Advances in Computer, Signals and Systems (2024)
Clausius Scientific Press, Canada

DOI: 10.23977/acss.2024.080702
ISSN 2371-8838 Vol. 8 Num. 7

Efficient multi-scale traffic object detection method based on RT-DETR

Songnan Zhang[1,a,*], Xiang Peng[1]

[1]*School of Information and Electronic Technology, Key Laboratory of Autonomous Intelligence and Information Processing in Heilongjiang Province, Jiamusi University, Jiamusi, China*
[a]*zhangsongnan16@163.com*
[*]*Corresponding author*

Keywords: Object detection, Transformer, Multi-scale network, Attention mechanism

Abstract: Traffic object detection is a crucial technological application with significant development potential. To address the limitations of current methods in multi-scale object detection, this paper introduces an Efficient Multi-scale Traffic Object Detection method based on RT-DETR. Specifically, we have designed an Efficient Multi-scale Network that incorporates Multi-head Mixed Convolution (MMC), Multi-scale Aggregation (MA), and an Efficient Multi-scale Module (EMM). This method integrates convolutional techniques with transformers to minimize the computational overhead of the model while enhancing the effectiveness of multi-scale detection. Experimental results demonstrate that, compared to the original method, the Average Precision (AP) and the Small Object Average Precision (APs) of our method have improved by 1.2% and 1.1%, respectively, indicating a notable advantage over similar approaches.

1. Introduction

Object detection is a fundamental task in computer vision, aimed at identifying the location and category of various targets. In particular, traffic object detection focuses on vehicles, pedestrians, signs, and other relevant elements. This application is extensively utilized in real-world scenarios and constitutes a critical technique for enhancing traffic safety.

Object detection methods are categorized into traditional methods and deep learning-based methods. Traditional methods primarily involve feature extraction and classification for detection. However, they predominantly rely on hand-designed features, which struggle to effectively handle multi-scale and multi-angle targets, resulting in poor detection performance and limited applicability. In contrast, early deep learning-based methods are primarily represented by Two-Stage and One-Stage approaches. Two-Stage methods, exemplified by the Fast R-CNN[1] series of algorithms, significantly enhance detection speed compared to R-CNN, yet they cannot circumvent the Non-Maximum Suppression (NMS) process, which limits their effectiveness in dense object detection tasks[2]. One-Stage methods, represented by the YOLO series of algorithms, reformulate object detection as a regression problem, allowing for the simultaneous completion of classification and localization tasks in a single step[3]. However, these methods also face challenges in effectively addressing dense object detection scenarios.

Recently, Transformers from the NLP field have demonstrated powerful global modeling capabilities and have shown strong performance in computer vision tasks. As the first detector based on Transformers, DETR eliminates the need for the NMS process, achieving true end-to-end target detection[4]. Anchor-DETR associates anchors with query vectors to enhance the relevance of the model's predictions for each position during inference[5]. Deformable-DETR introduces deformable convolutions into both the encoder and decoder, thereby improving the model's ability to detect small targets[6]. PnP-DETR offers a flexible and scalable framework for target detection through its modular design, allowing users to select and combine different modules based on their specific needs to adapt to various detection task scenarios[7]. DINO-DETR enhances detection accuracy and effectively reduces the model training time by employing contrastive methods for denoising training and hybrid query selection methods for anchor point initialization[8]. RT-DETR features numerous lightweight designs based on the original DETR, resulting in faster detection speeds suitable for low-latency applications such as autonomous driving and video surveillance[9].

Despite the ongoing advancements in these methods, several challenges persist in traffic object detection tasks. These approaches often lack effectiveness in multi-scale object detection, particularly for vehicles and pedestrians, and they tend to be computationally intensive, which complicates their applicability in real-world scenarios.

In summary, this study proposes an efficient multi-scale traffic object detection method based on RT-DETR. To address the model's limitations in effectively detecting objects at multiple scales, an efficient multi-scale network is employed. This approach not only enhances the model's capability for multi-scale object detection but also reduces computational requirements.

2. Related work

2.1 Vision transformer

Vision Transformer (ViT), as the first model to apply transformers to image classification, establishes a new paradigm for the use of transformers in computer vision[10]. ViT serializes images and demonstrates a strong capability for global modeling through its unique multiple self-attention (MSA) mechanism. DeiT enhances the training efficiency of the model by introducing a method of instructor model distillation, which reduces the dataset required for training[11]. The Swin Transformer builds upon this by constraining the MSA to a fixed window, thereby reducing computational overhead while incorporating window bias. Additionally, the Swin Transformer enhances information exchange between windows through the use of window offset operations[12]. T2T-ViT introduces a novel approach to generating more expressive token sequences layer by layer, which mitigates the loss of local information associated with image slicing[13]. MaxViT combines local and global window attention mechanisms to further enhance the performance of the transformer[14]. Despite these advancements, these models continue to impose a significant computational burden.

2.2 CNN-Transformer architecture in object detection

DETR represents the first pure transformer architecture for object detection. While it revolutionizes traditional architectures, it faces significant drawbacks, including poor multi-scale object detection, slow model convergence, and substantial computational demands. These limitations are intrinsically linked to the transformer structure and influence one another, providing a reference point for potential model improvements. The attention mechanism inherent in transformer models offers robust and comprehensive global modeling capabilities, which are essential for capturing image context information. However, this advantage comes at the cost of

considerable computational effort, often resulting in redundant calculations. Addressing multi-scale object detection within this framework poses additional challenges. Deformable-DETR enhances the Transformer module by incorporating deformable convolution, representing a notable CNN-Transformer architecture for object detection. Convolutional networks, with their varying kernel sizes, excel at extracting multi-scale information about targets while focusing locally on images, thereby reducing redundant computations within the Transformer architecture. This integration also helps to mitigate the inherent shortcomings of the transformer model. DAB-DETR [15], DN-DETR[16] and DINO-DETR build upon the foundational architecture of Deformable-DETR, demonstrating its potential. However, the ability to acquire multi-scale features remains inadequate. As research progresses, efficient architectures that combine these approaches have emerged, including MobileViT[17], EfficientFormer[18] and SMMA[19]. These works ingeniously integrate Convolutional Neural Network (CNN) into the Transformer architecture through various approaches. This integration facilitates the consideration of both long-range dependencies and local feature acquisition in images. In summary, these contributions offer valuable insights for enhancing multi-scale object detection models.

3. Method

3.1 Framework of the algorithm

As shown in Figure 1, the algorithm is mainly composed of three parts: feature extraction stage, encoder, decoder and detection head. In the feature extraction stage, Efficient Multi-scale Network is used and the structure is divided into four stages. In the first two phases, Efficient Multi-scale Module (EMM) is mainly deployed. The last two phases are Fusion Module (FM) and Swin Transformer Module (WMSA) in that order.

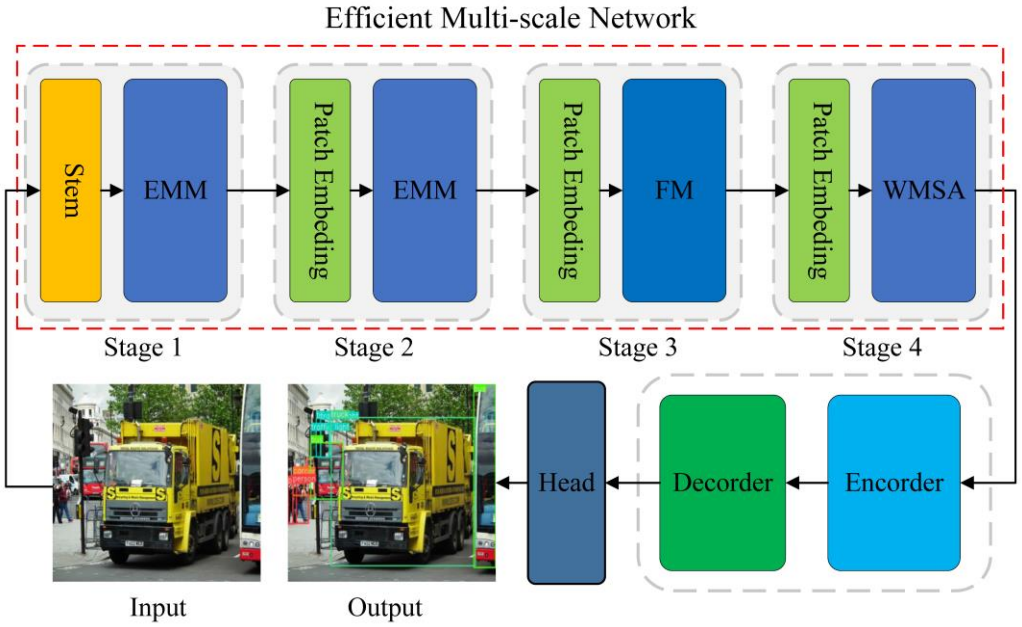

Figure 1: Framework of the algorithm.

3.2 Muti-head Mixed Convolution

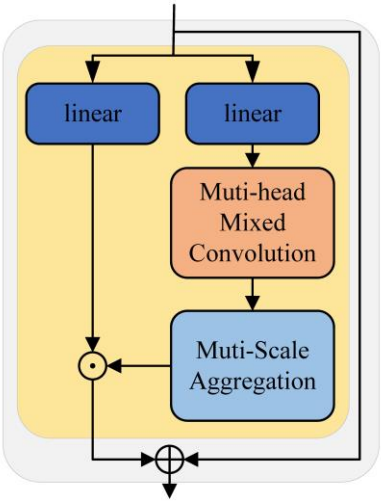

Figure 2: Efficient Multi-scale Module.

As shown in the Figure 2, for vector $X \in R^{C\times H\times W}$, after Muti-head Mixed Convolution (MMC), the number of head is *N*. The vector is divided into *n* parts according to the spatial dimensions to get $x=\{x_1,x_2,...,x_n\}$, which are entered into the convolution and then spliced respectively, as shown in Equation (1):

$$MMC(X)=Concat(DW_{k_1\times k_1}(x_1),...DW_{k_n\times k_n}(x_n)) \tag{1}$$

Where $k_i\times k_i=\{1,3,5...,2i-1\}$ denotes the size of the convolution kernel corresponding to different heads, $i\in[1,n]$, $n=C/N$.

Dividing the vector into different heads according to *C* is more conducive to learning the features of the target in different subspaces. Convolution operations of different sizes are performed on different vectors x_n to make full use of the characteristics of the convolution combination to obtain multi-scale features of the target. This method is easy to implement, not only combining Transformer and CNN successfully, but also easy to obtain the fine-grained multi-scale features of the target.

3.3 Muti-scale Aggregation

To enhance information exchange among different heads, Multi-Scale Aggregation (MA) is performed following MMC processing, as illustrated in Figure 2. Initially, a channel is selected within each head to form a group, after which an up-down feature fusion operation is conducted within each group utilizing the inverse bottleneck structure to improve the representation of multi-scale features. For vector $X \in R^{C\times H\times W}$, the number of groups is designated as $Groups=C/N$, and point-wise convolution is employed to facilitate the interaction of global information, with the MA calculated as shown in Equation (2):

$$\begin{cases} Q=W_a([G_1,G_2,...G_M]) \\ G_i=W_b([H_1^i,H_2^i,...H_N^i]) \\ H_j^i=DWconv_{k_j\times k_j}(x_j^i)\in R^{H\times W\times 1} \end{cases} \tag{2}$$

Where $i \in [1, M]$, $j \in [1, N]$, the number of head is $N, M = C / N$. $H_j \in R^{H \times W \times M}$ represents the j-th head with depth-wise convolution, H_j^i represents the i-th channel in the j-th head. W_a and W_b denote weight matrices of point-wise convolution.

3.4 Efficient Multi-scale Module

As illustrated in the Figure 2, the Efficient Multi-scale Module (EMM) is employed to modulate both the MMC and MA, resulting in the final output feature map. The calculation process for generating Z from the vector $X \in R^{C \times H \times W}$ is detailed in the accompanying Equation (3):

$$\begin{cases} Z = Q \odot V \\ V = W_v X \\ Q = MA(MMC(W_s X)) \end{cases} \tag{3}$$

where W_v and W_s denote linear transformations and $\odot$ denotes matrix dot product. Unlike traditional Transformers, EMA obtains Q through a convolutional structure that modulates spatial and channel-specific values after element wise multiplication, while reducing computational overhead.

4. Experimental results and analysis

4.1 Settings

The experiment utilizes the COCO2017 dataset, from which ten categories pertinent to traffic targets have been selected, including person, car, truck, and other common targets. The dataset comprises 35,784 training samples and 2,431 validation samples. The experimental metrics employed are AP, AP_{50} and AP_S. The Adam optimizer is utilized, featuring an initial learning rate of 0.0005, and a batch size of 8. The hardware environment is detailed in Table 1.

Table 1: Hardware environment.

Item	Setting
CPU	Intel(R) Xeon(R) Gold 6338
GPU	NVIDIA GeForce RTX 4090
RAM	32GB
Deep learning frameworks	Pytorch 2.2 Python 3.8
Operating system	Ubuntu 18.04

4.2 Comparison with other methods

To assess the effectiveness of this method, it was compared with other approaches under identical conditions, and the experimental results are presented in the accompanying Table 2. Among all algorithmic methods within the DETR class, the current method achieves the highest AP of 0.549, which is approximately 1.2% greater than that of the original method, while the AP_S roughly 1.1% higher than the original method as well. Although the AP and AP_S metrics of the current method are slightly lower than those of YOLOv8, it requires significantly fewer epochs and GFLOPs to achieve basic convergence compared to YOLOv8. In contrast to DINO-DETR, the current method necessitates more epochs for convergence. However, it ultimately demonstrates superior overall performance.

Table 2: The results of comparison

Model	AP	AP_{50}	AP_S	Epoch	GFLOPs
YOLOv7[20]	0.533	0.533	0.313	100	**104**
YOLOv8[21]	**0.552**	**0.552**	**0.342**	100	165
Deformable-DETR	0.504	0.504	0.316	50	173
DINO-DETR	0.531	0.531	0.323	**30**	279
RT-DETR	0.537	0.537	0.329	40	136
Proposed method	**0.549**	**0.549**	**0.340**	**60**	**125**

4.3 Visualization

We utilize visualization to assess the feasibility of this method in traffic scenarios, with the experimental results presented in the Figure 3. This figure includes the current mainstream methods: (a) Image to be detected (b) DETR (c) Deformable-DETR (d) DINO-DETR (e) RT-DETR (f) the proposed method. As illustrated in (f), our method demonstrates the fewest missed targets for pedestrians and traffic lights when compared to the other methods, showcasing its strong adaptability.

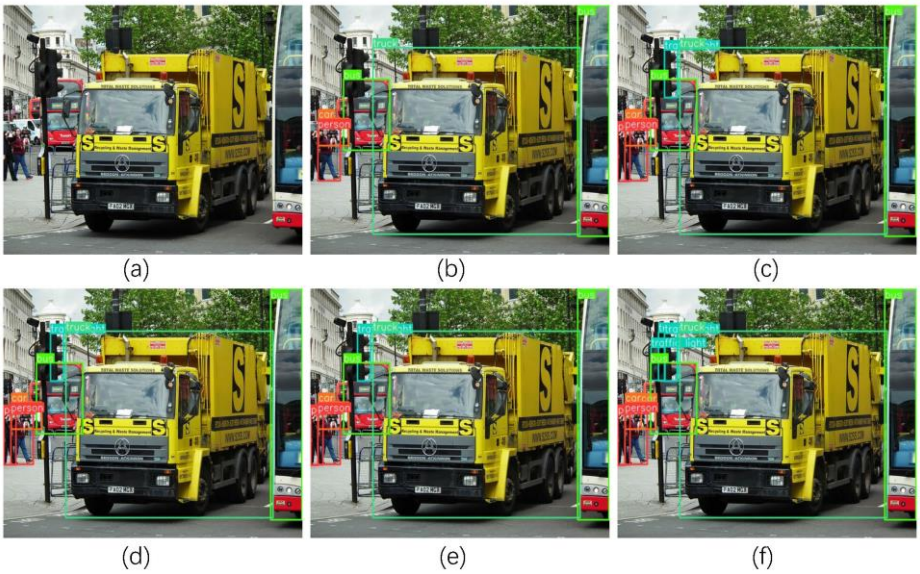

Figure 3: Result of visualization.

5. Conclusion

In this paper, we propose an efficient multi-scale traffic object detection method based on RT-DETR. Addressing the challenges associated with inadequate multi-scale object detection, we introduce an Efficient Multi-scale Network that incorporates MMC, MA and EMM. This method synergistically combines the strengths of Convolution and Transformer architectures to effectively capture multi-scale features of targets while minimizing computational overhead. Experimental results demonstrate that our approach exhibits significant advantages in traffic target recognition.

References

[1] Wang X, Shrivastava A, Gupta A. A-fast-rcnn: Hard positive generation via adversary for object detection[C]// Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 2606-2615.

[2] Bodla N, Singh B, Chellappa R, et al. Soft-NMS--improving object detection with one line of code[C]//Proceedings of the IEEE international conference on computer vision. 2017: 5561-5569.
[3] Wang A, Chen H, Liu L, et al. Yolov10: Real-time end-to-end object detection[J]. arXiv preprint arXiv:2405.14458, 2024.
[4] Carion N, Massa F, Synnaeve G, et al. End-to-end object detection with transformers[C]//European conference on computer vision. Cham: Springer International Publishing, 2020: 213-229.
[5] Wang Y, Zhang X, Yang T, et al. Anchor detr: Query design for transformer-based detector[C]//Proceedings of the AAAI conference on artificial intelligence. 2022, 36(3): 2567-2575.
[6] Zhu X, Su W, Lu L, et al. Deformable detr: Deformable transformers for end-to-end object detection[J]. arXiv preprint arXiv:2010.04159, 2020.
[7] Wang T, Yuan L, Chen Y, et al. Pnp-detr: Towards efficient visual analysis with transformers[C]//Proceedings of the IEEE/CVF international conference on computer vision. 2021: 4661-4670.
[8] Zhang H, Li F, Liu S, et al. Dino: Detr with improved denoising anchor boxes for end-to-end object detection[J]. arXiv preprint arXiv:2203.03605, 2022.
[9] Zhao Y, Lv W, Xu S, et al. Detrs beat yolos on real-time object detection[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2024: 16965-16974.
[10] Dosovitskiy A, Beyer L, Kolesnikov A, et al. An image is worth 16x16 words: Transformers for image recognition at scale[J]. arxiv preprint arxiv:2010.11929, 2020.
[11] Touvron H, Cord M, Jégou H. Deit iii: Revenge of the vit[C]//European conference on computer vision. Cham: Springer Nature Switzerland, 2022: 516-533.
[12] Liu Z, Lin Y, Cao Y, et al. Swin transformer: Hierarchical vision transformer using shifted windows[C]// Proceedings of the IEEE/CVF international conference on computer vision. 2021: 10012-10022.
[13] Yuan L, Chen Y, Wang T, et al. Tokens-to-token vit: Training vision transformers from scratch on imagenet[C]// Proceedings of the IEEE/CVF international conference on computer vision. 2021: 558-567.
[14] Tu Z, Talebi H, Zhang H, et al. Maxvit: Multi-axis vision transformer[C]//European conference on computer vision. Cham: Springer Nature Switzerland, 2022: 459-479.
[15] Liu S, Li F, Zhang H, et al. Dab-detr: Dynamic anchor boxes are better queries for detr[J]. arXiv preprint arXiv:2201.12329, 2022.
[16] Li F, Zhang H, Liu S, et al. Dn-detr: Accelerate detr training by introducing query denoising[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2022: 13619-13627.
[17] Mehta S, Rastegari M. Mobilevit: light-weight, general-purpose, and mobile-friendly vision transformer[J]. arXiv preprint arXiv:2110.02178, 2021.
[18] Li Y, Yuan G, Wen Y, et al. Efficientformer: Vision transformers at mobilenet speed[J]. Advances in Neural Information Processing Systems, 2022, 35: 12934-12949.
[19] Lin W, Wu Z, Chen J, et al. Scale-aware modulation meet transformer[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2023: 6015-6026.
[20] Wang C Y, Bochkovskiy A, Liao H Y M. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2023: 7464-7475.
[21] Shen L, Lang B, Song Z. Infrared object detection method based on DBD-YOLOv8 [J]. IEEE Access, 2023.