applied sciences

MDPI

Article

YOLO-DPDG: A Dual-Pooling Dynamic Grouping Network for Small and Long-Distance Traffic Sign Detection

Ruishi Liang, Minjie Jiang and Shuaibing Li *

School of Computer Science, University of Electronic Science and Technology of China, Zhongshan Institute, Zhongshan 528400, China; liangruishi@foxmail.com (R.L.); jiangmj@zscbdic.cn (M.J.)
* Correspondence: lishuaibing777@gmail.com

Abstract

Traffic sign detection is a crucial task for autonomous driving perception systems, as it directly impacts vehicle path planning and safety decisions. Existing algorithms face challenges such as feature information attenuation and model lightweighting requirements in the detection of small traffic signs at long distances. To address these issues, this paper proposes a dual-pooling dynamic grouping (DPDG) module. This module dynamically adjusts the number of groups to adapt to different input features, combines global average pooling and max pooling to enhance channel attention representation, and uses a lightweight 3 × 3 convolution-based spatial branch to generate spatial weights. Based on a hierarchical optimization strategy, the DPDG module is integrated into the YOLOv10n network. Experimental results on the traffic sign dataset demonstrate a significant improvement in the performance of the YOLO-DPDG network: Compared to the baseline YOLOv10n model, mAP@0.5 and mAP@0.5:0.95 improved by 8.77% and 10.56%, respectively, while precision and recall were enhanced by 6.16% and 6.62%, respectively. Additionally, inference speed (FPS) increased by 11.1%, with only a 4.89% increase in model parameters. Compared to the YOLOv10-Small model, this method achieves a similar detection accuracy while reducing the number of model parameters by 64.83%. This study provides a more efficient and lightweight solution for edge-based traffic sign detection.

Keywords: YOLOv10; traffic sign detection; DPDG; attention mechanism; small object detection

Academic Editor: Andrea Prati

Received: 29 August 2025
Revised: 25 September 2025
Accepted: 2 October 2025
Published: 11 October 2025

Citation: Liang, R.; Jiang, M.; Li, S. YOLO-DPDG: A Dual-Pooling Dynamic Grouping Network for Small and Long-Distance Traffic Sign Detection. *Appl. Sci.* **2025**, *15*, 10921. https://doi.org/10.3390/app152010921

1. Introduction

Driven by the deep integration of intelligent technology innovation and the digital revolution, autonomous driving technology has achieved breakthrough progress and garnered significant attention from all sectors of society [1]. In the field of autonomous driving technology, intelligent traffic sign recognition is a core component of environmental perception and decision-making planning, and the optimization of its algorithm performance directly impacts the reliability and safety of the system. Efficient recognition algorithms can significantly enhance the detection accuracy and real-time performance of autonomous vehicles in identifying traffic signs, ensuring compliance with traffic regulations, and optimizing overall traffic efficiency. Additionally, improved recognition accuracy not only enhances the responsiveness of vehicle decision-making mechanisms but also effectively reduces safety risks caused by misdetection or missed detection, providing road users with more comprehensive safety protection [2]. However, real-world road scenarios require autonomous vehicles to have the ability to perceive traffic signs at long distances to support

advanced decision-making. The "long distances" referred to in this study correspond to the detection distances of 50 m to 100 m or more in real scenarios. Due to their multi-scale nature, low resolution, and susceptibility to complex background interference such as changes in lighting and partial obstruction, traffic signs often appear as small objects in sensor images, making recognition challenging [3].

Traditional traffic sign recognition methods primarily rely on color space analysis and shape contour extraction, utilizing predefined template matching or shallow classifiers for detection and classification [4,5]. Such methods heavily depend on the robustness of predefined features and the stability of the scene. However, in scenarios involving distant, low-resolution small objects, artificially designed features exhibit severe limitations in adaptability, resulting in limited feature representation capabilities, poor generalization, and significant fluctuations in detection accuracy [2,6,7]. The complex feature calculation and matching processes typically involve high computational complexity [8], making it difficult to meet the stringent real-time processing performance requirements of in-vehicle systems.

With the rapid development of deep learning technology, research on traffic sign recognition has gained new momentum. Mainstream deep learning detection algorithms can be broadly categorized into two types: two-stage detection frameworks represented by R-CNN, Fast-RCNN, and Faster-RCNN, and single-stage detection frameworks, including the You Only Look Once (YOLO) series [9–17] and SSD models. Due to their inherent multi-stage processing mechanism, two-stage algorithms often suffer from high computational complexity and slow inference speeds, making it difficult to meet the stringent real-time requirements of traffic sign recognition. In contrast, single-stage algorithms have a simpler model structure, significantly improving detection speed while maintaining acceptable accuracy, making them more suitable for practical application scenarios. Among the many single-stage algorithms, the YOLO series has become a widely adopted research foundation in the field due to its excellent performance balance achieved through continuous iterative optimization. This paper selects the YOLOv10 [17] version of the YOLO series and performs in-depth optimization of model efficiency and detection performance to address the practical needs of traffic sign recognition.

The core competitiveness of YOLOv10 has been widely validated. Its end-to-end deployment and model architecture optimization have achieved comprehensive breakthroughs in speed, accuracy, and parameter efficiency. Its innovative direction represents a major advancement in object detection, particularly suitable for latency-sensitive autonomous driving scenarios. Among these, the lightweight model YOLOv10n achieves an extremely lightweight design and inference speed while maintaining high accuracy, better meeting the real-time requirements of in-vehicle deployment. However, when detecting small objects such as distant traffic signs, simply improving inference speed is insufficient. Lightweight models still have weak feature extraction capabilities for small objects, making it difficult to improve detection accuracy. They face challenges such as loss of feature resolution, inadequate utilization of contextual information, and insufficient flexibility of attention mechanisms, leading to the risk of missing critical traffic information and constraining the reliability and widespread application of autonomous driving technology. While YOLOv10s significantly improves detection accuracy, it also leads to a sharp increase in parameters, significantly increasing computational complexity and memory usage. This results in higher inference latency on edge devices, impacting real-time decision-making requirements for autonomous driving. Additionally, edge devices would need hardware upgrades, leading to increased deployment costs.

These shortcomings limit the generalization ability of YOLOv10n in complex traffic scenarios, especially when dealing with areas with a high density of small targets, where

false negatives and false positives are likely to occur. YOLOv10s suffers from feedback delays and instability, and changing hardware increases costs, making it difficult to adapt to the demands of dynamic traffic environments. Traffic sign detection needs to optimize detection accuracy while maintaining high inference efficiency.

To address the above issues, this paper proposes a dual-pooling dynamic grouping module (DPDG). The lightweight improved network YOLO-DPDG integrates our newly designed DPDG module into YOLOv10n to form a collaborative system with dynamic feature aggregation capabilities, effectively balancing model accuracy and computational efficiency. The DPDG module serves as the core component, with its implementation incorporating three innovative mechanisms:

1. Coordinated adaptive dynamic grouping mechanism: Adaptively adjust the number of groups based on the number of input channels to ensure optimal channel division, improve the model's generalization ability and feature utilization, and reduce intra-group redundancy.
2. Dual-pooling channel attention: This component simultaneously employs global average pooling to capture global statistical features across channel dimensions and max pooling to aggregate prominent local features. Finally, it constructs a hybrid statistic through dual-path feature tensor concatenation and dimension compression, enhancing feature representation in complex scenarios.
3. Lightweight spatial branch: A 3×3 separable convolution with parameter sharing [15,17] is used to construct a spatial weight generator, with fewer parameters than traditional spatial attention. Through spatial compression operations, computational complexity is reduced while maintaining the receptive field.

The main contributions of this network are as follows: it proposes a dynamic grouping attention mechanism for small object detection, innovatively integrating dynamic grouping with dual pooling. Compared to traditional fixed grouping strategies, dynamic grouping can maintain optimal and stable channel division. Compared to single pooling designs, dual pooling can increase the receptive field. Furthermore, compared to mainstream attention mechanisms, DPDG performs better in detection accuracy and speed. The improved network significantly optimizes the extraction strength and efficiency of small object features, enhancing recognition accuracy. It effectively addresses the issues of false positives and false negatives in detecting small traffic sign objects while achieving a high balance between performance and speed, making it more practical for real-world applications.

The remainder of this paper is organized as follows: Section 2 focuses on the research evolution of the YOLOv10n architecture and attention mechanism. Section 3 systematically analyzes the network structure proposed in this study. Section 4 explains the experimental implementation from three aspects: experimental details, comparison with advanced modules, and ablation experiments, and quantitatively evaluates the algorithm performance. Section 5 discusses and explains the deeper significance of this study. Section 6 summarizes the innovative methods and looks ahead to possible future optimization directions and technical extensions.

2. Related Work

As a key technology in autonomous driving perception systems, traffic sign detection continues to drive innovation in detection network architecture and attention mechanisms due to the challenge of balancing lightweight design and accuracy.

2.1. YOLOv10

The basic object detection model adopted the YOLOv10 version proposed by Wang et al. from the Multimedia Intelligent Group of Tsinghua University in 2024. As an up-

graded version of YOLOv8, YOLOv10 has undergone a number of key optimizations and algorithmic improvements in network architecture, training process, and post-processing mechanisms, significantly improving detection accuracy while maintaining excellent real-time detection speed. The model achieves a higher mean average precision (mAP) than YOLOv8 across multiple sizes, including Nano, Small, Medium, Large, and Extra-large. The YOLOv10 architecture consists of three components: a backbone network based on the enhanced Cross-Stage Partial Network (CSPNet-enhanced) structure, which reduces redundant computations through partial convolution; a PAN neck that uses hierarchical feature aggregation to fuse shallow spatial information with deep semantic features; and a dual detection head structure that includes a one-to-many head for enriching positive samples during training and a one-to-one head for directly outputting redundant predictions during inference.

YOLOv10 achieves an NMS-free detection process through a consistent dual assignment strategy. Specifically, during training, the one-to-many head dynamically selects positive samples using a task-aligned assigner, while the one-to-one head determines the unique match through optimal transport assignment. During inference, only the one-to-one head is retained, eliminating the need for NMS post-processing. Extensive experiments validate that YOLOv10 achieves state-of-the-art performance while significantly reducing computational overhead through the above optimizations, continuing and enhancing the YOLO series' advantage of balancing speed and accuracy.

YOLOv10n, a lightweight version of the YOLOv10 series, achieves synergistic optimization of speed and accuracy through architectural innovation, such as the use of depthwise separable convolutions and gradient reparameterization, which significantly reduce the number of parameters and computations while maintaining performance. Its lightweight branch, YOLOv10n, further reduces computational overhead while maintaining nano-level parameters, making it the preferred benchmark model for edge deployment. Its detailed structure is shown in Figure 1.

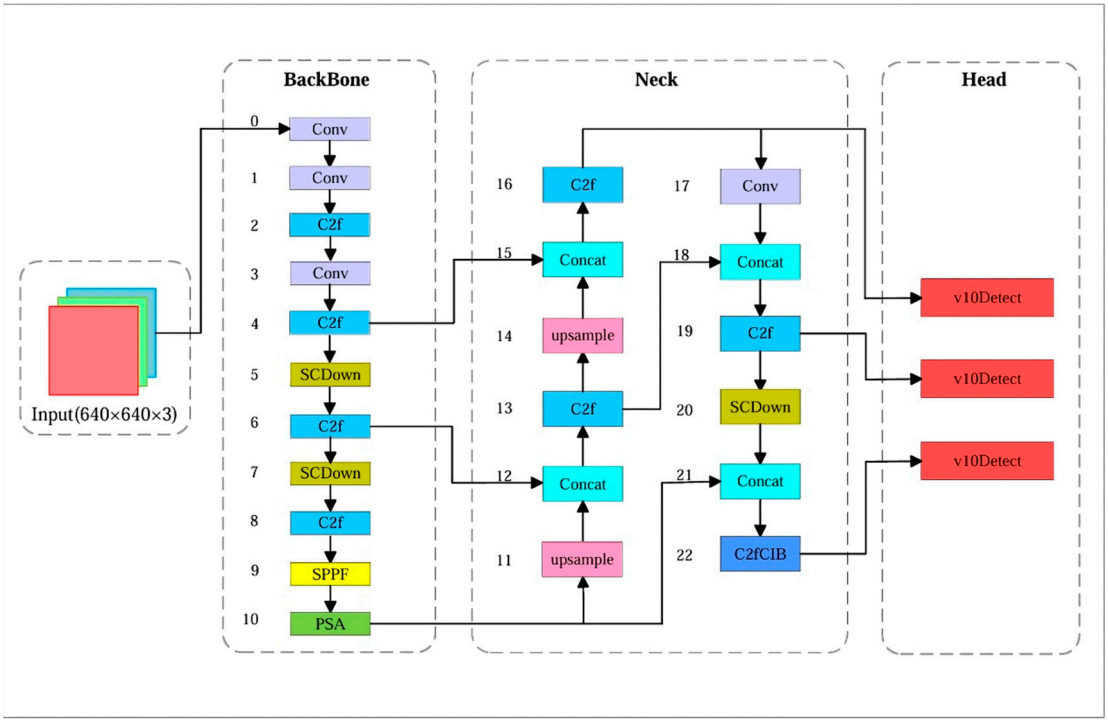

Figure 1. YOLOv10n network architecture.

The backbone network of YOLOv10n adopts a CSPNet-enhanced structure, achieving lightweight design through the synergistic integration of Depthwise Separable Convolution and Gradient Reparameterization. The network decomposes standard convolutions into a cascaded operation of depthwise convolutions and pointwise convolutions, significantly reducing the number of parameters. During training, multi-branch convolutions enhance feature diversity, which is then merged into a single path during inference to maintain efficiency. However, this design still has limitations in detecting small objects at long distances, especially in extracting low-pixel traffic sign features. The neck network combines reversible cross-stage connections (Reversible CSPConnect) with a hierarchical feature aggregation mechanism to fuse shallow spatial information and deep semantic features through bidirectional feature routing. Compared to the Feature Pyramid Network (FPN) proposed by Lin et al. [18], which enhances semantic perception through cross-scale feature fusion, the stacked structure of FPN significantly increases computational complexity. This design replaces feature concatenation with channel reordering, reducing memory usage while maintaining multi-scale fusion effects. However, its static grouping strategy has limited adaptability to the multi-scale dynamic changes in traffic signs, constraining detection stability in complex environments. The head network employs an Implicit Decoupled Head to optimize feature decoding for classification and regression tasks. By sharing the base convolutional layers and introducing task-specific weights at the terminal branches, it retains the accuracy advantages of the decoupled head while avoiding the computational overhead of an explicit multi-branch structure. Combined with a consistent dual-allocation strategy, during training, a pair of multi-branch structures is used to augment positive samples, and during inference, it switches to a single-branch structure to achieve NMS-free output. However, this mechanism is overly strict in filtering low-confidence small objects, leading to increased sign detection rates and bounding box prediction errors, highlighting the limitations of the existing architecture in detecting small objects.

2.2. Attention Mechanism

The attention mechanism suppresses interference and enhances responses in key regions through feature reweighting strategies, making it a core technology for improving small object detection performance. The channel attention mechanism was first proposed by Hu et al. [19] in SENet, which uses global average pooling (GAP) to generate channel weights. However, its single statistical measure struggles to capture the local salient features of traffic signs. To address this issue, Woo et al. [20] proposed CBAM, which combines channel and spatial attention and uses 7×7 convolutions to capture local context. However, the large convolution kernels introduce excessive computational load. Li et al. [21] further proposed a fixed grouping attention SGE, which uniformly divides channels into eight groups for parallel processing. Although this improves computational efficiency, the rigid grouping strategy causes information imbalance between groups when input channels dynamically change. As a result, dynamic attention has gradually become a research hotspot. Yang et al.'s [22] CondConv enhances feature discriminative power through sample-adaptive weight matrices, but the demand for dynamic parameter storage causes severe model bloat; Dai et al.'s [23] Deformable Convolution adapts the receptive field by learning spatial offsets, but it is sensitive to offset prediction errors for low-resolution targets. Sunkara et al.'s [24] SPD-Conv replaces downsampling with a spatial-to-depth transformation to preserve fine-grained features of small objects. However, introducing an additional transformation layer increases the number of parameters. In traffic sign detection, researchers have attempted to optimize attention design by incorporating domain knowledge. For example, Wang et al. [25] proposed color-aware attention based

on color priors, which enhances the response to sign colors in the HSV space; Zhu et al. [26] designed an orientation-sensitive spatial attention module to enhance the rotational robustness of triangular warning signs. Although the above methods have made some progress in small object detection, they still have limitations in many aspects. There is an imbalance between efficiency and accuracy, making it difficult to meet the lightweight requirements of edge devices; insufficient suppression of feature interference, with noisy features in complex backgrounds easily interfering with multi-scale fusion processes [27]; and rigid channel partitioning, with fixed-group attention leading to low information utilization between groups.

Therefore, using only the YOLOv10n model or existing attention mechanisms cannot fundamentally resolve the conflict between lightweight design and performance. An effective balance has yet to be established among dynamic adaptability, computational efficiency, and parameter control. Our method, YOLO-DPDG, does not require additional storage for dynamic weights. It achieves feature expression optimization solely through adaptive adjustment of the number of groups, providing a solution for designing lightweight detection networks.

3. Method

The YOLOv10 series of models has demonstrated outstanding performance in multiple computer vision tasks, including object detection, visual classification, and instance-level semantic segmentation. This series offers five model variants based on the balance between computational efficiency and detection accuracy, namely Nano (n), Small (s), Medium (m), Large (l), and Extra-large (x). These variants are designed to meet different resource limitations and accuracy requirements. The n version is specifically tailored for ultra-lightweight and high-speed deployment on edge devices, while the gradually larger types (s, m, l, x) will contain more parameters and computational complexity, achieving higher detection accuracy while increasing inference latency. In this study, we selected the computationally least intensive YOLOv10n as the base network architecture to meet the stringent requirements for real-time inference in practical application scenarios.

Aiming at the core issues of low feature resolution and strong background interference in traffic sign small object detection tasks, this paper constructs the Dual-Pooling Dynamic Grouping Network (YOLO-DPDG), a lightweight improved network based on YOLOv10n. By integrating a newly designed dual-pooling dynamic grouping module (DPDG) and restructuring the backbone with SPD-Conv and C2fCIB modules, the network forms a synergistic system with dynamic feature aggregation capabilities, thereby achieving a better balance between detection accuracy and computational efficiency. As shown in Figure 2, the overall model structure achieves coordinated optimization of accuracy and efficiency through multi-level modular design. Specifically, the input feature map is first decomposed into subregions and channel concatenation through a space-to-depth transformation, converting the downsampling process into a channel expansion operation. This design addresses the issue of small object detail loss caused by traditional strided convolutions while reducing the output resolution to one-quarter of the original, providing subsequent modules with rich spatial details, particularly enhancing edge and texture information of small objects. Second, the feature fusion layer is restructured into a cross-stage interaction bottleneck structure (C2fCIB), utilizing bidirectional cross-layer connections to enhance the interaction efficiency between shallow-layer localization information and deep-layer semantic features. Through a channel information bottleneck (CIB), redundant features are compressed to retain core semantic information across stages, thereby reducing noise interference in subsequent DPDG processing.

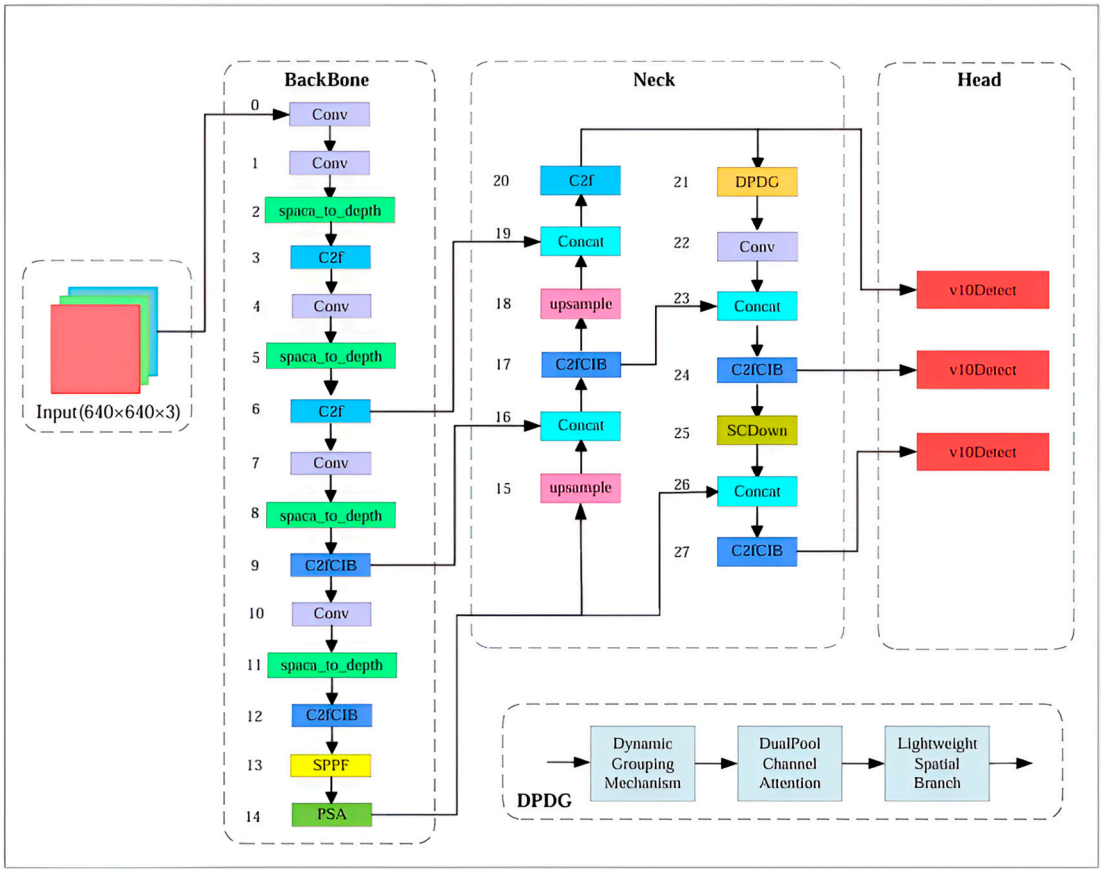

Figure 2. YOLO-DPDG network architecture.

To overcome issues such as channel redundancy, group hardening, and insufficient spatial perception in traditional attention mechanisms within lightweight models, this work proposes the DPDG. It achieves feature expression optimization through an adaptive channel partitioning strategy and a dual-dimensional attention coordination mechanism, ultimately deployed at the front end of the detection head. This module consists of three parts: the dynamic grouping mechanism adaptively adjusts the number of groups based on the input channel count to achieve a globally optimal solution for channel partitioning; the dual-pooling channel attention combines global average pooling and max pooling to generate hybrid statistics, enhancing feature discriminative power in complex scenes; and the lightweight spatial branch employs a 3×3 separable convolutional layer with parameter sharing to construct a spatial weight generator, resulting in lower computational complexity compared to traditional 7×7 convolutions. The entire network adopts an end-to-end optimization strategy, achieving dynamic aggregation of multi-scale features while maintaining lightweight characteristics.

3.1. SPD-Conv

Convolutional neural networks (CNNs) often lose detailed features when processing low-resolution images or small objects due to the coarse-grained downsampling operations of strided convolution and pooling layers. To address this issue, this study adopts the space-to-depth convolution module proposed by Sunaka et al. as an alternative to the standard downsampling layer. The SPD module consists of a spatial depth transformation layer and a non-strided convolution layer. Its core operation involves dividing an input

feature map X of size S × S × C into scale^2 sub-feature maps using a scaling factor scale, as shown in the following formula:

$$\begin{gathered} f_{0,0} = X[0{:}S{:}scale, 0{:}S{:}scale], f_{1,0} = X[1{:}S{:}scale, 0{:}S{:}scale], \ldots, \\ f_{scale-1,0} = X[scale - 1{:}S{:}scale, 0{:}S{:}scale]; \\ f_{0,1} = X[0{:}S{:}scale, 1{:}S{:}scale], f_{1,1}, \ldots, \\ f_{scale-1,1} = X[scale - 1{:}S{:}scale, 1{:}S{:}scale]; \\ f_{0,scale-1} = X[0{:}S{:}scale, scale - 1{:}S{:}scale], f_{1,scale-1}, \ldots, \\ f_{scale-1,scale-1} = X[scale - 1{:}S{:}scale, scale - 1{:}S{:}scale]; \end{gathered} \tag{1}$$

The sub-feature map is composed of all X(i,j) in the original feature map X that satisfy both i + x and j + y being divisible by scale. Therefore, each sub-feature map implements downsampling of X by a scale factor. For example, as shown in Figure 3, when scale = 2, the original 4 × 4 feature map is divided into four non-overlapping 2 × 2 subregions (f_{00}, f_{01}, f_{10}, f_{11}), each corresponding to a set of pixels with odd-even combinations of row and column indices in the original feature map. Each submap contains the spatial local information of the original feature map. The four sub-maps are concatenated along the channel dimension to form a temporary feature map of size 2 × 2 × 4C_1, which is then compressed to the target dimension C_2 via a 1 × 1 convolution. This operation reduces the resolution by a factor of 2 while fully preserving the original spatial information, addressing the detail loss issue in stride convolution and enabling the retention of fine details in small traffic signs. This approach demonstrates greater robustness in complex traffic scenes.

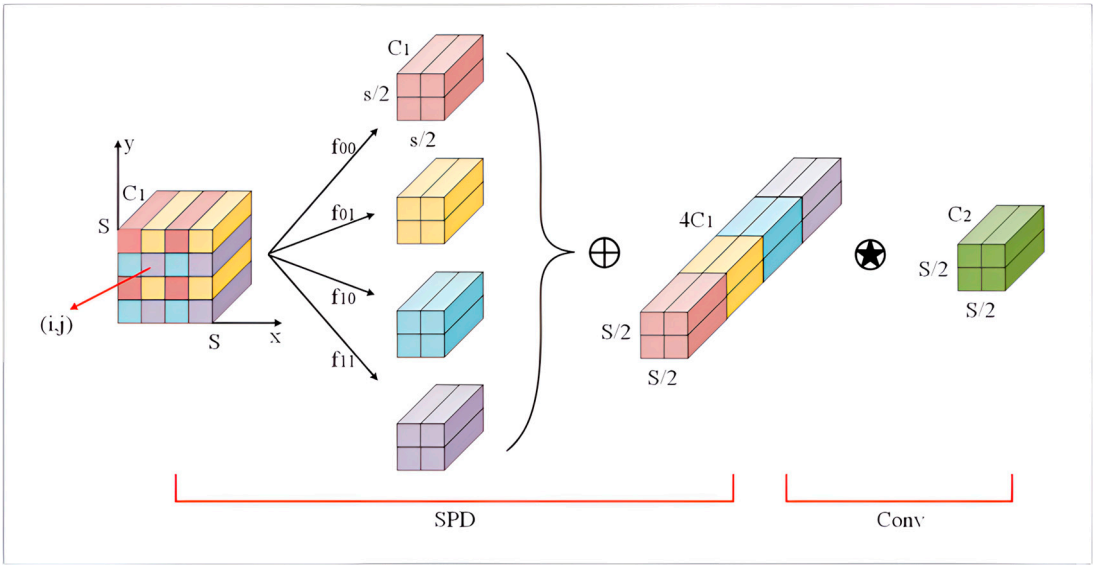

Figure 3. Illustration of SPD-Conv when scale = 2.

3.2. C2fCIB

To optimize model computational efficiency and adapt to edge device deployment requirements, this study adopts the Context Interaction Bottleneck Module (C2fCIB) based on deep separable convolutions proposed by Wang et al. This module is a lightweight modification of the original C2f module in the YOLO architecture, particularly suitable for processing deep features in the backbone network. Its core design involves constructing an inverted bottleneck structure using deep separable convolutions in the feature propagation path, as shown in Figure 4. First, a 1 × 1 convolution is used to expand the channel dimension, followed by a 3 × 3 deep convolution that processes each channel independently. Finally, a 1 × 1 convolution is used to compress the channel dimension. This structure, which first expands, then performs deep convolution, and finally compresses, significantly

reduces the number of model parameters and computational complexity. Specifically, deploying the C2fCIB module in areas with low feature map resolution and high channel counts can improve computational efficiency while maintaining model accuracy. Additionally, this module retains the cross-stage connection feature from the original C2f structure, integrating shallow-level detail features with deep-level semantic features to maintain multi-scale feature representation capabilities. This lightweight design enables the model to maintain traffic sign detection accuracy while improving inference speed, providing practical support for real-time processing requirements in actual traffic monitoring scenarios.

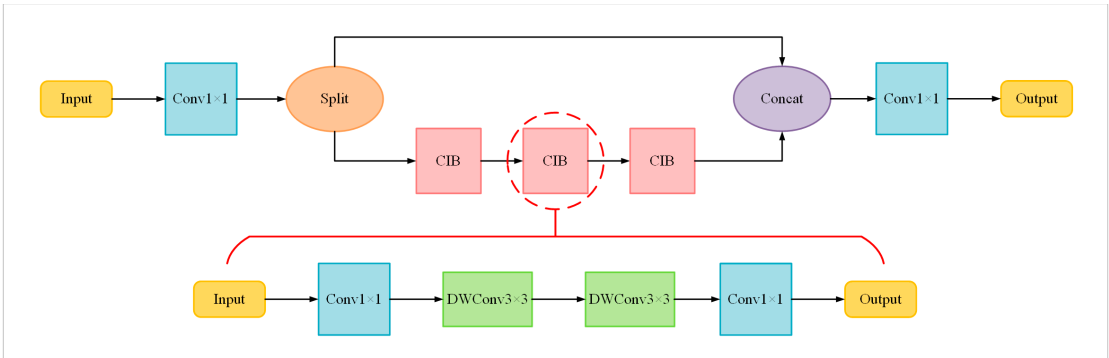

Figure 4. C2fCIB structure, containing CIB structure.

3.3. DPDG

3.3.1. Dynamic Grouping Mechanism

The dynamic grouping mechanism addresses the issues of empirical dependency and insufficient generalization in traditional fixed grouping methods (such as SGE's preset $G = 8$) in channel division. This mechanism abandons the prior assumption of a preset grouping number and instead dynamically calculates the optimal grouping number G based on the number of channels C in the input feature map. The mathematical expression is as follows:

$$G = \ argmax_g \left\{ g \in N^+ \middle| \frac{C}{g} \in Z^+, \ g \leq \left\lfloor \frac{C}{r} \right\rfloor \right\} \tag{2}$$

Among them, C is the channel dimension of the input feature map. By constraining $g \leq ⊚C/r⊚$, the grouping granularity and computational efficiency are balanced. Parameter r represents the reduction rate, and the empirical value is set to 16 to control the upper limit of the number of groups, prevent excessive grouping, and balance the grouping granularity and computational efficiency. The mechanism architecture is shown in Figure 5.

The algorithm begins by extracting the input feature map, initially setting the number of groups to $G_{init} \leq ⊚C/r⊚$, and restricting it to an integer. It then iteratively adjusts the number of groups to ensure that channels are evenly distributed within each group and no information is fragmented. The process continues in a loop until $C\%G = 0$. If the condition is not met after the final iteration, the number of groups will be set to 1 to avoid division-by-zero errors. The figure shows that when the input channel number C is 256, the calculation yields $G = 16$, which means the channels are divided into 16 groups, with each group containing 16 channels. Similarly, when $C = 512$ and $G = 32$, each group still maintains 16 channels. More typical examples are shown in Table 1. After dynamic grouping, the process proceeds to the dual-pooling channel attention stage. This design enables the module to adapt to feature maps of different scales, such as $C = 256$ for shallow layers and $C = 1024$ for deep layers in the backbone, thereby avoiding performance degradation caused by fixed grouping during cross-scale feature fusion.

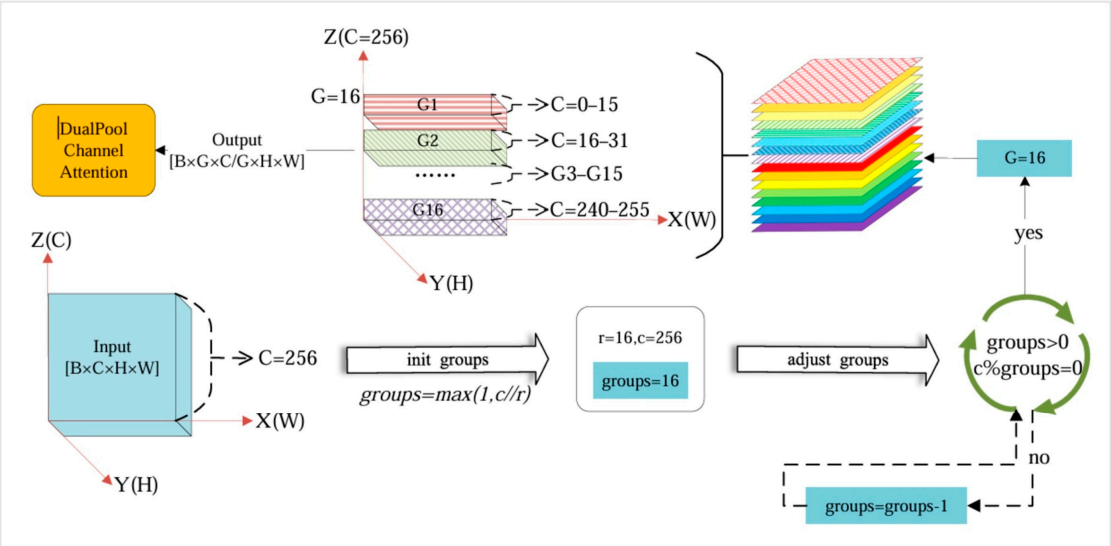

Figure 5. Dynamic grouping mechanism architecture diagram.

Table 1. Typical input channel number grouping result.

Channels	Reduction	Groups
256	16	16
512	16	32
300	16	15
127	16	1

The theoretical advantage of the dynamic grouping mechanism lies in its adaptive channel division strategy, which maximizes the consistency of features within a group. Traditional fixed grouping methods tend to result in two extremes when the number of channels changes: when there are few channels, grouping becomes redundant, with too few channels within a group, limiting feature expression capabilities; when there are many channels, grouping is insufficient, with redundant channels within a group, leading to the loss of detailed information. Dynamic grouping ensures stable channel counts within groups through mathematical constraints, such as maintaining a constant count of 16 in the example above. This enhances local texture responses in shallow features and captures fine-grained semantic information in deep features.

The setting of a reduction rate r = 16 in the dynamic grouping mechanism is based on systematic research findings that involve a thorough trade-off between model complexity and feature representation capability [28,29]. A larger r value (e.g., r = 32) limits the maximum number of groups, leading to excessive channels within each group, thereby weakening the network's ability to extract fine-grained features such as traffic sign textures and edges. A smaller r value (e.g., r = 8) may introduce too many groups, resulting in information redundancy and computational resource waste during feature interaction. Through grid search and empirical analysis, r = 16 was the optimal compromise, better balancing accuracy and efficiency. Additionally, the computational complexity of dynamic grouping is $O(C/r)$, which adds almost no latency under GPU parallel architecture, laying the foundation for hardware compatibility in edge deployment.

3.3.2. Dual-Pooling Channel Attention

The design of the dual-pooling channel attention module stems from an in-depth analysis of traditional single-pooling strategies. Existing methods typically rely solely on global average pooling (AvgPool) to generate channel weights, which essentially reflect fea-

ture importance through the mean distribution across the channel dimension. However, mean pooling tends to smooth out feature responses, potentially weakening the contribution of locally significant regions, especially in small object detection tasks where the target region accounts for a low proportion, and background features easily dilute its response. To address this issue, this paper proposes a dual-pooling channel attention mechanism, whose working principle is illustrated in Figure 6.

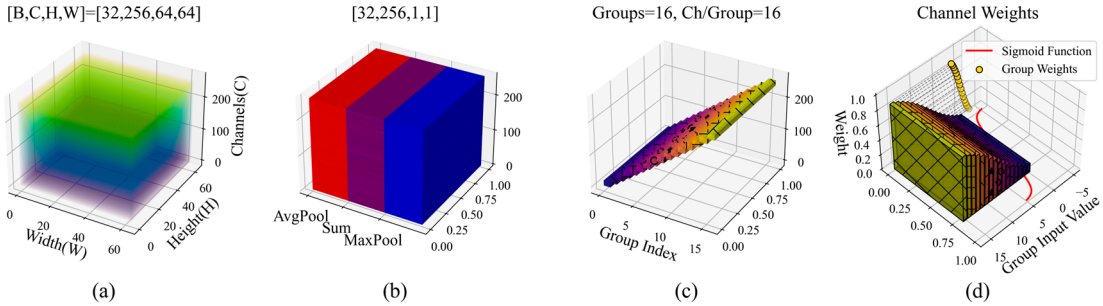

Figure 6. Dual-pooling channel attention: (**a**) input feature; (**b**) dual-pooling results; (**c**) grouping mean; (**d**) sigmoid activation.

This mechanism combines global average pooling and global MaxPool in parallel, balancing the overall statistical characteristics of the channel dimension with local salient responses, thereby covering more comprehensive feature information. Specifically, given an input feature map $X \in R^{B\times C\times H\times W}$, the module first performs global average pooling and max pooling independently on each channel, respectively, obtaining the mean vector representing the overall activity of the channel $AvgPool(X) \in R^{B\times C\times 1\times 1}$ and the response vector $MaxPool(X) \in R^{B\times C\times 1\times 1}$ that focuses on local extrema. The two are fused into the initial channel weights by adding them element by element:

$$W_{channel} = \sigma(AvgPool(X) + MaxPool(X)) \tag{3}$$

Here, σ is the sigmoid function, which normalizes the weights to the range [0, 1]. This design is mathematically equivalent to imposing dual constraints on the channel features: the mean weights ensure the global stability of the feature distribution, while the maximum weights enhance the saliency of key regions. To further adapt to the dynamic grouping mechanism, the module divides the fused weight tensor $W_{channel}$ into G subgroups according to the dynamic group number G, with each group containing C/G channels, and generates refined weights $W_{channel} \in R^{B\times G\times 1\times 1}$. through intra-group mean calculation. This strategy significantly reduces computational complexity while preserving channel differences and enhancing intra-group feature consistency.

Theoretical analysis shows that the dual-pooling strategy can cover a broader range of channel information entropy. According to the Shannon entropy formula in information theory [30], the information entropy $H(X)$ of a feature map can characterize its information richness:

$$H(X) = -\sum_{i=1}^{C} p(x_i)\log p(x_i) \tag{4}$$

where $p(x_i)$ is the normalized response probability of the i-th channel. Compared to single pooling, dual-pooling fusion weights can improve information entropy coverage, thereby enhancing the model's adaptability to complex scenes and improving its robustness in detecting small objects in low-light conditions. This improvement can be attributed to the dual-pooling strategy's multidimensional modeling of channel features, where mean pooling suppresses background noise and max pooling enhances target edges. The synergistic

effect of these two mechanisms enables the network to more accurately locate and classify small-scale objects.

In addition, the double pooling module has a significant computational efficiency advantage. Since the global pooling operation only involves simple tensor compression and addition operations, its computational overhead is negligible. It is also decoupled from the dynamic grouping mechanism's iterative computation process, ensuring the module's efficiency in real-time inference.

3.3.3. Lightweight Spatial Branch

The design philosophy behind lightweight spatial branches is to enhance the model's ability to perceive key regions in the spatial dimension without significantly increasing computational overhead, particularly addressing common challenges such as edge blurring and background interference in traffic sign detection tasks. Traditional spatial attention mechanisms typically employ large-sized convolution kernels or multi-branch structures, which, although they improve feature discriminative power, introduce additional parameter counts and reduce inference speed, making them unsuitable for the efficient deployment requirements of lightweight models. To address this, this work proposes a minimalist spatial weight generation strategy that balances computational efficiency and feature enhancement effects through the synergistic design of channel compression and local convolution. Specifically, given the input feature map $X_{max} \in R^{B\times C\times H\times W}$, the branch first performs two compression operations along the channel dimension: the first is maximum value compression, which takes the maximum response value of all channels at each spatial position (h, w) to generate the feature map $X_{max} \in R^{B\times 1\times H\times W}$, with the mathematical expression as follows:

$$X_{max}(b, 1, h, w) = max_c X(b, c, h, w) \tag{5}$$

This operation effectively preserves the prominent edge features of traffic signs, such as the red circular outline of prohibition signs. The second is mean compression, which calculates the mean of the channel dimension for each spatial position to generate the feature map $X_{max} \in R^{B\times 1\times H\times W}$, whose formula is:

$$X_{avg}(b, 1, h, w) = \frac{1}{C}\sum_{c=1}^{C} X(b, c, h, w) \tag{6}$$

Mean compression suppresses random noise, such as road reflections or leaf shadows, through smoothing while preserving the overall spatial distribution characteristics of the target. To further integrate the advantages of the two compression features, X_{max} and X_{avg} are concatenated along the channel dimension to obtain the multi-scale spatial feature $X_{compress} \in R^{B\times 1\times H\times W}$.

Subsequently, a single 3 × 3 convolution operation is applied to $X_{compress}$ to model local spatial relationships. The convolution kernel parameters are shared across all spatial positions, yielding a single-channel spatial weight map $W_{spatial} \in R^{B\times 1\times H\times W}$, which is normalized to the [0, 1] interval via the Sigmoid function:

$$W_{spatial} = \sigma\big(ConV_{3\times 3}(X_{compress})\big) \tag{7}$$

The number of parameters in the overall spatial branch is strictly limited by combining the channel weight control at the group level in the dynamic grouping mechanism. Compared to the YOLOv10n model, this design can reduce the false detection rate in complex urban scenes, especially under uneven lighting or partial occlusion conditions. The spatial

weights can precisely enhance the edge response of the target, and the lightweight spatial branch effect in traffic sign scenes is shown in Figure 7. The red-highlighted area precisely captures the edge contours of traffic signs, while the internal areas of the signs maintain moderate responses. The blue areas represent the background regions, which are effectively suppressed, and the road areas exhibit only weak responses, thereby significantly reducing false positives. Additionally, by omitting the multi-scale fusion or pyramid structure used in traditional methods, the inference time of this branch increases only minimally, achieving an optimal balance between parameter count and computational efficiency, fully meeting the real-time detection requirements of edge devices.

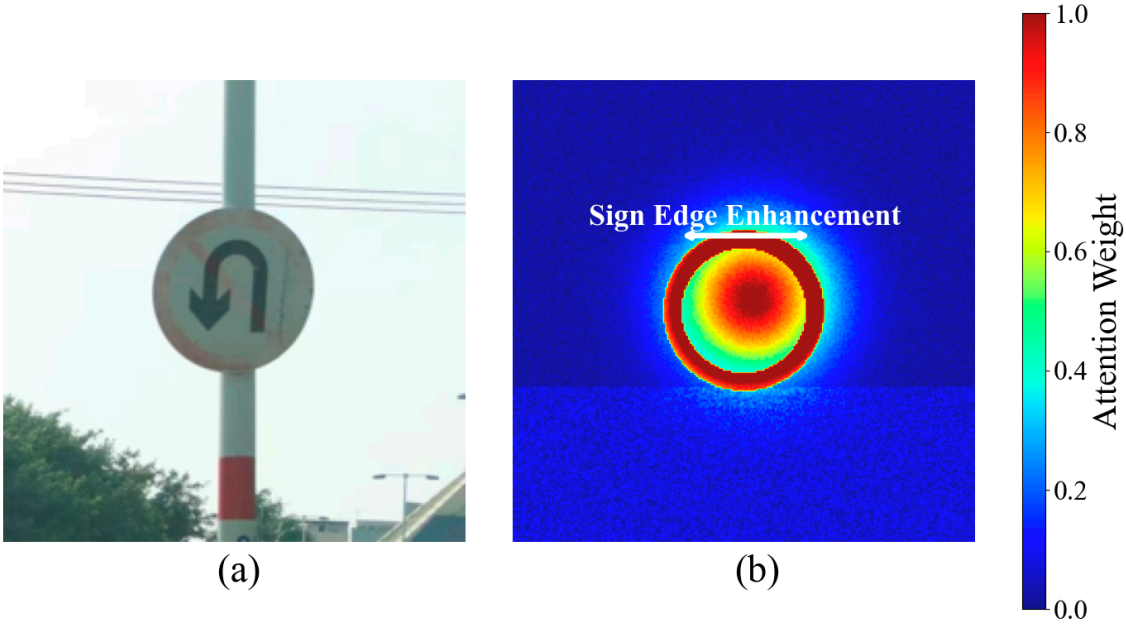

Figure 7. Lightweight spatial branch visualization: (**a**) original traffic sign image; (**b**) spatial attention heatmap.

It is worth noting that the lightweight spatial branch complements the dynamic grouping mechanism and dual-pooling channel attention: channel weights $W_{channel} \in R^{B\times G\times 1\times 1}$ focus on "which channels are important," while spatial weights $W_{spatial} \in R^{B\times 1\times H\times W}$ answer "which positions are important." This fusion strategy enables the model to adaptively select features in both the channel and spatial dimensions, with the two being jointly optimized through element-wise multiplication:

$$X_{out} = X \cdot W_{channel} \cdot W_{spatial} \tag{8}$$

4. Experiment

4.1. Experiment Setup

To verify the practical effectiveness of the proposed method, we conducted a large number of experiments on a large-scale real dataset. The experiment used the CCTSDB 2021 [31] traffic sign dataset created by the Changsha University of Science and Technology team, which provides 20,492 traffic sign images covering three typical categories of signs: warning, mandatory, and prohibitory. These categories comprehensively cover the core functional categories of China's traffic regulatory system. Figure 8 shows examples of the three categories of signs. Among these, small objects with pixel areas less than 50×50 account for as much as 42.7%, a feature that accurately simulates the challenges of detecting traffic signs at long distances and low resolutions in real-world road scenarios, enhanc-

ing the model's difficulty. To ensure the reliability and robustness of the experimental results, all reported performance indicators are derived from a large independent test set consisting of 1500 images. This test set was strictly retained during the training process. Consistent and significant improvements observed in all key indicators provide strong evidence that the performance enhancement brought by our method is not due to random fluctuations but is statistically significant and repeatable. After the test set was separated, the total number of training images became 18,992. The remaining dataset was divided into a training set of 15,194 images and a validation set of 3798 images at a ratio of 4:1.

Figure 8. Examples of three types of signs in the dataset: (**a**) warning; (**b**) mandatory; (**c**) prohibitory.

We use a high-performance computing environment to ensure experimental efficiency, equipped with Intel® Xeon® Platinum 8352V processors (Intel Corporation, Santa Clara, CA, USA) and NVIDIA GeForce RTX 4090 graphics cards (NVIDIA Corporation, Santa Clara, CA, USA), running on the Windows 11 operating system. The deep learning framework selected is PyTorch 2.0.1, and CUDA 11.7 technology is used to accelerate model training and inference processes, thereby improving the computational efficiency and data processing capabilities of the experiment.

This experiment was conducted over 250 training iterations, using a training configuration with an input resolution of 640 × 640 pixels (imgsz) and a batch size of 32. The optimizer selected was Stochastic Gradient Descent (SGD), which is known for its convergence stability and strong generalization capabilities. The initial learning rate was set to 0.01. Mosaic and MixUp data augmentation were disabled to reduce interference from complex scenes on learning basic features, enabling the model to focus more on the essential features of traffic signs. The training process enabled 8-thread data loading and automatic mixed-precision acceleration for computation.

This experiment uses six standard metrics from the field of object detection to comprehensively evaluate model performance: mAP@0.5, mAP @ 0.5–0.95, precision (P), recall (R), parameters (Params), and speed (FPS). Among them, mAP@0.5 is the average precision (AP) value under a single intersection over union (IoU) threshold, which is relatively simple to calculate. However, mAP@0.5:0.95 considers multiple IoU thresholds and is the core metric for evaluating the overall performance of the model, better reflecting its comprehensive capabilities. Higher P and R values indicate fewer false positives and false negatives, respectively, directly reflecting the reliability of the detection results; the number of parameters reflects the computational complexity of the model; and FPS quantifies the model's inference speed by measuring the number of images processed per unit of time, making it a key indicator for assessing real-time performance.

4.2. Comparison of DPDG with Mainstream Attention Modules

To comprehensively evaluate the effectiveness of the DPDG module, we compare it with current mainstream attention mechanisms. All attention modules are inserted into the same position at the end of the P3/8-small branch of the neck layer of YOLOv10n to

ensure fairness in the comparison. The training configuration is strictly unified, and the experimental results are shown in Table 2.

Table 2. Comparison of attention module performance.

Model	mAP@0.5	mAP@0.5:0.95	P	R	Params/M	Incremental Params	FPS	FLOPs
YOLOv10n	0.730	0.464	0.844	0.680	2.695586	-	1000	8.4
YOLOv10n + SE	0.722	0.454	0.839	0.658	2.696098	+0.000512	1111	8.4
YOLOv10n + CBAM	0.726	0.443	0.841	0.661	2.699844	+0.004258	1000	8.4
YOLOv10n + SGE	0.741	0.468	0.831	0.696	2.945202	+0.249616	1250	8.7
YOLOv10n + DPDG	0.740	0.473	0.878	0.676	2.695604	+0.000018	1250	8.7

Experimental data shows that the YOLOv10n model with DPDG achieves a 1.94% improvement in mAP@0.5:0.95 and a 25% improvement in inference speed while maintaining almost the same number of parameters, and the FLOPs increase slightly, with significantly higher accuracy than YOLOv10n with other modules added. Compared to the SGE module, which has nearly equivalent speed, DPDG achieves a 1.07% increase in mAP@0.5:0.95 with fewer parameters. As shown in Figure 9, in the Pareto front diagram [32,33] composed of model complexity (incremental parameter count ΔParams/M) and detection accuracy (mAP@0.5:0.95), the DPDG module is located in the upper-left region of the Pareto front. The Pareto frontier represents the boundary solution set where all objectives cannot be improved simultaneously in a multi-objective optimization problem. The upper-left region signifies the ideal direction of minimizing model parameter counts while maximizing accuracy. This Pareto optimality clearly demonstrates the effectiveness of the DPDG module in balancing accuracy and efficiency, i.e., significantly enhancing the robustness of sign detection in complex traffic scenarios without significantly increasing computational overhead.

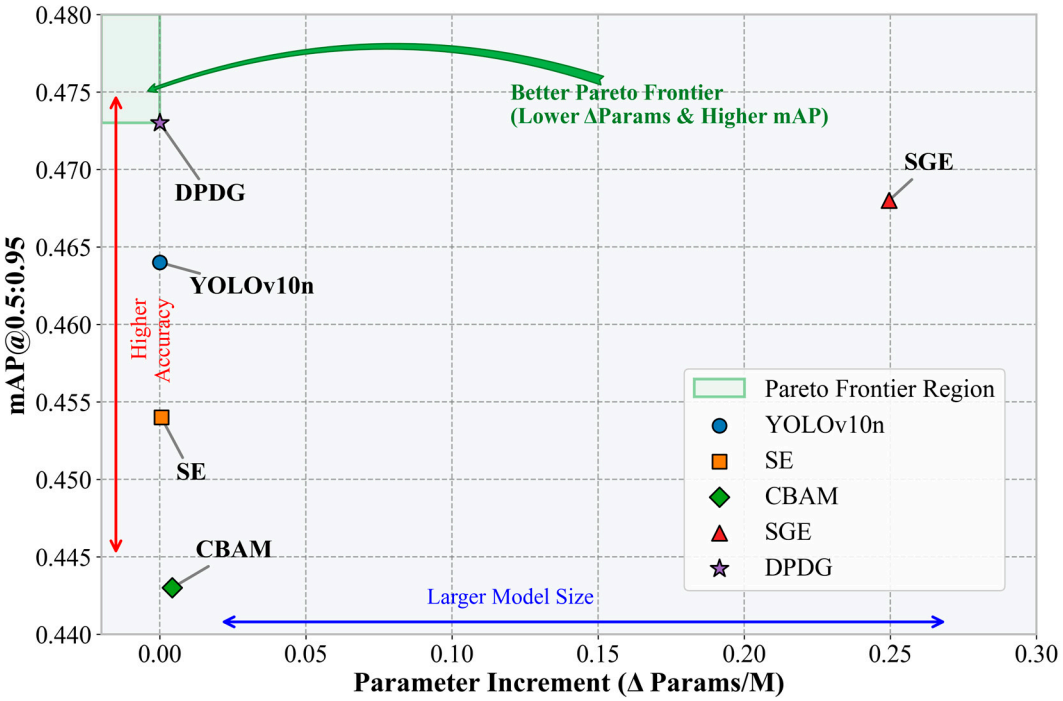

Figure 9. Pareto frontier analysis of attention modules.

The significant performance advantages demonstrated by DPDG in multi-module system comparison evaluations are mainly due to its efficient collaboration and high re-

source utilization design. The SGE module enhances local features through grouped spatial attention, improving recall rates and demonstrating greater robustness in detecting occluded and deformed signs. However, this also increases computational complexity, and the fixed grouping leads to channel information fragmentation, resulting in higher false positive rates. In contrast, DPDG's dynamic grouping mechanism adapts the number of groups to effectively avoid channel redundancy issues caused by fixed grouping, and reasonable adjustment of channel division enables more balanced feature representation. The SE module reweights feature maps through channel attention, but overemphasizing the channel dimension may destroy the integrity of spatial features, especially in traffic sign detection, where spatial location information is critical to positioning accuracy, leading to a significant decrease in mAP@0.5:0.95. However, the DPDG's dual-pooling strategy combines global statistical features with local salient responses to enhance target discrimination in complex backgrounds. Experimental results also validate the effectiveness of our lightweight module design. DPDG uses a single 3×3 convolution layer to generate spatial weights, resulting in lower computational overhead compared to the complex spatial branch of CBAM. Additionally, the dynamic grouping mechanism further improves efficiency by reducing redundant computations.

In summary, DPDG achieves the greatest accuracy improvement with the smallest increase in parameters, providing an efficient attention method for lightweight models.

4.3. Ablation Experiment

To validate the independent contributions of each module and the effectiveness of the algorithm improvements, this experiment uses the YOLOv10n algorithm as the base framework and conducts ablation experiments on the CCTSDB2021 dataset. Through a progressive integration strategy, the SPD-Conv, C2fCIB, and DPDG modules are systematically validated, and our method is compared with the larger YOLOv10s model. All experiments strictly adhere to the single-variable principle, with consistent training configurations and dataset splits to ensure rigorous experimental results. The experimental results are shown in Table 3.

Table 3. Ablation experiment.

YOLOv10n	SPD-Conv	C2fCIB	DPDG	YOLOv10s	mAP@0.5	mAP@0.5:0.95	P	R	Params/M	Incremental Params	FPS
√	×	×	×	×	0.730	0.464	0.844	0.680	2.695586	-	1000
√	√	×	×	×	0.785	0.510	0.893	0.722	3.280546	+0.584960	1111
√	√	√	×	×	0.787	0.508	0.886	0.714	2.827298	+0.131712	1000
√	√	√	√	×	0.794	0.513	0.896	0.725	2.827316	+0.131730	1111
√	×	×	×	√	0.804	0.509	0.895	0.733	8.037282	+5.341696	769

4.3.1. Key Findings and Analysis

Analysis of the experimental data table reveals a key finding: replacing stride convolution and pooling layers with spatial-to-depth convolution significantly improves mAP@0.5 and mAP@0.5:0.95 by 7.53% and 9.91%, respectively, with P improving by 5.81%, R by 6.18%, and FPS by 11.1%. However, this resolution preservation comes at the cost of a 21.7% increase in parameter count. These results demonstrate that in object detection, the SPD-Conv module can maintain feature map resolution by converting the spatial dimension into the depth dimension, effectively mitigating the common issue of small object detail loss during downsampling. Additionally, parallel operations enhance speed, but the number of parameters tends to surge. Therefore, maintaining resolution via SPD-Conv requires balancing detection accuracy (Accuracy) with the number of parameters.

Therefore, the experiment further increased C2fCIB to achieve cross-stage optimization and enhance the fusion capabilities of features at different scales. After introducing

C2fCIB on top of SPD, the number of parameters decreased by 13.82%, mAP@0.5 improved by 0.25%, but mAP@0.5:0.95 slightly decreased by 0.39%, P decreased by 0.78%, R decreased by 1.11%, and FPS fell back to the baseline level. This phenomenon indicates that C2fCIB reduces redundant computations through channel compression, albeit at the cost of slightly sacrificing accuracy. However, it reduces the number of parameters, laying the foundation for subsequent lightweight networks, and to some extent addresses the trade-off issues introduced by SPD-Conv.

Although the current P, R, and FPS are still superior to YOLOv10n, given the importance of accuracy and efficiency in traffic sign detection, we require a more comprehensive model. We have already validated the effectiveness of the DPDG module in Section 4.2, and its dynamic enhancement effect is evident. After improving the neck layer small object detection head with DPDG, the complete YOLO-DPDG model achieves an 8.77% increase in mAP@0.5 and a 10.56% increase in mAP@0.5:0.95, with a 6.16% increase in P and a 6.62% increase in R, compared to the YOLOv10n model, with an incremental parameter count of 0.13M. And an 11.11% improvement in FPS.

At this point, the ablation experiments for the YOLO-DPDG network model are nearing completion. However, we have included an additional dataset for YOLOv10s at the end to highlight the comparative advantages of our research against larger models. The visualization of the experimental results comparing YOLOv10n, YOLOv10s, and our network is shown in Figure 10. Compared to YOLOv10n, YOLOv10s achieves an increase of 10.14% and 9.70% in mAP@0.5 and mAP@0.5:0.95, respectively, with 8.04 million parameters, while P improves by 6.04% and R by 7.79%. However, the number of parameters increases by 198.1%, and FPS decreases by 23.1%. Further calculating the accuracy-parameter ratio (mAP@0.5:0.95/Params), YOLO-DPDG achieves 0.181, far exceeding YOLOv10s' 0.063. This result demonstrates the effectiveness of our algorithm improvements and shows that through targeted design of lightweight modules, we can significantly reduce computational resource requirements while approaching or even surpassing the performance of large models. In summary, for small object detection, whether a lightweight model like YOLOv10n or a high-performance model like YOLOv10s is required, we recommend using the YOLO-DPDG network model proposed in this study.

4.3.2. Module Synergy Effects

From ablation experiments, it was found that adding the first two modules consecutively resulted in a slight decrease in performance compared to adding only the first module. However, when all modules were integrated into the network, performance improved and surpassed the previous results, which is worth further exploration.

We conducted combination experiments on the modules. Saltelli et al. [34] proposed that by decomposing the variance contributions of model outputs, the interactive effects of multi-scale features can be quantified. To provide an interpretable assessment of multi-module synergistic effects, we adopted Sobol's sensitivity index [35] for quantification. Additionally, we referenced the adversarial generation method proposed by Wang and Gupta [36] to design occlusion experiments. We randomly generated rectangular or irregular masks covering 50% of the test images to simulate partial occlusion in real-world scenarios, verifying the feature compensation capability of the dynamic grouping mechanism and providing reliability assurance for high-risk scenarios such as autonomous driving. The experimental results are shown in Table 4. We speculate that the performance improvements are primarily attributed to the synergistic effects of component collaboration.

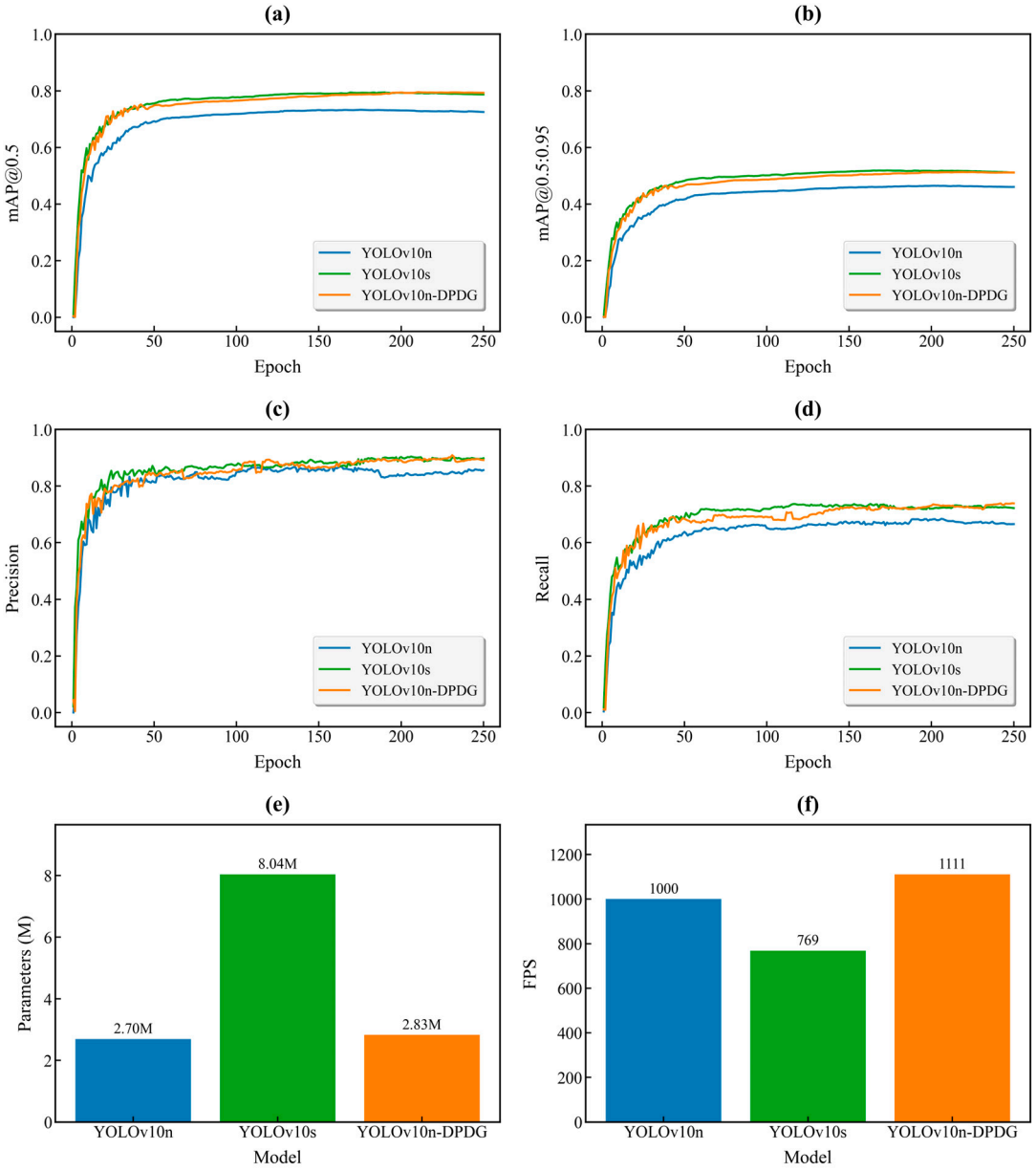

Figure 10. Visualization of model performance: (**a**) mAP@0.5; (**b**) mAP@0.5:0.95; (**c**) precision; (**d**) recall; (**e**) parameters; (**f**) FPS.

Table 4. Quantitative comparison of synergistic effects.

SPD-Conv	C2fCIB	DPDG	mAP@0.5	mAP@0.5:0.95	P	R	Params/M	Incremental Params	FPS	Cross-Scale Feature Correlations	mAP@0.5:0.95 When Occluded by 50%
√	√	×	0.787	0.508	0.886	0.714	2.827298	+0.131712	1000	0.61	0.426
√	×	√	0.784	0.510	0.877	0.715	3.280564	+0.584978	1111	0.68	0.445
×	√	√	0.741	0.477	0.833	0.678	2.491956	−0.203630	1250	0.52	0.392
√	√	√	0.794	0.513	0.896	0.725	2.827316	+0.131730	1111	0.76	0.471

Experimental data indicate that pairwise combinations of modules yield significant improvements in mAP or speed. However, the deep collaborative design of the three modules in this study enables comprehensive optimization of the entire network, with interaction effects significantly outperforming the simple stacking of single or pairwise modules.

The SPD-Conv module effectively preserves the core semantic information and fine-grained spatial features of the input, providing a rich information foundation for subsequent processing, and the C2fCIB module reduces computational complexity through its

bottleneck structure, which performs channel compression and reorganization of features. The two modules exhibit high correlation at the feature level, indicating that C2fCIB effectively maintains the key information extracted by SPD-Conv during the compression process. The fine-grained features retained by SPD-Conv provide high-resolution input for DPDG, and DPDG adaptively enhances the response of key regions at different scales of SPD-Conv through dynamic weight allocation. Specifically, when processing deep low-resolution features (such as the P5 layer, which is responsible for distant small objects), the channel attention branch of DPDG exhibits a more concentrated weight distribution, effectively focusing on discriminative channel information and improving small object detection capabilities. When processing shallow high-resolution features (such as the P3 layer, responsible for close-range targets), the spatial attention branch of DPDG plays a dominant role, reinforcing responses to local details and edges. SPD-Conv and DPDG exhibit strong cross-scale correlation, validating the critical role of SPD-Conv as a high-quality input source for DPDG's dynamic refinement effects. Although the correlation between C2fCIB and DPDG is lower than that of the aforementioned combinations, their synergistic contribution to model efficiency is crucial. The bottleneck structure of C2fCIB not only reduces computational burden but also standardizes the distribution of features across different scales. This standardization helps the attention weights learned by DPDG maintain better semantic consistency across different layers, thereby enhancing the model's overall robustness. The significant computational efficiency optimization achieved by C2fCIB and DPDG together offers an undeniable advantage for detection tasks with high real-time requirements.

To address the issue of traffic signs being easily obstructed and leading to missed detections in real-world road scenarios, this paper designed robustness verification experiments under extreme obstruction conditions. The experimental results show that the YOLO-DPDG model demonstrates significant advantages under occlusion conditions due to its collaborative enhanced feature compensation capabilities. Specifically, the SPD-Conv module retains fine-grained spatial information in high-resolution feature maps, effectively capturing key local details such as edges and textures in the visible regions of partially occluded targets, thereby providing reliable foundational information for subsequent processing. The core advantage of the DPDG module lies in its adaptive attention mechanism, which actively focuses on unobstructed effective regions and dynamically enhances their response weights based on the saliency features of these regions. This mechanism partially compensates for information loss caused by occlusion, guiding the model to focus on the distinguishable parts of the target. The C2fCIB module utilizes its CIB mechanism to dynamically identify and suppress background noise and redundant features introduced by occlusion by constraining information flow and optimizing mutual information between feature channels. This effectively enhances the model's adaptability to local feature loss or mutation. In high-risk scenarios such as autonomous driving, target occlusion and partial visibility are commonplace. The synergistic effect of the three modules significantly enhances the model's ability to extract and utilize features under incomplete information conditions. Quantitative evaluations show that under extreme conditions with a target occlusion rate of 50%, the mAP@0.5:0.95 of YOLO-DPDG reaches 0.471, fully validating the model's robustness advantage in complex occlusion scenarios.

Overall, the three-module joint architecture significantly outperforms the two-module combination scheme in all metrics, demonstrating the synergistic gain effect between modules. The YOLO-DPDG network is not simply stacking modules but achieves an optimal balance between computing resources and feature representation through cascaded optimization of high-resolution feature retention, cross-stage information purification, and dynamic attention enhancement.

4.3.3. Detection Effect Comparison

As shown in Figure 11, we conducted a comparative analysis of the detection results of YOLOv10n and its improved model, YOLO-DPDG. From sample (a), it can be seen that YOLO-DPDG demonstrates higher detection accuracy. From sample (b), it can be seen that when handling small objects at long distances, YOLO-DPDG exhibits more comprehensive detection capabilities, successfully identifying traffic signs that the original model missed. From sample (c), it can be seen that in low-light nighttime scenes, YOLO-DPDG effectively suppresses false detections caused by headlight reflections. At the same time, the original model mistakenly identifies reflections as traffic signs. In summary, the improved algorithm proposed in this paper effectively alleviates issues such as insufficient feature expression capabilities and difficulties in identifying small objects, thereby enhancing the model's robustness and accuracy.

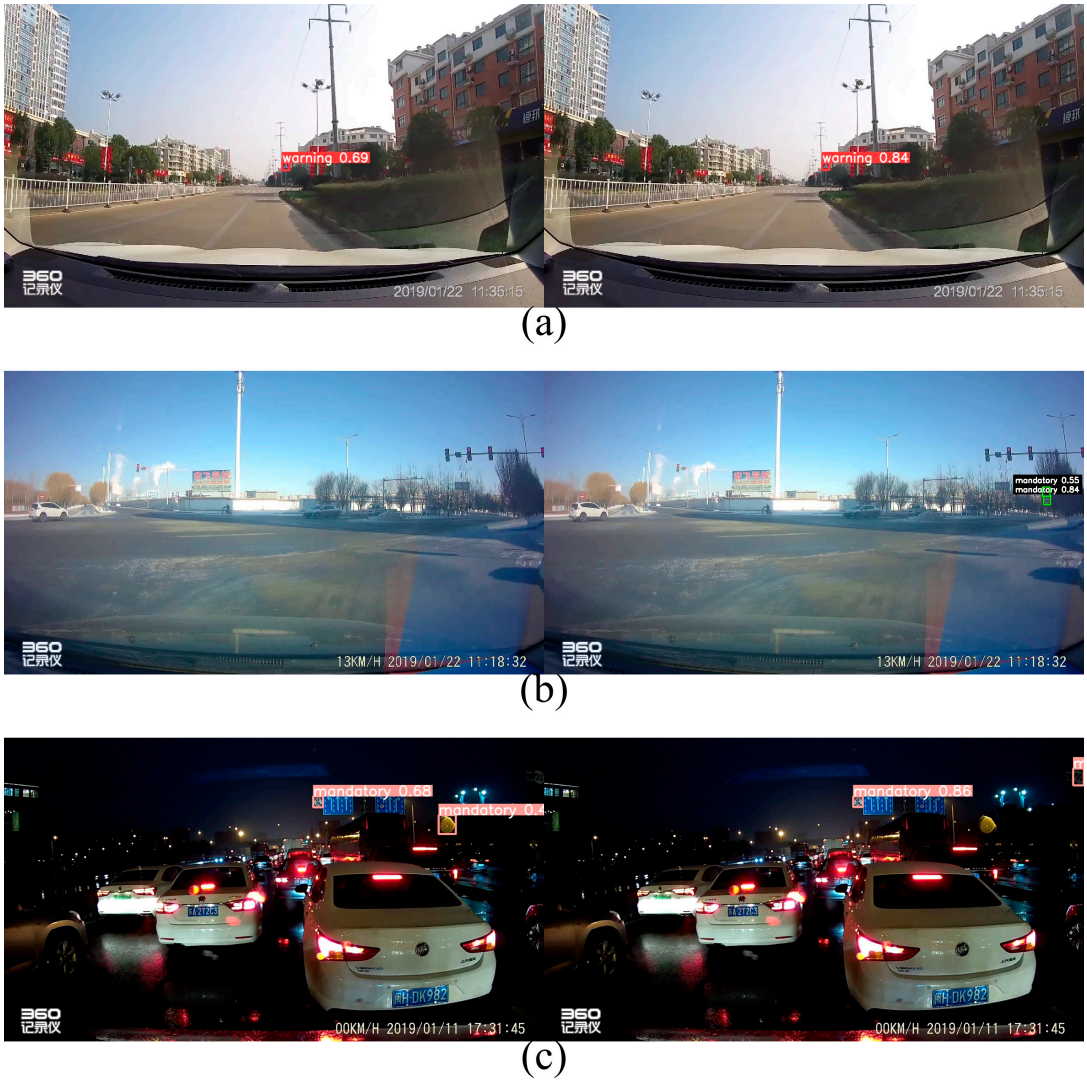

Figure 11. Comparison of detection results. The left column shows the detection results of YOLOv10n, and the right column shows the detection results of YOLO-DPDG: (**a**) comparison of detection accuracy; (**b**) false negatives of the original model; (**c**) false positives of the original model. In the picture, the non-English characters "360 记录仪" refer to an electronic device installed in a vehicle, which captures real-time video, audio and environmental data during the driving process through high-definition cameras.

4.4. Generalization Experiment

To verify the generalization ability of the YOLO-DPDG model on different datasets, this study conducted supplementary experiments on the Tsinghua-Tencent 100K (TT100K) dataset [26]. TT100K is a large-scale traffic sign dataset compiled by Tsinghua University and the Tencent Joint Laboratory, containing 100,000 high-resolution street view images, covering various scenarios such as urban roads, highways, and rural roads, as well as various lighting and weather conditions. The dataset includes 30,000 traffic sign instances and 221 different types of traffic signs. We selected 45 categories with more than 100 instances for the experiment to eliminate the influence of class imbalance. After preprocessing, the dataset was divided into 6793 training images, 1949 validation images, and 996 test images, with a small target proportion of up to 84%, providing an ideal platform for verifying the model's generalization ability in challenging scenarios.

The experimental setup is consistent with that in Section 4.1, using the same hyperparameters and training strategies. We trained the YOLOv10n baseline model and the YOLO-DPDG model separately on the TT100K dataset and evaluated their performance on the test set. The experimental results are shown in Table 5.

Table 5. The experimental results on the TT100K dataset.

Model	mAP@0.5	mAP@0.5:0.95	P	R	Params/M	Incremental Params	FPS	FLOPs
YOLOv10n	0.67	0.501	0.731	0.579	2.711966	-	769.23	8.3
YOLOv10n + DPDG	0.695	0.529	0.698	0.61	2.843696	+0.131730	625	11.1

As can be seen from Table 5, YOLO-DPDG achieved consistent performance improvements over the baseline model YOLOv10n on the TT100K dataset. Specifically, the mAP@0.5 and mAP@0.5:0.95 metrics increased by 2.5% and 2.8%, respectively, and the recall rate (R) increased by 3.1%. Although the precision slightly decreased on the TT100K dataset, the mAP metric, which measures the overall detection performance, significantly improved, demonstrating the enhancement of the model's generalization ability.

In conclusion, YOLO-DPDG not only performed well on the main experimental dataset CCTSDB2021 but also demonstrated excellent generalization performance on the more diverse and challenging dataset TT100K. This verifies the strong adaptability of the proposed method to different traffic sign datasets and scenarios, providing strong support for its deployment in practical applications.

5. Discussion

The core objective of this study is to address the performance-efficiency trade-off challenge in small object detection of traffic signs by improving the network and designing a new attention mechanism. Based on experimental validation and theoretical analysis, this section discusses the findings from four dimensions: methodological innovation, practicality, limitations, and implications for the field, revealing the more profound significance of the research results.

Compared with existing work, the experimental data in Section 4.2 show that the DPDG module outperforms current mainstream attention mechanisms. The DPDG module effectively combines dynamic grouping with dual-pooling channel attention. Additionally, its lightweight design maintains performance while offering advantages in real-time-critical traffic detection scenarios. The dynamic grouping mechanism breaks free from the fixed structural constraints of traditional attention modules, reducing channel redundancy through adaptive adjustment of group counts and enhancing system robustness via dynamic parameter optimization. Furthermore, the introduction of dual-pooling channel attention integrates global statistical features with local salient responses, enabling more

comprehensive information extraction than single-pooled attention. This validates the critical role of information completeness in feature discriminative power.

In edge computing scenarios, YOLO-DPDG demonstrates significant application value. The balance between parameter count and inference speed makes it suitable for low-power devices and maintains advantages over YOLOv10s. This feature is crucial for real-time perception in autonomous driving systems, particularly for detecting small traffic signs that frequently appear on urban roads. Additionally, the dynamic feature compensation mechanism effectively mitigates the impact of partial information loss. By preserving visible edge details and enhancing responses in unobstructed areas, the model reduces occlusion misclassification rates, enhancing its robustness in occlusion scenarios and its practical application value.

Although YOLO-DPDG performs well in most scenarios, it still has the following limitations: First, YOLO-DPDG still has room for improvement in some extreme cases. As suggested by the occlusion experiment (Table 4, with 50% occlusion, mAP@0.5:0.95 drops to 0.471), severe occlusion remains a challenge. Additionally, although the performance has improved under low-light conditions (as shown in Figure 11c), the harsh weather conditions not widely covered in our dataset, such as heavy rain or fog, may reduce performance due to the introduction of noise and decreased contrast. Chen et al. [37] pointed out that integrating infrared or thermal imaging data can significantly improve target discrimination in low-light conditions, and this issue can be optimized in the future through multimodal expansion. Second, the dynamic grouping mechanism has theoretical limitations when the number of input channels is prime. For example, when C = 257, grouping must be forced to 1, which may affect the balance of feature representation. Based on the differentiable architecture search method proposed by Liu et al. [38], we will further design a continuous relaxation grouping strategy in the future. Finally, although the dynamic grouping mechanism is quite robust, it may still perform poorly for extremely rare and complex shapes of markers that are underrepresented in the training data. Introducing deformable convolutions or adaptive receptive field modules can enhance the model's feature extraction capability for non-rectangular targets, further improving generalization performance.

This study proposes a lightweight object detection model optimization method comprising a three-stage processing workflow of "resolution preservation," "feature purification," and "dynamic enhancement." The design of this framework provides a feasible approach and modular reference for constructing efficient and accurate small models. Extending these design concepts to other visual tasks, for example, introducing a dynamic grouping mechanism in instance segmentation can optimize feature aggregation during the mask generation stage. This objective is similar to the adaptive processing of features at different scales in the multi-scale ROIAlign of Mask R-CNN [39], but the focus is on the grouping strategy. In real-time video analysis, drawing inspiration from feature refinement and dynamic enhancement concepts and designing lightweight attention modules can effectively reduce the computational overhead of temporal feature fusion. Additionally, in this study, cross-scale feature correlation analysis provides a quantitative tool for evaluating the synergistic effects between modules. Future research could combine causal inference models [40] to further explore the universal mechanisms of causal interactions between modules, thereby guiding model optimization design.

6. Conclusions

Long-range traffic sign detection is critical for autonomous driving. This paper proposes an improved efficient detection network, YOLO-DPDG, based on YOLOv10n, which significantly improves small object detection accuracy while maintaining real-time inference capabilities. Compared to the original YOLOv10n algorithm, our improved algo-

rithm, YOLO-DPDG, has been optimized in all metrics, resulting in better detection performance. Even when compared to the larger YOLOv10s algorithm, this network still demonstrates a clear advantage in balancing performance and efficiency.

Based on the analysis of the limitations discussed, future work will focus on the following directions: optimizing model performance in extreme scenarios such as occlusion; utilizing differentiable architecture search to improve dynamic grouping strategies to address prime channel constraints; introducing deformable convolutions to enhance feature extraction capabilities for irregularly shaped traffic signs; simultaneously extending core mechanisms such as dynamic grouping and feature refinement to visual tasks like instance segmentation and real-time video analysis; and conducting in-depth exploration of universal interaction mechanisms between modules using causal inference models to further enhance the generalization and interpretability of lightweight models.

Author Contributions: Conceptualization, R.L., M.J. and S.L.; methodology, M.J.; software, M.J.; validation, R.L., M.J. and S.L.; formal analysis, M.J.; investigation, M.J.; resources, M.J.; data curation, M.J.; writing—original draft preparation, M.J.; writing—review and editing, R.L., M.J. and S.L.; visualization, M.J.; supervision, R.L.; project administration, S.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Major Science and Technology Foundation of Zhongshan City (2019B2006, 2019A40027, 2021A1003, 2023AJ002), the First-Class Course Program of Guangdong Province (YLKC202202), the Science and Technology Commissioner Project of Guangdong Province (GDKTP2021025700), and Guangdong Basic and Applied Basic Research Foundation (2024A1515140093).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The datasets generated and/or analyzed during the current study are available in the GitHub repository, accessible via https://github.com/csust7zhangjm/CCTSDB2021 (accessed on 10 May 2025).

Acknowledgments: We acknowledge with thanks the support from the School of Computer Science, University of Electronic Science and Technology of China, Zhongshan Institute. The use of the institute's facilities and the stimulating research atmosphere are gratefully appreciated.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Grigorescu, S.; Trasnea, B.; Cocias, T.; Macesanu, G. A Survey of Deep Learning Techniques for Autonomous Driving. *J. Robot. Syst.* **2020**, *37*, 362–386. [CrossRef]
2. Tabernik, D.; Šela, S.; Skvarč, J.; Skočaj, D. Segmentation-based deep-learning approach for surface-defect detection. *J. Intell. Manuf.* **2020**, *31*, 759–776. [CrossRef]
3. Liu, L.; Ouyang, W.; Wang, X.; Fieguth, P.; Chen, J.; Liu, X.; Pietikäinen, M. Deep Learning for Generic Object Detection: A Survey. *Int. J. Comput. Vis.* **2020**, *128*, 261–318. [CrossRef]
4. Mogelmose, A.; Trivedi, M.M.; Moeslund, T.B. Vision-Based Traffic Sign Detection and Analysis for Intelligent Driver Assistance Systems: Perspectives and Survey. *IEEE Trans. Intell. Transp. Syst.* **2012**, *13*, 1484–1497. [CrossRef]
5. Ruta, A.; Li, Y.; Liu, X. Towards Real-Time Traffic Sign Recognition by Class-Specific Discriminative Features. In Proceedings of the British Machine Conference, Coventry, UK, 10–13 September 2007; pp. 416–430. [CrossRef]
6. Chen, C.; Liu, M.Y.; Tuzel, O.; Xiao, J. R-CNN for Small Object Detection. In Proceedings of the Asian Conference on Computer Vision (ACCV), Taipei, Taiwan, 20–24 November 2016; pp. 214–230.
7. Soans, R.; Fukumizu, Y. Custom Anchorless Object Detection Model for 3D Synthetic Traffic Sign Board Dataset with Depth Estimation and Text Character Extraction. *Appl. Sci.* **2024**, *14*, 6352. [CrossRef]
8. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.

9. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016. [CrossRef]
10. Redmon, J.; Farhadi, A. YOLO9000: Better, Faster, Stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 6517–6525. [CrossRef]
11. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *arXiv* **2018**, arXiv:1804.02767. [CrossRef]
12. Bochkovskiy, A.; Wang, C.Y.; Liao, H.Y.M. YOLOv4: Optimal Speed and Accuracy of Object Detection. *arXiv* **2020**, arXiv:2004.10934. [CrossRef]
13. Ge, Z.; Liu, S.; Wang, F.; Li, Z.; Sun, J. YOLOX: Exceeding YOLO Series in 2021. *arXiv* **2021**, arXiv:2107.08430. [CrossRef]
14. Li, C.; Li, L.; Jiang, H.; Weng, K.; Wei, X. YOLOv6: A Single-Stage Object Detection Framework for Industrial Applications. *arXiv* **2022**, arXiv:2209.02976. [CrossRef]
15. Wang, C.Y.; Liao, H.Y.M.; Bochkovskiy, A. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 18–22 June 2023.
16. Ultralytics. YOLOv8. Available online: https://github.com/ultralytics/ultralytics (accessed on 8 May 2025).
17. Wang, A.; Chen, H.; Liu, L.; Chen, K.; Lin, Z.; Han, J.; Ding, G. YOLOv10: Real-Time End-to-End Object Detection. *arXiv* **2024**, arXiv:2405.14458.
18. Lin, T.Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature Pyramid Networks for Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017. [CrossRef]
19. Hu, J.; Shen, L.; Sun, G.; Albanie, S. Squeeze-and-excitation networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141.
20. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. *CBAM: Convolutional Block Attention Module*; Springer: Cham, Switzerland, 2018. [CrossRef]
21. Li, X.; Hu, X.; Yang, J. Spatial Group-wise Enhance: Improving Semantic Feature Learning in Convolutional Networks. *arXiv* **2019**. [CrossRef]
22. Yang, B.; Bender, G.; Le, Q.V.; Ngiam, J. CondConv: Conditionally Parameterized Convolutions for Efficient Inference. *arXiv* **2019**. [CrossRef]
23. Dai, J.; Qi, H.; Xiong, Y.; Li, Y.; Zhang, G.; Hu, H.; Wei, Y. Deformable Convolutional Networks. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 764–773.
24. Sunkara, R.; Luo, T. No More Strided Convolutions or Pooling: A New CNN Building Block for Low-Resolution Images and Small Objects. *arXiv* **2022**, arXiv:2208.03641. [CrossRef]
25. Wang, J.; Chen, K.; Yang, S.; Loy, C.C.; Lin, D. Region Proposal by Guided Anchoring. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019; pp. 2965–2974.
26. Zhu, Z.; Liang, D.; Zhang, S.; Huang, X.; Li, B.; Hu, S. Traffic-Sign Detection and Classification in the Wild. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 2110–2118.
27. Liu, S.; Qi, L.; Qin, H.; Shi, J.; Jia, J. Path Aggregation Network for Instance Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018; pp. 8759–8768.
28. Hu, E.J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Chen, W. LoRA: Low-Rank Adaptation of Large Language Models. *arXiv* **2021**, arXiv:2106.09685.
29. Pei, J.; Han, Y.; Zhang, X. Model Complexity of Deep Learning: A Survey. *Knowl. Inf. Syst.* **2021**, *63*, 2585–2619. [CrossRef]
30. Cover, T.M.; Thomas, J.A. *Elements of Information Theory*, 2nd ed.; Wiley-Interscience: Hoboken, NJ, USA, 2006; p. 748.
31. Zhang, J.; Zou, X.; Kuang, L.D.; Wang, J.; Sherratt, R.S.; Yu, X. CCTSDB 2021: A More Comprehensive Traffic Sign Detection Benchmark. *Hum.-Centric Comput. Inf. Sci.* **2022**, *12*, 23. [CrossRef]
32. Deb, K.; Pratap, A.; Agarwal, S.; Meyarivan, T. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evol. Comput.* **2002**, *6*, 182–197. [CrossRef]
33. Zhang, Q.; Li, H. MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition. *IEEE Trans. Evol. Comput.* **2008**, *11*, 712–731. [CrossRef]
34. Saltelli, A.; Ratto, M.; Andres, T.; Campologo, F.; Cariboni, J.; Gatelli, D.; Saisana, M. *Global Sensitivity Analysis: The Primer*; John Wiley & Sons: Chichester, UK, 2008; p. 292.
35. Sobol, I.M. Sensitivity Estimates for Nonlinear Mathematical Models. *Math. Model. Comput. Exp.* **1993**, *1*, 112–118.
36. Wang, X.; Gupta, A. Generative Image Modeling Using Style and Structure Adversarial Networks. In Proceedings of the 14th European Conference on Computer Vision (ECCV), Amsterdam, The Netherlands, 11–14 October 2016. [CrossRef]
37. Chen, Y.; Li, W.; Sakaridis, C.; Dai, D.; Van Gool, L. Domain Adaptive Faster R-CNN for Object Detection in the Wild. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018; pp. 3339–3348.
38. Liu, H.; Simonyan, K.; Yang, Y. DARTS: Differentiable Architecture Search. *arXiv* **2018**, arXiv:1806.09055.

39. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask R-CNN. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *42*, 386–397. [CrossRef] [PubMed]
40. Pearl, J. *Causality: Models, Reasoning, and Inference*, 2nd ed.; Cambridge University Press: Cambridge, UK, 2009; p. 464.