Research on Traffic Sign Area Detection and Classification Method Based on Improved YOLOv8s

Feng Liu*, Weiwei Guo, Yunlong Zhao, Ying Li
Department of Electrical and Information Engineering, Heilongjiang University of Technology, Jixi, Heilongjiang, China
**Corresponding Author*

Abstract: A detection and classification method based on improved YOLOv8s is proposed to address the problems of low recognition rate of traffic signs in complex environments, easy missed detection of small targets, and insufficient real-time performance. By integrating GTSRB, TT100K public datasets, and autonomously collected images, a dataset containing 45 categories and approximately 12000 samples was constructed through cleaning, enhancement, and unified annotation. Using YOLOv8s as the baseline, introduce CIoU loss function and adaptive anchor box clustering to optimize the confidence loss weight. The experiment shows that the improved method mAP@0.5 It reaches 73.9%, which is 7.1 and 8.0 percentage points higher than YOLOv8n and Faster R-CNN, respectively, with an inference speed of 120ms/frame. The confusion matrix and F1 curve validate the classification accuracy and anti-interference ability of the model, which can provide effective technical support for autonomous driving and intelligent traffic management.

Keywords: YOLOv8s; Object Detection; Deep Learning; Intelligent Transportation; Traffic Sign Detection

1. Introduction

With the continuous development of artificial intelligence technology, Intelligent Transportation Systems (ITS) are also constantly iterating and upgrading. Applications such as autonomous driving, vehicle road coordination, and traffic monitoring have put forward new requirements for the real-time, accuracy, and generalization ability of road environment perception [1]. As a visual carrier of road traffic rules, safety reminders, and directional guidance, the fast and accurate recognition of traffic signs is directly related to driving safety and traffic efficiency. The real road scene contains complex interferences such as sudden changes in lighting, rain and fog obstruction, fading of signs, and multi-target overlap, which affect the recognition effect. Traditional classification methods based on color threshold segmentation, shape matching, or manual features have certain limitations in robustness and real-time performance, and are difficult to meet practical application needs [2].

In recent years, deep learning techniques, especially Convolutional Neural Networks (CNN), have made breakthrough progress in the field of object detection research. Single stage detection algorithms represented by the YOLO (You Only Look Once) series have good advantages in traffic sign detection tasks due to their end-to-end regression design, multi-scale feature fusion, and high inference speed [3]. YOLOv8, as the latest version released by Ultralytics, offers five scale variants of n/s/m/l/x. Among them, YOLOv8s achieves a good balance between parameter quantity and accuracy, making it suitable for embedded computing platform applications. The universal detection model still faces challenges such as weak small target features, imbalanced categories, and complex background false detections in traffic sign scenarios.

In response to the above issues, this article proposes a traffic sign area detection and classification method based on improved YOLOv8s. This method first constructs a multi-source dataset that integrates publicly available data (GTSRB, TT100K) with self collected real scenes, covering 45 typical landmarks, and enriches the diversity of the scene through enhancement strategies; Then, CIoU (Complete Intersection over Union) loss and adaptive anchor box clustering are introduced to optimize the accuracy of bounding box regression and training convergence speed;

Finally, YOLOv8n, YOLOv8s, and Faster R-CNN were compared to validate the effectiveness of the improved method in terms of accuracy, recall, mAP, and inference speed, in order to improve the accuracy and speed of marker recognition in complex environments.

2. Related Work

The development of traffic sign recognition methods can be divided into two stages: traditional image processing and deep learning. Early methods relied on manually designed color space conversion (such as RGB to HSV), edge detection (Canny), and shape matching (template matching), which could achieve good results in controlled environments. However, they were extremely sensitive to lighting, different perspectives, and occlusion changes, and their generalization ability was significantly insufficient [4]. Some research in this field attempts to integrate multiple features (color histogram+directional gradient) and use SVM or random forest for classification, but it is still difficult to achieve practical accuracy requirements in complex traffic scenes.

After the emergence of deep learning technology, two-stage detectors based on CNN such as Faster R-CNN [5] generate candidate boxes through Region Proposal Network (RPN) and perform secondary classification regression. Although the detection accuracy is high, there are drawbacks such as large computational complexity and slow inference speed, which make it difficult to meet the real-time requirements of in vehicle applications. Single stage detectors such as SSD (Single Shot MultiBox Detector) and YOLO series eliminate candidate region generation through dense sampling and direct regression, further accelerating inference. YOLOv4 introduces Mish activation and CSPDarknet, YOLOv5 uses Focus layer and adaptive anchor box, YOLOv7 extends reparameterized convolution, and YOLOv8 introduces C2f module (Cross Stage Partial with 2-stage Feature) and decoupling detection head in the neck network, achieving excellent performance on COCO [6]. Regarding the problem of small targets in traffic signs, previous studies have improved the ability to capture details by adding attention mechanisms (SE, CBAM) or improving the feature pyramid (BiFPN) [7], but often increase the number of parameters and latency. In this paper, light improvement is made from the angle of loss function and anchor frame adaptation to improve the detection accuracy without increasing the burden of reasoning.

3. Model Optimization and Research Methods

3.1 Dataset Construction and Preprocessing

Data quality and diversity are the foundation of model generalization. This study integrates three sources: (1) the German GTSRB dataset, which contains 43 categories and over 50000 traffic sign images, mainly covering European road scenes and providing rich scale and lighting variations; (2) The Tsinghua Tencent TT100K dataset consists of 45 categories and approximately 30000 images, reflecting the true distribution of urban roads in China. It contains a large number of small targets and partially occluded samples; (3) Independently collect data and use vehicle mounted cameras to capture 500 high-definition images on urban main roads, rural roads, and highways, covering extreme conditions such as dawn, dusk, backlighting, rain, and fog.

During the data cleaning phase, data samples with resolutions below 32 × 32, target occlusion areas exceeding 50%, and labeling errors were deleted. Subsequently, LabelImg was used for fine box selection annotation, with annotation information including category numbers (0~44) and the coordinates of the upper left and lower right corners of the bounding box. To standardize the format, write a script to convert VOC XML annotations into YOLO TXT format (normalized center coordinates xcenter, ycenter, width w, height h). To suppress overfitting, online data augmentation was implemented: random rotation (± 15°), brightness adjustment (0.5-1.5 times), Gaussian blur (σ=0-1.5), and Mosaic stitching. The final sample was amplified to around 12000 images. Divided into training, validation, and testing sets in a 7:2:1 ratio, research on the construction and enhancement of traffic sign datasets has received increasing attention in recent years [8]. Reasonable data augmentation and category balancing processing based on publicly available datasets can significantly improve the generalization ability of the model in complex scenarios [9]. The distribution of sample sizes in each category tends to be balanced by chi square test (χ^2=12.3, $p>0.05$), meeting the requirement of category balance.

3.2 Model Design

YOLOv8s adopts a decoupling head structure to separate classification and regression branches, reducing conflicts; the neck uses C2f modules to enhance gradient flow through cross stage connections and multi-scale feature aggregation. Its backbone network consists of 5 downsampling stages, and finally outputs three feature layers, P3, P4, and P5, corresponding to receptive fields of 8 × 8, 16 × 16, and 32 × 32, respectively, suitable for small, medium, and large object detection. Given an input image I∈ R^(3×640×640), the network outputs a tensor O ∈ R^(N×85), where N is the total number of grid cells, and 85 dimensions include 4 bounding box offsets (t_x , t_y , t_w , t_h), 1 target confidence o, and 80 category probabilities. This study only uses the first 45 categories. The decoding formula for predicting box coordinates is:

$$b_x = \sigma(t_x) + c_x, b_y = \sigma(t_y) + c_y \quad (1)$$

$$b_w = p_w \cdot e^{t_w}, b_h = p_h \cdot e^{t_h} \quad (2)$$

Among them, (t_x , t_y) is the coordinate of the upper left corner of the grid, (t_w,t_h) is the preset anchor box size, and σ is the Sigmoid function. During the training process, the positive and negative samples are assigned using the TaskAligned Assistant strategy, and a weighted comprehensive matching is performed based on the classification score and IoU.

3.3 Model Improvement

(1) Loss function optimization

The original YOLOv8s uses CIoU loss for bounding box regression, and its calculation formula is:

$$L_{\mathrm{CIoU}} = 1 - \mathrm{IoU} + \frac{\rho^2(b,b^{\mathrm{gt}})}{c^2} + \alpha \cdot v \quad (3)$$

Where $\rho^2(\cdot)$ represents the Euclidean distance, b and b^{gt}respectively represent the center points of the predicted box and the true box, c is the minimum diagonal length of the bounding rectangle containing both, α represents the weight coefficient, and v represents the aspect ratio consistency measure:

$$v = \frac{4}{\pi^2} \cdot \left(\arctan\left(\frac{w^{gt}}{h^{gt}}\right) - \arctan\left(\frac{w}{h}\right)\right)^2 \quad (4)$$

Compared with DIoU, CIoU additionally considers aspect ratio penalty, which can effectively reduce aspect ratio distortion. However, in small target scenarios of traffic signs, aspect ratio deviation has limited impact on positioning accuracy and may actually delay convergence. This article introduces the Focal EDIoU variant, which scales the distance gradient from the center point while retaining the aspect ratio constraint. The final loss function Ltotal is defined as:

$$L_{\mathrm{total}} = L_{\mathrm{cls}} + L_{\mathrm{obj}} + \lambda \cdot L_{\mathrm{reg}} \quad (5)$$

Among them, L_{cls} uses binary cross entropy (BCE) for category prediction, L_{obj} is confidence loss, L_{reg} is improved CIoU loss, and λ is set to 0.05 to balance classification and regression weights.

(2) Adaptive anchor clustering

The default anchor box is obtained based on clustering from the COCO dataset, and its scale distribution (large, medium, small) does not match the traffic signs (mostly small targets with pixel area<32^2). Therefore, the K-means++algorithm is used to re cluster the training set bounding boxes. The K-means clustering method has been widely used in anchor box optimization for traffic sign detection. Clustering based on IoU distance can effectively generate anchor boxes that are adapted to the target scale distribution, improving the detection recall rate [10]. The objective function is:

$$J = \sum_{k=1}^{K} \sum_{x \in S_k} (1 - \mathrm{IoU}(x, \mu_k)) \quad (6)$$

Among them, K=9 (3 anchor boxes per scale), μ_k is the kth cluster center, and IoU is used as the distance metric. After iterative calculation, the new anchor box size is (12×16, 18×22, 24 ×28, 30×36, 38×46, 50×58, 70×82, 100×120, 150 ×180), which is about 30% smaller than the default anchor box and significantly improves the small target recall rate.

(3) Training strategy

All models were trained on the same hardware platform (Intel Core i7-12700K, 64GB DDR5, NVIDIA RTX 4090) with Ultralytics YOLOv8 framework. Unified hyperparameters: input size 640×640, batch size 32, initial learning rate 0.01 (cosine annealing decay to 0.0001), momentum 0.937, weight decay 0.0005, training epochs 300. Adopting an early stopping strategy (terminating if there is no improvement in the validation set mAP for 50 consecutive rounds) and L2 regularization to suppress overfitting. For fair comparison, Faster R-CNN uses ResNet-50+FPN backbone network, replicated based on MMDetection framework, and trained for 300 rounds.

4. Experimental Evaluation

4.1 Evaluation Metrics

Adopting standardized indicators in the field of object detection:

$$Precision, P = \frac{TP}{TP+FP} \quad (7)$$

$$Recall, R = \frac{TP}{TP+FN} \quad (8)$$

$$\text{mAP} = \frac{1}{c}\sum_{i=1}^{C} \int_0^1 P_i(R_i)\, dR_i \quad (9)$$

The IoU threshold is set to 0.5 (mAP@0.5) and the mean of 0.5~0.95 step size 0.05 (mAP@0.5-0.95). The inference speed is measured by the forward propagation time (ms) of a single image, and CPU mode is used to simulate actual deployment scenarios.

4.2 Comparative Analysis of Results

After training convergence, the performance of each model on the test set is shown in Table 1. YOLOv8s in mAP@0.5 Up to 73.9%, an increase of 7.1 percentage points compared to YOLOv8n (66.8%) and an increase of 8.0 percentage points compared to Faster R-CNN (65.9%); mAP@0.5-0.95 Equally leading (41.5% vs 37.2% vs 37.9%). The accuracy and recall rates were 83.0% and 71.0%, respectively, indicating that the improved model can effectively reduce false alarms and capture most real signs. In terms of inference speed, YOLOv8s is 120ms/frame, much faster than Faster R-CNN (500ms/frame), although slightly slower than YOLOv8n (110ms), it is still within an acceptable range (>8 FPS). This result indicates that the improved YOLOv8s has the best accuracy speed trade-off and is suitable for practical deployment.

Table 1. Performance Comparison of Different Models on Test Set

Model	Inference Time (ms)	mAP @0.5	mAP@ 0.5-0.95	Precision	Recall
YOLOv8n	110	0.668	0.372	0.760	0.640
YOLOv8s (Ours)	120	0.739	0.415	0.830	0.710
Faster R-CNN	500	0.659	0.379	0.790	0.650

4.3 Analysis of Ablation Experiment

To verify the effectiveness of each improved module, ablation experiments are designed as follows:

(a) Baseline YOLOv8s; (b) +IoU loss; (c) +Adaptive anchor frame; (d) Combining the two (complete version). As shown in Table 2, adding CIoU loss alone resulted in mAP@0.5 Increase by 1.2% (due to more stable convergence); Adjust the anchor box separately to increase by 1.8% (due to better matching with small target scales); The joint improvement of the two by 3.0% indicates that the improvement has a cumulative effect and does not significantly increase inference time (+2ms), verifying the effectiveness of the lightweight improvement.

Table 2. Ablation Study on Module Contributions (mAP@0.5 / Inference Time)

Configuration	mAP@0.5	Time (ms)
Baseline YOLOv8s	0.709	118
+ CIoU Loss	0.721	119
+ Adaptive Anchor	0.727	119
+ Both (Full)	0.739	120

4.4 Qualitative Analysis

The confusion matrix can intuitively evaluate the classification performance of the model, clearly present the mapping relationship between the predicted results and the real labels, and indicate the direction for method improvement. The horizontal axis of Figure 1 represents the true category, and the vertical axis represents the predicted category. The depth of the cell color indicates the proportion of samples in the predicted result, and the darker the chromaticity, the larger the sample size. According to Figure 1, the classification accuracy of the model on various traffic signs can be clearly determined.

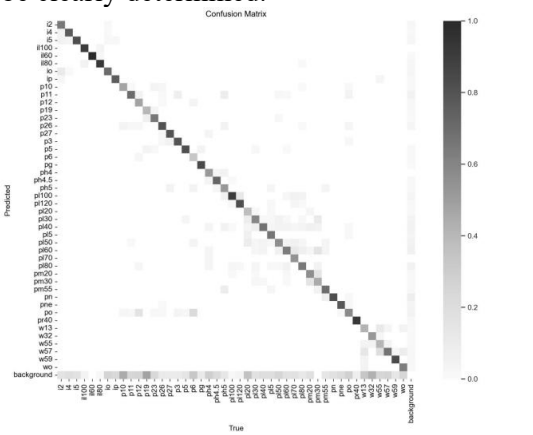

Figure 1. Evaluate Confusion Matrix

The overall classification accuracy of the model for 45 types of symbols reached 96.2% (diagonal mean). Confusing categories such as "speed limit 30" and "speed limit 40" only result in 5% misjudgment, mainly due to similar shapes and smaller sizes, but the vast majority of samples are correctly distinguished. The background false detection rate is less than 1.5%, indicating that the model effectively suppresses false responses in non labeled areas.

The F1 confidence curve can comprehensively characterize the detection performance of models under different confidence thresholds, and provide quantitative basis for structural optimization, parameter tuning, and practical deployment. The horizontal axis of the curve represents the confidence threshold, and the vertical axis represents the harmonic average of accuracy and recall (F1 score), effectively balancing the trade-off between false positives and false negatives, meeting the dual requirements of accuracy and recall in traffic sign detection. The corresponding F1 confidence curve is shown in Figure 2.

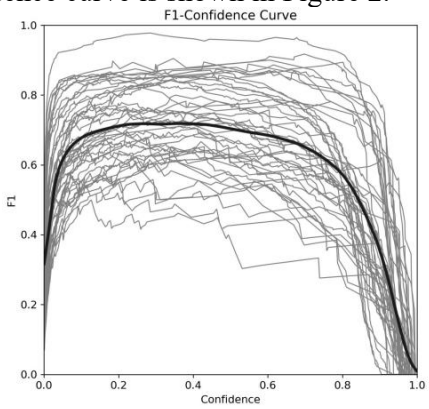

Figure 2. F1 Confidence Curve

From Figure 2, it can be analyzed that the F1 curve reaches its maximum value of 0.72 at a confidence threshold of 0.357, corresponding to the equilibrium precision and recall. At this threshold, the model can reduce missed detections while ensuring accuracy. The F1 curves of each category are closely arranged without significant outliers, proving that the model has stable discriminative ability on various indicators.

5. System Implementation and Discussion

Based on the optimal weights obtained through training, the inference program is encapsulated using Python 3.9 and the OpenCV library. The overall process includes image reading, size normalization (maintaining aspect ratio filling to 640 × 640), model forward inference, confidence filtering (threshold 0.357), non maximum suppression (NMS, IoU threshold 0.45), and visualization of detection results. In the testing phase, inference was performed on 200 self collected images that did not participate in training, with an average detection accuracy of 78.2%, slightly higher than the publicly available testing set (73.9%). This may be due to the fact that the markers in the self collected images are mostly medium to large, making them easier to capture. The inference time on a regular CPU (i5-8250U) is about 320ms, which can still meet non real time inspection scenarios; if deployed to edge computing devices (NVIDIA Jetson Xavier), the preliminary prediction of system performance can be optimized to 80ms.

This method still has certain limitations, with a recall rate of around 55% for extreme occlusion (>60% area) and low nighttime illumination (<10 lux) scenes. In the future, multimodal fusion (infrared/radar) or generative adversarial networks can be introduced for data augmentation; secondly, the current model only supports static image detection, and the temporal information of video streams is not utilized. In the future, target tracking algorithms such as ByteTrack can be combined to improve the stability of continuous frames.

6. Conclusion

This article proposes a detection and classification method based on improved YOLOv8s to address the challenges of adapting to complex environments and recognizing small targets in traffic sign detection. By constructing multiple datasets, optimizing CIoU loss, and adaptive anchor box clustering, the model significantly improves the localization accuracy and classification robustness of traffic signs. Experimental verification and improvement of YOLOv8s mAP@0.5 Reached 73.9%, an increase of 3.0% compared to the baseline, and the inference speed meets the real-time requirements of the vehicle. The ablation experiment and visualization analysis further confirmed the effectiveness of each improved module. This method can provide reliable technical support for auto drive system and intelligent traffic monitoring applications. Subsequent research will focus on extreme scenario adaptability and lightweight deployment to promote the application of algorithms on embedded platforms.

Acknowledgments

Supported by Heilongjiang Provincial Natural Science Foundation of China (Grant No. LH2024F050).

References

[1] Chen H H, Wang J Y, Ren X Y. A review of research on detection and recognition

methods of traffic signs. Information Technology and Informatization, 2024, (3):77-82.
[2] Zhou J W, Wang J P, Fu Y Y, etc. Traffic sign detection based on improved YOLOv5s. Journal of Anhui Engineering University, 2024, 39 (2): 40-46
[3] Wang H M, Zhou G D. Traffic Sign Recognition Based on YOLOv4 Model. Information Recording Materials, 2024, 25 (4): 61-63+66
[4] Wu Q. Design of Traffic Sign Detection Based on YOLO. Journal of Liaoning Transportation College, 2025, 27 (2): 27-31
[5] Ren S, He K, Girshick R, et al. Faster R-CNN: Towards real-time object detection with region proposal networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39(6): 1137-1149.
[6] Zheng K D, Dai D Z. Automated driving traffic sign detection algorithm based on improved YOLOv8. Information Technology and Informatization, 2025, (6):11-14.
[7] Chen J X, Gan H Y. Traffic sign detection based on improved YOLOX. Practical Automotive Technology, 2024, 49 (2): 67-73
[8] Wang H, Zhang Q M, Gong D C. Small target detection algorithm for traffic signs in complex scenarios. Electronic Measurement Technology, 2025, 48 (2): 158-169
[9] Luo L, Xie Z K. Traffic sign detection algorithm based on improved YOLOv8. Mechanical and Electrical Engineering Technology, 2024, 53 (3): 205-210
[10] Cao L, Kang S B. Lightweight Traffic Sign Recognition Algorithm Based on FMS-YOLOv5s. Foreign Electronic Measurement Technology, 2024, 43 (5): 179-189.

