A Dangerous Driving Behavior Detection Method Based on Improved YOLOv8s

Tong Zhou, Xiaoxia Zhang, Huilong Chen

Abstract—**Detecting dangerous driving behavior is a critical research area focused on identifying and preventing actions that could lead to traffic accidents, such as smoking, drinking, yawning, and drowsiness, through technical methods. Advanced computer vision and machine learning technologies enable efficient detection models to monitor and analyze driver behavior, improving road safety. Due to challenges posed by complex environments, this paper introduces an enhanced detection algorithm, YOLOv8s-CDS, to improve the identification of dangerous driving behaviors. First, the ConvNeXt V2 module is integrated with the C2f module to form C2fNeb2, optimizing feature extraction for behaviors like smoking or phone use. Second, the DASI (Dimension-Aware Selective Integration) module enhances detection accuracy through multi-scale fusion and dimension perception. Additionally, the SCConv module replaces the Conv module in the Bottleneck, forming C2fSCConv, which reduces spatial redundancy and improves detection efficiency.A comprehensive experimental analysis of dangerous driving image datasets demonstrates that the mean average precision (mAP) of the YOLOv8s-CDS algorithm is 91.20%, which is 2.4% higher than that of the YOLOv8s algorithm. Compared to other object detection algorithms, such as Faster R-CNN, YOLOv5s, YOLOv7s, YOLOX and YOLOv8, YOLOv8s-CDS demonstrates greater practicality in detecting dangerous driving behaviors, contributing to a reduction in traffic accidents and enhancing the safety of life and property.**

Index Terms—**dangerous driving, YOLOv8s, ConvNeXt V2, DASI, SCConv**

I. Introduction

IN recent years, the rapid increase in motor vehicles and the expansion of the logistics industry have led to a rise in traffic accidents, threatening public safety and social and economic development. Research shows that dangerous driving behaviors are a major cause of traffic accidents, resulting in significant harm to life, health, and social security. As driver-related factors are the primary cause of these accidents, studying technologies for detecting dangerous driving behaviors is of great scientific importance.

Studying and detecting dangerous driving behaviors [1] can improve road safety, reduce traffic accidents, and protect lives and property. This paper uses deep learning to train a model for detecting dangerous behaviors in driving images, with the goal of mitigating such behaviors and preventing accidents. These detection technologies are often integrated into vehicle safety or driver assistance systems, which can alert drivers to maintain safety or interact with vehicle controls to mitigate hazards. Additionally, they can monitor driver behavior and provide data to assess driving habits, promoting better driving practices and heightened safety awareness.

Researchers are increasingly focused on detecting dangerous driving behaviors, exploring both indirect and direct detection [2] methods. Indirect detection monitors factors like vehicle speed, steering wheel movements, and lane positions [3] using sensors. It tracks the vehicle's speed, observes steering changes, and analyzes surrounding conditions to detect behaviors such as illegal lane changes. While effective, these methods are influenced by road conditions and vehicle abnormalities, which can reduce accuracy. Direct detection, on the other hand, monitors the driver's physiological state, including heart rate, blood pressure, and brain activity, to assess potential danger. While theoretically feasible, this approach requires complex equipment in the vehicle, which can interfere with driving and presents challenges in practical implementation.

The driver behavior-based method assesses dangerous driving by monitoring behavior patterns [4], such as phone use, smoking, checking the back seat, talking to passengers, or falling asleep. With advances in deep learning and object detection, this approach is gaining popularity as it does not interfere with the driver and enables efficient data processing. A key example is using computer vision for dangerous driving detection, where image data is collected, driver behavior is extracted, and features are classified. This method overcomes the limitations of environmental sensitivity and the interference of wearable devices, making computer vision-based detection a superior approach for identifying dangerous driving behaviors.

Gao Wen [5] enhances the deep neural network Mask R-CNN for object detection by improving its ability to accurately locate object borders. Experimental results show that this method achieves nearly 90% accuracy in detecting dangerous driving behaviors, outperforming other algorithms, especially for overlapping and small targets. In recent years, first-order algorithms, which bypass candidate frame generation and directly extract features for object classification and localization, have gained popularity. These include SSD [6], the YOLO series, and RetinaNet [7], which are increasingly used in dangerous driving behavior detection. First-order algorithms offer faster detection and more efficient feature extraction. Ji Chenjin [8] introduced the Convolutional Block Attention Module (CBAM) into the YOLO layer, creating the improved C-YOLOv3 structure. This modification reduced background interference and significantly boosted detection accuracy. Using 8,000 images from the StateFarm distracted driver detection dataset and 1,000 real driving images, the

Manuscript received June 23, 2024; revised January 23, 2025.

T. Zhou is a postgraduate student of School of Computer Science and Software Engineering, University of Science and Technology LiaoNing, Anshan, 114051, China (e-mail: 3223464820@qq.com).

X. X. Zhang is a Professor of School of Computer Science and Software Engineering, University of Science and Technology LiaoNing, Anshan, 114051, China (corresponding author, phone: 86-0412-5929812; e-mail: aszhangxx@163.com).

H. L. Chen is a postgraduate student of School of Computer Science and Software Engineering, University of Science and Technology LiaoNing, Anshan, 114051, China (e-mail: changango12138@163.com).

model achieved a 94.41% accuracy rate, demonstrating its superior performance in detecting dangerous driving behaviors and minimizing background noise, making it highly suitable for real-world driver behavior detection.

The YOLO series algorithms offer significant theoretical and practical value in dangerous driving behavior detection. Their fully convolutional network structure enables global perception of the entire image, which is crucial for detecting behaviors that span the whole cab scene, not just a specific area. Additionally, YOLO transforms object detection into a regression problem, allowing for the simultaneous detection of multiple targets in one forward pass, which is especially useful for identifying various dangerous driving behaviors in a single image. As end-to-end systems, YOLO algorithms learn target representation and detection through a single training process, simplifying model design and maintenance. While YOLO is known for its speed, its accuracy, especially for small targets, can be limited. To address this, researchers have continuously refined and developed newer versions of YOLO to improve its performance in detecting dangerous driving behaviors.

In summary, this paper enhances the YOLOv8s [9]–[11] algorithm by improving both the Backbone feature extraction and Neck feature fusion components. Three additional modules are introduced to more accurately detect a range of dangerous driving behaviors, leveraging recent advancements in computer vision and deep neural network models.

II. TERIALS AND METHOD

A. YOLOv8 Object Detection Algorithm

YOLOv8 is an efficient real-time object detection algorithm introduced by Joseph Redmon et al. in 2020. The core concept of YOLOv8 is to frame the object detection task as an end-to-end regression problem. It directly predicts the location and category of objects within an input image using a single neural network model, streamlining the process from the original image to the detection result. This approach simplifies the training process, reduces computational costs, and minimizes the need for manual feature engineering. At the same time, YOLOv8 delivers high real-time performance and maintains excellent detection accuracy. It is capable of processing large-sized images while sustaining a high frame rate.

Through multiple iterations and enhancements, YOLOv8 has significantly improved in both performance and speed. It incorporates a deeper network structure and employs various optimization techniques, such as feature pyramid networks, cross-scale connections, and multi-scale prediction, to enhance detection accuracy and the ability to detect small targets. Additionally, YOLOv8 integrates technologies like Bag of Freebies and Bag of Specials, further boosting the model's performance and generalization capabilities.

B. ConvNeXt V2

This paper replaces the C2f module in YOLOv8 with the ConvNeXt V2 module [12], [13], which shows significant improvements over ConvNeXt V1. ConvNeXt V2 integrates Transformer-inspired design concepts while maintaining the efficiency of traditional CNNs. It aims to enhance CNN performance, particularly in tasks like dangerous driving behavior detection. In YOLOv8s, ConvNeXt V2 serves as the backbone network to improve target detection capabilities and overall performance on large-scale datasets.

This paper integrates the ConvNeXt V2 module into the YOLOv8s backbone to improve small target feature extraction and overall image feature learning. The module uses multi-layer convolution with progressively larger kernels for multi-scale feature learning, followed by a fully connected layer for classification.

ConvNeXt V2 introduces two innovations: the Fully Convolutional Masked Autoencoder (FCMAE) and Global Response Normalization (GRN). The GRN layer normalizes feature maps on each channel, enhancing feature competition. It offers two advantages over Batch Normalization (BN): no additional parameters and support for variable batch sizes without extra computational overhead.

The GRN layer is implemented in three steps: global feature aggregation using the L2 norm, feature normalization, and feature calibration. It requires only three lines of code and no learnable parameters. The GRN layer has low computational complexity, making it easy to integrate into CNNs to enhance feature competition and improve performance.

ConvNeXt V1 encountered a feature collapse problem due to redundant activations, such as dead or saturated neurons, between channels. To address this issue, ConvNeXt V2 introduces the GRN layer, which promotes feature diversity and mitigates the feature collapse problem. Diagrams of the ConvNeXt V1 and ConvNeXt V2 models are shown in Fig. 1.

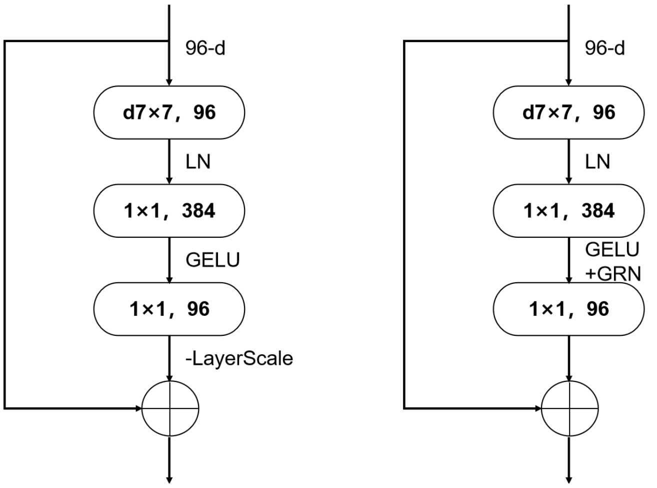

Fig. 1. ConvNeXt V1 (left) and ConvNeXt V2 (right) Model Diagram

Applying ConvNeXt V2 to YOLOv8s involves several steps: loading pre-trained models, modifying the YOLOv8 backbone, training and fine-tuning the model, and optimizing performance. ConvNeXt V2 enhances YOLOv8's feature extraction, improving detection precision. The integration process is divided into three parts: replacing YOLOv8's C2f module with ConvNeXt V2 for high-level feature extraction, using a hierarchical ConvNeXt V2 architecture with convolution blocks, and passing the extracted features to YOLOv8's head for final detection results. The model is further optimized through transfer learning and end-to-end training.

Integrating ConvNeXt V2 as the backbone in YOLOv8

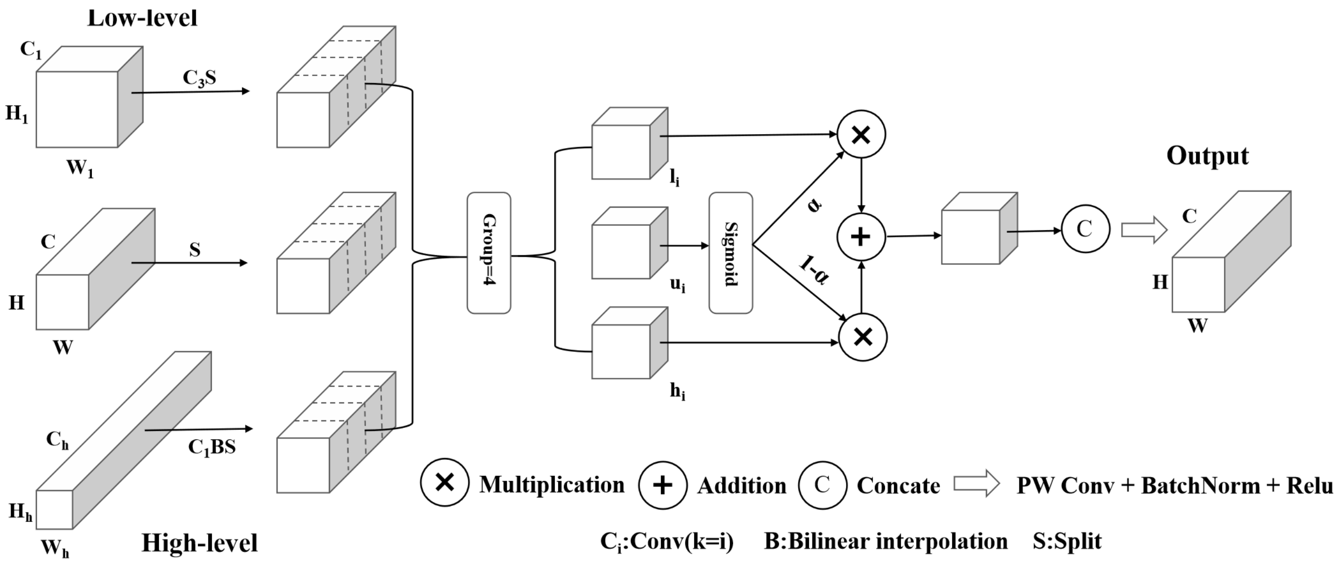

Fig. 2 DASI Module Structure Diagram.

enhances feature extraction, improving accuracy and efficiency in dangerous driving target detection. This combination leverages ConvNeXt V2's efficient convolution design and YOLOv8's powerful detection capabilities, excelling in complex dangerous driving behavior detection tasks.

C. Dimension-Aware Selective Integration (DASI)

The DASI (Dimension-Aware Selective Integration) module [14], [15] is designed to enhance feature expression capabilities in computer vision models, particularly for tasks such as object detection and image classification. The primary purpose of the DASI module is to selectively integrate multi-scale features through a dimension-aware mechanism, thereby improving model performance. The DASI module design is inspired by attention mechanisms and multi-scale feature fusion. By incorporating a dimension-aware mechanism, DASI effectively selects and integrates features across different scales, enhancing the model's feature expression capability and generalization performance. The structure of the DASI module is illustrated in Fig. 2.

The core part of the DASI module includes multiple components: firstly, multi-scale feature extraction is performed by using convolution kernels of different scales to capture objects of different sizes and detailed information in the input features; secondly, a dimensionality awareness mechanism is introduced to analyze the channels of the feature map. The dimensions and spatial dimensions are weighted to selectively emphasize important features, thereby paying more attention to the dimensional information of the features; finally, the multi-scale features that have been dimensionally weighted are fused to form the final output feature map.

The DASI module enhances dangerous driving target detection tasks by effectively integrating multi-scale features, which improves model accuracy. It is well-suited for processing complex scenes and multi-scale object detection tasks. The DASI module improves feature expression through its dimension-aware mechanism and multi-scale feature fusion, demonstrating strong performance in handling complex target detection tasks and significantly enhancing both target detection and image classification performance.

D. SCConv

The convolutional Conv module performs effectively in detecting dangerous driving behaviors but demands substantial computational resources and incurs high costs due to redundant features. To address these issues, a related paper introduced the SCConv module [16] , which stands for Spatial and Channel Reconstruction Convolutional Module. SCConv comprises two key components: the Spatial Reconstruction Unit (SRU) and the Channel Reconstruction Unit (CRU).

The Spatial Reconstruction Unit (SRU) mitigates spatial redundancy using a separation-reconstruction approach, while the Channel Reconstruction Unit (CRU) employs a segmentation-conversion-fusion strategy to reduce channel redundancy. The SRU applies convolution operations to merge information across the spatial dimension, capturing spatial features at various scales and enhancing the spatial structure of the input feature map. The Spatial Reconstruction Unit (SRU) extracts multi-scale spatial information using convolution kernels of various sizes and receptive fields, integrating this information into a feature map. The Channel Reconstruction Unit (CRU) performs fusion in the channel dimension to efficiently manage information interactions between channels, aiming to enhance the expressiveness of feature maps in this dimension. By adjusting channel weights, CRU strengthens important channel features while suppressing less relevant ones. Through spatial and channel reconstruction, the SCConv module extracts rich feature information across different scales, enhancing its ability to capture complex contextual relationships. By combining dual information reconstruction of both spatial and channel dimensions, SCConv improves the expressiveness of feature maps and demonstrates strong adaptability. Additionally, SCConv can be integrated into existing convolutional neural networks as an independent module, enhancing the performance of models for detecting dangerous driving behaviors.

SCConv enhances the feature extraction capabilities of convolutional neural networks by combining spatial and channel reconstruction strategies. Its approach to multi-scale and multi-dimensional information fusion enriches and refines the expression of feature maps, making it an effective method for improving model performance.

III. THE PROPOSED ALGORITHM

This section offers a thorough introduction to the enhanced YOLOv8s algorithm, designated as the YOLOv8s-CDS algorithm. It commences by delineating the primary structure of the algorithm and its detection methodology. Subsequently, this section elaborates on the architecture of ConvNeXt V2, emphasizing its advantages relative to other architectures and elucidating its integration into the model. Furthermore, it discusses the principles underlying the DASI module and its contributions within the framework. Additionally, the SCConv module is incorporated to augment the precision of detecting hazardous driving behaviors.

A. Improved YOLOv8s-CDS Network Structure

While YOLOv8s shares similarities with the stable YOLOv5, it exhibits a more streamlined design. This algorithm incorporates the Path Aggregation Network (PAN) concept, which effectively mitigates issues related to information loss and resolution degradation in semantic segmentation, thereby enhancing both accuracy and detail retention within the model. PAN employs a dual mechanism of feature propagation that includes both top-down and bottom-up pathways. The top-down pathway facilitates the transfer of high-level semantic information to low-level features, enabling the acquisition of more detailed data; conversely, the bottom-up pathway conveys low-level features to high-level semantic representations to furnish global context. By leveraging feature reuse and reorganization from the network's intermediate layers, PAN significantly reduces computational demands while improving efficiency. During the path aggregation phase, feature reorganization recombines information across various scales, resulting in richer and more precise semantic details. Through multi-scale feature fusion, PAN enhances its capacity to capture intricate object details alongside broader contextual information, thus bolstering performance, robustness, and generalization capabilities in semantic segmentation tasks. This methodology adeptly addresses challenges posed by variations in object scale and perspective.

The YOLOv8s algorithm initiates feature extraction through the backbone network, subsequently enhancing these features via both top-down and bottom-up pathways before generating predictions based on feature maps of three distinct sizes. The process unfolds as follows: a 640×640 three-channel image is inputted, followed by a convolutional layer that reduces its dimensions by half while augmenting the channel count to 64. This is succeeded by a series of convolutional layers combined with C2fneb2 modules, further diminishing the image size to 160×160 and increasing the number of channels to 128. This procedure is iterated four times, ultimately reducing the dimensions to 20×20 while elevating the channel count to 1024. Subsequently, the SPPF module executes pooling cascade operations. Features extracted from the backbone layer are then upsampled to double their original size and concatenated with corresponding feature maps obtained from the C2f module within the backbone architecture. The C2f module further refines these features, with this enhancement process repeated twice to achieve comprehensive top-down feature fusion. Finally, for processing bottom-level feature maps, these maps are scaled and concatenated with those derived from previous iterations of the C2f module at equivalent sizes. Following this scaling and enhancement using both C2f modules and convolutions results in feature maps across three different scales: small-sized feature maps predict large targets; medium-sized ones cater to medium targets; whereas large-sized ones focus on small targets. The structures of Conv, C2f, and SPPF modules are depicted in Fig. 3.

This paper presents an integration of the ConvNeXt V2, DASI, and SCConv modules into YOLOv8s, based on the algorithms and modules previously discussed. The architecture of the enhanced algorithm is depicted in the figure below. By incorporating these advanced modules and strategies, the YOLOv8s-CDS algorithm markedly improves the accuracy of detecting dangerous driving behaviors. The model structure for YOLOv8s-CDS is illustrated in Fig. 4.

B. Feature Extraction

To integrate ConvNeXt V2 into the YOLOv8s model and enhance its feature extraction capabilities, this study first prepares a pre-defined ConvNeXt V2 model, which is an advanced convolutional neural network specifically designed as an efficient and robust feature extractor for various computer vision tasks. Subsequently, it replaces the backbone module in YOLOv8s with this model. The next step involves adjusting the feature size and connections to ensure that the output dimensions of ConvNeXt V2 align with the input requirements of YOLOv8s. Finally, the improved YOLOv8s model undergoes training and debugging to confirm proper convergence and satisfactory performance on the validation set.

In the original YOLOv8s architecture, the C2f module is employed for feature extraction using standard convolutional layers. Furthermore, within C2f, a Bottleneck module implements a conventional bottleneck design that reduces computational complexity by utilizing two convolutional layers. To address overfitting and improve generalization capabilities, DropPath functionality is incorporated; this mechanism randomly discards portions of network paths during training.

This paper substitutes the Bottleneck module in C2f with ConvNeXt V2. The ConvNeXt V2 module utilizes 7×7 deep convolutions along with layer normalization and an activation function for effective feature extraction. Additionally, it integrates a path drop layer to randomly eliminate certain paths during training while combining inputs and outputs post-path dropping through residual connections.

Subsequently, we invoke the CNeB2 class to adjust channel dimensions across different convolutional layers so that features of varying dimensions are standardized to a common dimension. After introducing n instances of ConvNeXtV2Block2d modules, consistency among channels is ensured.

Finally, we define the C2fNeb2 module which calls upon its parent class CNeB2 to facilitate replacement within the overall model structure.

C. Feature Fusion

The modules are initialized at three distinct resolutions—high, medium, and low—to extract features accord-

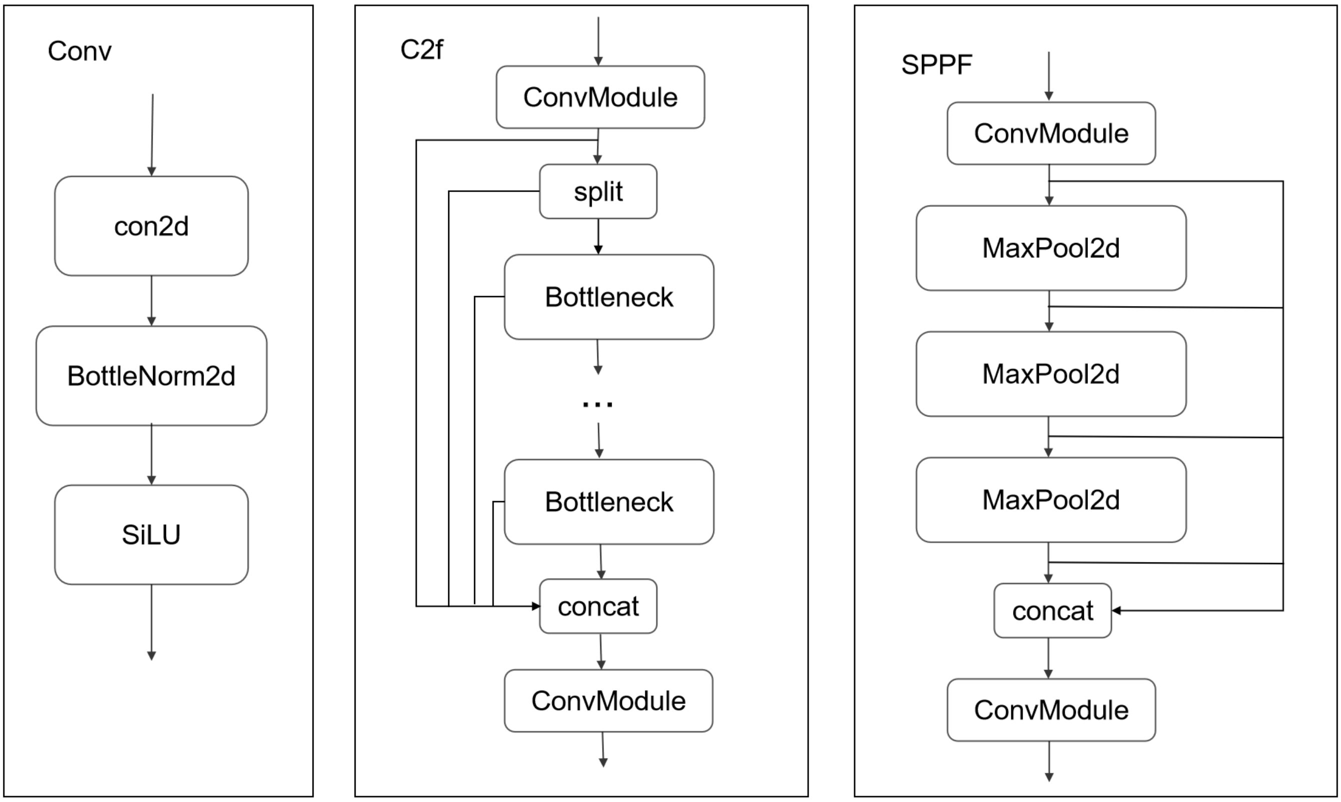

Fig. 3. The Structures of Conv, C2f and SPPF

ingly. The features obtained from the processed high-resolution and medium-resolution modules are separated into channels and stored in x_high_blocks and x_blocks, respectively. For the low-resolution module, bilinear interpolation is applied post-feature extraction to resize it before channel separation and storage in x_low_blocks. When high-resolution features are available while low-resolution features are absent, the high- and medium-resolution features are concatenated followed by convolutional processing. Conversely, if only low-resolution features exist without corresponding high-resolution data, the medium-resolution and low-resolution features undergo concatenation prior to convolution. In scenarios where both feature sets coexist simultaneously, aggregation of these two feature types occurs with repetition of the aforementioned steps. After concatenating all channel-specific features, residual connections integrate original feature information to prevent loss during processing. Finally, batch normalization along with activation functions is applied to yield the final output of extracted features.

This module enhances the model's capacity for capturing and fusing multi-scale resolution features through comprehensive multi-scale feature fusion techniques combined with convolution operations. Particularly within complex scenes, effective integration of these diverse feature representations significantly bolsters detection accuracy and robustness.

The SCConv module further augments feature extraction capabilities via a self-calibration mechanism that improves performance across convolutional neural networks (CNNs). Specifically designed to mitigate feature redundancy effectively, this module not only reduces model parameters but also computational demands while enhancing representational efficacy of extracted features. By offering a novel perspective on CNNs' feature extraction processes alongside methods for optimizing spatial and channel redundancies, SCConv contributes positively towards improving overall model performance whilst minimizing redundant elements.

Experimental results indicate that models incorporating the SCConv module achieve superior performance characterized by substantial reductions in complexity as well as computational costs due to diminished redundant characteristics within their architectures.

This study implements the SCConv module which facilitates decreased computational expenses alongside reduced model storage requirements while concurrently enhancing CNN performance through alleviation of prevalent spatial and channel redundancies found in standard convolutions. Through Spatial Redundancy Utilization (SRU) and Channel Redundancy Utilization (CRU), redundancy within feature maps is minimized leading to significant enhancements in performance coupled with marked reductions in computational load. This approach integrates seamlessly with the C2f module resulting in a composite structure termed C2fSCConv; a schematic representation illustrating both C2fSCConv's architecture along with its internal Bottleneck component can be found in Fig. 5.

IV. EXPERIMENT

A. Experimental Data

The dataset used in this paper is a collection of open source datasets of dangerous driving behaviors from the Internet, including the Advance Driver Monitoring System Dataset (ADMS) dataset, the Driver Inattention Detection Dataset and some online images. This dataset focuses on specific driver behaviors, such as drinking water, falling

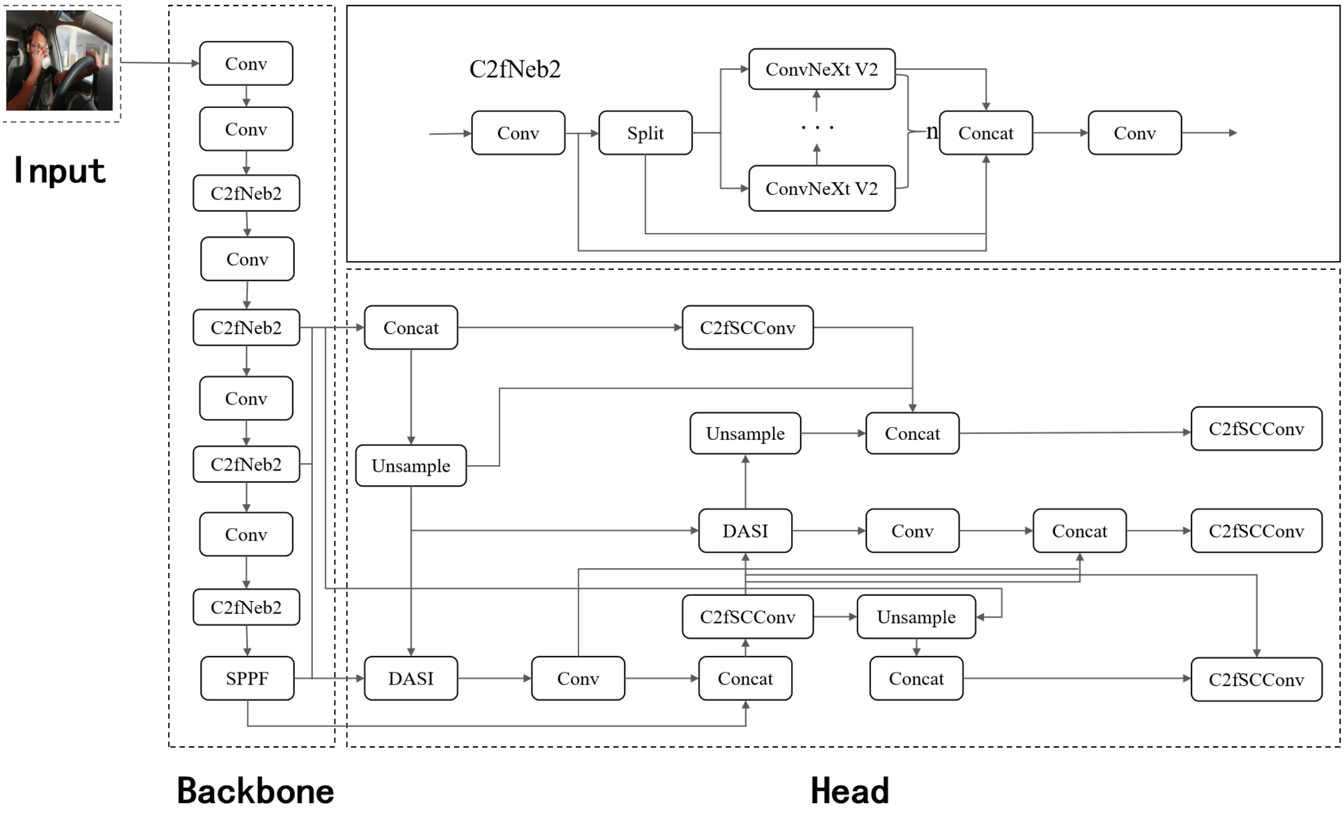

Fig. 4. The YOLOv8s-CDS Model Structure Diagram

asleep, playing with mobile phones, smoking, and yawning. The dataset contains 13,000 images in 5 different categories, providing a diverse and extensive collection for training and validation purposes. In order to train the model, we downsampled it, that is, randomly extracted about half of the 6,518 images, and divided the training set and validation set into a ratio of 8:2, including 5,215 images in the training set and 1,303 images in the validation set. The specific distribution of these five dangerous driving behaviors in the dataset is shown in Table I.

TABLE I
DISTRIBUTION TABLE OF DIFFERENT TYPES OF IMAGES IN THE DATASET

Dangerous Driving Behavior Image	train	val	total
drinking	673	170	843
drowsy	932	239	1171
phone	1840	425	2265
smoking	953	218	1171
yawning	817	251	1068
total	5215	1303	6518

B. Experimental settings

In this experiment, all input dangerous driving behavior images are uniformly resized to 640×640×3, and the epoch is 200. This configuration is shown in Table II.

TABLE II
SYSTEM CONFIGURATION

Configuration	Configuration Parameter
Operating System	Ubuntu 20.04.6 LTS
CPU	Intel® Core™ i7-10700F CPU @ 2.90GHz × 16
GPU	RTX 3090
Memory	24GB
Deep learning framework	PyTorch 1.10.1
CUDA version	Cuda 11.3
Integrated dev. environment	PyCharm

C. Performance Indicators

In order to evaluate the performance of the model proposed in this paper in more detail, this study uses precision, recall, mAP@0.5, frames per second (FPS), number of parameters and model size as evaluation indicators.

Precision refers to the proportion of dangerous driving behaviors detected that are actually dangerous driving, where true positives (TP) refer to the number of samples where the model correctly detects dangerous driving behaviors and represent the number of samples correctly classified as positive examples, that is, they are actually positive examples. False positives (FP) refer to the number of samples where the model incorrectly detects dangerous driving behaviors. The calculation formula is as follows (1):

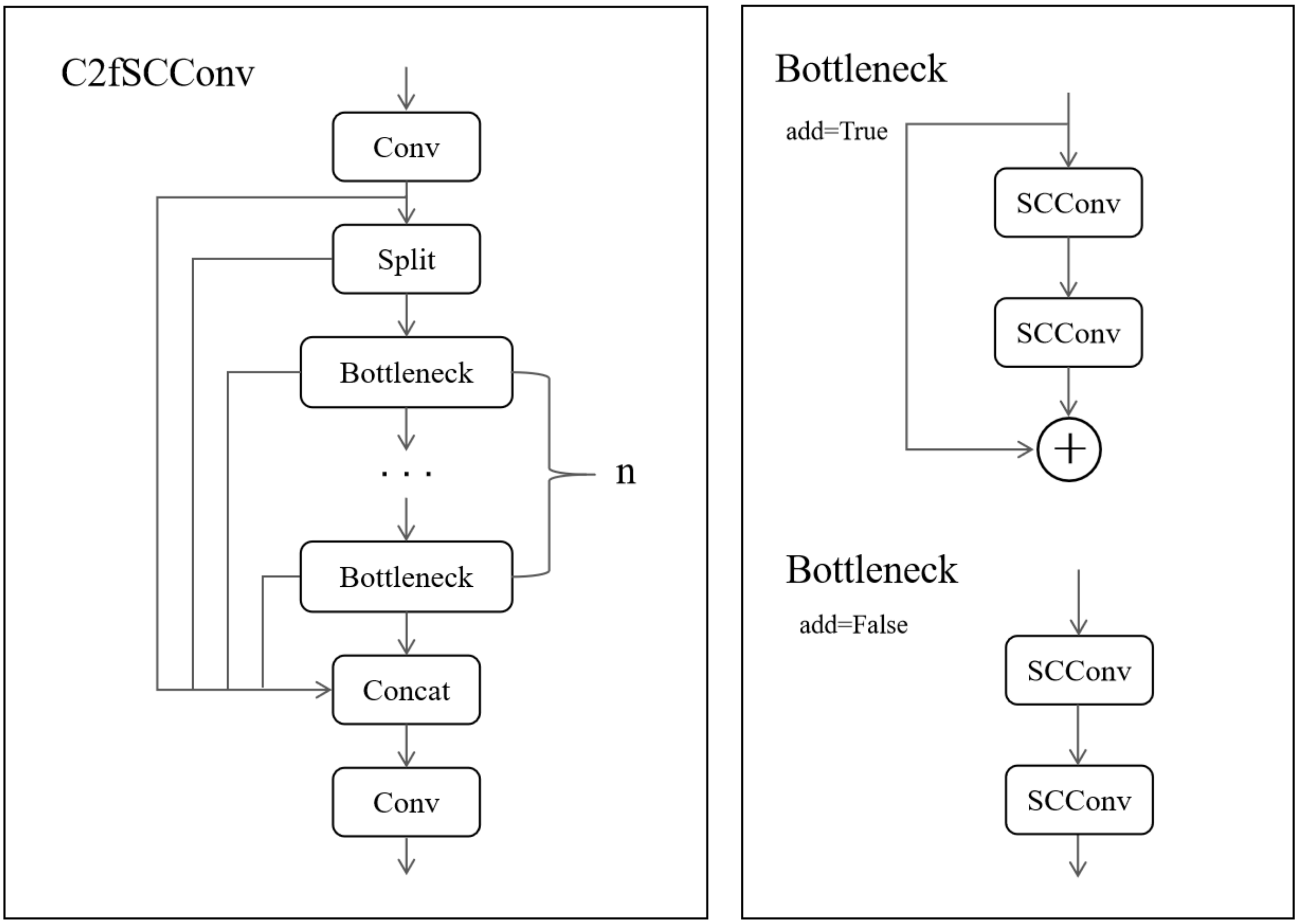

Fig. 5. The Structures of C2fSCConv and Bottleneck

TABLE III
COMPARISON OF DETECTION EFFECTS OF VARIOUS MODELS

Object Detection Algorithm	Params(M)	GFLOPS (G)	Precision	Recall	mAP0.5	mAP0.5-0.95
Faster-RCNN	41.37	241.95	86.93	82.78	85.81	40.70
YOLOv5s	7.03	16.00	89.71	85.83	88.10	47.80
YOLOv7s	37.62	100.39	86.20	80.82	84.20	43.60
YOLOX	8.97	26.93	87.69	84.66	86.56	44.80
YOLOv8n	2.62	7.00	86.90	83.40	87.10	48.40
YOLOv8s	9.59	24.30	89.42	85.30	88.80	49.60
YOLOv8s-CDS	10.00	28.40	91.40	88.20	91.20	50.80

$$Precision = \frac{TP}{TP + FP} \tag{1}$$

Recall refers to the proportion of all real dangerous driving behaviors detected by the model, where false negatives (FN) refer to the number of samples of real dangerous driving behaviors that the model failed to detect and represent the number of samples incorrectly classified as negative examples, that is, they are actually positive examples. The calculation formula is as follows (2):

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

AP is a comprehensive evaluation index for evaluating precision and recall, which is used to calculate the average accuracy of a single-class model. Among them, P is the vertical coordinate precision of the P-R curve, and R is the horizontal coordinate recall of the P-R curve. The value of AP is the area under the interpolated precision and recall curve, which represents the average accuracy of the model at all recall values. The higher the AP, the better the performance of the model in the target detection task. The calculation formula is as follows (3):

$$AP = \int_0^1 P(R)dR \tag{3}$$

mAP is the average of the AP values of all categories. The calculation formula is as follows (4):

$$mAP = \frac{\sum_{i=0}^{n} AP_i}{n} \tag{4}$$

Where n represents the total number of categories, and APi is the AP value of the i-th category. FPS (Frames Per Second) represents the number of frames processed per second to measure the speed of the algorithm in the image processing task. Higher FPS values indicate faster the algorithm in processing images and is able to analyze and detect more images in a shorter time.

D. Experimental Results Analysis

YOLOv8 provides users with multiple different versions to meet various application requirements. These versions mainly differ in model size, computational complexity, architecture design, weights, and performance. Common YOLOv8 versions include YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l, and YOLOv8x. This paper comprehensively considers the speed and accuracy of the model, that is, fast processing but still high accuracy is required. Therefore, we initially select YOLOv8s as the baseline network model of this paper.

In order to further verify the effectiveness of the dangerous driving behavior detection method based on the YOLOv8s model, we conducted comparative experiments in the same environment to evaluate the performance of this method with the current mainstream target detection model. This paper compares it with Faster-RCNN [17], the same series of YOLOv5s [18], YOLOv7s [19], YOLOX [20] and YOLOv8n salgorithms. As can be seen from Table III, compared with the two-stage algorithm such as Faster RCNN, the one-stage algorithm has a faster detection speed. Compared with the first-stage algorithms such as YOLOv5s, YOLOv7s, YOLOX and YOLOv8n, the detection accuracy of YOLOv8s has been further improved.

The table shows the detection performance of these algorithms on five sets of images. We observed that although the Faster-RCNN algorithm showed high detection accuracy, its two-stage nature led to a slower detection speed, and its large network parameters made it very challenging to use on mobile devices with limited computing and storage resources. In contrast, although the number of parameters of YOLOv7s has been greatly improved compared to the two-stage Faster-RCNN algorithm, the YOLOv5s ,YOLOX and YOLOv8n algorithms have been further improved compared to YOLOv7s, but based on the dataset used in this experiment, they are not as accurate as the YOLOv8s model. In summary, compared with other target detection methods, the method proposed in this paper has superior advantages in comprehensive performance. Therefore, this experiment was finally determined to be based on YOLOV8s.

Through the improvement of the method of adding modules to the head and neck network model in this paper, the number of parameters in the model can be greatly reduced, and the detection accuracy can also be greatly improved.

E. Ablation Experiment

As can be seen from Table IV, the detection accuracy of YOLOv8s is relatively high, but the number of network parameters is large, and the amount of computation required by the network is also large, which is not conducive to use on embedded mobile terminals. Therefore, this paper makes improvements based on the YOLOv8s model. The improvements include: using the ConvNeXt V2 module to improve the backbone network; using the DASI module to improve the feature fusion module; and introducing the SCConv module through the improvements of the above methods, the number of parameters in the model can be greatly reduced, and the detection accuracy can also be greatly improved. In order to verify the effectiveness of each module, we used the same data set and the same training hyperparameters to conduct an ablation study and compared it with the original model. The experimental results are shown in Table IV.

Table IV highlights the good performance of YOLOv8s on various enhancement points. First, integrating the ConvNeXt V2 module into the original YOLOv8s model can learn global and local features of the image, thereby improving its generalization ability, resulting in a 0.80% increase in mAP@0.5. Adding DASI and SCConv modules to enhance the neck network further improves the accuracy, with a 1.20% increase in mAP@0.5. By combining the advantages of each module, the algorithm finally achieves 91.20% of mAP@0.5, 2.4% higher than the YOLOv8s algorithm.

Fig.6 is the normalized confusion matrix of the experiment, that is, the performance of the model on different categories after the prediction results of each category are standardized to relative proportions. The diagonal elements represent the correct prediction ratio of the model on each category, that is, the recall rate. A higher diagonal value indicates that the model performs well on this category. For example, the recall rate of the drinking category is 0.88, which means that the model correctly classifies 88% of the drinking samples. The non-diagonal elements represent the confusion between categories. For example, the proportion of drinking being misclassified as background is 0.23, which means that 23% of drinking behaviors are misclassified as background.

Fig. 7 compares the mAP_0.5 curves of the improved YOLOv8s-CDS algorithm with those of the YOLOv8s, YOLOv8s-C, and YOLOv8s-DS algorithms. As shown in Fig. 7, the introduction of the ConvNeXt V2 module, the DASI module, and the SCConv convolution module increases the mAP_0.5 of the YOLOv8s-CDS algorithm to 91.2%, which is 2.4% higher than that of the YOLOv8s algorithm.

The mAP of all models in the figure shows an upward trend with the increase in the number of training rounds, which indicates that the model continuously improves its detection performance during the training process. The mAP curve rises rapidly in the first 50 rounds and then enters a gradual improvement stage, which indicates that the model gradually converges. By the 200th round, the mAP values of all models tend to be stable, indicating that the model has converged and there is no obvious performance improvement. The enlarged graph of the mAP_0.5 curve in the ablation experiment is shown in Fig. 8.

In summary, YOLOv8s-CDS is the optimal model in this experiment because it has the fastest convergence speed and the highest final mAP, which shows that it is suitable for dangerous driving detection and can improve the performance of

TABLE IV
INFLUENCE OF DIFFERENT INPUT SIZES ON THE DETECTION EFFECT

Method	ConvNeXt V2	DASI+	SCConv	mAP0.5
YOLOv8s	-	-	-	88.80
Proposed method1	✓	✓	-	89.60
Proposed method2	✓	✓	-	90.00
Proposed method3	✓	✓	✓	91.20

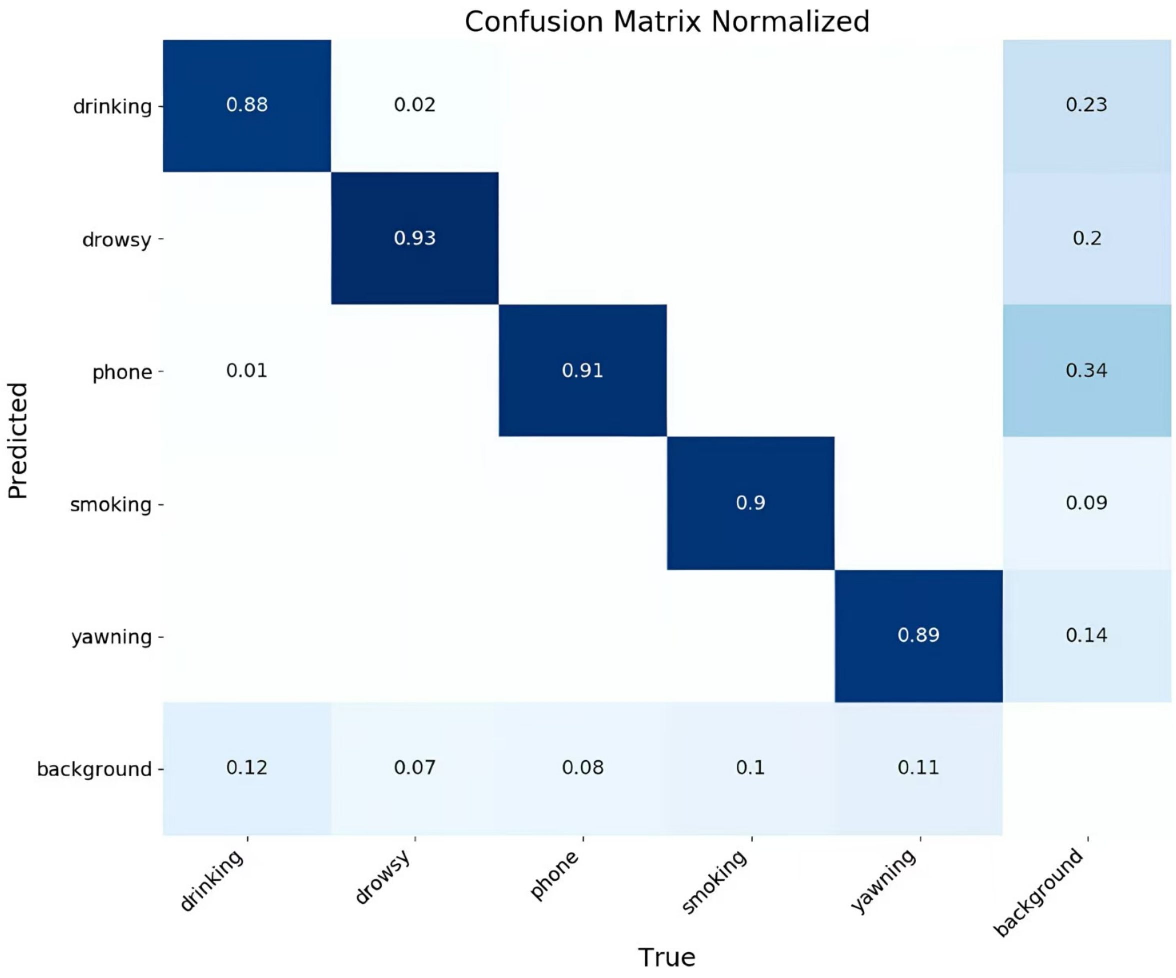

Fig. 6. Model normalized confusion matrix plot

dangerous driving detection projects.

Fig. 9 is a comparison curve of mAP:0.5_0.95 and epoch between the improved YOLOv8s-CDS algorithm and the YOLOv8s, YOLOv8s-C and YOLOv8s-DS algorithms. The figure shows that the YOLOv8s model gradually converges with the increase of epochs, and the final mAP reaches about 0.45. The performance of the YOLOv8s-C model is significantly higher than that of the YOLOv8s. Its rising speed in the early stage is close to that of the YOLOv8s, but in the middle and late stages, the mAP maintains a higher level, and finally slightly exceeds 0.47. The performance of the YOLOv8s-DS model is close to that of the YOLOv8s-C, but it is more stable throughout the training process, and the final mAP is close to 0.48. The YOLOv8s-CDS model has the best performance. During the entire training process, the mAP rises the fastest, and the final mAP reaches close to 0.5, which is the best performing model.

All models converged around 75-100 rounds, at which time the mAP began to stabilize without obvious increases or fluctuations. This indicates that the model has learned the best features in this range, and continued training may have limited performance improvements. In the first 25 rounds, the mAP of all models experienced a rapid increase, that is, in the initial stage, the model quickly learned some basic features from the data. However, as the rounds increased, the speed of model performance improvement gradually slowed down, especially after 50 rounds, the mAP growth of all models became relatively slow. Among them, the YOLOv8s-CDS model performed outstandingly, with the fastest convergence speed, while other models required more epochs to reach a similar performance level, that is, they could achieve higher accuracy in less training time.

The enlarged graph of the mAP_0.5:0.95 curve in the ablation experiment is shown in Fig. 10. It can be clearly seen that YOLOv8s-CDS improves the detection performance of the model, especially in the dangerous driving detection task.

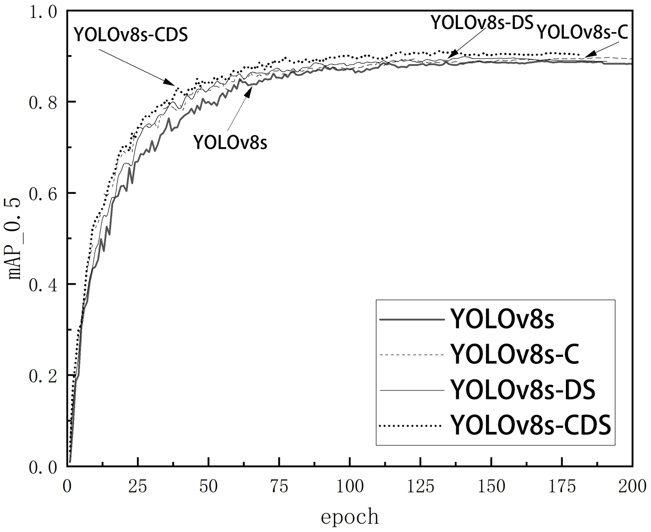

Fig. 7. Comparison curve of mAP_0.5 and epoch of the improved YOLOv8s-CDS model

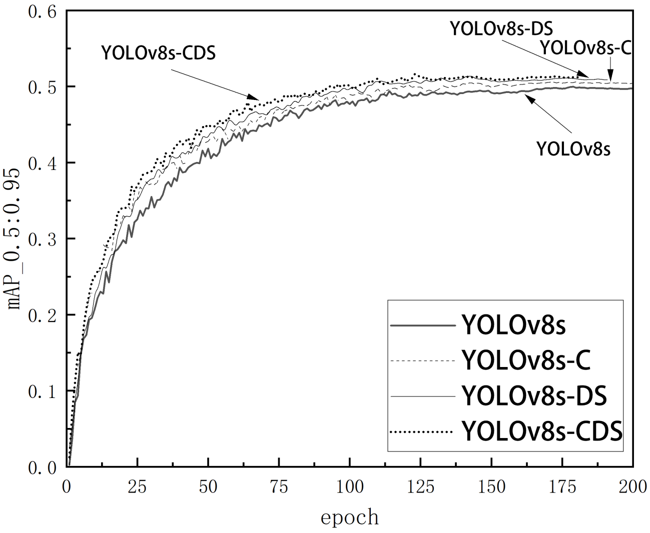

Fig. 9. Comparison curve of mAP_0.5:0.95 and epoch of the improved YOLOv8s-CDS model

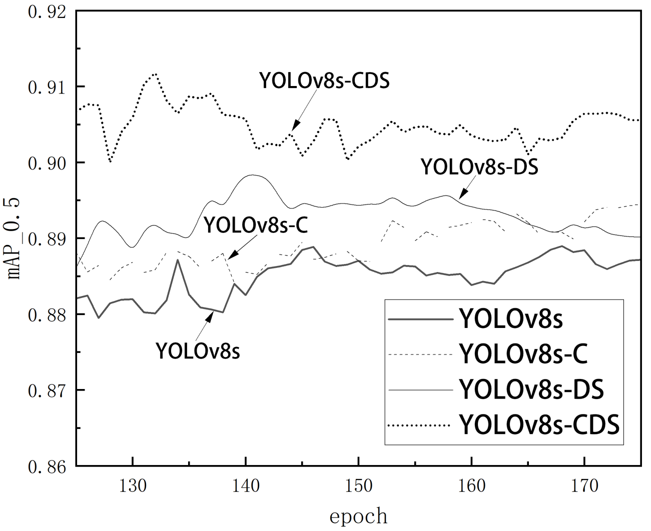

Fig. 8. Enlarged map_0.5 comparison curve

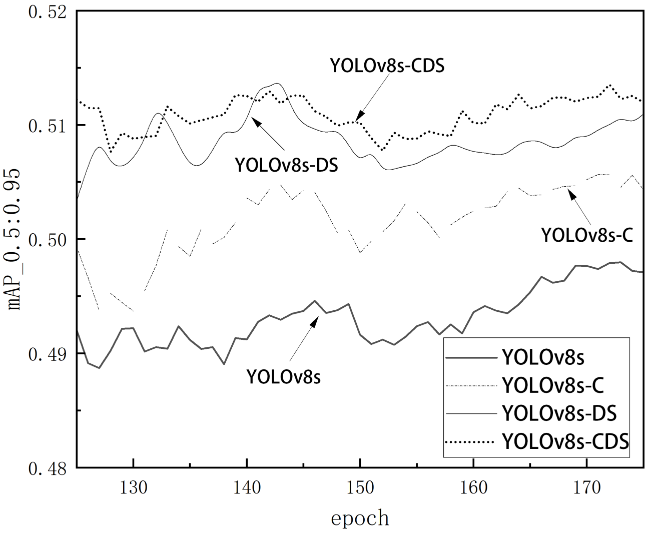

Fig. 10. Enlarged map_0.5:0.95 comparison curve

The results of the enhanced dangerous driving detection model are presented in Fig. 11, which provides clear evidence of the model's outstanding performance in both efficiency and accuracy. This advanced model demonstrates its capability to successfully identify a wide range of dangerous driving behaviors, such as speeding, abrupt lane changes, tailgating, and distracted driving.

By accurately detecting and distinguishing various forms of hazardous driving, the model empowers traffic management authorities and law enforcement agencies to take timely and targeted preventive measures. These measures could include issuing early warnings, implementing stricter enforcement, or conducting driver education programs, all of which work together to reduce the frequency of traffic accidents. Ultimately, this improved detection model contributes to creating safer roadways, protecting the lives and well-being of every road user, and fostering a more responsible driving culture in society.

V. CONCLUSIONS

This study conducted a high-precision optimization of the YOLOv8s target detection algorithm for dangerous driving scenarios. Through in-depth analysis and experimental verification, it was concluded that the improved YOLOv8s method enhances detection capabilities in dangerous driving behaviors. The method maintains high detection accuracy and a low false alarm rate, effectively meeting the needs for dangerous driving behavior detection. Additionally, the proposed detection method demonstrates high practical value in real-world applications by identifying dangerous driving behaviors and effectively preventing traffic accidents.

However, this method still has certain limitations. This research primarily focuses on improving accuracy, with less emphasis on accelerating detection speed. In the future, we aim to explore and develop a more lightweight deep learning model to enhance the detection speed. Additionally, future detection systems could leverage in-vehicle cameras and Internet of Vehicles technologies to collect and analyze driving behavior data in real-time, thereby providing safety guarantees for intelligent transportation systems.

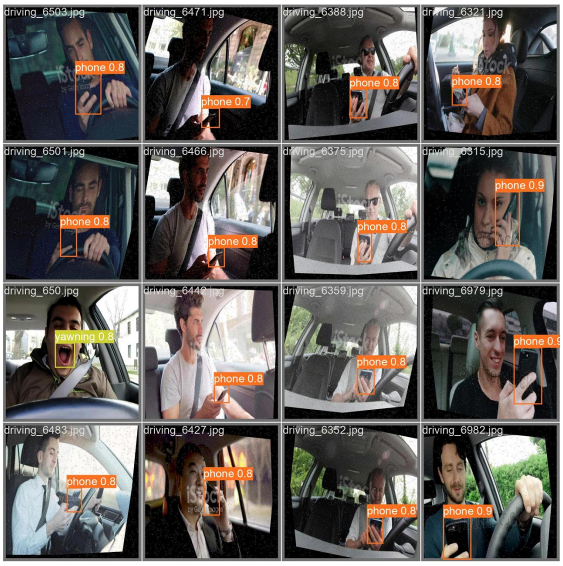

Fig. 11. Dangerous driving detection results diagram

REFERENCES

[1] S. Bouhsissin, N. Sael, F. Benabbou, "Driver behavior classification: a systematic literature review," IEEE Access, vol. 11, pp 14128-14153, 2023.

[2] W. Song, G. Zhang, Y. Long, "Identification of dangerous driving state based on lightweight deep learning model," Computers and Electrical Engineering, vol. 105, pp 108509, 2023.

[3] N. J. Zakaria, M. I. Shapiai, R. Abd Ghani, M. N. M. Yassin, M. Z. Ibrahim and N. Wahid, "Lane detection in autonomous vehicles: A systematic review," IEEE access, vol. 11, pp 3729-3765, 2023.

[4] N. M. Negash, J. Yang, "Driver behavior modeling toward autonomous vehicles: Comprehensive review," IEEE Access, vol. 11, pp 22788-22821, 2023.

[5] W. Gao, "Implementation of MASK R-CNN model for driver behavior detection," Electronic World, vol. 06, pp 29-30, 2021.

[6] J. Ni, K. Shen, Y. Chen and S. X. Yang, "An improved ssd-like deep network-based object detection method for indoor scenes," IEEE Transactions on Instrumentation and Measurement, vol. 72, pp 1-15, 2023.

[7] N. I. Nife, M. Chtourou, "A Comprehensive Study of Deep Learning and Performance Comparison of Deep Neural Network Models (YOLO, RetinaNet)," International Journal of Online & Biomedical Engineering, vol. 19, no. 12, 2023.

[8] C. Ji , "Research on driver behavior detection method based on convolutional neural network," Jilin University, 2021.

[9] B. Xiao, M. Nguyen, W. Q. Yan, "Fruit ripeness identification using YOLOv8 model," Multimedia Tools and Applications, vol. 83, no. 9, pp 28039-28056, 2024.

[10] L. Shen, B. Lang, Z. Song, "DS-YOLOv8-Based object detection method for remote sensing images," IEEE Access, vol. 11, pp 125122-125137, 2023.

[11] X. Zou, S. Jiang, J. Yang, X. Huang, "Concrete Bridge Crack Detection Based on YOLO v8s in Complex Background," International Conference on Image, Vision and Intelligent Systems, pp 436-443, 2023.

[12] S. Woo, S. Debnath, R. Hu, X. Chen, Z. Liu, I. S. Kweon and S. Xie, "Convnext v2: Co-designing and scaling convnets with masked autoencoders," Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 16133-16142, 2023.

[13] S. Cheng, E. Sun, W. Qian and Y. Han, "A Breast Cancer Detection Model Based on Modified Convnext v2," Proceedings of the 2023 3rd Guangdong-Hong Kong-Macao Greater Bay Area Artificial Intelligence and Big Data Forum, pp 556-559, 2023.

[14] S. Xu, S. Zheng, W. Xu, R. Xu, C. Wang, J. Zhang, X. Teng, A. Li and L. Guo, "HCF-Net: Hierarchical Context Fusion Network for Infrared Small Object Detection," arXiv preprint arXiv: 2403. 10778, 2024.

[15] P. Gong, J. Liu, X. Zhang, X. Li and Z. Yu, "Circulant-interactive transformer with dimension-aware fusion for multimodal sentiment analysis," Asian Conference on Machine Learning, pp 391-406, 2023.

[16] J. Li, Y. Wen, L. He, "Scconv: Spatial and channel reconstruction convolution for feature redundancy," Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 6153-6162, 2023.

[17] J. Hou, C. Yang, Y. He and B. Hou, "Detecting diseases in apple tree leaves using FPN-ISResNet-Faster RCNN," European Journal of Remote Sensing, vol. 56, no. 1, pp 2186955, 2023.

[18] Z. Lin, L. Zhu, J. Zhang, Y. Zhang, and X. Liu, "Research on Improving YOLOv5s Algorithm for Defect Detection in Cylindrical Coated Lithium-ion Batteries," Engineering Letters, vol. 32, no. 7, pp 1521-1528, 2024.

[19] H. Zhuang, W. Liu, "Underwater Biological Target Detection Algorithm and Research Based on YOLOv7 Algorithm," IAENG International Journal of Computer Science, vol. 51, no. 6, pp 594-601, 2024.

[20] C. Song, F. Zhang, J. Li, J. Xie, Y. Chen, Z. Hang and J. Zhang. "Detection of maize tassels for UAV remote sensing image with an improved YOLOX model", Journal of Integrative Agriculture, vol. 22, no. 6, pp 1671-1683, 2023.