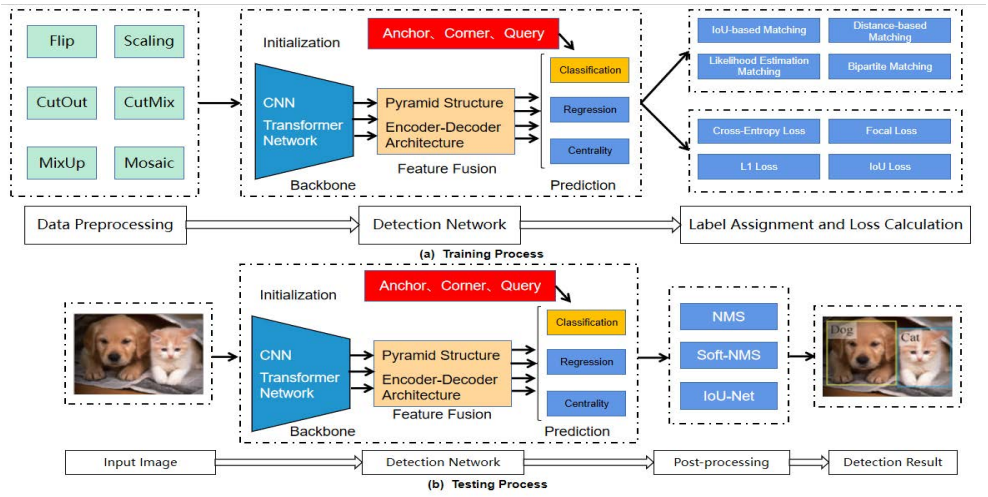

Fig. 1 Flowchart of the object detection algorithm

2.2.2 Development of the YOLO series models

The YOLO model was initially proposed to address limitations in existing two-stage algorithms, notably the complex pipeline and slow inference speed. The core concept of the YOLO model can be summarized into three key aspects: dividing the input image into grids, predicting bounding boxes with associated confidence scores and class probabilities for each grid, and applying post-processing operations such as Non-Maximum Suppression to refine the output bounding boxes. This concept significantly streamlines the detection pipeline, enabling end-to-end learning of the mapping from input images to detection outcomes while leveraging global contextual information for simultaneous object localization and classification. Furthermore, across subsequent iterations from the initial YOLOv1 to the state-of-the-art YOLOv8, researchers have progressively enhanced detection accuracy and efficiency through the integration of advanced feature extraction networks and multi-scale prediction strategies. Consequently, the YOLO series models have been extensively deployed in real-time critical applications such as autonomous driving, fatigue detection, and robotic vision, where high computational efficiency and low latency are paramount. For this experiment, the YOLOv8 model—the latest iteration in the YOLO series—was selected as the detection framework. YOLOv8 incorporates an anchor-free design philosophy, reformulates the loss function and label assignment strategy, and streamlines the model architecture. Additionally, YOLOv8 introduces the CBS, C2f, and SPPF modules, which collectively reduce

computational overhead, enhance multi-branch feature extraction capabilities, and improve recognition accuracy.

3. Real-Time and Lightweight Algorithm for Fatigue Driving Detection Based on YOLOv8

3.1 Architecture of the YOLOv8 Model

YOLOv8 offers multi-resolution object detection networks and instance segmentation models[8]. The improved activation function and optimized loss function in the model enhance both the detection speed and accuracy of YOLOv8 when handling complex tasks and scenarios. The architecture of the YOLOv8 model is illustrated in Figure 2[9].

The YOLOv8 detection model can be divided into three primary components: a feature extraction network, a feature fusion network, and a detection head[9]. The feature extraction network employs the CSP-Darknet53 architecture, which comprises a series of convolutional layers and residual blocks. It is primarily structured into three modules: the Convolutional Module, the CSP Module, and the SPPF Module. Inspired by the ELAN structure of YOLOv7, the CSP module is introduced as an improvement over the C3 module in YOLOv5. The introduction of the CSP module enhances the model's feature fusion capability, significantly improving the performance of the convolutional neural network in multi-scale and high-complexity tasks. The SPPF module is retained and refined from the Spatial Pyramid Pooling layer in the YOLOv5 architecture. Its application effectively integrates local and global features, thereby enhancing the YOLOv8 model's accuracy in detecting objects at various scales.

The feature fusion network of YOLOv8 employs a PAN-FPN architecture. This architecture enhances the model's perception of object features at various scales by constructing a multi-scale feature pyramid and facilitating beneficial information flow across different layers. The detection head of YOLOv8 employs a mainstream decoupled design, which utilizes two parallel branches for bounding box regression and classification, respectively. This prediction paradigm enables the YOLOv8 model to achieve superior detection accuracy with lower computational costs, thereby optimizing computational efficiency and enhancing performance in complex scenarios.

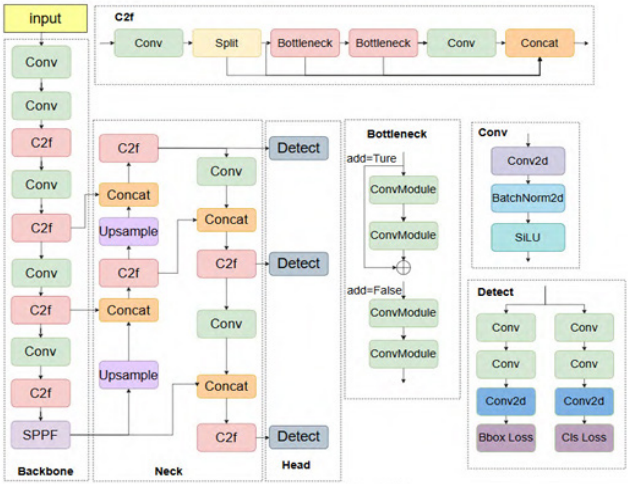

Fig. 2 Architecture of the YOLOv8 Model

3.2 YOLOv8: Algorithm and Function Optimization

Although the YOLOv8 model has achieved notable success in the general field of object detection, its application to driver fatigue detection still presents certain limitations, including suboptimal operational efficiency, high computational cost, and insufficient real-time performance. To address the challenges of lightweight and real-time pilot fatigue detection, this paper proposes an optimized architecture based on modifications to the YOLOv8 framework. Specifically, the C2f module in the CSP-Darknet53 backbone is replaced with a VanillaNet block, effectively reducing computational complexity and significantly improving inference speed. Furthermore, this paper introduces the Wise-IoU loss function with a dynamic non-monotonic focusing mechanism to enhance localization accuracy. Compared to the CIoU loss used in the baseline model, WIoU achieves superior prediction performance, leading to an overall improvement in detection capability. The architecture of the improved model is illustrated in Figure 3[10,4].

ISSN 2959-6157

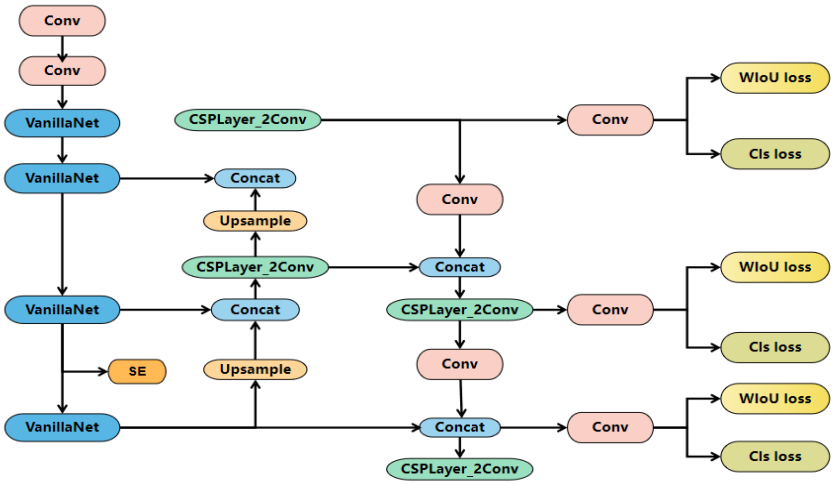

Fig. 3 Framework of the improved YOLOv8 Model

3.2.1 VanillaNet backbone

VanillaNet is a novel neural network architecture that emphasizes simplicity and elegant design. VanillaNet avoids deep structures by comprising only fundamental convolutional and pooling layers, while eschewing complex connections and operations. Experimental results demonstrate that the 13-layer VanillaNet model achieves an accuracy of 83% on the ImageNet dataset, indicating that its performance in computer vision tasks is comparable to that of deep and complex networks[10]. Furthermore, the architecture of VanillaNet reduces computational cost and the number of parameters, which enhances detection efficiency and allows the model to maintain competitive performance while achieving a lightweight design.

Taking the 6-layer VanillaNet as an example, the architecture comprises 5 convolutional layers, 5 pooling layers, 1 fully connected layer, and 5 activation functions. This architecture can be divided into three components: the Stem Block, the backbone network, and the fully connected layer. In the Stem Block, a convolutional layer is employed to project the input image from 3 channels to C channels, with a stride of 4. The backbone network comprises four stages. Each of the first three stages consists of a 1×1 convolutional layer followed by a 2×2 max-pooling operation with a stride of 2, which halves the spatial size and doubles the number of channels of the feature maps stage by stage. In contrast, the fourth stage employs a 1×1 convolutional operation followed by an average pooling layer, while the number of output channels remains unchanged. The final layer is a fully connected layer that produces the classification output by mapping the high-dimensional features to class probabilities. The VanillaNet architecture is shown in Figure 4 [11].

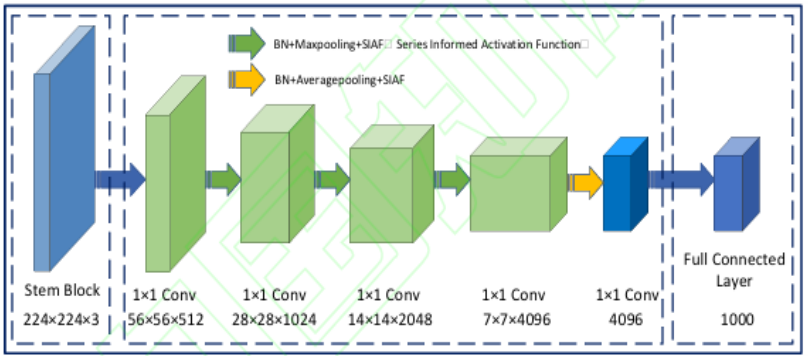

Fig. 4 The VanillaNet architecture

3.2.2 Improvement of the WIoU loss function

In the YOLOv8 model, the loss function primarily consists of three components: classification loss, bounding box regression loss, and object confidence loss[4]. The classification loss is the binary cross-entropy loss, computed as the average of the BCE losses over the N objects. The bounding box regression loss is primarily composed of the IoU Loss and the DFL Loss, which are employed to quantify the discrepancy between the predicted and ground-truth target bounding boxes. The confidence loss,

implemented via Focal Loss or BCE Loss, mitigates gradient interference from low-quality detection boxes, thereby enhancing the trade-off between recall and precision. However, in processing pilot fatigue detection data, the presence of low-quality samples is often unavoidable. Since both IoU Loss and DFL Loss rely on calculating geometric parameters such as distance and aspect ratio to measure the discrepancy between predicted and ground-truth values, the presence of such samples can amplify the computed loss values. This may lead to slower convergence and reduced training efficiency, thereby adversely affecting the model's generalization performance.

Therefore, this paper adopts the Wise-IoU loss function to replace the original IoU-based bounding box regression loss. The Wise-IoU loss function incorporates a dynamic non-monotonic focusing mechanism that utilizes outliers instead of IoU for anchor box quality assessment, along with a rationally designed gradient gain allocation strategy. This strategy mitigates large or detrimental gradients arising from low-quality examples, thereby enabling the model to focus more on average-quality samples and ultimately enhancing the overall model performance.

Building upon WIOUv1, which incorporates an attention mechanism, WIOUv3 is developed by introducing a non-monotonic focusing factor, thereby constructing a dynamic non-monotonic focusing mechanism. The formulation of the WIOUv1 loss function is given by Eqs. (2) to (4).

$$L_{WIOUv1} = R_{WIOU} \times L_{IOU} L_{WIOUv1} = R_{WIOU} \times L_{IOU} \tag{2}$$

$$R_{WIOU} = \exp(\frac{(a-a_{gt})^2 + (b-b_{gt})^2}{(C_w^2 + C_h^2)}) \tag{3}$$

$$L_{IOU} = 1 - IOU \tag{4}$$

In the above equations, R_{WIOU} denotes the penalty coefficient of the relative position deviation between the predicted and ground-truth boxes on the IoU loss. L_{IOU} represents the non-overlapping degree between the boxes. The coordinates (a,b) and (a_{gt}, b_{gt}) are the center points of the predicted and ground-truth boxes, respectively. C_w and C_h are the width and height of the smallest enclosing rectangle covering both boxes. IOU is the ratio of the intersection area to the union area of the two boxes.

The WIOUv3 loss is designed with a dynamic non-monotonic mechanism for anchor box quality assessment. This design encourages the model to prioritize ordinary-quality boxes, which significantly improves localization accuracy. The formulation of the WIOUv3 loss function is given by Eqs. (5) to (7).

$$L_{WIOUv3} = r \times L_{WIOUv1} \tag{5}$$

$$r = \frac{\beta}{\delta \alpha^{\beta - \delta}} \tag{6}$$

$$\beta = \frac{L_{IOU}^*}{L_{IOU}} \in [0, +\infty) \tag{7}$$

In the above equation, L_{IOU}^* denotes the monotonic focusing coefficient, r represents the gradient gain, β denotes the non-monotonic focusing coefficient, and α、δ represents the hyperparameter.

The WIoU v3 loss function is a superior choice compared to the original YOLOv8 loss for driver fatigue detection scenarios. This approach facilitates the training of low-quality samples and balances their impact against high-quality ones, leading to an overall performance improvement.

4. Model Training

4.1 Dataset and Experimental Setup

The dataset is of paramount importance in research on deep learning algorithms and models. This study employs the publicly available Fdd-Dataset, which was collected using an in-vehicle camera that recorded drivers' facial expressions across various states, including expressions of fatigue such as eye closure and yawning. The entire dataset consists of 2,914 images, including 545 images representing a fatigue state. For the experiments, the dataset was partitioned into training, validation, and test sets following a 3:1:1 ratio[12].

The study selected Python as the programming language and PyCharm as the development environment for model training and validation. The specifications of the experimental setup are detailed in Table 2[10].

Table 2. Experimental Environment Configuration

Experimental Environment	Configuration
Operating System	Windows10
CPU	Inter(R)Core(TM)i7-13620
GPU	Nvidia GeForceRTX4060Ti
Memory	16GB
Video Memory	8GB

ISSN 2959-6157

Programming Language	Python3.9
Deep Learning Framework	Pytorch-GPU2.0.1

4.2 Evaluation Metric

In this experiment, the following performance metrics are employed to evaluate the fatigue driving detection model: Precision, Recall, and mean Average Precision. Precision refers to the proportion of samples predicted as fatigued driving that are truly fatigued. Recall, on the other hand, is defined as the proportion of actual fatigued driving samples that are correctly identified by the model[13]. Average Precision is obtained by integrating the Precision-Recall curve, reflecting the model's precision performance across the entire range of recall. The mean Average Precision is the average of the AP values over all categories, calculated as[12]:

$$P = \frac{TP}{TP + FP} \tag{8}$$

$$R = \frac{TP}{TP + FN} \tag{9}$$

$$AP = \int_0^1 P(R)dR \tag{10}$$

$$mAP = \frac{\sum_{n=0}^{N} AP_n}{N} \tag{11}$$

In the above equation, TP represents the number of samples correctly identified as fatigue driving, FP denotes the number of non-fatigue driving samples incorrectly classified as fatigue driving by the model, FN indicates the number of actual fatigue driving samples that the model failed to detect, N is the total number of categories, and AP_n refers to the Average Precision value for the n-th category.

4.3 Experimental Results and Analysis

4.3.1 Comparative experiments

To further minimize the model's parameter count and improve computational efficiency, this paper conducts a comparative experiment pitting the lightweight network VanillaNet against the original YOLOv8n model. The experimental data are shown in Table 3[10]. Comparative analysis reveals that the lightweight network VanillaNet significantly reduces the number of model parameters to merely 1.2 million. Although slight declines are observed across the three core performance metrics, both precision and recall remain at relatively high levels. This indicates that VanillaNet achieves a notable improvement in computational efficiency by sacrificing marginal performance while maintaining high precision and recall.

Table 3. Comparison Table of Backbone Networks

Model	Params(M)	P(%)	R(%)	mAP(%)
YOLOv8n	3.2	84.9	76.6	83.1
VanillaNet	1.2	84.3	73.2	80.1

4.3.2 Ablation study

To quantitatively analyze whether the proposed WIoU loss function improves fatigue detection performance, the paper employs an ablation study for validation. A comparative test was conducted by training the baseline YOLOv8 on the experimental dataset with the only modification being the introduction of the WIoU loss function, ensuring a single-variable premise. The results are summarized in Table 4 [4]. The ablation study reveals that the introduction of the WIoU loss function improves the model accuracy by 6.1% compared to the baseline YOLOv8, albeit at the cost of increased computational overhead.

Table 4. Ablation Study Results

Model	Params(M)	mAP(%)
YOLOv8s	14.54	83.5
YOLOv8s+Wiou	14.81	88.6

The above experimental results demonstrate that integrating the lightweight VanillaNet into the YOLOv8 architecture effectively reduces computational complexity and the number of parameters. Concurrently, the improved WIoU loss function enhances model sensitivity to regions of interest, thereby increasing detection accuracy. The synergistic application of these two components not only further boosts the detection precision of the YOLOv8 model

but also significantly reduces its computational load, ultimately leading to a lightweight and real-time fatigue detection model.

5. Summary

In recent years, fatigue driving has become a significant concern, coinciding with the growing emphasis on aviation safety. However, existing fatigue detection systems often suffer from high computational cost and poor real-time performance. Therefore, this paper proposes an improved, lightweight, real-time fatigue detection model based on YOLOv8. By incorporating the lightweight design of VanillaNet to reconstruct the backbone of YOLOv8, the model effectively reduces computational complexity and the number of parameters. Additionally, by replacing the original bounding box regression loss with the improved WIoU loss, the issue of low-quality training samples is effectively addressed, leading to enhanced detection accuracy. Consequently, the model maintains high detection accuracy while significantly reducing both the model size and computational demands, leading to an efficient, lightweight, and real-time solution for fatigue detection.

References

[1] W. Zhang, “Research on fatigue driving detection method based on deep learning and system implementation”, M.S. thesis, Xi’an University of Science and Technology, 2023.

[2] Y. You, «Research on pilot fatigue monitoring technology based on machine vision», M.S. thesis, University of Electronic Science and Technology of China, 2011.

[3] L. Ma, W. Qi, Y. Chen, and X. Han, «Lightweight fatigue driving detection method based on improved Yolov8», in Proceedings of the 43rd Chinese Control Conference (13), Kunming, 2024, pp. 249-254.

[4] H. Li, “Design and implementation of a driver fatigue detection system based on improved YOLOv5”, M.S. thesis, Hebei University of Science and Technology, 2024.

[5] T. Zhu, C. Zhang, T. Wu, Z. Ouyang, H. Li, X. Na, J. Liang, and W. Li”, Research on a real-time driver fatigue detection algorithm based on facial video sequences”, Applied Sciences. 2022; 12(4):2224.

[6] X. Zou, “Research on driver’s multi-feature fatigue detection system based on machine vision”, M.S. thesis, Shandong Jiaotong University, 2025.

[7] B. Yang, “Research and design of an alarm for driver fatigue in automobile drivers”, M.S. thesis, Fuzhou University,2005.

[8] Z. Guo, J. Liu, D. Qiu, and Y. Li, “A survey of YOLO algorithms for 2D medical image detection”, Journal of Frontiers of Computer Science and Technology, early access, pp. 1-22, Jun. 6, 2025. [Online]. Available: https://link.cnki.net/urlid/11.5602.TP.20250605.1513.004.

[9] Z. He, M. Jiang, M. Xie, et al. “Design of driver fatigue recognition algorithm based on YOLOv8 and facial keypoint detection”, Computer & Telecommunication, no. 11, pp. 1-6+13, 2023.

[10] Z. Ding, “Research on lightweight fatigue driving detection based on improved YOLOv8”, M.S. thesis, Dalian Jiaotong University, 2025.

[11] J. Tang and A. Pang, «Traffic police gesture recognition based on improved YOLOv8 algorithm”, Journal of Wuhan Textile University, early access, pp. 1-7, Jul. 17, 2025. [Online]. Available: https://link.cnki.net/urlid/42.1818.Z.20250717.1211.002.

[12] S. Yang and S. Peng, “Train driver fatigue detection algorithm based on YOLOv8n”, Information Technology and Informatization, no. 4, pp. 70-73, 2025.

[13] Y. Yang, L. Li, and H. Lin, “A survey of fatigue driving detection by extracting driver’s facial features”, Journal of Frontiers of Computer Science and Technology, vol. 17, no. 6, pp. 1249-1267, 2023.