GSD-YOLO: a Lightweight Framework for Traffic Sign Detection

Jianbang Ding, Tianwei Shi*, Zhehao Zhang, Ruiqi Wang.

Abstract—**To address the challenges of detecting small traffic signs, low detection accuracy, complex detection network models, and large parameter counts, this paper proposes a lightweight traffic sign detection framework called GSD-YOLO. This framework integrates the GCSconv module, SENetV2 module, and DySample module. Specifically, the Slim-net architecture, formed by combining the GCSconv and VOVGSCP modules, is employed to replace the Conv and C2f modules in the original YOLOv8 neck network. This substitution not only reduces the number of parameters and computational load during training but also achieves a lightweight design. The lightweight SENetV2 module is incorporated into the backbone of the detection model, further lightening the backbone. In the neck network, the DySample module replaces the original upsampling module, enhancing the model's upsampling capability while reducing the parameter count. Compared to the original YOLOv8 detection network, Grad-CAM visualization shows that traffic signs in the detected regions appear darker and more concentrated. On the CCTSDB dataset, the proposed framework achieves an mAP@0.5 of 95.5%, which is 2.3% higher than the original YOLOv8 (93.3%). The model size is reduced to 4.2MB, which is 32.2% smaller than the original YOLOv8. The GFLOPs are reduced by 35%, and the overall parameter count is decreased by 40%. The lightweight detection framework proposed in this paper effectively reduces model size, simplifies model complexity, decreases parameter count, and improves detection accuracy, achieving significant improvements in the field of traffic sign detection.**

Index Terms— **GSConv, Lightweight Framework, Traffic sign Detection, YOLOV8**

I. Introduction

As a crucial element of unmanned driving technology, traffic sign detection plays an essential role in ensuring driving safety and the efficiency of intelligent transportation systems. It enables driverless vehicles to accurately understand road regulations and make informed decisions. The accuracy of traffic sign detection is of paramount importance in unmanned driving, as it indirectly affects vehicle safety and can significantly reduce the likelihood of traffic accidents. Enhancing the detection accuracy of traffic signs is therefore vital for the success of unmanned driving. However, deploying detection models on embedded devices presents unique challenges. Due to the limitations of embedded devices, models cannot be excessively large or complex.

To tackle the challenges posed by the limitations of embedded devices, this paper introduces a lightweight detection framework named GSD-YOLO. This framework integrates three key technologies: GSConv, SENetV2, and DySample. Specifically, the GSConv + Slim-neck architecture is employed to replace the conventional Conv and C2f modules in the original YOLOv8, thereby significantly reducing computational complexity and achieving a lightweight operation. Additionally, the lightweight SENetV2 model is incorporated into the backbone to further minimize its computational load. Lastly, the dynamic up-sampling module DySample is utilized to replace the original up-sampling module, thereby enhancing the model's up-sampling capabilities.

II. Related Work

In recent years, deep learning models have come to dominate the field of object detection, with convolutional neural networks (CNNs) playing a particularly prominent role and achieving numerous significant milestones [1][2]. These models can be broadly categorized into two types. The first type is the two-stage detection algorithm based on region proposals, with notable examples including Region-based Convolutional Neural Network (R-CNN) [3], Fast R-CNN [4], and Faster R-CNN [5]. While these two-stage algorithms excel in accuracy, they face limitations in speed. Their complex and bulky network architectures, coupled with a large number of parameters, lead to lengthy recognition processes. Thus, their computational latency exceeds the permissible thresholds for time-sensitive traffic sign identification applications.

The second classification encompasses efficient single-shot detectors, particularly YOLO [6] and SSD [7] frameworks. These models achieve real-time performance by eliminating region proposal stages, instead predicting object classes and positions directly from convolutional features. Comparative studies, including the work by Zuo's team [8], have evaluated these against Faster R-CNN in traffic sign

Manuscript received December 23, 2024;revised April 13, 2025

This work was supported by the Department of Science and Technology of Liaoning-Province (2022JH1/10400005).

This work was supported by Open Project of Liaoning Province Key Laboratory of Intelligent Construction IoT Application Technology (2024KFKT-08).

Jianbang Ding is a postgraduate student of School of Computer Science and Software Engineering, University of Science and Technology Liaoning, Anshan, 114051, China (email: jianbang0219@163.com).

Tianwei Shi is an associate professor at the School of Computer Science and Software Engineering, University of Science and Technology Liaoning, Anshan, China (corresponding author to provide phone: +86-139-9805-3962; e-mail: tianweiabbcc@163.com).

Zhehao Zhang is a postgraduate student of School of Computer Science and Software Engineering, University of Science and Technology Liaoning, Anshan, 114051, China (email:421753234@qq.com).

Ruiqi Whang is a postgraduate student of School of Computer Science and Software Engineering, University of Science and Technology Liaoning, Anshan, 114051, China (email:1046913352@qq.com).

detection scenarios. Building on the SSD algorithm, You et al. [9] developed a network structure specifically designed to reduce computational complexity and successfully validated it in traffic sign detection tasks. Yang [10] and Yuan [11] integrated attention mechanisms into convolutional neural networks to efficiently identify regions of interest in input images, thereby optimizing the feature extraction process for traffic signs in complex backgrounds. Zhang [12] utilized image enhancement techniques and incorporated the Spatial Pyramid Pooling (SPP) module into YOLOv3 to fully leverage fine-grained features and achieve precise target localization. Additionally, Kai Han and Yunhe Wang from Huawei's Noah's Ark Laboratory [13] proposed GhostNet, a lightweight feature extraction backbone network that even outperforms MobileNet in classification tasks [14].

In summary, deep learning methods have established the cornerstone of object detection challenges, with traffic sign detection being no exception. In terms of inference speed, two-stage networks are generally inadequate for real-time detection caused by their complex architectures. Consequently, single-stage networks have emerged as the primary focus for research and development. Specifically, in the domain of traffic sign detection, the straightforward and efficient YOLO-TINY series, as well as lightweight enhanced networks based on YOLO, are increasingly becoming the research focal points. However, despite significant progress, lightweight networks remain relatively underexplored. For instance, Chakkritt et al. [15] investigated neural network architecture search (NAS) using multi-objective evolutionary algorithms, emphasizing the enhancement of processing speed, reduction of storage space, and maintenance of high accuracy to develop efficient and robust convolutional neural network architectures. Zhang et al. [16] introduced the Ghost-YOLO lightweight model and proposed the C3Ghost module to exchange feature extraction module in YOLOv5, aiming to accelerate inference speed. Hu et al. [17] developed the Micro-YOLO algorithm based on YOLOv3-TINY, using compressed excitation blocks instead of inverted residual bottleneck convolution algorithms to significantly reduce parameters and computational load while preserving detection performance. Li et al. [18] proposed the edge-to-YOLO algorithm, employing lightweight ShuffleBlock and strip depth convolutional attention modules to replace the backbone network of YOLOv5M, achieving faster detection while maintaining accuracy. Ding et al. [19] engineered a lightweight network model using depthwise separable convolution in the head and neck sections to avoid excessive resource consumption.

Building on these efforts, Zhang et al. [20] optimized the original convolution operation in YOLOv5 through convolutional stacking and depthwise convolution within the Ghost module. Wu et al. [21] presented the DET-YOLO enhancement algorithm based on YOLOv4, leveraging the detached-oriented adaptive pyramid architecture for optimizing Multi-Scale feature blending. He et al. [22] combined the Convolutional Block Attention Model into YOLOv5, augmenting feature weights in challenging images and improving the network's feature expression capabilities, thereby refining detection accuracy. Chen [23] combined the high-performance MobileNetV3-Large with CBAM and Focal Loss, demonstrating superior overall performance compared to other convolutional neural networks. These advancements highlight the ongoing efforts to balance efficiency and accuracy in lightweight network architectures for object detection.

To deal with challenges highlighted earlier, this paper puts forward a lightweight traffic sign detection framework named GSD-YOLO, which integrates three key components: the GCSconv module, the SENetV2 module, and the DySample module. Specifically, the Slim-net network structure, formed by combining the GCSconv and VOVGSCP modules, is used to exchange the Conv and C2f modules in the original YOLOv8 neck network. This substitution reduces the number of parameters and computational load, and it also improves detection accuracy, thereby achieving a lightweight operation. Additionally, the lightweight SENetV2 model is incorporated into the backbone of the detection framework, further streamlining the network structure and making the backbone more efficient. In the neck network, the dynamic DySample module exchange the original upsampling module, improving the model's upsampling capability while simultaneously reducing the overall number of parameters in the detection model.

The main contributions of this paper are as follows:

1. Use the Slim-net network structure composed of GCSconv + VOVGSCP modules to replace the Conv and C2f modules in the original YOLOV8 neck network.

2. Using SENetV2 module, the lightweight model SENetV2 is integrated into the detection model backbone to simplify the network structure and make the backbone more lightweight.

3. The dynamic module DySample is used in the neck network to replace the original up-sampling module, which improves the up-sampling ability of the model.

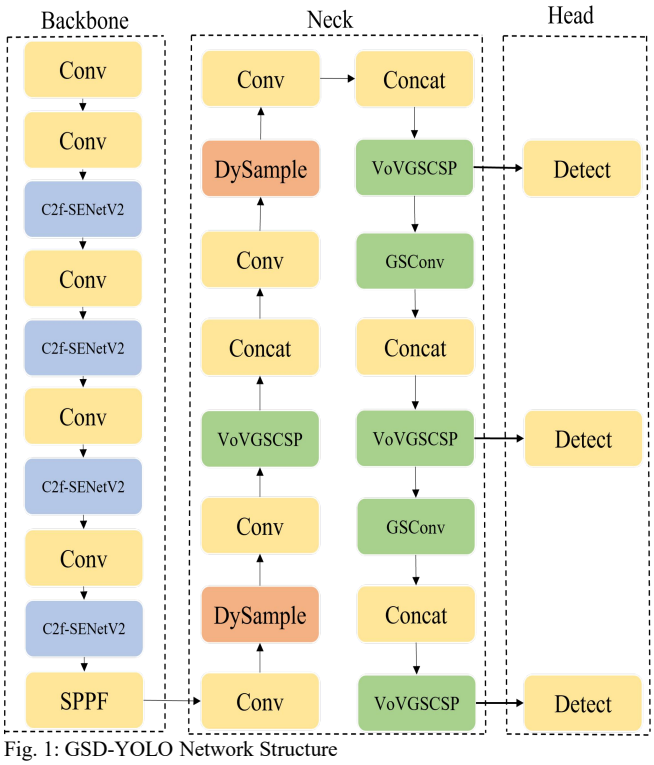


Fig. 1: GSD-YOLO Network Structure

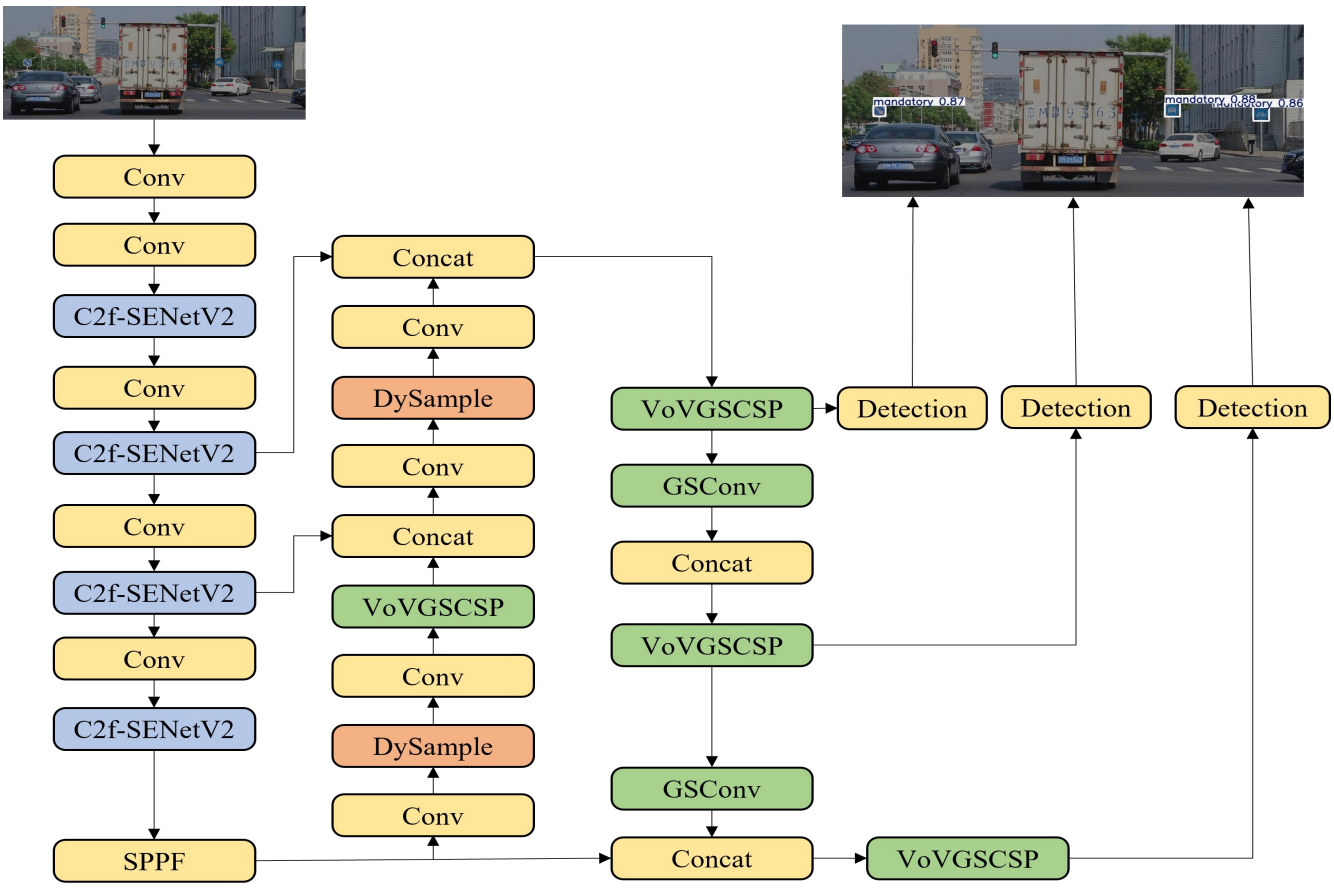


Fig. 2: Architecture of GSD-YOLO

III. RESEARCH METHODOLOGY

In this paper, we propose a novel detection framework named GSD-YOLO. Particularly, the C2f_SENetV2 module is incorporated within the backone network to boost performance, DySample upsampling operator is incorporated into the neck to optimize feature fusion, and an attention mechanism is introduced into the detection head via the Slim-neck network structure. These innovations collectively simplify the overall network architecture, reduce computational complexity, and minimize model size while preserving the accuracy of the detection framework.

A. GSConv

In this thesis, the Slim-neck network structure is employed in the neck section, significantly enhancing performance in the realm of unmanned driving. The core concept is to streamline the neck while maintaining a robust and reliable backbone, thereby reducing maintaining detection accuracy while reducing the model size. The Slim-neck network structure is characterized by its ability to mitigate the computational complexity of the neck and optimize its architecture, effectively reducing model size and maintaining detection precision. The Slim-neck architecture is composed of GSConv and VoVGSCSP modules.

The GSConv structure, as illustrated in the figure, integrates lightweight techniques from GhostNet and ShuffleNetv. Specifically, the GSConv module [24] represents a unique convolution operation designed to approximate the output of depthwise separable convolution to that of standard convolution. The GSConv module combines depthwise separable convolution (DConv) [25], standard convolution (SConv) [26], and Shuffle modules. By integrating DConv, SConv, and Shuffle mixed convolution, GSConv performs intensive convolution computations while maximally preserving inter-channel connections. This approach significantly reduces computational cost compared to SConv while achieving the same output effect. Incorporating the GSConv module into the model effectively reduces computational load and cost while maintaining model performance.

The VoVGSCSP module is a cross-stage partial-architecture network structure on the basis of GSConv and Conv. Its introduction aims to enhance the model's nonlinear capabilities, improve parameter transfer efficiency, and thereby strengthen the model's generalization ability.

$$\mathrm{YOLO}V8(input) = GSConv\left(SC\left(C2/2\times 2\right)\right) \quad (1)$$

In this article, the input feature map (denoted as "input") denotes the data, while "(C2/2 × 2)" signifies the result after two rounds of depthwise convolution. To achieve efficient and intensive convolution computations, GSConv—a specialized convolution approach—integrates standard convolution (SC), depthwise separable convolution (DSC), and shuffle hybrid convolution. This method maximizes the preservation of hidden inter-channel connections, significantly reducing computational costs while achieving equivalent output performance to standard convolution (SC).

Additionally, a cross-stage partial-architecture network module named VoV-GSCSP is introduced, on the basis of GSConv. The integration of convolution techniques such as GSConv and VoV-GSCSP enhances the model's nonlinear capacity promotes parameter sharing, Consequently improving the model's transferability performance. This

algorithm efficiently diminishes computational complexity and streamlines the network structure while maintaining sufficient accuracy, thus minimizing the risk of model overfitting.

In this thesis, the Slim-neck network structure is employed in the neck section to replace the original C2f module of YOLOv8 with GSConv and VoVGSCSP modules. This modification allows the detection model to maintain its accuracy while simplifying the network architecture, reducing computational load and cost, decreasing model size, and rendering the detection model more lightweight.

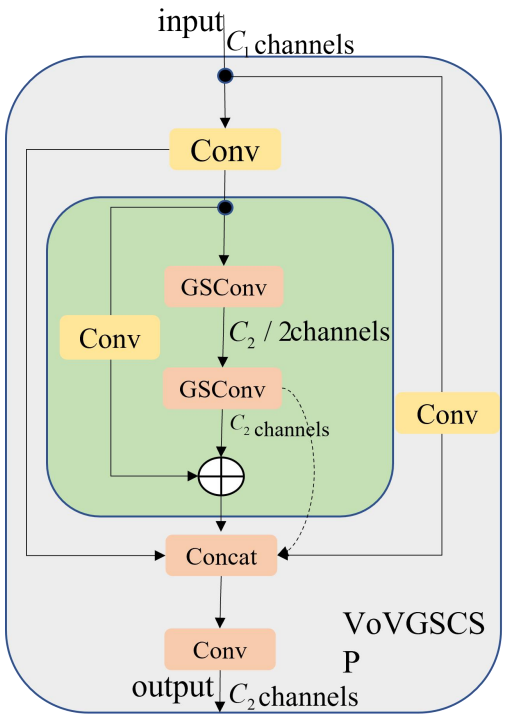


Fig. 3: VOVGSCSP module structure diagram

B. SENetV2

Traffic signs are typically small and diverse, and their recognition becomes particularly challenging in low-light environments, which can considerably impact the comprehensive performance of model detection. The introduction of a focus-enhancing module can boost the model's feature extraction proficiency, allowing it to concentrate on salient feature details while ignoring complex background noise [27]. The SENetV2 network structure integrates a squeezing-excitation module (SaE) with dense layers, thus enhancing the model's capacity for capture channel patterns and global context. In contrast to alternative attention mechanisms, such as the traditional squeezing-excitation (SaE) [28] and Efficient Channel Attention (ECA) [29], SENetV2 demonstrates greater efficiency and achieves superior feature representation for traffic signs.

SENet, put forward by Jie Hu et al. [30], innovates through the introduction of a "feature recalibration" mechanism. This mechanism bolsters the network's ability to capture key features by evaluating the significance of individual channels. SaE (Squeeze and Excitation), a lightweight attention technique, can be flexibly incorporated within any level of convolutional neural networks to boost performance. The fundamental concept underlying SENet is to use SaE modules to dynamically adjust the weights of every individual channel, thereby amplifying retain valuable traits while curbing extraneous attributes.

The SENetV2 attention module combines concepts from the ResNeXt network and the SaE module. ResNeXt integrates the multi-branch design of the initial module with subsequent modules to form a unified architecture. In the SaE module, the feature map is first reduced in dimensionality via global average pooling after standard convolution. It then calculates the weight of each channel using a pair of fully connected layers succeeded by a Sigmoid activation. These weights acquired through learning are subsequently multiplied element-wise with the feature map fed into produce a weighted feature map.

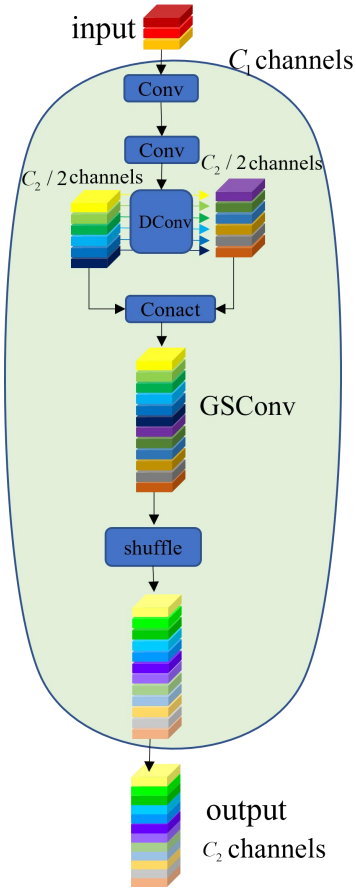


Fig. 4: GSConv module structure diagram

SENetV2 refines this process by employing a more sophisticated strategy during the squeezing phase to capture richer global information. It then excites the global features, analyzing the correlations between channels to derive their respective weights. Compared to the original SENet, SENetV2 introduces a multi-branch fully connected layer design in the excitation phase. This enhancement strengthens the network's capacity for expressing global features and further improves its feature recalibration capabilities.

SENetV2 is adept at efficiently identifying and extracting image regions that are rich in information pertinent to the target while automatically filtering out background features. Consequently, SENetV2 enables the model to focus on meaningful feature information, effectively eliminating the interference of irrelevant environmental details. This

enhances detection precision and dependability spanning a broad spectrum of complex scenarios.

Thus, incorporating the C2f_SENetV2 module into the backbone, as proposed in this paper, significantly boosts the model's capability to extract traffic sign features in complex environments, resulting in an enhancement of the overall detection performance of traffic signs.

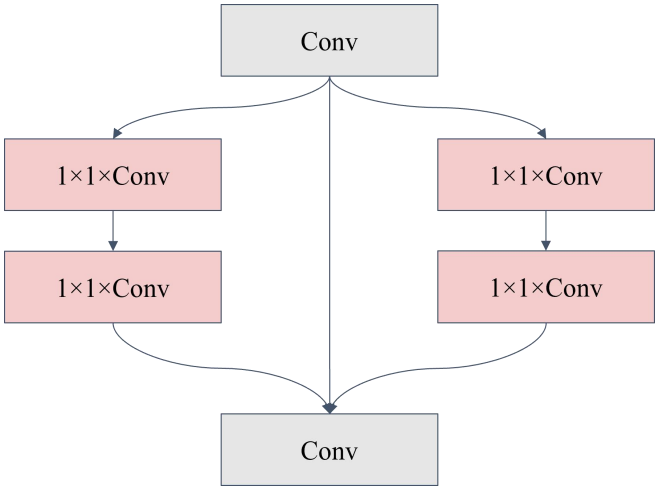


Fig. 5: Structure of the SENetV2 module

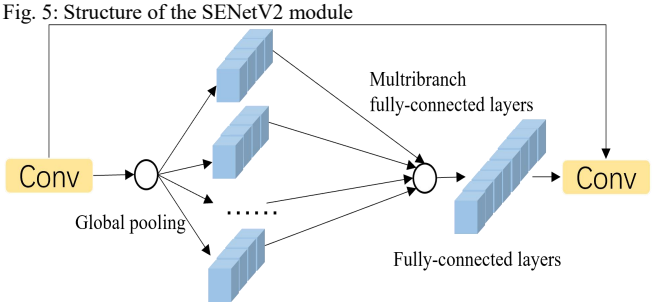


Fig. 6: Structure diagram of SaE module

C. *DySample*

To enhance the efficiency and quality of image processing, especially in scenarios like high-resolution reconstruction and traffic sign detection, conventional upsampling techniques such as nearest-neighbor interpolation interpolation and linear interpolation are often limited by detail loss and blurring. In order to tackle these limitations, this thesis presents the DySample module [31] as an alternative. DySample is an extremely lightweight and highly efficient dynamic upsampler that can flexibly downsample input feature maps by dynamically generating offsets. Its adaptive computational adjustments make it highly efficient and versatile when handling various input feature maps.

Compared to models that use transposed convolution for upsampling, DySample demonstrates superior performance and resource utilization. It employs a differential sampling strategy, which selectively samples only the most significant changes in the data distribution. This approach significantly reduces computational and storage demands. The key advantage of DySample lies in its unique differential sampling (DS) strategy. By precisely targeting the most differentiated portions of the data, this strategy minimizes unnecessary data processing, Consequently, it elevates both the precision and effectiveness of upsampling.

The grid_sample function resamples the input feature map X using the coordinates specified by the sample set δ. This process involves employing bilinear interpolation to generate a new feature map X′ of size C×H2 ×W2 .

$$X' = grid\_sample(X, \delta) \qquad (2)$$

Here, the feature map X has a size of C×H1 ×W1 , while the sample set δ has a size of 2×H2 ×W2 , where the first two dimensions correspond to the x and y coordinates, respectively.

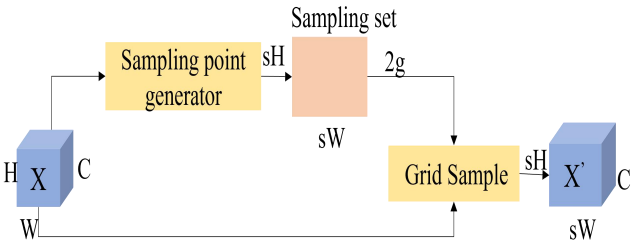


Fig. 7: Sampling based dynamic upsampling

Assume that the upsampling scale factor is represented by s, and the dimensions of the feature map X are C×H×W. An output offset O with dimensions 2s2×H×W is generated through a linear layer, where the input channel size is C and the output channel size is 2s2. Afterward, the offset O is reshaped to dimensions 2×sH×sW using a pixel rearrangement algorithm. Ultimately, the sampling set δ is acquired by superposing the offset O onto the original sampling grid G. The process is defined as follows:

$$O = linear(X) \qquad (3)$$

Ultimately, The upsampled feature map X′ , which has dimensions C × sH × sW, is produced by employing the sampling set δ in conjunction with the grid_sample function. In the YOLOv8 algorithm, the integration of DySample effectively leverages its strengths. Traditional upsampling operations, when applied to tasks such as traffic sign detection, often fail to maintain the model's lightweight nature due to significant computational demands and a substantial number of parameters. This can negatively impact the detection performance of traffic signs. In contrast, DySample, as a lightweight and efficient dynamic upsampler, not only addresses this issue but also significantly enhances image resolution. Particularly in handling small and easily distorted traffic sign images, DySample demonstrates a notable improvement in recognition capability.

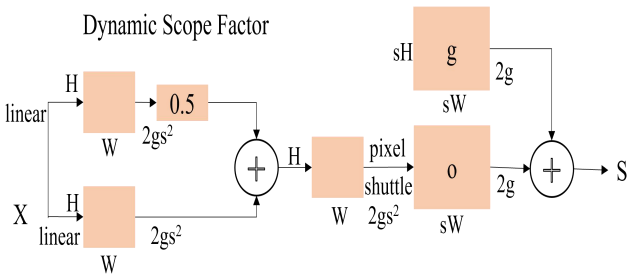


Fig. 8: DySample module structure diagram

IV. Experiment and Analysis

A. *Experimental Settings*

In this paper, the improved YOLOV8 detector serves as the core detection network, and the input image is uniformly adjusted to a size of 640×640. In the course of the training

TABLE I
COMPARING WITH OTHER METHODS ON CCTSDB

Approaches	Size	Parameter	GFlOPs	P	R	mAP
YOLOV3	120	112M	150	0.896	0.771	0.812
YOLOV5s	14.3	13.2M	14.2	0.883	0.752	0.801
YOLOX	69	8.96M	26.6	0.813	0.795	0.802
YOLOV7-tiny	74.5	6.2M	13.3	0.965	0.939	0.928
YOLOV8	6.2	3.0M	8.1	0.945	0.899	0.933
PVF-YOLO	80	10.36M	32.1	0.852	0.848	0.848
Ours	4.2	1.9M	6.3	0.959	0.913	0.955

TABLE II
ABLATION EXPERIMENTS ON CCTSDB

Approaches	GSConv	SENetV2	DySample	Size	Parameter	GFLOPs	mAP
YOLOV8				6.2	3006233	8.1	0.933
A	√			5.9	2802025	7.4	0.949
B	√	√		4.2	1950841	6.3	0.95
Ours	√	√	√	4.2	1816479	6.0	0.955

procedure, the initial learning rate is configured as 0.001, the training momentum is assigned a value of 0.98, and the weight decay parameter is set at 0.0001. The number of training epochs is established at 100, and the batch size is determined to be 8. The PyTorch framework was utilized in the experiment. The model training and test codes were run in a Windows environment with CUDA 11.2. The processor was an Intel® CPU Core i7-11100, the graphics card was an RTX 3060Ti, and the graphics memory capacity was 8 GB.

B. Introduce datasets

In this study, the CCTSDB 2021 dataset is utilized to thoroughly assess the improved YOLOV8 detection network. This dataset gathers over 1,000 car dashcam videos, yielding 17,856 frames in total. It enriches the dataset by extracting and storing key frames featuring traffic signs. Building upon CCTSDB 2017, CCTSDB 2021 adds 5,268 new traffic scene images, including 3,268 training images and 2,000 test images. Additionally, it contains 5,000 low-light traffic sign images of various types and shapes.

The dataset is further categorized based on three dimensions: category meaning (three types), sign size (five types), and weather conditions (six types). Per common road traffic sign definitions, signs are grouped into prohibited, mandatory, and warning signs. By size, they are classified as XS, S, M, L, and XL. Weather-wise, images are categorized as foggy, snowy, rainy, night-time, sunny, or cloudy.

C. Evaluation index

In order to validate the superiority of the proposed method in traffic sign detection, the CCTSDB 2021 dataset was contrasted with other models such as the original YOLOX, YOLOV7-tiny, YOLOV8, and PVF–YOLO. The comprehensive performance of the proposed method was appraised by means of indicators like model size, detection accuracy (P), recall rate (R), mean average precision (mAP), number of parameters, and FLOPS.

$$\mathrm{Re}call = \frac{TP}{TP = FN} \tag{4}$$

$$\mathrm{Pr}ecision = \frac{TP}{TP + FP} \tag{5}$$

$$mAP = \frac{1}{K}\sum_{1}^{k}(\mathrm{Pr}ecision) \tag{6}$$

$$\mathrm{FLOPs}=2\times \mathrm{H}\times W\left(C_i \times K^2 + 1\right)\times C_o \tag{7}$$

Model capacity is gauged by parameter count and FLOPs, reflecting computational demand and complexity. Model size, related to parameters, denotes learnable component magnitude. Evaluation metrics include TP (correct positive identification), FN (incorrect negative classification), and FP (wrong positive categorization). Precision, recall, and mAP are calculated from these, with mAP derived from average precision across classes. Notations like C_i, C_o, H, W, and K are also key.

D. Evaluation index

Four methods, namely YOLOV3, YOLOV5s, YOLOX, YOLOV7-tiny, YOLOV8, and PVF - YOLO, were selected to be compared with the method proposed in this paper. The CCTSDB dataset was utilized to conduct comparative experiments under the same training strategy and hyperparameters. As presented in the following table, the model size, number of parameters, FLOPs, accuracy rate, recall rate, and mAP0.5 of the proposed method all exhibit optimal values.

As a result, the detection capability of the model achieved by the proposed approach outperforms that of the alternative comparison methods. Despite preserving the detection accuracy, the proposed method achieves a reduction in model size, parameter count, and overall complexity, thereby rendering the model more streamlined and lightweight.

Fig. 10: Comparison of prediction effect between YOLOv8 and GSD-YOLO

E. Ablation experiment

To assess the efficacy of the enhanced method presented in this paper, comparative experiments are carried out using an identical training strategy and hyperparameters. The effectiveness and practicality of the improved method are confirmed through a comparison of the experimental outcomes.

As depicted in the figure below, when solely employing GSConv, there is a notable decrease in model size, parameter count, and FLOPs, albeit with a slight reduction in model detection performance. When both GSConv and SENetV2 are employed, there is a significant reduction in model size, parameter count, and FLOPs, resulting in a more streamlined model. When the three modules—GSConv, SENetV2, and Dysample—are implemented, in comparison to the original YOLOV8, there is a 32.2% reduction in model size, a 40% decrease in the number of parameters, a 35% drop in GFLOPs, and a 2.3% increase in average accuracy.

F. .Heat map results

In this paper, the Grad-CAM [32] heat map method is employed to represent the weight of traffic sign area detection classification via the color depth of the region generated by the heat map. In the original YOLOV8 model, the regions produced by the heat map are not concentrated on traffic signs, and the weight of detection classification is relatively low. In contrast to the original YOLOV8, the method proposed in this paper is more beneficial for model detection, as the color of the heat map generated area is darker and the weight of detection classification is greater in the traffic sign area.

G. Experimental results

As illustrated in the figure, the detection outcomes are visually represented on the CCTSDB dataset and contrasted with YOLOV8 to showcase the impact of the method introduced in this paper. According to the results shown in the picture below, The proposed method demonstrates superior detection accuracy compared to the original YOLOV8 model.

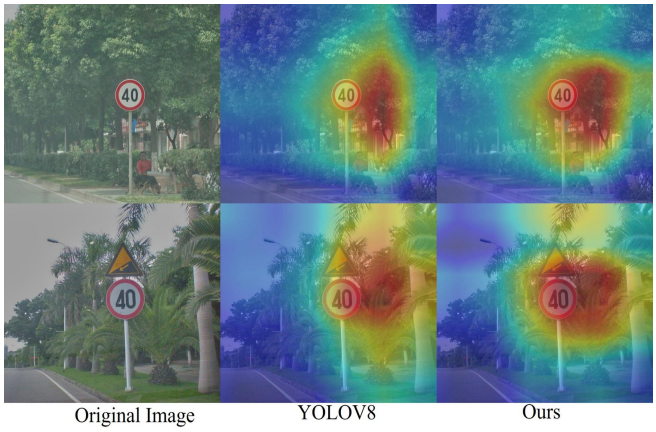
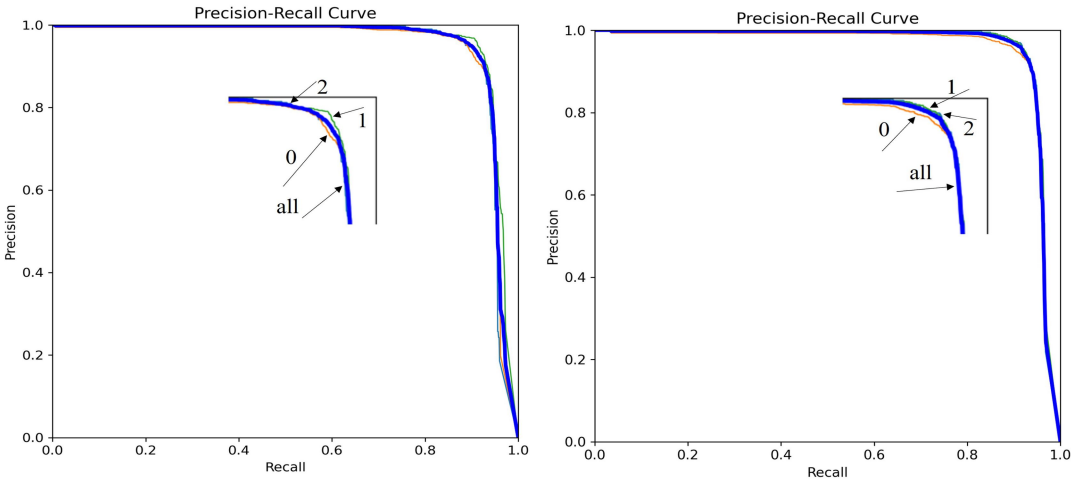


Fig. 9: Comparison results of heat map

H. P-R result

As depicted in the figure, based on the results of the training conducted on the CCTSDB dataset, the accuracy rate (P) and the recall rate (R) are plotted within the plane coordinate system, with the vertical axis representing the accuracy rate and the horizontal axis representing the recall rate.

As observed from the figure, when compared to the original YOLOV8 model, the Precision-Recall (PR) curve of the proposed approach is situated nearer to the upper right quadrant of the coordinate plane.This indicates that the model demonstrates better performance and higher accuracy when it comes to identifying positive sample.



YOLOV8 Ours

Fig. 11: PR curve

V. Conclusion

Addressing the challenges of detection difficulty, low accuracy in identifying small traffic signs, and the complexity of detection network models with excessive parameters, this study introduces a lightweight traffic sign detection framework known as GSD-YOLO. This framework integrates the GCSconv module, SENetV2 module, and Dysample module to achieve a balance between network lightweighting and detection efficacy.To achieve a more streamlined network while maintaining detection performance, the Slim-net architecture, comprising GCSconv + VOVGSCP modules, is employed to replace the Conv and C2f modules in the original YOLOV8 neck network. This substitution not only reduces the parameter count and computational overhead during training but also enhances detection accuracy, facilitating lightweight operation. As a result, the model size and computational complexity are significantly reduced.The lightweight SENetV2 module is incorporated into the backbone of the detection model to simplify the network architecture and promote a more lightweight backbone. This allows the model to adaptively modulate the significance of feature channels, thereby improving sensitivity and accuracy for small targets and minimizing detection errors and omissions.

Furthermore, the dynamic module DySample is implemented in the neck network to replace the original up-sampling module, enhancing the model's up-sampling capabilities and reducing the parameter count in the detection model.

In contrast to the original YOLOV8 detection network, the proposed framework in this study visualizes gradient information in the Grad-CAM method through a heat map. The traffic signs within the detected region appear darker and more concentrated, with a more focused distribution of gradient information in the image, which is advantageous for target detection.

When evaluated on the CCTSDB dataset, the proposed framework achieves an mAP@0.5 value of 95.5% (93.3), surpassing the original YOLOV8 model by 2.3%. The precision (P) is 0.945, and the recall (R) is 0.953, both of which outperform the original YOLOV8's 0.94 and 0.95, respectively. The model size is reduced to 4.2MB, representing a 32.2% decrease compared to the original YOLOV8. The number of parameters is 1,816,479, a 40% reduction from the original YOLOV8, and the GFLOPs are 6.0, a 35% decrease compared to the original YOLOV8. The lightweight detection framework presented in this study effectively reduces model size, simplifies model complexity, decreases parameter count, and enhances detection accuracy, yielding promising results in the realm of traffic sign detection.

References

[1] D. Cireşan, U. Meier, J. Masci, and J. Schmidhuber, ‘‘A committee of neural networks for traffic sign classification,’’ in Proc. Int. Joint Conf.Neural Netw., San Jose, CA, USA, Jul./Aug. 2011, pp. 1918–1921.1

[2] D. Ciresan, U. Meier, J. Masci, and J. Schmidhuber, “Multi-column deep neural network for traffic sign classification,’’ Neural Netw., vol. 32,pp. 333–338, Aug. 2012.

[3] R. Girshick, J. Donahue, T. Darrell, and J. Malik, ‘‘Rich feature hierar-chies for accurate object detection and semantic segmentation,’’ in Proc.IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Columbus, OH, USA,Jun. 2014, pp. 580–587.

[4] R. Girshick, ‘‘Fast R-CNN,’’ in Proc. IEEE Int. Conf. Comput. Vis. (ICCV),Dec. 2015, pp. 1440–1448.

[5] S. Ren, K. He, R. Girshick, and J. Sun, ‘‘Faster R-CNN: Towards real-time object detection with region proposal networks,’’ IEEE Trans. Pattern Anal. Mach. Intell., vol. 39, no. 6, pp. 1137–1149, Jun. 2017.

[6] Huaixu Gao, and Ying Tian, "Research on Road-Sign Detection Algorithms Based on Depth Network," *Engineering Letters*, vol. 31, no. 1, pp136-142, 2023.

[7] Xiaoming Zhang, Ying Tian. “Improved YOLOv5s Traffic Sign Detection”, *Engineering Letters*, vol. 31, no. 4, pp1883-1893, 2023 .

[8] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, ‘‘SSD: Single shot multibox detector,’’ in Proc. Eur. Conf.Comput. Vis. (ECCV), vol. 9905. Cham, Switzerland: Springer, Oct. 2016,pp. 21–37.

[9] Z. Zuo, K. Yu, Q. Zhou, X.Wang, and T. Li, ‘‘Traffic signs detection based on faster R-CNN,’’ in Proc. IEEE 37th Int. Conf. Distrib. Comput. Syst.Workshops (ICDCSW), Atlanta, GA, USA, Jun. 2017, pp. 286–288.

[10] S. You, Q. Bi, Y. Ji, S. Liu, Y. Feng, and F. Wu, ‘‘Traffic sign detection method based on improved SSD,’’ Information, vol. 11, no. 10, p. 475,Oct. 2020.

[11] T. Yang, X. Long, A. K. Sangaiah, Z. Zheng, and C. Tong, ‘‘Deep detection network for real-life traffic sign in vehicular networks,’’ Comput. Netw.,vol. 136, no. 8, pp. 95–104, May 2018.

[12] Y. Yuan, Z. Xiong, and Q. Wang, ‘‘VSSA-NET: Vertical spatial sequence attention network for traffic sign detection,’’ IEEE Trans.Image Process.,vol. 28, no. 7, pp. 3423–3434, Jul. 2019.

[13] H. Zhang, L. Qin, J. Li, Y. Guo, Y. Zhou, J. Zhang, and Z. Xu, “Real-time detection method for small traffic signs based on YOLOv3,’’ IEEE Access,vol. 8, pp. 64145–64156, Mar. 2020.

[14] Han K, Wang Y, Tian Q, Guo J, Xu C, Xu C (2020) Ghostnet: More features from cheap opera-tions. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp1580–1589.
[15] Howard A, Sandler M, Chu G, Chen LC, Chen B, Tan M, Wang W, Zhu Y, Pang R, Vasudevan V,V.Le Q, Adam H (2019) Searching for mobilenetv3. In: Proceedings of the IEEE/CVF internationalconference on computer vision, pp 1314–1324.
[16] Termritthikun C, Jamtsho Y, Muneesawang P, Zhao J, Lee I (2023) Evolutionary neural architec-ture search based on efficient CNN models population for image classification. Multimed Tools Appl 82(16):23917–23943.
[17] Zhang S, Che S, Liu Z, Zhang X (2023) A real-time and ightweight traffic sign detection method based on ghost-YOLO. Multimed Tools Appl 82:26063–26087.
[18] Hu L, Li Y (2021) Micro-YOLO: Exploring Efficient Methods to Compress CNN based Object Detection Model. In: ICAART (2), pp 151–158.
[19] Li J, Ye J. Edge-YOLO: Lightweight infrared object detection method deployed on edge devices. Appl Sci, 2023, 13(7):4402.
[20] Ding P, Qian H, Zhou Y, Chu S (2023) Object detection method based on lightweight YOLOv4 and attention mechanism in security scenes. J Real-Time Image Proc 20(2):34.
[21] Zhang J, Jin J, Ma Y, Ren P (2023) Lightweight object detection algorithm based on YOLOv5 for unmanned surface vehicles. Front Mar Sci 9:1058401.
[22] Wu Y, Li J. YOLOv4 with deformable-embedding-transformer feature extractor for exact object detection in aerial imagery. Sensors 2023, 23(5):2522.
[23] He L, Wei H (2023) CBAM-YOLOv5: a promising network model for wear particle recognition. Wirel Commun Mob Comput 2023:2520933.
[24] Chen L, Yao H, Fu J, Ng CT. The classification and localization of crack using lightweight convolutional neural network with CBAM. Eng Struct 2023, 275:115291.
[25] Li, H.; Li, J.; Wei, H.; Liu, Z.; Zhan, Z.; Ren, Q. Slim-neck by GSConv: A better design paradigm of detector architectures for autonomous vehicles. arXiv 2022, arXiv:2206.02424.
[26] Chollet, F. Xception: Deep learning with depthwise separable convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1251–1258.
[27] Chen J, Wang X, Guo Z, Zhang X, Sun J. Dynamic region-aware convolution. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 8064–8073.
[28] Yin, J.; Huang, P.; Xiao, D.; Zhang, B. A Lightweight Rice Pest Detection Algorithm Using Improved Attention Mechanism and YOLOv8. Agriculture 2024, 14, 1052. [CrossRef].
[29] Hu, J.; Shen, L.; Sun, G. Squeeze-and-Excitation Networks. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141.
[30] Wang, Q.; Wu, B.; Zhu, P.; Li, P.; Zuo, W.; Hu, Q. ECA-Net: Efficient Channel Attention for Deep Convolutional Neural Networks.In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 11531–11539.
[31] Hu, J.; Li, S.; Gang, S. Squeeze-and-excitation networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018.
[32] Liu, W.; Lu, H.; Fu, H.; Cao, Z. Learning to Upsample by Learning to Sample. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Paris, France, 2–6 October 2023; pp. 6027–6037.
[33] Selvaraju, R. R, Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localizatio, 2017 IEEE International Conference on Computer Vision (ICCV) 2017 pp. 618–626