Mathematical Modeling and Algorithm Application
ISSN: 3006-0842 | Vol. 8, No. 3, 2026

Research on a Lightweight and Efficient Vehicle Detection Approach Based on YOLO Algorithm

Chengfeng Su, Qiang Xiang*

School of Electronic Information, Southwest Minzu University, Chengdu 610225, China

* **Corresponding author**: Qiang Xiang

Abstract: In recent years, intelligent transportation systems have developed rapidly, and autonomous driving systems, as key solutions to traditional traffic problems, have received increasing attention. Vehicle and pedestrian detection serves as a fundamental component within these systems. However, in practical application scenarios, it is challenging to achieve a good balance between algorithm accuracy and resource consumption; furthermore, accuracy degrades under complex conditions such as occlusion and illumination variations. To address these challenges, this paper proposes an efficient and lightweight vehicle and pedestrian detection model named YOLO-MDT, based on the YOLOv8n architecture. Firstly, we introduce a Feature Dynamic Task-Sharing Detection Head (DT-Head). This head reduces the number of parameters through weight sharing and enhances detection performance by facilitating feature interaction between classification and localization tasks. Secondly, we design a C2f-MEE module to replace the original C2f module. This new module improves multi-scale feature extraction and edge awareness capabilities, thereby boosting object detection performance. Finally, DySample upsampling and the Focal-PIoUv2 loss function are employed to replace the original upsampling method and loss function, respectively. This improves model convergence speed, enhances bounding box accuracy, and reduces computational overhead. The proposed model was validated on the SODA10M and KITTI datasets. It achieves a parameter count that is only 70% of the original network and reduces computational complexity by 0.1 GFLOPs. Significant improvements in mean Average Precision (mAP) were observed: mAP50 increased by 4.3% and 3.1%, while mAP50-95 increased by 2.9% and 3.2% on the SODA10M and KITTI datasets, respectively.

Keywords: Vehicle and Pedestrian Detection, YOLOv8n, YOLO-MDT, DT-Head, C2f-MEE

1. Introduction

According to statistics from the National Bureau of Statistics of China, by the end of 2023, the total number of civilian vehicles in China reached 336.18 million, an increase of 17.14 million vehicles compared to the previous year [1]. With the continuous advancement of urbanization and the sharp increase in the number of vehicles, traffic management and safety issues have increasingly become the focus of public attention. Research on autonomous driving technology has received widespread attention. Vehicle and pedestrian target detection, as a core fundamental part, provides the basis for subsequent decision-making. Therefore, quickly and accurately locating and identifying vehicles ahead has become a research focus. Moreover, vehicle and pedestrian detection algorithms also play an important role in traffic management, intelligentization, and transportation systems [2]. However, in practical scenarios, interference from factors such as multiple scales, illumination changes, weather, occlusion, and complex backgrounds poses greater challenges for the accurate detection of vehicle and pedestrian targets [3-4]. This requires detection algorithms to possess strong robustness, maintaining stability and reliability in these changing environments. Secondly, model design must also balance detection performance and algorithmic complexity, enabling high-precision detection on some resource-limited edge devices [5-7].

Currently, vehicle and pedestrian detection algorithms based on deep learning are divided into single-stage detection algorithms and two-stage detection algorithms by process. Two-stage algorithms are represented by Faster R-CNN [8], while the other category includes YOLO, SSD [9], etc. In the target scenario of vehicle and pedestrian detection, although two-stage detection algorithms have good accuracy, they are slower and do not meet practical application scenarios. Therefore, most research focuses on improving single-stage detection algorithm models to achieve faster, more accurate, and lighter models. Sommer et al. [10] improved Faster R-CNN by using deconvolution for upsampling on deeper convolutional layers to effectively extract detailed features of small targets, then fused them with shallow feature layers to enhance the detection accuracy of small target vehicles. Zhang et al. [11] improved the YOLOv7 algorithm by using the Res3Uint structure to reconstruct the backbone feature extraction network, improving its nonlinear capability. They then used the ACmix attention mechanism to enhance the network's attention to vehicles, and finally used the Gaussian receptive field method of the RFLA module between the feature fusion network and the detection network, improving the detection accuracy of vehicles and pedestrians on urban roads. Wang [12] et al. improved the YOLOv8 algorithm by embedding BiFormer attention in the Neck layer, adding a small object detection layer, and using Wiou v3 to increase detection accuracy and improve small object detection capability. Wang [13] et al. improved the model through the Rep-ResNeXt reparameterization structure and designed a feature enhancement module FEM. By introducing an attention mechanism to improve the model's feature extraction effect in foggy conditions, the parameter count was better than the original model, but the accuracy did not improve. Luo [14] et al., in solving the problem of infrared vehicle and pedestrian detection, added a small object layer to YOLOv5 to improve small target detection performance, and combined FocalIoU and GIoU to enhance the detection

effect of occluded targets.

The above studies have improved some problems to a certain extent, but in-depth research reveals that they still face issues such as low accuracy, high missed detection rates, insufficient feature extraction capability for small targets, and difficulty in balancing resources and accuracy. To address these problems, in this paper, we optimize the YOLOv8n model. The optimization further reduces the number of parameters, improves the feature extraction capability for small targets through edge information enhancement, and reduces the impact of false detection and missed detection in various complex environments.

The main contributions of this study are as follows:

1.The YOLOv8 detection head has a large number of parameters, and the separate classification and localization branches lead to a lack of feature interaction. A DT detection head is designed, which reduces the number of parameters through weight sharing and designs interaction between classification and localization branches to enhance model performance.

2.The C2f module is improved by designing C2f-MEE, which strengthens edge information through multi-scales, highlights small target details, and enhances the network's ability to extract small target features.

3.DySample improves upon the nearest neighbor interpolation upsampling method, enhances feature information after upsampling, and reduces the blurring of small targets in complex scenes.

4.Focaler-IoU and Powerful-IoUv2 strengthen bounding box regression and focus on difficult samples.

2. YOLO-MDT Network

2.1. The pattern is rich YOLO-MDT Network Structure

The YOLO-MDT architecture is a lightweight and efficient vehicle and pedestrian detection model proposed based on the YOLOv8 core framework, as shown in Figure 2. This model improves upon YOLOv8n by incorporating some lightweight module structures, making it more suitable for traffic scenes, enhancing performance while maintaining a reduced parameter count. Firstly, to reduce the number of parameters, the DT detection head is proposed, which not only reduces parameters but also enhances the interaction between the two tasks of localization and classification, improving model performance. Secondly, replacing the original C2f with C2f-MEE reduces computational complexity while simultaneously improving multi-scale feature extraction capability and enhancing the model's ability to learn edge information, thereby improving the accuracy of small target detection. Furthermore, DySample is introduced. In dense scenes where target feature information highly overlaps, DySample's adaptive sampling position weight generation mechanism can retain more complete details of cross-level feature maps, ensuring the robustness of feature representation. Finally, Focaler-PIoUv2 is used to replace CIoU to solve the problem of excessive bounding box enlargement caused by unreasonable factors in the loss function's penalty term, and to focus on different samples to adjust the attention level to easy and hard samples, thus promoting more stable model convergence

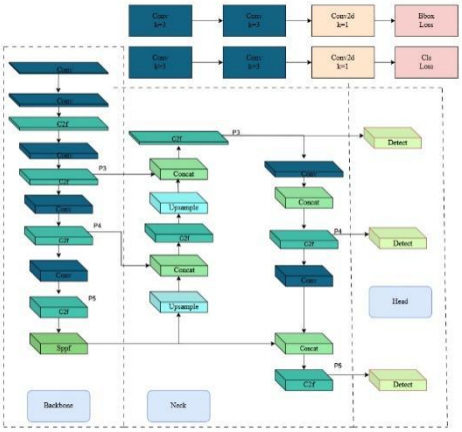

Fig. 1 YOLO-MDT network structure

2.2. C2f-MEE

The C2f-MEE module is designed to overcome the limitations of YOLOv8's C2f module, which often loses fine-grained details of small objects and lacks explicit edge information processing, by integrating multi-scale feature extraction, edge information enhancement, and efficient convolution. It first applies adaptive average pooling to generate four different scales (e.g., 3×3, 6×6, 9×9, 12×12) from the input feature map, capturing local details across varied receptive fields while reducing parameters. Each scaled feature undergoes convolution for feature extraction and upsampling to restore original resolution. An Edge Enhancer, inspired by HWD, then extracts high-frequency edge information by subtracting low-frequency components (obtained via average pooling) from the convolved features, reinforces these edges through convolution, and adds them back to the original features. Finally, the edge-enhanced multi-scale features are concatenated and passed through another convolution layer for further fusion and optimization, producing a feature map rich in multi-scale edge information that enhances small object detection and localization accuracy.

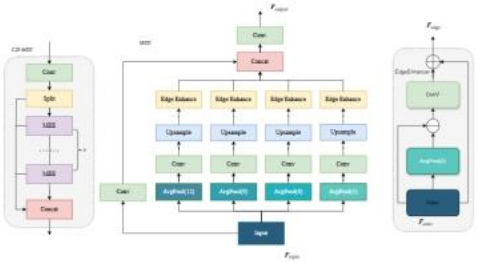

Fig. 2 C2f-MEE structure

2.3. DT Detection Head

The proposed DT detection head addresses the limitations of YOLOv8's decoupled head, which suffers from a lack of interaction between classification and regression tasks leading to prediction inconsistencies, and a large parameter count. To mitigate these issues, we first employ a feature extractor with shared-weight convolutions to reduce parameters. We replace Batch Normalization with Group Normalization for more stable feature distributions and introduce a dynamic alignment mechanism inspired by TOOD. This mechanism explicitly decomposes interactive features into task-specific features for classification and regression. For the regression branch, Deformable Convolution v2 (DCNv2) is utilized for spatial alignment to better adapt to varying object scales, while a learnable scaling

factor refines predictions across different scales. For the classification branch, dynamic feature selection is applied. By incorporating task-specific feature decomposition and dynamic interaction within a shared structure, this design not only reduces the overall parameter count compared to the original YOLOv8 head but also enhances the synergy between classification and localization, thereby improving both accuracy and robustness in complex scenes.

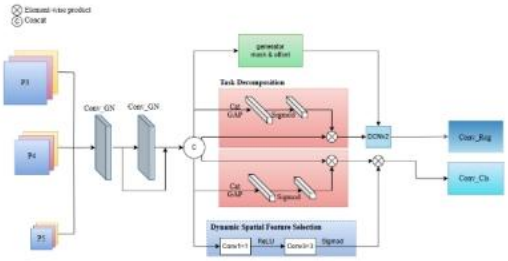

Fig. 3 DT Detection Head

2.4. Dysample

To address the limitations of nearest-neighbor interpolation in YOLOv8—such as discontinuous pixel changes and the loss of small object features in complex traffic scenes with dense targets—this study introduces the DySample dynamic upsampling strategy. DySample generates an offset tensor via linear transformation, constrains the offset range by multiplying with a static factor (0.25) to avoid sampling overlap artifacts, and then applies pixel shuffle to reshape offsets to the target resolution. These offsets are added to the initial sampling grid to produce adaptive sampling positions, enabling more accurate upsampling. In vehicle and pedestrian detection tasks, DySample preserves fine details of small objects by dynamically adjusting sampling points, resulting in sharper boundaries, improved feature consistency, and higher model stability under challenging conditions.

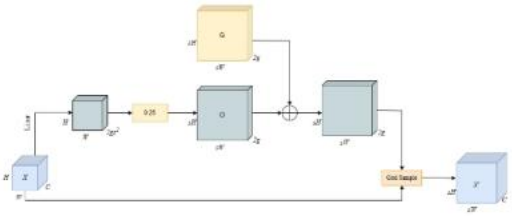

Fig. 4 Dysample structure

2.5. Focal-PIoUv2

To overcome the limitations of CIoU in YOLOv8—particularly its instability with drastic scale variations or occlusions, and its inability to balance hard and easy samples—this study introduces a novel loss function called Focal-PIoUv2. This loss is synthetically designed by integrating Focal-IoU and PIoUv2. Focal-IoU employs a piecewise function to remap the original IoU, enabling the loss to focus on different regression samples to address sample imbalance. Concurrently, PIoUv2 incorporates an anchor-quality evaluation mechanism that uses an adaptive penalty factor and a non-monotonic attention function to mitigate the bounding box expansion issue and emphasize medium-quality anchors. By combining these elements, Focal-PIoUv2 applies a linear interval mapping method to enhance the distinction between anchors of varying qualities. Its nonlinear formulation ensures smooth gradient transitions during optimization, preventing oscillations and promoting more stable convergence. Consequently, this loss function exhibits superior adaptability and performance across diverse sample qualities compared to PIoUv2.

3. Experimental Results and Analysis

3.1. Experimental Configuration

The experimental environment is Windows 11 64-bit operating system, using an Intel(R) Core(TM) i7-14700HX central processing unit and an NVIDIA GeForce RTX 4070 graphics processing unit. The experimental environment was built under Python version 3.9, deep learning framework PyTorch 1.13.2, and CUDA version 11.6. During the training process, no pre-trained models were used. The input image size was set to 640×640 pixels. The model was trained for 200 epochs with a batch size of 16. The initial learning rate was 0.01, and the learning rate momentum was 0.937. Stochastic Gradient Descent (SGD) was used for parameter optimization. The mosaic data augmentation algorithm was applied during most of the training process but was turned off during the last 10 epochs to avoid interfering with the images.

3.2. Ablation experiments

Ablation experiments were conducted on the SODA10M dataset. Table 1 shows the performance of the improved modules. It can be seen that after replacing the original detection head with Detect-DT, although the computational complexity increased by 0.5 GFLOPs, the parameter count decreased by 25%, and both mAP50 and mAP50-95 increased by 1.7%. This has a certain effect on model lightweighting. Replacing the original C2f module with the C2f-MEE module increased mAP50 by 0.5% and decreased GFLOPs by 0.6. Replacing the original upsampling operator with DySample increased mAP50 and mAP50-95 by 0.6% and 0.4%, respectively. Replacing CIoU with Focal-PIoUv2 increased mAP50 and mAP50-95 by 1.5% and 1%, respectively. The improvements from each module enhanced the overall performance of the model, and the fully improved model compared to the original baseline showed a 30% decrease in parameter count and a 4.3% increase in mAP50.

Table 1 Ablation experiments

Head	Dysample	Focal Power IoU2	C2f-MEE	mAP50	mAP 50-95	Parameters	GFLOPs
x	x	x	x	64.5	41.9	3.0M	8.2
√	x	x	x	66.2	43.6	2.24 M	8.7
x	√	x	x	651	42.3	3.0M	8.2
x	x	√	x	66.0	42.9	3.0M	8.2
x	x	x	√	65.0	41.9	2.85 M	7.6
√	√	x	x	66.7	43.6	2.15 M	8.1
√	√	√	x	67.2	43.9	2.15 M	8.1
√	√	√	√	68.8	44.8	2.1M	8.1

3.3. Comparison results of different models on SODA10M

To verify the performance of the improved detection algorithm, this model was compared with several current mainstream detection algorithms on the SODA10M dataset.

The table shows the experimental results of different models on the SODA10M dataset, mainly comparing some YOLO series models from recent years. Compared with the baseline model, the parameter count decreased by 30%, and mAP50 increased by 4.3%. It can be seen that compared with

the Transformer-based RT-DETR-L , its parameters are only 7% of it, but mAP50 is only 1.1% behind. Compared with the latest algorithm Hyper-YOLO-t , the parameters are 70% of it, but mAP50 is increased by 3.4%, and the computational complexity is also reduced. The number of parameters and high FLOPS of deep learning models directly affect the model's computational resource requirements and practical performance. Larger parameter counts and FLOPS tend to increase the model's computational overhead, especially in resource-constrained environments. Our proposed algorithm achieves higher accuracy with fewer parameters and lower computational complexity. This demonstrates that our method is suitable for devices with limited computing resources, achieving a balance between lightweight and accuracy.

Table 2 Comparison results of different models on SODA10M

Model	mAP50	mAp50-95	Parameters	GFLOPs
RT-DETR-L	69.9	46.7	29.3M	105.2
YOLOv5n	63.9	41.2	2.5M	7.1
YOLOv8n(base)	64.5	41.9	3.0M	8.2
YOLOV8s	69.6	46.5	11.1M	28.7
YOLOV8-P2	69.1	45.8	2.92M	12.2
YOLOv9s	69.3	46.6	7.28M	27.4
YOLOv10n	63.8	41.6	2.27M	6.5
YOLOv11n	63.6	41.3	2.58M	6.4
YOLOv12n	61.6	39.8	2.51M	5.8
Hyper-YOLO-t	65.4	42.7	3.01M	9.0
YOLO-MDT	68.8	44.8	2.1M	8.1

4. Conclusion

This paper proposes an improved vehicle and pedestrian detection algorithm based on the YOLOv8n model to address the challenges of false detections and missed detections caused by background interference in complex road scenes, as well as the issue that feature information of small objects is easily overlooked and disturbed in multi-scale scenarios. The core contributions of this paper lie in designing the C2f-MEE module to enhance the model's edge enhancement for target features and strengthen multi-scale feature extraction capabilities. Additionally, the DT detection head is designed, which not only reduces the parameter count but also facilitates feature interaction between classification and localization, thereby improving the performance of the detection head. This is complemented by Dysample and Focal-PIoUv2 to form a complete detection algorithm. Experimental results validated on the SODA 10M dataset and KITTI dataset show that the parameter count is 70% of the original network, and the computational complexity is reduced by 0.1 GFLOPs. The mAP50 and mAP50-90 metrics improved by 4.3% and 2.9%, respectively, on the SODA 10M dataset. Although the reduction in GFLOPs is modest, it achieves a favorable balance between accuracy and parameter count. Future research will focus on exploring the collaborative application of distillation and pruning techniques to further reduce computational complexity through the aforementioned methods.

References

[1] Dai T. Research on Traffic Target Detection Algorithm Based on Improved YOLOv7 [D]. Xihua University, 2024. (in Chinese)

[2] Wang J M, Pi J Y, Huang K, et al. Multi-scale Pedestrian and Vehicle Detection Algorithm for Complex Scenes [J]. Modern Electronics Technique, 2025, 48(09): 143-153. DOI: 10.16652/j.issn.1004-373x.2025.09.022. (in Chinese)

[3] Li J, Zou J, Chen C, et al. Vehicle and Pedestrian Detection Algorithm Based on Attention Scale Sequence Fusion [J]. Journal of Chongqing Jiaotong University (Natural Science Edition), 2025, 44(07): 75-82. (in Chinese)

[4] Li H R. Vehicle Target Detection in Haze Weather Based on Deep Learning [D]. Chang'an University, 2024. DOI: 10.26976/d.cnki.gchau.2024.000481. (in Chinese)

[5] Tian D, Wei X, Yuan J. Lightweight Vehicle Target Detection Algorithm Based on Improved YOLOv5 [J]. Computer Applications and Software, 2024, 41(12): 240-246. (in Chinese)

[6] X. Liu, Y. Wang, D. Yu and Z. Yuan, "YOLOv8-FDD: A Real-Time Vehicle Detection Method Based on Improved YOLOv8," in IEEE Access, vol. 12, pp. 136280-136296, 2024, doi: 10.1109/ACCESS.2024.3453298.

[7] Liao Y H, Wan X J, Zhao Z Z, et al. RO-YOLOv9 Vehicle and Pedestrian Detection Algorithm [J]. Computer Engineering and Applications, 2025, 61(11): 144-155. (in Chinese)

[8] Ren, Shaoqing et al. “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks.” IEEE Transactions on Pattern Analysis and Machine Intelligence 39 (2015): 1137-1149.

[9] Liu, W. et al. “SSD: Single Shot MultiBox Detector.” European Conference on Computer Vision (2015).

[10] Sommer L, Schumann A, Schuchert T, et al. Multifeature deconvolutional faster r-cnn for precise vehicle detection in aerial imagery[C]//2018 IEEE winter conference on applications of computer vision (WACV). IEEE, 2018: 635-642

[11] Zhang, Yuanhang et al. “YOLOv7-RAR for Urban Vehicle Detection.” Sensors (Basel, Switzerland) 23 (2023): n. pag.

[12] Wang B, Li Y-Y, Xu W, Wang H, Hu L. Vehicle–Pedestrian Detection Method Based on Improved YOLOv8. Electronics. 2024; 13(11):2149.

[13] H. Wang, Y. Xu, Y. He, Y. Cai, L. Chen, Y. Li, M. A. Sotelo, and Z. Li, “YOLOv5-Fog: A multiobjective visual detection algorithm for fog driving scenes based on improved YOLOv5,” IEEE Trans. Instrum. Meas., vol. 71, pp. 1–12, 2022.

[14] Xiao Luo, Hao Zhu, Zhenli Zhang, IR-YOLO: Real-Time Infrared Vehicle and Pedestrian Detection, Computers, Materials and Continua, Volume 78, Issue 2,2024, Pages 2667-2687, ISSN 1546-2218.