An object detection algorithm for complex urban scenarios based on YOLOv11

Wenshuai Yuan 1098803024@QQ.COM
School of information engineering Shanghai Maritime University Shanghai, China

Changming Zhu[*] CMZHU@SHMTU.EDU.CN
School of information engineering Shanghai Maritime University Shanghai, China
[*]*Corresponding author*

Editors: Nianyin Zeng, Ram Bilas Pachori and Dongshu Wang

Abstract

In recent years, with the rapid development of fields such as autonomous driving and intelligent transportation, the detection of pedestrians and vehicles in complex urban scenarios has become a hot topic in the field of object detection. However, these complex urban scenarios pose significant challenges to object detection. This paper proposes an improved algorithm based on YOLOv11, namely the YOLOv11 - APAS - MDC algorithm for object detection in complex urban scenarios on the Urban Environment Detection dataset. The aim is to enhance the accuracy and robustness of detecting pedestrians and vehicles under conditions such as occlusion and multi - scale targets in complex urban environments. This paper proposes a multi - scale edge information enhancement module called APAS based on the YOLOv11 basic model. This module highlights important edge feature information and can improve the model's perception ability of multi - scale features. Secondly, this paper presents the MDC module. By using convolutional layers with different dilation rates, this module can extract features at different scales. In addition, this paper introduces the RepGFPN feature network. This network re - parameterizes the structure and reduces redundant operations. Through a more complex cross - layer connection mechanism, it enhances the interaction of features at different levels, thereby improving the performance and efficiency of the detection model. The experimental verification results on the Urban Environment Detection dataset show that the improved algorithm in this paper outperforms the traditional YOLOv11 algorithm in terms of detection accuracy and robustness under conditions such as occlusion and multi - scale targets.

Keywords: YOLOv11; Urban Environment Detection Dataset; Pedestrian and vehicle detection; Dilation rate; Edge feature information

1. Introduction

With the rapid development of technologies such as driverless driving and intelligent transportation, the detection of pedestrians and vehicles in complex urban environments has become a current research hotspot. However, in complex urban scenes, there may be various factors such as pedestrians and vehicles being occluded, multi-scale targets, and dynamic lighting. These different factors pose huge challenges to target detection. Therefore, how to improve the accuracy and robustness of target detection in complex urban scenes has become the main research direction of this paper.

In recent years, deep learning technology has become a powerful method for automatically learning feature representations from data. Such technology has achieved significant improvements in the field of target detection (Hmidani and Ismaili Alaoui, 2022). The methods of target detection are mainly divided into two categories. The first category is two-stage target detection. The core idea is to first generate candidate regions, and then classify the candidate regions and perform bounding box regression. Typical algorithms include Cascade R-CNN (Cai and Vasconcelos, 2018), etc. The second category is single-stage target detection, which directly predicts the bounding box and category in the network without generating candidate regions. Its popular algorithms mainly include the YOLO (You Only Look Once) (Redmon et al., 2016) series . Compared with two-stage target detection algorithms, single-stage target detection algorithms have the advantages of high efficiency, low demand for computing resources, and a simplified design, making them dominant in scenarios with high real-time requirements.

For the detection of complex urban scenes, some traditional target detection algorithms may have deficiencies such as insufficient feature representation ability, poor processing ability for multi-scale and occluded targets, and poor robustness in complex scenes. This paper takes YOLOv11 as the basic framework. YOLOv11 is a newly released target detection algorithm by Ultralytics. Compared with previous versions, YOLOv11 has made innovations and improvements in multiple aspects. By improving the Backbone and Neck architectures and adding new components such as C3k2 and C2PSA, it enhances the feature extraction ability of images. This improvement makes YOLOv11 perform more excellently in some complex tasks (multi-targets, with occlusions). A more efficient architecture and training process have been redesigned, optimizing efficiency and speed while having higher accuracy and fewer parameters. In this project, improvements are made based on YOLOv11, and a YOLOv11-APAS-MDC algorithm for detecting pedestrians and vehicles in complex urban scenes is proposed, ultimately improving the detection accuracy and robustness for pedestrians and vehicles.

2. Relevant Methods

This paper proposes the YOLOv11-APAS-MDC algorithm, whose structure is shown in Figure 1. It is an optimized algorithm based on the basic model YOLOv11. The C3K2 module in the backbone network structure of the model is replaced with the APAS module. Different from traditional edge enhancement methods, APAS achieves the adaptive enhancement of edge information through hierarchical feature extraction and edge perception mechanism, addressing the problem of insufficient perception and representation capabilities of object edges in complex scenes. Then, the SPPF module in the backbone network structure is replaced with the MDC module. In contrast to traditional multi-scale feature extraction methods that usually rely on fixed dilation rates, MDC dynamically adapts to the context information of objects at different scales through hierarchical dilation rate design. Experiments show that its feature expression ability in complex scenes is significantly better than the benchmark methods. In addition, the RepGFPN feature network is introduced in the Neck part. Through the optimization of structural reparameterization and feature fusion strategies, it obtains a richer multi-scale feature expression with lower computational costs.

2.1. APAS Module

This module achieves the adaptive enhancement of edge information through the mechanism of low-frequency-high-frequency information separation and multi-scale feature fusion, as shown in Figure 2. Its core design can be decomposed into the following levels: low-frequency information extraction and high-frequency edge enhancement, multi-scale feature fusion mechanism, and lightweight design.

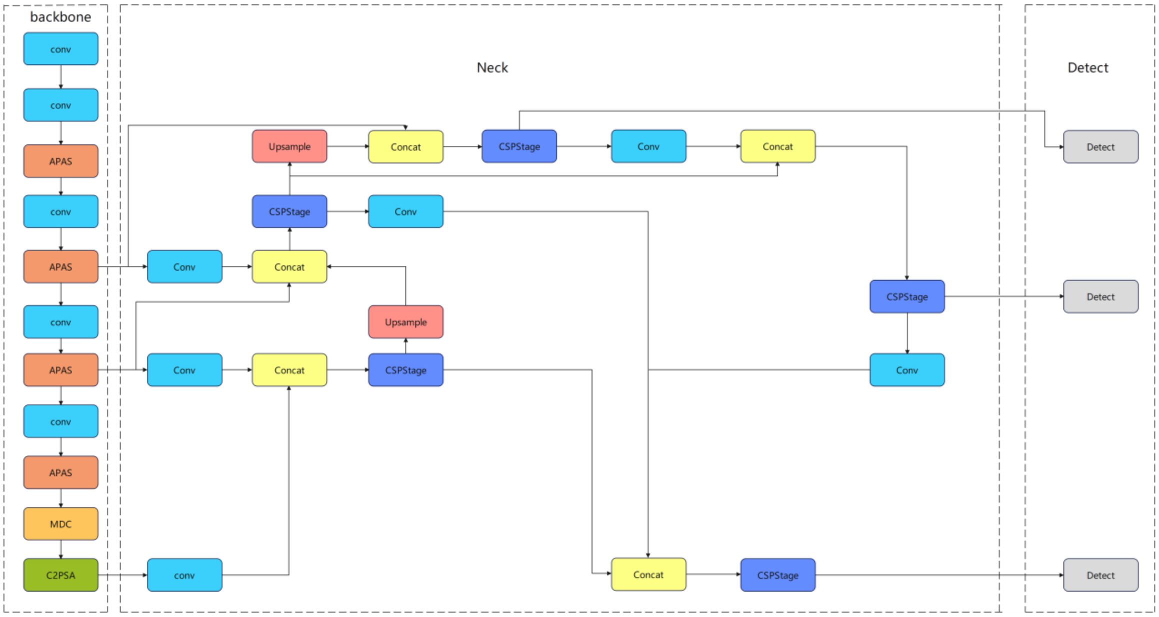

Figure 1: Structure Diagram of YOLOv11-APAS-MDC.

Low-frequency information extraction and high-frequency edge enhancement mean that in the PAS sub-module, first, a 3×3 average pooling (nn.AvgPool2d) is used to smooth the input feature map and extract low-frequency background information. Subsequently, high-frequency edge components are separated through residual calculation (x-edge). The high-frequency information generates a weight map through a convolutional layer (including the Sigmoid activation function). By performing pixel-wise weighting, it suppresses noise and enhances effective edges. Finally, the enhanced high-frequency features are added to the original input to achieve feature reconstruction, improving the edge contrast while retaining semantic information.

The APAS module generates multi-resolution feature maps (such as 3×3, 6×6, etc.) through adaptive average pooling (AdaptiveAvgPool2d), and combines grouped convolutions to extract scale-specific features. After the features of each branch are upsampled to the original resolution through bilinear interpolation, edge enhancement is carried out through PAS. Finally, cross-scale information interaction is achieved through feature concatenation and convolutional fusion (final_conv). Furthermore, the Dual-Domain Selection Mechanism (DSM) is introduced to dynamically screen key edge features through channel and spatial attention. In the lightweight design, depth-wise separable convolutions and grouped convolutions are adopted to reduce the computational complexity. For example, the 3×3 convolutional layer in the branch reduces the computational amount to 1/g of that of the traditional convolution through the grouping parameter g while maintaining the feature expression ability.The comparative analysis between the APAS module and other methods is shown in Table 1.

Table 1: Comparison between the APAS module and other methods

method	Edge extraction method	Multi-scale processing	Computational efficiency	adaptability
Traditional operator	Fixed gradient calculation	NO	Low	highly dependent on the scenario
single - scale edge enhancement	Data-driven	Single resolution	Moderate	Small target performance is constrained
Ordinary FPN fusion	No explicit edge enhancement	Multi-scale	Height	Edge information is prone to loss
APAS Method	Low-frequency - high-frequency separation + DSM	Multi-scale + dynamic selection	Moderate (lightweight design)	Strong task adaptability

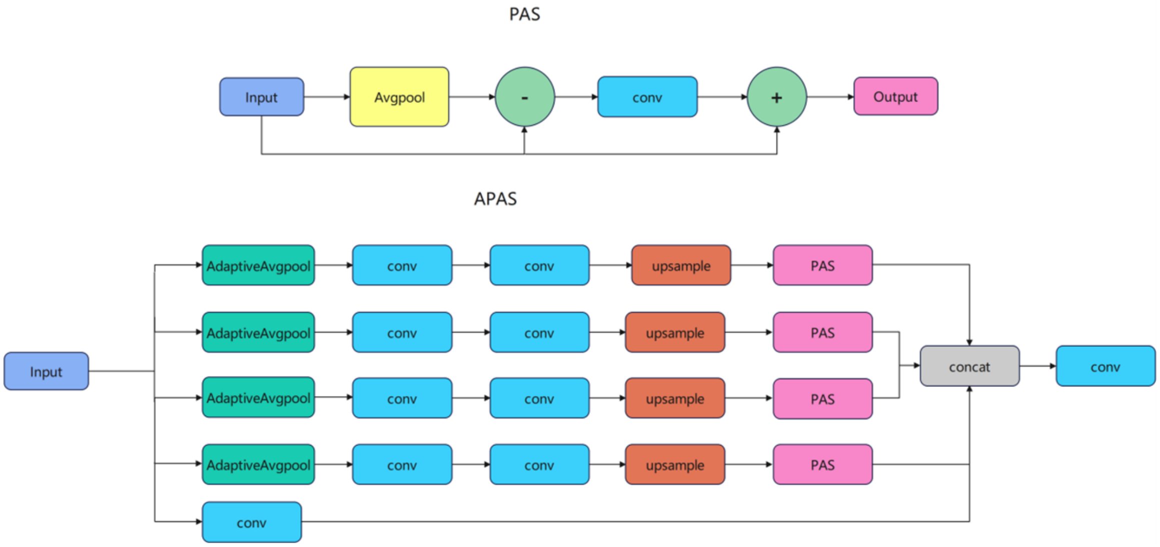

Figure 2: Overall Structure Diagram of APAS.

2.2. MDC Module

This module designs a multi-scale feature pyramid structure based on weight sharing, aiming to efficiently aggregate the context information of different receptive fields. As shown in Figure 3, the input features are first compressed in channels through a 1×1 convolutional layer to generate the initial feature representation. Subsequently, multi-scale features are generated in parallel using 3×3 convolutional kernels with shared weights, where the dilation rates are set to 1, 3, 5 in sequence. The design basis is that this combination follows the geometric series interval principle (1, 3, 5), balancing the computational efficiency and the feature coverage through exponential growth of the receptive fields. The theoretical basis can be traced back to the Atrous Convolution Pyramid theory (Chen et al., 2018), and its mathematical expression is:

$$RF_d = 2 \times \lfloor \frac{d \times (k-1)}{2} \rfloor + 1 \tag{1}$$

When $k = 3$, the effective receptive fields corresponding to each dilation rate are 3×3, 7×7, and 11×11 respectively, forming a multi-granularity feature coverage. Experimental studies have shown (Yu and Koltun, 2016) that this kind of design can significantly improve the model's robustness to scale changes. After the multi-branch features are concatenated along the channel dimension, 1×1 convolution is used to achieve cross-channel information interaction and dimension recalibration. This structure enhances the perception of local details through the dense sampling characteristics of the atrous convolution. At the same time, it uses the weight sharing mechanism to balance the computational efficiency and the multi-scale representation ability, and finally forms a hierarchical context feature pyramid.

2.3. RepGFPN Feature Network

In this paper, innovatively, the RepGFPN (Xu et al., 2022) module from DAMO-YOLO is introduced into the Neck part of the model, aiming to improve the model performance through hierar-

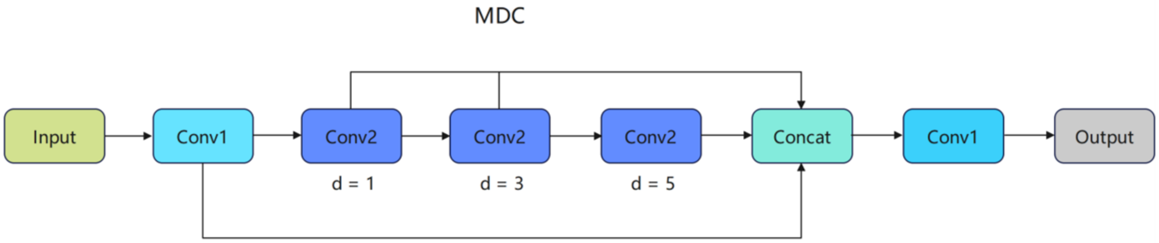

Figure 3: Structure Diagram of MDC.

chical feature aggregation and cross-stage information interaction. The network adopts a modular design, and its core components include the RepConv structure, the basic residual module, and the stage-wise feature fusion module (CSPStage).

The reparameterizable convolution RepConv structure, as shown in Figure 4, uses a dynamic activation function (SILU) and batch normalization (BN) to optimize the training stability, and supports deployment flexibility through a switchable inference mode (train/infer).

The basic residual module is designed with an inverted residual structure and consists of two convolutional layers: The first layer uses a lightweight reparameterizable convolution (RepConv) for feature mapping, and the second layer achieves channel dimension alignment through a standard 3×3 convolution (Conv). The module adds the input and output through a shortcut connection, enhancing the ability to express local details while retaining the original features. Among them, the channel scaling coefficient dynamically adjusts the dimension of the middle layer to balance computational efficiency and feature expression ability.

The stage-wise feature fusion module (CSPStage) adopts a channel splitting strategy, and its structure diagram is shown in Figure 5. The input features are proportionally divided into the main path and the auxiliary path. The main path directly extracts shallow features through a 1×1 convolution, and the auxiliary path mines deep features by stacking N basic residual modules. Finally, the multi-level features are concatenated in the channel dimension and then fused by a 1×1 convolution to achieve cross-stage information interaction. This structure improves efficiency by reducing redundant calculations and enhances semantic expression by using the feature reuse mechanism.

3. Results and Analysis

3.1. Experimental Configuration

The computer system used in this study is equipped with a high-performance NVIDIA RTX A5000Ti GPU and a CPU (AMD Ryzen 7 5800H with Radeon Graphics). The operating system is Windows 11. The software environment includes PyTorch 2.6.0, Torchvision 0.21.0, Python 3.10.16, and CUDA 11.8 (paired with cuDNN 8.9) to support the operation of the deep learning framework.

3.2. Datasets

In this experiment, the Urban Environment Detection dataset is used, which consists of a total of 5,648 images. Among them, there are 3,956 images in the training set, 848 images in the validation set, and 846 images in the test set. This dataset has four categories, namely cars, buses, pedestrians,

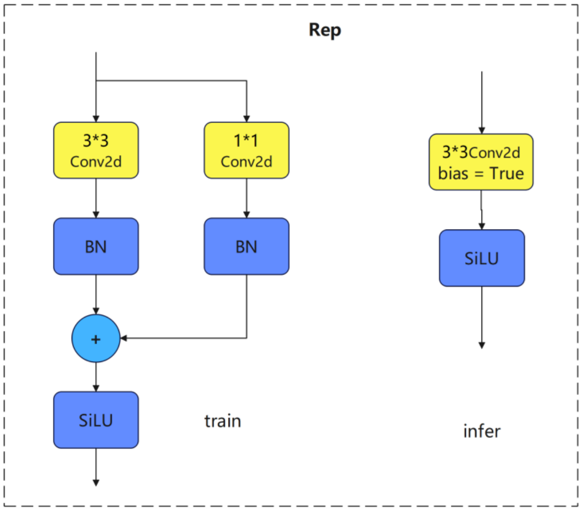

Figure 4: Structure Diagram of MDC.

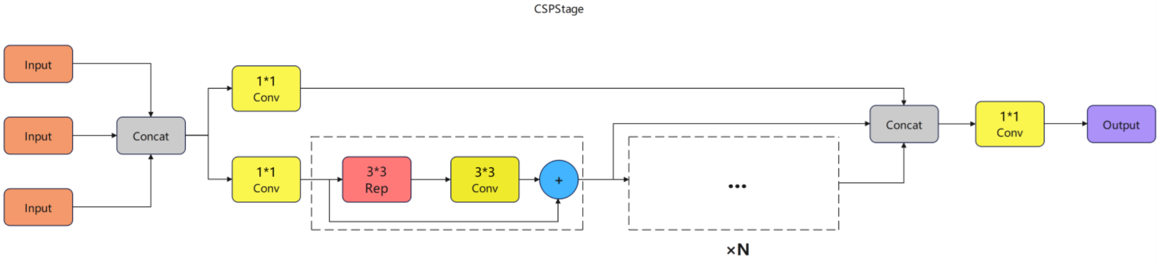

Figure 5: Structure Diagram of CSPStage.

and trucks. In particular, a subset of difficult samples accounting for 20% has been designed, including challenging scenarios such as severe occlusion (occlusion area $> 40\%$), small targets (pixel area $< 32 \times 32$), and motion blur.

The size of each image is 640*640, and the total number of annotations is 121,175. To prevent the model from overfitting, we adopt the gradient descent optimization algorithm. After repeated verification, we determine that the initial learning rate is 0.01, the batch size is 16, and the total number of training epochs is 300.

3.3. Experimental Results and Analysis

In this study, we use the mean average precision (mAP), frames per second (FPS), number of parameters (Params), and floating-point operations (FLOPs) as the core indicators to measure the detection accuracy, real-time performance, model lightweightness, and computational complexity. Among them, the average precision (AP) is used to evaluate the model's detection ability for a single category, and its value is calculated from the area under the precision-recall (P-R) curve. True positives (TP) represent the number of bounding boxes correctly predicted by the model, false positives (FP) are the number of bounding boxes wrongly predicted, and false negatives (FN) are the number of targets that have not been detected. AP is obtained by integrating the precision values under all recall rates, and mAP is the average result of the AP values of all categories. The calculation formula is as follows:

$$P = \frac{TP}{TP + FP} \tag{2}$$

$$R = \frac{TP}{TP + FN} \tag{3}$$

$$AP = \int_0^1 P\left(R\right) dR \tag{4}$$

$$mAP = \frac{1}{N} \sum_{i=1}^{N} AP_i \tag{5}$$

In this experiment, we conducted a comparative experiment between common object detection algorithms and the detection algorithm proposed in this paper. The results are shown in Table 2.

Table 2: Model Comparison Experiment.

	Precision	Recall	mAP50	mAP50-95
yolov5n (Ultralytics, 2024)	0.617	0.451	0.491	0.296
yolov8n (Ultralytics, 2023)	0.625	0.46	0.498	0.301
yolov10n (Wang et al., 2024)	0.605	0.455	0.498	0.3
yolov12n (Ultralytics, 2025)	0.62	0.461	0.502	0.31
Ours	0.643	0.478	0.519	0.325

As can be seen from Table 2, our algorithm has significant improvements in performance indicators compared with YOLOv5n, YOLOv8n, YOLOv10n and YOLOv12n. In terms of accuracy, it is 2.6%, 1.8%, 3.8%, and 2.3% higher respectively. In terms of recall rate, it is increased by 2.7%, 1.8%, 2.3%, 1.7%, and 14.4% respectively. In terms of mAP50, our model is 2.8%, 2.1%, 2.1%, 1.7%, and 15.9% higher than the other models. In terms of mAP50-95, it is 2.9%, 2.4%, 2.5%, 1.5%, and 8% higher respectively. These experimental data show that the detection algorithm proposed in this paper is superior to the above common object detection algorithms.

Table 3: Ablation Experiments of the Modules

APAS	MDC	RepGFPN	Precision	Recall	mAP50	mAP50-95	Params	GFLOPS
			0.617	0.469	0.501	0.307	2,582,932	6.3
✓			0.62	0.477	0.513	0.316	2,531,116	6.3
	✓		0.63	0.462	0.505	0.314	2,730,388	6.3
		✓	0.609	0.481	0.51	0.32	3,660,388	8.2
✓	✓		0.616	0.481	0.516	0.322	2,678,572	6.3
✓	✓	✓	0.643	0.478	0.519	0.325	3,756,028	8.2

Ablation experiments between modules were conducted in this study. As shown in Table 3, we can conclude that each improved module has improved the detection performance to varying degrees. Compared with the basic model, after adding the APAS module, the mAP50 and mAP50-95 have increased by 1.2% and 0.9% respectively. After adding the MDC module, the mAP50

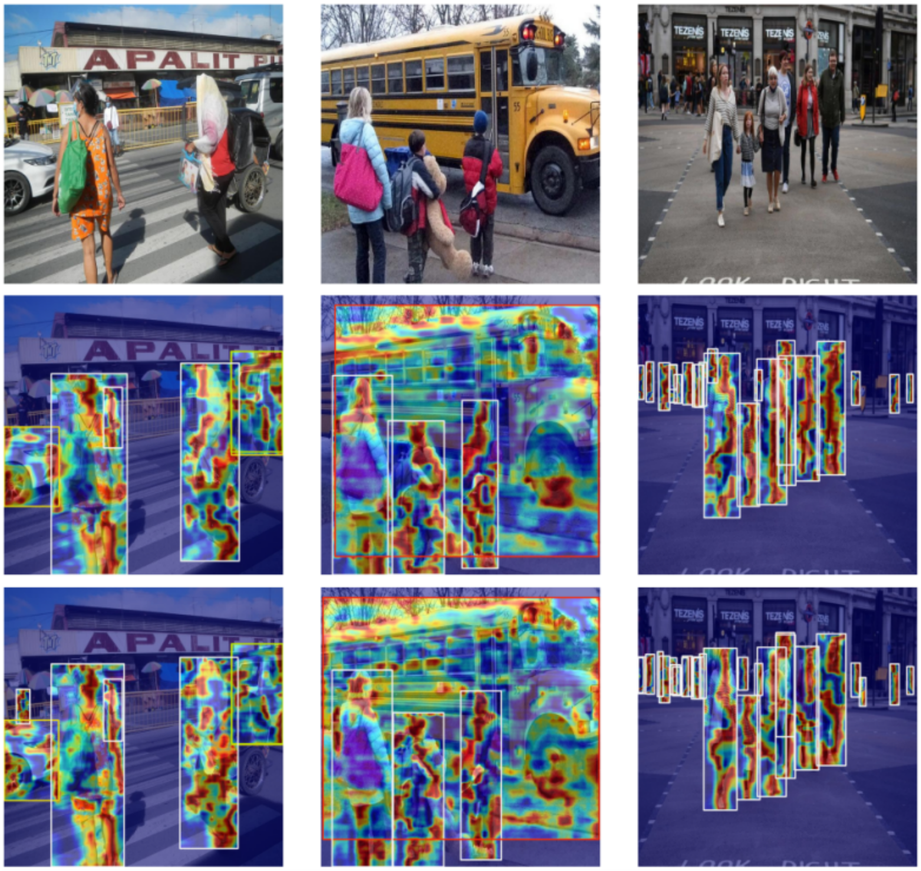

Figure 6: Heat map (compared with the basic model).

and mAP50-95 have increased by 0.4% and 0.7% respectively. After introducing the RepGFPN feature network, the mAP50 and mAP50-95 have increased by 0.9% and 1.3% respectively. After fusing the APAS module and the MDC module and adding them to the basic model, the model's ability to process edge information and handle features of different scales has been enhanced. At this time, both the mAP50 and mAP50-95 have increased by 1.5%. Finally, after introducing the RepGFPN feature network and fusing it into the model, both the mAP50 and mAP50-95 have increased by 1.8%. Figure 6 shows a comparison of heat maps. The first row of this figure is the original image, and the second and third rows are the heat maps of the basic model and the algorithm proposed in this study respectively. For the situation of image occlusion, if the target is partially occluded, the heatmap of our algorithm can still focus on the unoccluded area and there is no wrong shift towards the background, indicating that the algorithm in this paper has good robustness to occlusion. For large-sized targets in this study, by observing the heat map, it can be found that the targets are completely and reasonably covered. For small-sized targets, they are not obscured by the background in the heatmap, and there is a concentrated high-brightness area, indicating that the model has a strong adaptability to size changes. This shows that compared with the YOLOv11 basic model, the YOLOv11-APAS-MDC algorithm proposed in this paper has significantly improved the target detection performance in complex urban scenes.

4. Conclusion

In this paper, an algorithm named YOLOv11-APAS-MDC for pedestrian and vehicle detection in complex urban scenes is proposed, and this algorithm is experimented on the Urban Environment Detection dataset. The multi-scale edge information enhancement module APAS is added, which focuses on highlighting the key edge features and improves the model's ability to perceive and represent the object edges in complex scenes. The MDC module is added, which can accurately capture features of various scales by virtue of convolutional layers with different dilation rates. Additionally, the RepGFPN feature network is introduced, and it uses a more sophisticated cross-layer connection mechanism to enhance the performance and efficiency of the model's detection. Experimental data show that, compared with the YOLOv11 basic model, the accuracy, recall rate, mAP50, and mAP50-95 of this experiment have all been significantly improved. This indicates that the improved algorithm can better adapt to the object detection tasks in complex urban scenes, providing more reliable and efficient technical support for practical application fields such as autonomous driving and intelligent transportation.

Acknowledgments

This work was supported in part by the Natural Science Foundation of China under Grant 62276164; in part by theScience and Technology Innovation Action Plan Natural Science Foundation of Shanghai under Grant 22ZR1427000;and in part by the Shanghai Oriental Talent Program-Youth Program. The authors would like to thank their supports.

References

Z. Cai and N. Vasconcelos. Cascade r-cnn: Delving into high quality object detection. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6154–6162. IEEE, 2018. doi: 10.1109/CVPR.2018.00644.

L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4):834–848, 2018. doi: 10.1109/TPAMI.2017.2699184.

O. Hmidani and E. M. Ismaili Alaoui. A comprehensive survey of the r-cnn family for object detection. In *2022 5th International Conference on Advanced Communication Technologies and Networking (CommNet)*, pages 1–6. IEEE, 2022. doi: 10.1109/CommNet56067.2022.9993862.

J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 779–788. IEEE, 2016. doi: 10.1109/CVPR.2016.91.

Ultralytics. Ultralytics yolov8. GitHub, 2023. URL https://github.com/Pertical/YOLOv8.

Ultralytics. Yolov5[computersoftware]. GitHub, 2024. URL ttps://github.com/ultralytics/yolov5?tab=readme-ov-file.

Ultralytics. yolov12. GitHub, 2025. URL https://github.com/ultralytics/yolov12.

A. Wang, H. Chen, L. Liu, K. Chen, Z. Lin, J. Han, and G. Ding. Yolov10: Real-time end-to-end object detection. *arXiv preprint arXiv:2405.14458*, 2024.

X. Xu, Y. Jiang, W. Chen, Y. Huang, Y. Zhang, and X. Sun. Damo-yolo: A report on real-time object detection design. *arXiv preprint arXiv:2211.15444v2*, 2022.

F. Yu and V. Koltun. Multi-scale context aggregation by dilated convolutions. In *International Conference on Learning Representations (ICLR)*, 2016.