Chain Chain, vol. 2, no. 2, pp. 1–XX, 2025
DOI:10.23919/CHAIN.2025.000008

ISSN 2097-3470
CN 10-1915/TB

Research Article

Improved YOLO algorithm based on multi-scale object detection in haze weather scenarios

Junqing Shi[1], Sui Ruan[1], Yanhong Tao[1], Yingxu Rui[1,✉], Jun Deng[2], Peng Liao[3], and Peng Mei[4]

[1] *College of Engineering, Zhejiang Normal University, Jinhua 321000, China*
[2] *Department of Wuhan Comprehensive Transportation Research Institute, Co., Ltd, Wuhan 430056, China*
[3] *Department of College of Engineering, Ocean University of China, Qingdao 266110, China*
[4] *Department of Mechanical Engineering, Politecnico di Milano, Milan, Lombardy 20133, Italy*

Received: 29 March 2025
Revised: 9 April 2025
Accepted: 9 April 2025
Online: 15 April 2025

KEYWORDS
convolutional network,
object detection,
self-attention mechanism,
YOLO algorithm.

ABSTRACT

Computer vision-based traffic object detection plays a critical role in road traffic safety. Under hazy weather conditions, images captured by road monitoring systems exhibit three main challenges: significant scale variations, abundant background noise, and diverse perspectives. These factors lead to insufficient detection accuracy and limited real-time performance in object detection algorithms. We propose AMC-YOLO an improved YOLOv11-based traffic detection algorithm to address these challenges. In this work, we replace the C3k block's bottleneck module with our novel attention-gate convolution (AGConv), which improves contextual information capture, enhances feature extraction, and reduces computational redundancy. Additionally, we introduce the multi-dilation sharing convolution (MDSC) module to prevent feature information loss during pooling operations, enhancing the model's sensitivity to multi-scale features. We design a lightweight and efficient cross-channel feature fusion module (CCFM) for the path aggregation neck to adaptively adjust feature weights and optimize the model's overall performance. Experimental results demonstrate that AMC-YOLO achieves a 1.1% improvement in mAP@0.5 and a 2.7% increase in mAP@0.5:0.95 compared to YOLOv11n. On graphics processing unit (GPC) hardware, it achieves real-time performance at 376 (FPS) with only 2.6 million parameters, ensuring high-precision traffic detection while meeting deployment requirements on resource-constrained devices.

Address correspondence to yingxurui@zjnu.edu.cn

有科出版 YOUKE PUBLISHING
EXCLUSIVELY AVAILABLE VIA IEEE Xplore® DIGITAL LIBRARY

1 Introduction

With the rapid increase in private car ownership and the growing complexity of public transportation systems, traffic congestion, vehicle collisions, and other traffic safety issues have become increasingly severe. Intelligent transportation systems, relying on deep learning-based object detection, can accurately and in real-time acquire road traffic flow information. This supports informed decision-making by managers, effectively reducing congestion and improving road safety. However, under haze weather conditions, reduced visibility increases image noise, making it difficult to correctly identify and detect vehicles, pedestrians, and other targets (Fig. 1). This presents a significant challenge to convolutional network-based object detection systems.

In recent years, convolutional neural network (CNN) technology has developed rapidly and widely applied in various fields, particularly in image processing and video-related tasks [1]. CNNs, with their unique network structures and convolutional operations, can automatically extract features from input data and perform effective classification and regression tasks. Depending on the processing flow, common object detection methods are categorized into two-stage and single-stage object detection. Two-stage object detection first generates candidate regions through a region proposal network (RPN), then classifies and precisely locates these regions. For instance, three representative approaches include: (a) the seminal R-CNN framework [2]; (b) spatial pyramid pooling networks (SPP-Net) [3], introducing spatial pyramid pooling to mitigate feature redundancy; and (c) region-based fully convolutional networks (R-FCN) [4] employing a fully convolutional architecture. These methods can better filter background noise for accurate predictions, but are slower in detection speed. Single-stage object detection achieves faster processing speeds. This approach typically uses a single neural network model to complete the task, including object localization and classification (e.g., YOLO [5], single shot multiBox detector (SSD) [6], RetinaNet [7]). However, it faces the challenge of low detection accuracy.

In 2017, Vaswani et al. [8] proposed the Transformer model, which became dominant in natural language processing. The Transformer model introduced the self-attention mechanism, replacing traditional recursive and convolutional structures. The self-attention mechanism allows the model to globally interact with any position in the input sequence during decoding, enhancing the model's expressive power and generalization ability. Recent studies by introduced Vision Transformers (ViT), successfully applying the Transformer architecture to computer vision. This study demonstrated that pure Transformers applied directly to image patch sequences can perform exceptionally well in image classification tasks. Transformers are not limited by the locality of convolutional operations and can globally focus on dependencies between image feature patches, while requiring fewer computational resources than CNNs.

On the other hand, object size variation analysis in road surveillance videos poses significant challenges, where most targets appear at medium scales while many others exhibit extreme (either very small or large) dimensions. Traditional single-layer convolutional neural networks produce feature maps with inherent limitations, including uniform receptive fields and constrained multi-scale representation capacity. Therefore, developing scale-adaptive feature representation frameworks has become a critical research challenge. Existing studies typically employ static fusion strategies via direct stacking or channel-wise concatenation

Figure 1 The traffic detection challenges: (a) small target, (b) shape change, and (c) object occlusion.

of shallow detail features with deep semantic features. However, this approach fails to consider the non-uniform contribution patterns of multi-scale features in both channel and spatial dimensions, potentially causing fine-grained information loss or semantic ambiguity. Our key innovation introduces learnable weights to dynamically assess the importance of different input features, while implementing iterative multi-scale fusion through coordinated top-down and bottom-up processing pipelines. This architecture enables the network to autonomously learn cross-scale spatial/temporal correlations and inter-level channel-wise relationships, establishing a more robust and discriminative scale-invariant feature representation system.

The degradation of sample quality under hazy conditions and the complexity of traffic scenes necessitate detection systems that balance recognition accuracy with computational efficiency. Consequently, the model must accomplish high-precision multi-scale object detection within constrained computational resources. This paper introduces several enhancements to YOLOv11n aimed at improving model performance while reducing inference time. The principal contributions are as follows.

(1) We propose the attention-gate convolutional (AGConv) block, which incorporates both the convolutional additive self-attention (CAS) [9] module and convolutional gated linear units (CGLU) [10] module to augment the C3k2 module in YOLOv11n. The enhanced C3k2 module significantly improves feature extraction capabilities while simultaneously reducing the model's computational burden.

(2) We present the multi-dilation shared convolution (MDSC) module, which employs convolutional layers with varying dilation rates for multi-scale feature extraction. This design enhances sensitivity to objects at different scales and improves contextual information integration. To optimize parameters, we implement shared convolutional kernels across dilation rates rather than using separate layers for each rate. This approach effectively minimizes parameter redundancy while enhancing model efficiency.

(3) The cross-channel fusion module (CCFM) represents an advanced feature fusion mechanism designed to enhance YOLOv11's feature pyramid network (FPN). By incorporating contextual guidance and adaptive feature adjustment during multi-scale fusion, it ensures robust object detection in complex scenarios. The module employs channel-wise squeezing and excitation to selectively compress and amplify feature representations, enabling dynamic contextual information utilization. Through learned adaptive weight allocation for feature reorganization, the framework improves representation efficacy and directs attention to target-relevant patterns. This mechanism substantially enhances detection accuracy by emphasizing discriminative features while suppressing irrelevant information.

2 Related work

In recent years, the rapid advancement of deep learning technologies has led to widespread applications of object detection across various domains. Contemporary object detection models are generally categorized into two-stage detectors (e.g., R-CNN, SPP-NET, and R-FCN [2]–[4]) and single-stage detectors (e.g., SSD, RetinaNet, and YOLO [5–7]). The YOLO, as a single neural network-based detection framework, demonstrates remarkable efficiency for real-time applications. However, earlier versions exhibited limitations in generalization capability and detection accuracy, prompting numerous researchers to focus on architectural improvements.

Early YOLO iterations (v1–v3 [11]) employed Darknet backbones with pyramid pooling layers to balance efficiency and accuracy, albeit with increased computational demands. Subsequent versions (v4–v6 [12]) adopted CSPDarknet-53 backbones and transitioned to anchor-free mechanisms, significantly reducing model complexity while maintaining state-of-the-art performance. Nevertheless, these models still face challenges in fully leveraging contextual information for precise detection in complex scenarios. Recent advancements (v8–v11) have preserved effective historical designs while incorporating cutting-edge technologies, achieving further improvements in both lightweight design and accuracy. For instance, YOLOv8 replaced the C3 module with a C2f module and implemented a decoupled head structure to reduce computational redundancy. YOLOv11 introduced enhanced backbone and neck architectures, improving the C3k2 module and adding a C2PSA module to strengthen feature extraction capabilities, though multi-scale target detection in extreme scenarios remains challenging.

Current object detection models predominantly rely on CNN for feature extraction. While effective at local feature processing, CNNs' limited receptive fields constrain their ability to capture global contextual information. Additionally, important shallow features may be lost through successive convolutional and pooling operations, potentially degrading model performance. The ViT architecture addressed these limitations by abandoning traditional CNN designs in favor of image patching and encoding mechanisms that better capture global relationships. ViT achieved top-tier performance on large-scale datasets like ImageNet [13], establishing the foundation for self-attention mechanisms in computer vision. Notably, the Detection Transformer (DETR) [14] became the first fully end-to-end trained detection model, combining CNN and Transformers to reformulate object detection as a set prediction problem. This innovation eliminated traditional anchor mechanisms, simplified the detection pipeline, and improved accuracy.

However, as model performance improves, structural complexity increases, leading to higher training and deployment costs. Recent research has addressed these challenges through various innovations. The SMCA model [15] introduced a spatial-modulated co-attention mechanism that improved DETR's convergence, reducing required training epochs from 500 to 108 while enhancing performance. Similarly, the Swin Transformer [16] implemented a shifted window mechanism to reduce computational complexity and improve local feature extraction.

A fundamental challenge in object detection involves effective multi-scale feature representation and processing. While low-level features offer high resolution and detailed information, high-level features provide stronger semantic representations with reduced noise. Multi-scale feature fusion enhances model comprehension by integrating information across different layers, enabling the capture of complex patterns. Early fusion methods relied on computationally intensive image pyramids [6], which generated multi-resolution inputs through iterative downsampling. With deep learning advancements, Lin et al. [17] proposed the feature pyramid network (FPN), which creates multi-scale representations through top-down and lateral connections between different feature levels, significantly improving detection accuracy. Building on this foundation, Hu et al. [18] incorporated squeeze-and-excitation (SE) attention mechanisms to adaptively weight channel features, emphasizing the most discriminative information.

3 Proposed method

This paper presents AMC-YOLO, an enhanced hybrid model based on the YOLOv11n architecture. As illustrated in Fig. 2, AMC-YOLO incorporates three key improvements over the original YOLOv11n: (1) The enhanced AGConv module replaces the original C3k2 module to improve feature extraction capability and robustness while reducing computational complexity; (2) in the super position polynomial factorization (SPPF) layer, we implement the MDSC module, which utilizes shared 3×3 convolutions with dilation rates of 1, 3, and 5 to progressively expand the receptive field while minimizing computational redundancy, thereby effectively capturing multi-scale contextual information. (3) A novel CCFM is designed to optimize feature fusion through adaptive feature reorganization. It emphasizes discriminative elements while suppressing less relevant features, consequently enhancing the representational power of feature maps. Compared to YOLOv11n, AMC-YOLO maintains deployment efficiency while significantly improving detection accuracy and robustness in traffic scenarios. The following sections detail these architectural innovations.

3.1 Attention-gate convolution

The YOLOv11 network is widely used in real-time object detection tasks due to its fast detection speed and high accuracy. The diversity of traffic scenes introduces complex visual features into the detection video stream, which leads to limited model recognition features. In contrast, Vision Transformers can comprehensively capture dependencies between image feature blocks and retain sufficient spatial information through their

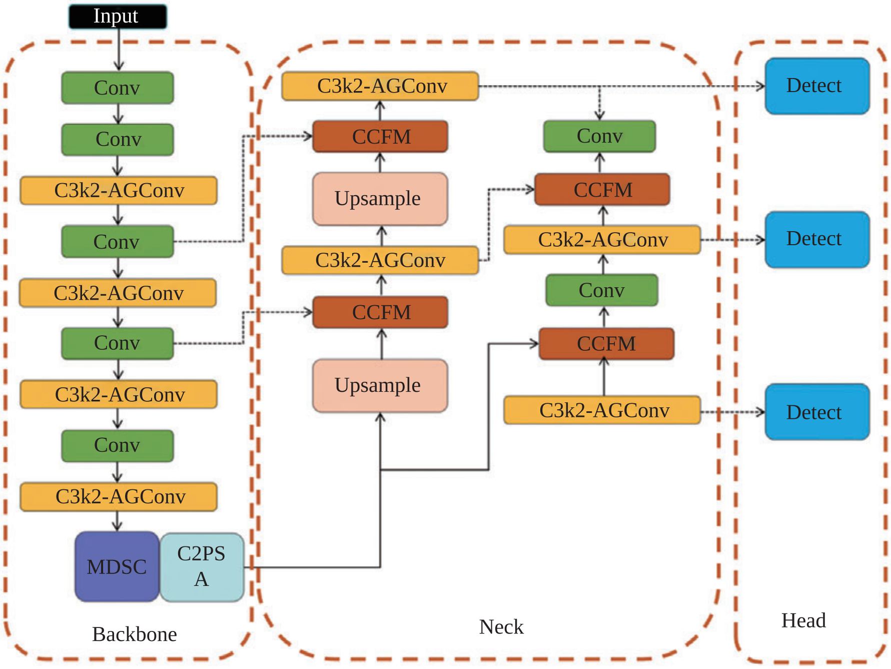

Figure 2 The overall structure of the AMC-YOLO model network proposed in this paper.

unique multihead self-attention mechanism, which is crucial for object detection tasks. To enhance the feature extraction capability and robustness of the model while reducing computational redundancy, we design the attention-gate convolution (AGConv) module, as shown in Fig. 3. The AGConv module is a feature enhancement module composed of a convolutional additive self-attention and a Convolutional gated linear unit. In this work, the feature is first divided into blocks. Then processed in parallel along the channel dimension through the query and key branches for similarity computation, generating attention weights. This method maintains original feature dimensions across branches and reduces computational costs. The operational procedure is described as follows:

$$\mathrm{Sim}(Q,K)=\phi(Q)+\phi(K) \tag{1}$$

Among them, Q, K, and V are obtained through independent linear transformations, such as Q=Wqx, K=Wkx, V=Wvx. where, $\Phi(\cdot)$ represents the context mapping function, which incorporates basic information interactions.

In addition, we incorporate both spatial and channel attention mechanisms to perform hierarchical processing on feature maps, enhancing the model's sensitivity to critical features. The input feature maps

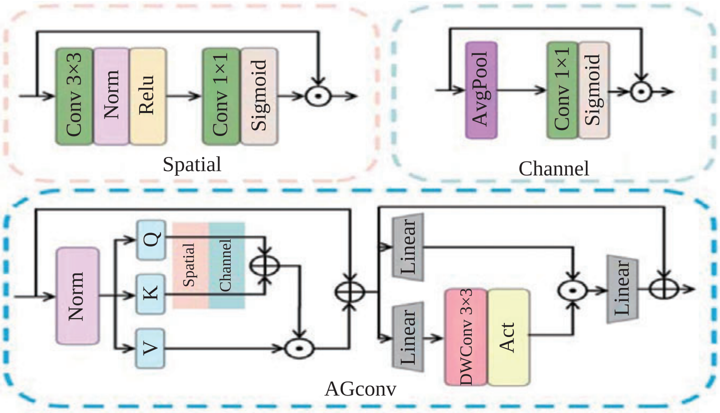

Figure 3 The AGConv structure diagram.

first undergo a 3×3 convolution to extract local spatial characteristics, complemented by normalization. A ReLU activation function is introduced to improve adaptability to complex spatial patterns. For channel dimensionality reduction, a 1×1 convolution is applied to eliminate redundant information. The spatial attention mechanism employs multiple stacked convolutions to progressively abstract spatial information, emphasizing target region saliency through edge and texture enhancement. This is achieved by generating spatial attention weight maps using sigmoid activation that maps outputs to [0, 1]. Meanwhile, the channel attention mechanism applies global average pooling to the input feature map, compressing the spatial dimension and aggregating global information. It then restores information density using a 1×1 convolution. A sigmoid function outputs the channel attention weight vector, identifying channels with significant texture and semantic importance. This method enhances feature representation efficiency by promoting information complementarity among channels.

3.2 C3k2-AGConv

The C3k2 module serves as an efficient and robust feature extraction component in YOLOv11n, employing variable convolution kernels (including 3×3 and 5×5 sizes) combined with a channel separation strategy to enhance feature extraction capabilities. However, under hazy weather conditions, the module exhibits limitations in feature extraction due to complex image backgrounds and substantial variations in target shapes. Since the backbone network's primary function involves extracting both shallow features and global context from raw input images, which is essential for comprehensive scene understanding, we enhance the original C3k2 module by integrating the AGConv component, proposing the improved C3k2-AGConv variant. This enhanced module retains AGConv's advantages in effectively capturing and emphasizing edge details within feature maps while simultaneously aggregating global feature information. In our implementation, we systematically replace all original C3k2 modules in YOLOv11's backbone network with the proposed C3k2AGConv modules, as illustrated in Fig. 4.

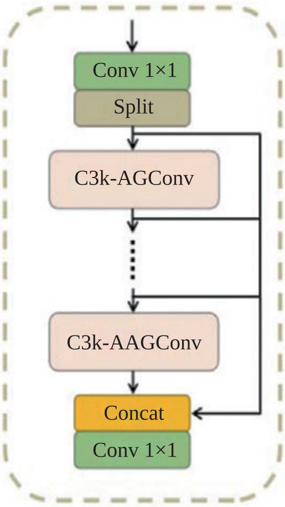

Figure 4 The C3k2-AGConv structure diagram.

The integration of AGConv substantially enhances the C3k2 module's sensitivity to gradient variations, enabling the model to capture more discriminative feature representations. This improvement leads to superior performance in identifying boundaries and fine details of multi-scale targets, particularly in complex scenarios. The processing pipeline operates as follows: Input feature maps first undergo 1×1 convolution to double the channel dimensionality. A channel-wise splitting operation then divides these maps into two parallel branches with equal channels, allowing independent enhancement of edge and texture features. Each branch is subsequently processed by the AGConv module, which comprises two key components: (1) the convolutional additive self-attention mechanism for edge and texture extraction, and (2) the convolutional gated linear unit for effective feature aggregation and representation.

Following feature enhancement, both branches pass through a 1×1 convolutional layer for channel dimension adjustment before being fused with the original feature maps. The resulting output contains significantly enriched edge details and texture information, providing more discriminative inputs for subsequent detection tasks. This split-process-fuse architecture enables the C3k2-AGConv module to independently optimize and combine edge and texture features, ultimately improving multi-scale target detection performance under challenging hazy conditions.

3.3 Multi-dilation shared convolution

Traffic detection targets frequently exhibit significant deformations in video streams or images. Since single convolutional neural network layers possess limited feature representation capacity, robust multi-scale processing capabilities become essential for target detection models. The SPPF module, a crucial component in YOLOv11, effectively captures contextual information through efficient spatial pyramid pooling operations, enabling precise object detection while maintaining high processing speed. However, YOLOv11's strict image size requirements during training lead to partial semantic information loss during pooling operations. Moreover, the SPPF module's structural complexity exceeds that of conventional convolutional layers, imposing greater hardware resource demands.

To overcome these limitations, we propose the MDSC module (Fig. 5). Our design incorporates three 3×3 convolutional kernels with progressively increasing dilation rates (1, 3, 5), systematically expanding the receptive field [19]. As demonstrated in Fig. 6, this configuration enables effective multi-scale feature extraction and broader contextual information capture. Rather than maintaining separate weights for each dilation rate, we implement shared convolutional layers where all three kernels utilize identical weight parameters, significantly reducing trainable parameters while enhancing model efficiency. The module further employs dual 1×1 convolutional kernels for efficient channel dimension adjustment and feature fusion, preserving critical feature information while minimizing parameter overhead.

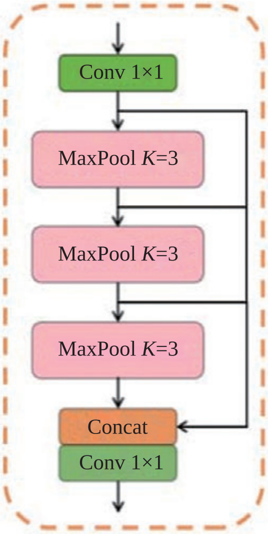

Figure 5 The MDSC model structure diagram.

3.4 Cross-channel fusion module

Under hazy weather conditions, target objects frequently experience occlusion, overlap, or reduced visibility, significantly challenging detection accuracy. The YOLOv11 architecture employs a PANet-based neck network (combining FPN and PAN structures) [20], where the feature pyramid network (FPN) augments shallow features with high-level semantic information through top-down propagation, facilitating multi-scale and small target detection. However, YOLOv11's conventional concatenation-based feature fusion suffers from equal weighting limitations, potentially losing critical semantic information during processing. To resolve this limitation, we propose the CCFM that performs adaptive feature reweighting, emphasizing discriminative features while suppressing less informative ones. As demonstrated in Fig. 7, the CCFM module enhances feature map discriminability through the following computational pipeline [21]:

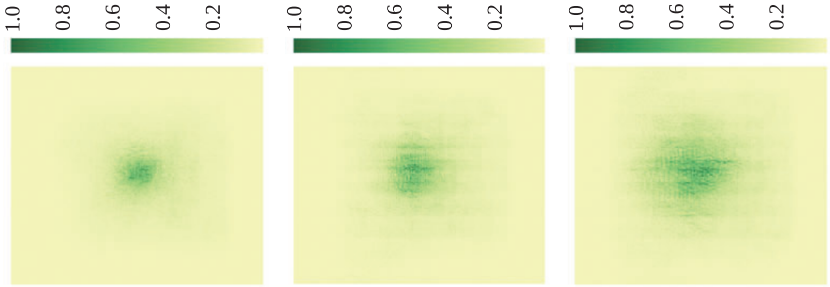

Figure 6 Visualization of effective receptive fields for dilated convolution kernels with varying dilation rates: (a) dilation rate=1, (b) dilation rate=3, and (c)dilation rate=5.

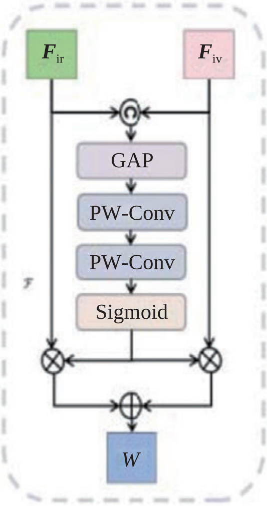

Figure 7 The CCFM model structure diagram.

$$\boldsymbol{F} = \boldsymbol{F}_{\text{ir}} \oplus \boldsymbol{F}_{\text{iv}} \tag{2}$$

Then, feed $\boldsymbol{F}$ into an attention module composed of convolution and pooling operations to generate new attention weights $\hat{\boldsymbol{F}}$. This process effectively preserves the main features of F while suppressing noise. The computational process is as follows:

$$\hat{\boldsymbol{F}} = \text{PW} - \text{Conv}^{n}(\text{GAP}(\boldsymbol{F})) \tag{3}$$

After generating the attention weights, we apply element-wise multiplication to weight the original features. The resulting weighted features are then concatenated to enhance their representation. The computational process is as follows:

$$\hat{\boldsymbol{F}}_{\text{ir}} = \hat{\boldsymbol{F}} \otimes \boldsymbol{F}_{\text{ir}} \tag{4}$$

$$\hat{\boldsymbol{F}}_{\text{iv}} = \hat{\boldsymbol{F}} \otimes \boldsymbol{F}_{\text{iv}} \tag{5}$$

$$W = \delta(\hat{F}_{\text{ir}} \oplus \hat{F}_{\text{vi}}) \tag{6}$$

where, ⊕ denotes element-wise summation, ⊗ represents element-wise multiplication, and PW−Conv^{n} indicates n cascaded point-wise convolutional layers. The operation $C(\cdot)$ performs channel-wise concatenation, while $\delta(\cdot)$ and GAP$(\cdot)$ denote the sigmoid activation function and global average pooling operation, respectively. The CCFM module employs residual connections to strengthen feature information flow, enabling effective integration of detailed spatial features with high-level semantic context. This architecture is specifically designed to guide the model in learning discriminative features for target detection, consequently enhancing the overall detection performance.

4 EXPERIMENTS

4.1 Datasets

The methods and models presented in this paper are trained and evaluated on our customized dataset. The dataset consists of 8792 images collected from road surveillance equipment at different times, capturing scenes such as urban expressways and highways. Due to variations in collection times and weather conditions, the original images exhibit significant differences in noise, brightness, and contrast, ensuring the model's generalization capability. We annotated the bounding boxes of objects in the dataset images using LabelImg in the Microsoft common objects in context (MS COCO) [22] dataset format, with a total of 6 categories: car, truck, pedestrian, bus, motorcycle, and bicycle. The dataset was split into training and validation sets in a 7:2 ratio.

4.2 Experimental environment

The hardware and environment settings used in this experiment are shown in Table 1 below. The graphics processing unit is equipped with an Nvidia GeForce RTX 4070 GPU, featuring 12 GB of dedicated memory to support large-scale parallel computing tasks. To ensure sufficient model convergence while preventing overfitting, the experiment was conducted with 300 training epochs. The batch size was set to 32 to balance computational efficiency and memory usage. During data loading, the CPU utilized 2 worker threads. The

Table 1 The hardware and software environment configuration of this experiment.

Software/hardware	Version
Operating system	Windows 11 professional
Central processing unit (CPU)	Intel Core i7-13790F (2.1GHz)
Graphics card (GPU)	Nvidia GeForce RTX 4070
Memory	12 GB
Programming language version	Python-3.10.14
GPU parallel computing platform	CUDA 12.1
Deep learning framework	Torch-2.2.2+cu121

learning rate was set to 0.01.

4.3 Model results and analysis

In this experiment, five evaluation metrics were selected to measure model performance: precision, recall, mean average precision (mAP), Parameters, and frames per second (FPS). Precision is the most fundamental evaluation metric for object detection models, representing the proportion of correctly predicted samples out of the total samples. Recall indicates the proportion of true positive examples that are correctly predicted as positive by the classifier. The specific expressions are as follows:

$$P = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}} \tag{7}$$

$$R = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}} \tag{8}$$

Here, TP represents the number of true positives, FP represents the number of false positives, and FN represents the number of false negatives. The mAP (i.e., mean average precision) value is applicable to various types of datasets, including custom datasets. It comprehensively evaluates the model's performance by calculating the average area under the precision-recall curves for all categories. The specific expression is as follows:

$$\mathrm{AP} = \int_0^1 P(R)\mathrm{d}r \tag{9}$$

$$\mathrm{mAP} = \frac{\sum_{i=1}^{k} AP_i}{k_{\mathrm{classes}}} \tag{10}$$

IoU represents the ratio of the intersection to the union of the model's predicted bounding box and the ground truth bounding box, and it is used to determine the accuracy of the bounding boxes in object detection models. The mAP@0.5 and mAP@0.5:0.95 are commonly used to evaluate the overall performance of a model. mAP@0.5 is the mean average precision calculated with the intersection over union (IoU) threshold set to 0.5, while mAP@0.5:0.95 is the average mAP computed across different IoU thresholds ranging from 0.5 to 0.95 with a step size of 0.05.

Parameters refer to the adjustable weights learned by the model during training. These weights control the model's response to input data, determining its capabilities, and are a key indicator of the complexity of deep learning models. Generally, the more complex the model, the larger the number of parameters, which means the model can store and process more information. However, this also requires more computational resources and training time. Therefore, balancing model complexity and performance is crucial.

FPS refers to the number of images the model can process per unit of time, typically expressed as frames per second. This metric is particularly important in real-time processing scenarios such as video surveillance and autonomous driving. A higher FPS value indicates stronger real-time responsiveness of the model.

We conducted comprehensive ablation experiments to further analyze the effects of each improvement. Tables 2 and 3 show the changes in precision for the six target categories and the overall performance feedback under different enhancements. The C3k2 module is a critical feature extraction module in the YOLOv11 model.

4.4 C3k2-AGCov Comparative Experiments

To thoroughly evaluate the effectiveness of our proposed C3k2-AGConv module, we conducted

Table 2 The comparison of different modifications to C3k2 module.

Model	P_{all}	R_{all}	mAP@0.5	mAP@0.5:0.95	Params (M)	FPS (GPU)
YOLOv11n	0.882	0.832	0.901	0.754	2.58	322.6
YOLOv11n-C3k2-faster	0.889	0.825	0.898	0.732	2.28	348.0
YOLOv11n-C3k2-faster(EMA)	0.887	0.825	0.901	0.765	2.30	341.5
YOLOv11n-C3k2-DBB	0.891	0.828	0.902	0.756	2.6	317.8
YOLOv11n-C3k2-AGConv	0.908	0.839	0.909	0.768	2.45	384.1

Table 3 The ablation experiment of the AGConv module.

Model	P_{all}	R_{all}	mAP@0.5	mAP@0.5:0.95	Parameters (M)	FPS (GPU)
YOLOv11n	0.882	0.832	0.901	0.754	2.58	322.6
YOLOv11n-C3k2-CAS	0.906	0.835	0.908	0.761	2.62	337.0
YOLOv11n-C3k2-AGconv	0.908	0.839	0.909	0.768	2.45	384.1

systematic architectural modifications to the YOLOv11 baseline and performed comparative experiments with alternative enhancement approaches. Our investigation focused on improving the Bottleneck component within the C3k2 module through three distinct methodologies.

The diverse branch block (DBBNet) approach implements a multi-branch architecture with heterogeneous receptive fields, combining convolutional sequences, multi-scale convolutions, and average pooling to enrich feature representation. While this method achieved a marginal improvement in mAP@0.5:0.95 from 0.901 to 0.902 with enhanced Recall and Precision metrics, it incurred increased computational costs reflected in reduced FPS performance.

The partial convolution (FasterNet) strategy employs selective channel processing to reduce computational overhead while maintaining original channel dimensionality. Experimental results demonstrated an 11.6% reduction in parameters (from 2.58 M to 2.28 M) and improved FPS, though at the cost of decreased mAP@0.5 and mAP@0.5:0.95 performance. Subsequent attempts to incorporate EMA attention mechanisms yielded only limited improvements.

Our proposed AGConv method integrates multi-head self-attention with convolutional gating mechanisms to enable parallel learning of complementary attention patterns across kernel space dimensions. This approach delivered comprehensive performance enhancements, including a 2.6% increase in Precision (0.882 to 0.908), 1.4% improvement in mAP@0.5:0.95 (0.754 to 0.768), and significant FPS gains, while simultaneously reducing parameter redundancy and ineffective computations. Based on these superior results, we adopted AGConv as our optimal enhancement strategy, naming the improved module C3k2AGConv. The complete comparative experimental data are presented in Table 2.

4.5 Ablation experiment

4.5.1 Improved strategy ablation

This paper presents the AGConv module as an enhancement to YOLOv11's feature extraction capability through the replacement of the original C3k2 module with our proposed C3k2-AGConv variant. Comprehensive ablation studies compared the C3k2-AGConv against both the baseline C3k2 module and a C3k2-CAS variant (lacking the CGLU component), with results detailed in Table 3. The experimental data reveal that while the C3k2-CAS module improves model accuracy and robustness, confirming the effectiveness of multi-head self-attention mechanisms for inter-layer information flow, it incurs significant parameter inflation. In contrast, the C3k2-AGConv module achieves superior performance across all accuracy metrics while actually reducing parameter counts below the original model's level. This demonstrates that integrating convolutional gating units with self-attention mechanisms enables efficient dimensionality reduction and feature reorganization, preserving critical information without increasing computational overhead.

To assess the MDSC module's multi-scale detection capability, we evaluated three dilation rate configurations ([1,1,1], [3,3,3], [5,5,5]) against the original YOLOv11, as presented in Table 4. The [1,1,1] variant improved P_{all} from 0.882 to 0.889 and R_{all} from 0.832 to 0.838, validating dilated convolution's ability to capture fine details while maintaining essential information. However, higher uniform dilation rates showed diminishing returns, with [3,3,3] and [5,5,5] configurations yielding negligible gains or performance degradation. Significantly, the mixed-rate [1,3,5] scheme achieved optimal results: mAP@0.5 increased to 0.906 and mAP@0.5:0.95 reached 0.762, with across-the-board metric improvements.

Further ablation studies comparing CCFM with alternative feature pyramid structures (Table 5)

Table 4 Comparison of different dilation rates in the MDSC module.

Model	Dilation rate	P_{all}	R_{all}	mAP@0.5	mAP@0.5:0.95
YOLOv11n	[0,0,0]	0.882	0.832	0.901	0.754
YOLOv11n-MDSC	[1,1,1]	0.889	0.836	0.902	0.744
YOLOv11n-MDSC	[3,3,3]	0.885	0.833	0.902	0.752
YOLOv11n-MDSC	[5,5,5]	0.886	0.833	0.903	0.750
YOLOv11n-MDSC	[1,3,5]	0.898	0.845	0.906	0.762

Table 5 Comparison of different feature pyramid networks.

Model	Network	P_{all}	R_{all}	mAP@0.5	mAP@0.5:0.95
YOLOv11n	FPN	0.882	0.832	0.901	0.754
YOLOv11n	PAFPN	0.893	0.838	0.902	0.753
YOLOv11n	BIFPN	0.891	0.833	0.901	0.746
YOLOv11n	CCFM	0.905	0.847	0.908	0.768

demonstrated consistent performance gains over the baseline FPN. Notably, CCFM outperformed both PAFPN and BIFPN, confirming its superior capability in fusing shallow and deep features through cross-channel integration and adaptive reorganization. This enriched semantic representation enables precise multi-scale target detection. In conclusion, our AGConv, MDSC, and CCFM modules collectively represent optimal solutions for balanced model optimization, demonstrating superior detection performance, computational efficiency, and overall system balance.

4.5.2 Model global ablation

We conducted comprehensive ablation experiments to systematically analyze the contributions of each improvement. Tables 6 and 7 present the precision changes for six object categories and overall performance metrics under different modifications. Module with the C3k2-AGConv, the results indicate that the overall mAP@0.5 increased from 0.901 to 0.909, and mAP@0.5:0.95 rose from 0.754 to 0.768 compared to the baseline model (Table 5). Additionally, the precision for trucks improved from 0.825 to 0.852, for buses from 0.854 to 0.882, and for motorcycles from 0.868 to 0.918 (Table 6). These results demonstrate that for traffic object detection in large-scale and complex scenarios, the self-attention mechanism can capture global contextual relationships in images through neighborhood learning, enhancing semantic discriminability and mitigating category confusion.

When we replaced the SPPF module with the MDSC module, mAP@0.5 increased from 0.901 to 0.906,

Table 6 The ablation experiments on the comparative results of six categories.

C3k2-AGConv	MDSC	CCFM	Car	Truck	Person	Bus	Motor	Bicycle	P_{all}	R_{all}
-	-	-	0.912	0.825	0.931	0.854	0.868	0.905	0.882	0.832
√	-	-	0.909	0.852	0.933	0.882	0.918	0.904	0.908	0.839
-	√		0.921	0.831	0.937	0.912	0.893	0.899	0.898	0.845
-	-	√	0.922	0.853	0.942	0.902	0.894	0.919	0.905	0.847
√	√		0.915	0.850	0.940	0.908	0.908	0.877	0.905	0.836
√	-	√	0.928	0.863	0.948	0.916	0.910	0.925	0.910	0.838
-	√	√	0.924	0.846	0.945	0.915	0.901	0.916	0.903	0.847
√	√	√	0.937	0.860	0.953	0.929	0.931	0.924	0.923	0.847

Table 7 The ablation experiments on collaborative effects of AMC-based Multi-module Components.

C3k2-AGConv	MDSC	CCFM	mAP@0.5	mAP@0.5:0.95	Params (M)	FPS (GPU)
-	-	-	0.901	0.754	2.7	322.6
√	-	-	0.909	0.768	2.5	384.1
	√	-	0.906	0.762	2.4	360.5
-	-	√	0.908	0.768	2.6	373.8
√	√	-	0.907	0.773	2.5	380.7
√	-	√	0.911	0.770	2.6	359.4
-	√	√	0.909	0.766	2.8	312.5
√	√	√	0.912	0.781	2.6	376.2

and mAP@0.5:0.95 rose from 0.754 to 0.762 (Table 7), while the number of parameters decreased by 0.3 M. Under the condition of keeping other model structures unchanged, we used the SPP, SPPF, and MDSC modules as the feature pyramid pooling modules of the model, respectively, and calculated their receptive field sizes, as shown in Table 8. The results in Table 8 show that setting the dilation rate significantly expands the effective receptive field of the convolutional kernel, thus enhancing its ability to capture global contextual information.

4.6 Comparative experiments

To evaluate the feasibility, authenticity, and robustness of the ACM-YOLO model, we trained and tested the ACM-YOLO model alongside other popular object detection models on our proposed custom dataset [23–25]. The experimental results of Table 9 show that the YOLO series exhibits significant performance advantages compared to current state-of-the-art methods. The parameter count and detection frame rate of YOLO models are significantly lower than those

Table 8 The comparison of receptive field size.

	t=20%	t=30%	t=50%	t=99%
SPP	1.5%	2.8%	7.4%	61.78%
SPPF	1.6%	2.9%	7.4%	91.2%
MDSC	2.7%	4.7%	10.6%	92.3%

Table 9 The comparison of different modifications.

Model	P_{all}	R_{all}	mAP@0.5	mAP@0.5:0.95	Params (M)	FPS (GPU)
SSD	0.862	0.737	0.880	0.745	24.1	186.2
Faster R-CNN	0.881	0.805	0.875	0.695	41.3	57.9
Cascade R-CNN	0.885	0.810	0.896	0.764	61.7	53.3
RT-DETR-ResNet50	0.879	0.798	0.878	0.763	61.5	97.3
YOLOv5n	0.864	0.778	0.870	0.682	2.64	425.6
YOLOv7	0.872	0.744	0.861	0.695	37.2	253.8
YOLOv7-tiny	0.868	0.789	0.873	0.737	6.0	312.4
YOLOv8n	0.860	0.796	0.875	0.703	3.1	344.8
YOLOv10n	0.830	0.766	0.854	0.665	2.3	290.3
YOLOv11 (Baseline)	0.882	0.832	0.901	0.754	2.6	322.6
ACM-YOLO (ours)	0.923	0.847	0.912	0.781	2.7	376.2

of two-stage object detection models such as SSD and Faster R-CNN, while maintaining model accuracy and robustness. Notably, compared to the baseline model YOLOv11, our proposed ACM-YOLO model improved overall precision and recall from 0.882 and 0.832 to 0.923 and 0.847, respectively. The mAP@0.5 and mAP@0.5:0.95 reached 0.912 and 0.781, representing improvements of 1.1% and 3.3%, respectively, demonstrating a significant enhancement in overall performance.

4.7 Visualization analysis

To validate the superior feature extraction capability of the proposed improved AMC-YOLO model in this paper, we visualized the heatmaps and detection results of the YOLOv11n and AMC-YOLO models across different scenarios. These visual feature maps represent the density or magnitude of values in the data through color mapping. As shown in Fig. 8, compared to YOLOv11n, AMC-YOLO displays a broader range of bright colors, indicating enhanced feature representation and

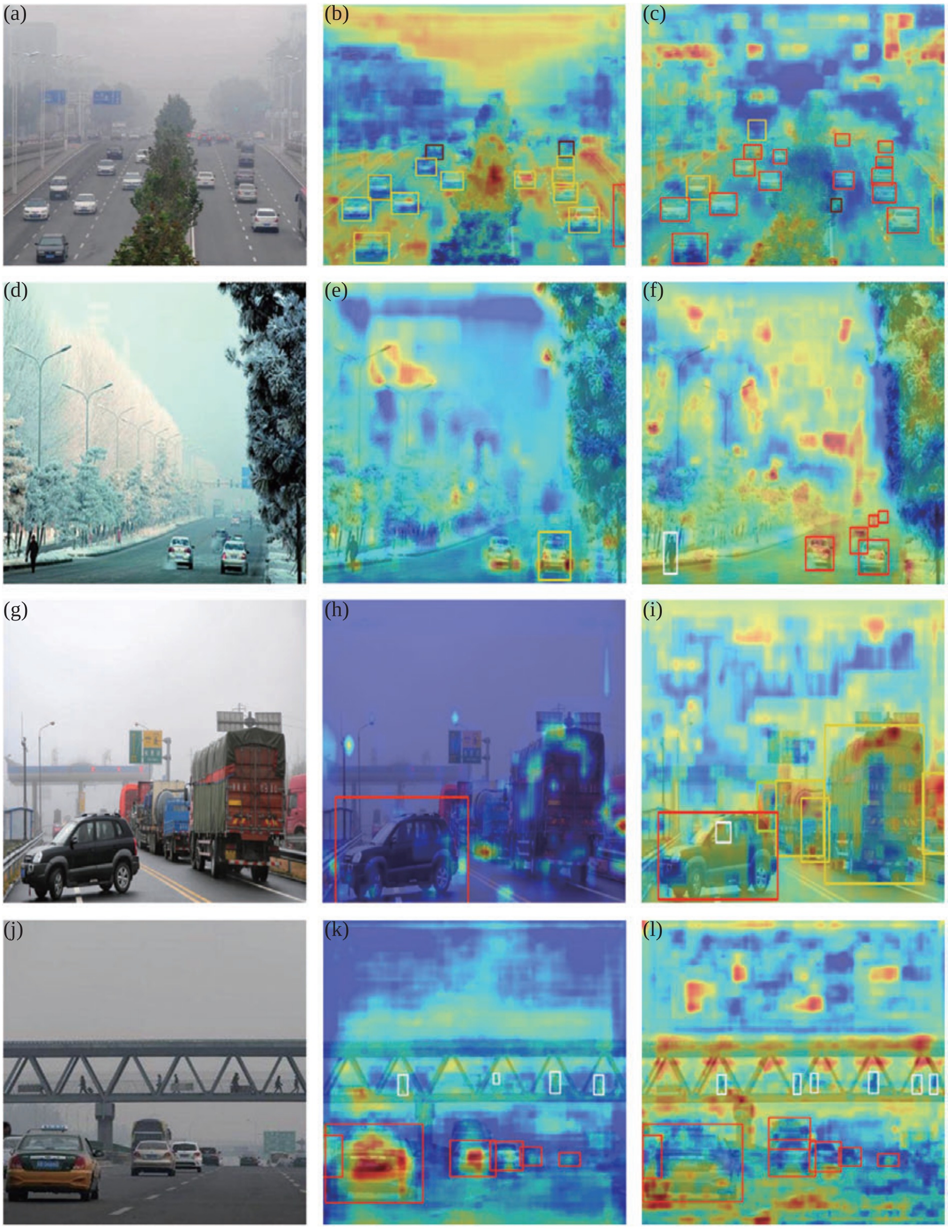

Figure 8 The visualization analysis of different models: (a) telephoto: original, (b) telephoto: YOLOv11n, (c) telephoto: AMC-YOLO, (d) shortphoto: original, (e) shortphoto: YOLOv11n, (f) shortphoto: AMC-YOLO, (g) ImgDef: original, (h) ImgDef: YOLOv11n, (i) ImgDef: AMC-YOLO, (j) UltraSP: Original, (k) UltraSP: YOLOv11n, and (l) UltraSP: AMC-YOLO.

reflecting the model's attention to image regions. From various perspectives, the robustness and accuracy of the AMC-YOLO model are significantly higher than those of YOLOv11n. In high-angle views, YOLOv11n misclassifies a car as a truck, while AMC-YOLO correctly distinguishes between cars and trucks. In normal-angle views, the AMC-YOLO model can detect targets at greater distances. In low-angle views, AMC-YOLO demonstrates stronger detection capabilities for heavy objects and accurately detects pedestrians on overpasses. In nighttime scenarios, AMC-YOLO still achieves accurate multi-target recognition. Therefore, the improved AMC-YOLO model proposed in this paper exhibits better feature extraction capabilities and stronger robustness.

5 Conclusion

Under haze weather conditions, images captured by road surveillance equipment exhibit overwhelming characteristics, including significant scale variations and complex backgrounds filled with distractors, posing substantial challenges for general object detectors based on conventional convolutional networks. This paper proposes an improved traffic object detection algorithm, AMC-YOLO, based on YOLOv11. Experimental results demonstrate that AMC-YOLO achieves higher accuracy and generalization capabilities across diverse traffic scenarios while excelling in model parameter count, computational load, detection accuracy, and speed. It significantly enhances the reliability of traffic detection under haze weather conditions. Furthermore, the model achieves a reasonable balance between lightweight design and high precision, providing robust support for deploying traffic detection systems on roadside devices. Although AMC-YOLO has achieved notable results, its performance can be further improved through new model architectures, optimization algorithms, and hardware advancements. With further research, future applications could expand to real-time tracking, accident identification, and other areas to enhance traffic safety.

Acknowledgments

This paper is supported by the Wuhan Pilot construction of a strong Transportation Country Science and Technology Joint Research Project (No. 2024-1-10).

References

[1] X. Zhao, L. Wang, Y. Zhang, X. Han, M. Deveci, and M. Parmar, "A review of convolutional neural networks in computer vision," *Artificial Intelligence Review*, vol. 57, no. 4, p. 99, 2024.

[2] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, H, USA, 23–28 June 2014, pp. 580–587.

[3] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 37, no. 9, pp. 1904–1916, 2015.

[4] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: Object detection via region-based fully convolutional networks," 30th Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain, 5–10 December 2016, pp. 379–387.

[5] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," 2016 IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016, pp. 779–788.

[6] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot multibox detector," Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, 11–14 October 2016, Part I 14, Springer, 2016, pp. 21–37.

[7] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Doll´ar, "Focal loss for dense object detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, pp. 2980–2988, 2018.

[8] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in Neural Information Processing Systems*, vol. 30, pp. 6000–6010, 2017.

[9] T. Zhang, L. Li, Y. Zhou, W. Liu, C. Qian, J.-N. Hwang, and X. Ji, "CAS-ViT: Convolutional additive self-attention Vision Transformers for efficient mobile applications," *arXiv preprint arXiv:2408.03703*, 2024.

[10] D. Shi, "Transnext: Robust foveal visual perception for Vision Transformers," 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, USA, 17–21 June 2024, pp. 17773–17783.

[11] J. Redmon and A. Farhadi, "Yolo9000: better, faster, stronger," IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017, pp. 7263–7271.

[12] N. Li, M. Wang, G. Yang, B. Li, B. Yuan, and S. Xu, "DENS-

YOLOv6: A small object detection model for garbage detection on water surface," *Multimedia Tools and Applications*, vol. 83, no. 18, pp. 55751–55771, 2024.

[13] J. Deng, W. Dong, R. Socher, L. -J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009, pp. 248–255.

[14] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with Transformers," in *European Conference on Computer Vision*. Springer, 2020, pp. 213–229.

[15] S. Cui and H. Deng, "PMG-DETR: Fast convergence of DETR with position-sensitive multi-scale attention and grouped queries," *Pattern Analysis and Applications*, vol. 27, no. 2, p. 58, 2024.

[16] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin Transformer: Hierarchical Vision Transformer using shifted windows," 2021 IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021, pp. 10 012–10 022.

[17] T.-Y. Lin, P. Doll'ar, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," 2017 IEEE Conference on Computer Vision Venice, Italy, 22–29 October 2017, pp. 2117–2125.

[18] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," 2018 IEEE conference on computer vision and pattern recognition, Salt Lake City, UT, USA, 18–23 June 2018, pp. 7132–7141.

[19] P. Wang, P. Chen, Y. Yuan, D. Liu, Z. Huang, X. Hou, and G. Cottrell, "Understanding convolution for semantic segmentation," 2018 IEEE Winter Conference on Applications of Computer Vision (WACV). Salt Lake City, UT, USA, 18–23 June 2018, pp. 1451–1460.

[20] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," 2018 IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018, pp. 8759–8768.

[21] L. Tang, H. Zhang, H. Xu, and J. Ma, "Rethinking the necessity of image fusion in high-level vision tasks: A practical infrared and visible image fusion network based on progressive semantic injection and scene fidelity," *Information Fusion*, vol. 99, p. 101870, 2023.

[22] N. Yin, L. Shen, M. Wang, L. Lan, Z. Ma, C. Chen, X.-S. Hua, and X. Luo, "COCO: A coupled contrastive framework for unsupervised domain adaptive graph classification," 40th International Conference on Machine Learning. Honolulu Hawaii, USA, 23–29 July 2023, pp. 40 040–40 053.

[23] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2016.

[24] Z. Cai and N. Vasconcelos, "Cascade r-CNN: High quality object detection and instance segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 5, pp. 1483–1498, 2019.

[25] Y. Zhao, W. Lv, S. Xu, J. Wei, G. Wang, Q. Dang, Y. Liu, and J. Chen, "DETRS beat yolos on real-time object detection," 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, USA, 17–21 June 2024, pp. 16965–16974.