IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 20 April 2024, accepted 1 July 2024, date of publication 16 July 2024, date of current version 26 July 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3429282

RESEARCH ARTICLE

LTEA-YOLO: An Improved YOLOv5s Model for Small Object Detection

BO LI[1,2], **SHENGBAO HUANG**[3], **AND GUANGJIN ZHONG**[3]

[1]Faculty of Applied Sciences, Macao Polytechnic University, Macau, SAR, China
[2]College of Mechatronics Engineering, Zhongshan Institute, University of Electronic Science and Technology of China, Zhongshan, Guangdong 528402, China
[3]School of Mechanical and Electrical Engineering, University of Electronic Science and Technology of China, Chengdu, Sichuan 611731, China

Corresponding author: Bo Li (libo@zsc.edu.cn)

This work was supported in part by Guangdong Provincial Department of Education Fund Project under Grant 2021ZDZX4033, and in part by Zhongshan Public Science and Technology Research Project under Grant 2023B2007.

ABSTRACT Small target information has a lower proportion and severe background interference in the image, which significantly restrains the performance of small object detection algorithms. Most detection models today have a large size, making them unsuitable for deployment on mobile terminals. Based on YOLOv5s, we proposed a light-weight model, LTEA-YOLO, with a model size of only 13.2MB, which has a Light-weight Transformer and Efficient Attention mechanism for small object detection. Firstly, a new light-weight Transformer module, called the inverted Residual Mobile Block (iRMB), is employed as a back-bone network to extract features. Secondly, we created a DBMCSP module (Diverse Branch Modules are inserted into Cross-Stage Partial network), which takes the place of all $C3$ modules in the fusion section, to extract a wider range of feature information without compromising the speed of inference. We then employ $WIoU_{v3}$ as the loss function of box regression to accelerate training convergence and improve positioning precision. Finally, we developed a light-weight and efficient Coordinate and Adaptive Pooling Attention (CAPA) module, which performs better than the Coordinate Attention (CA) module, to be embedded into the SPPF module to enhance detection accuracy. Our model gets 97.8% at mAP@0.5 on the NWPU VHR-10 dataset, which is 3.7% better than YOLOv8s and 6% better than the baseline model YOLOv5s-7.0. In experiments with the VisDrone 2019 dataset, its mAP@0.5 reached 35.8%, outperforming other comparison models. Our LTEA-YOLO, with its small model size, demonstrates superior overall performance in detecting challenging small objects.

INDEX TERMS Attention mechanism, lightweight transformer, YOLOv5, small object detection.

I. INTRODUCTION

A small target is a specific area inside a picture that occupies just a small proportion of the overall image. Generally, there are two commonly accepted definitions for a small target. One is defined by the perspective as a whole. A target is considered small if it is less than one-tenth the size of the image. The other is distinguished by a regional viewpoint. If the target covers fewer than 32 × 32 pixels, it can be considered a small target. Small target identification technology has significantly increased in several fields, such as remote sensing, insect pest detection in agriculture, and the operation of unmanned aerial vehicles. Deep learning technology advancements are responsible for this growth. Due to the minimal zone, poor resolution, and inadequate feature expression in the image, detecting small targets is more challenging than identifying general objects, and the resulting detection effects are not ideal. Hence, a pivotal inquiry in the realm of computer vision pertains to the precise identification and positioning of small objects.

Recently, researchers have developed numerous ways to address the difficulty of detecting small targets. These solutions are primarily based on two-stage and single-stage target recognition algorithms. Two-stage target detection algorithms first detect the background and target, preserve the information area of the target, and detect the target instance

The associate editor coordinating the review of this manuscript and approving it for publication was Rajeeb Dey.

and category. These algorithms exhibit great accuracy but lack sufficient detection speed. Representative algorithms in this category include Faster-RCNN [1] and others. Single-stage target detection algorithms have the ability to directly detect the category and position of the target. These algorithms are known for their speed, however, they may sacrifice a certain level of accuracy. YOLO [2] series and SSD [3] series are the representative algorithms. In this paper, we proposed a lightweight and effective small target detection algorithm based on YOLOv5s-7.0 [4], which is called LTEA-YOLO (Lightweight Transformer and Efficient Attention mechanism), aiming to further increase the accuracy of small target detection and lighten the model size in order to deploy it to embedded mobile terminals with insufficient computing resources. The following are its significant innovations and contributions:

- The iRMB [5], a revolutionary lightweight Transformer module, is efficiently stacked in the backbone network. Its sequence-to-sequence properties can better extract information about obstructed and crowded objects, improving model detection effectiveness.
- The DBMCSP (Diverse Branch Modules are inserted into Cross-Stage Partial network) module serves as a replacement for all $C3$ modules in the feature fusion section. A multi-branch structure extracts a greater quantity of information during training. DBMCSP can be integrated into a convolution block during inference, improving the algorithm's overall performance without affecting the inference time.
- By replacing the CIoU regression loss function with the latest $WIoU_{v3}$ [6] function, the training convergence becomes quicker and the algorithm's performance is improved.
- Create a lightweight and efficient attention mechanism known as CAPA and integrate it into the SPPF module. For simplicity, we denote it as SPPFA in the network structure diagram.

II. RELATED WORK

Koyun et al. [7] proposed a two-stage target detection architecture that produces target clusters using a Gaussian mixture model before accessing the target detection network, and it detects tiny targets in aerial images well. Zhou et al. [8] balanced throughput and computational speed while improving the ability of the network to identify small objects by using a novel backbone called "RepDarkNet" and a multi-scale cross-layer detector. Kang et al. [9] developed an alignment matching strategy based on the SSD model that took into consideration not only IoU but also aspect ratio and central focal length, solving the problem of the model's poor efficiency in learning to detect small objects, and increasing its mAP by 60%. Jiao et al. [10] designed a feature pyramid network combining attention mechanism with feature fusion factors and a soft weighted loss function that can focus on shallow small objects adaptively and reduce false positives, but the network missed some tiny and fuzzy objects. Li et al. [11] applied an adaptive multi-feature fusion algorithm for small target detection based on YOLOv3. Its mAP is 2.6% higher than that of YOLOv3, but the number of parameters has increased by 13%. Liu et al. [12] used YOLOv5, incorporating FEBlock and SCEP networks, to detect small objects with complicated backgrounds; however, the number of parameters and computations in the model grows significantly. Based on SSD, Zhang et al. [13] presented a recursive attention-enhanced bidirectional feature pyramid network (RA-BiFPN). Although the mAP value has greatly improved, the model has become more complex. Liu et al. [14] enhanced the prediction headers using YOLOv5 and stacked residual transformer and residual attention modules. Compared to YOLOv5, mAP rises by 14.5%, decreasing the complexity of the network. In order to detect construction site employees wearing helmets, Liang et al. [15] developed a single-stage high-precision attention-weighted fusion network with a high-precision Swin Transformer module as the backbone network. Zhu et al. [16] used YOLOv5 to create a micro-target detector that used predictive perceptual loss and extended feature pyramid-enhanced representation to find micro-targets in UAV pictures. By integrating a detector and an attention mechanism, Zhao et al. [17] adopted the YOLOv7-SEA detector to identify tiny objects at sea, resulting in a 7% improvement in AP compared to YOLOv7. Liu et al. [18] combined Coordinate Attention and mixed dilated convolution into YOLOv5 for small target detection in remote sensing, and the mAP increased by 8.1%, which significantly improved the model performance.

The aforementioned research has made significant contributions to the field of small target detection. However, the detection models employed are relatively large, making them unsuitable for use on mobile edge devices. In the next section, we introduce our lightweight detection model.

III. PROPOSED METHODS

This section first introduces the benchmark model YOLOv5s-7.0, then describes the framework of our improved algorithm, and finally details the contents of the four improved components.

YOLOv5 is one of the simplest and most efficient single-stage target detection models, with high reliability and easy deployment. Its detection idea is to divide the input image into grids, and when there is target information in the center of the grid, it is responsible for detecting the target. YOLOv5 has five kinds of models, including YOLOv5n, YOLOv5s, YOLOv5m, YOLOv5l, and YOLOv5x, among which the YOLOv5s network is the second-smallest in YOLOv5 series. Since the smallest model, YOLOv5n, has poor information extraction ability for dense and small targets, we chose YOLOv5s as the benchmark model, which is conducive to achieving a good balance between lightweightness and accuracy in the network model. The latest version of YOLOv5s, YOLOv5s-7.0, features a classic single-stage network structure, divided into three parts: the

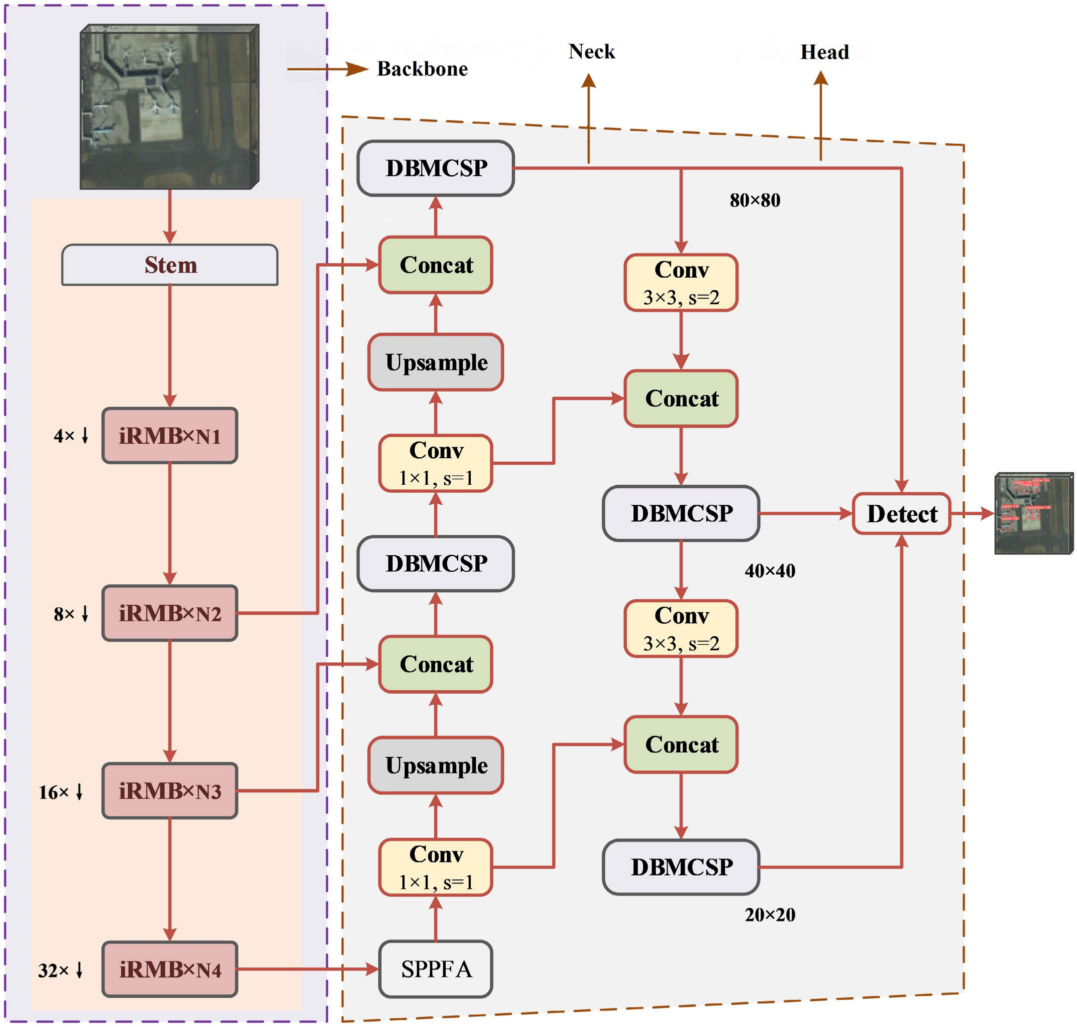


FIGURE 1. The overall framework of LTEA-YOLO.

backbone, neck, and head. To facilitate efficient model training and reasoning, the image in the training dataset needs to be processed to enhance the data complexity using the Mosaic data enhancement method and resized to 640 × 640. The backbone mainly consists of Conv (convolution + standardization + activation function) and $C3$ (a cross-stage partial network) modules, which is used to extract image features. To improve feature representation, the neck adopts the FPN (feature pyramid network) structure, which enables top-down and bottom-up fusion of feature layers of various resolutions. And the detection head utilize three $C3$ modules, processing 80 × 80, 40 × 40, and 20 × 20, respectively, to detect small, medium, and large targets. Fig. 1 presents the overall structure of our LTEA-YOLO model in this paper. Firstly, we introduce the effective stacking of the iRMB as a replacement for the YOLOv5s-7.0 backbone network. Secondly, in the neck fusion phase, our improved DBMCSP modules replace the $C3$ modules. Fig. 1 does not show the loss function, but we can see it in Section C. Finally, we design the CAPA attention mechanism, which is lightweight and effective, and integrate it with SPPF to generate the SPPFA module.

A. IRMB MODULE

Since ViT [19] first introduced Transformer to the object detection field, developers have developed a great deal of object detection methods based on Transformer, such as SMCA [20], Swin Transformer [21], and DESTR [22]. Because of its sequence-to-sequence nature, the Transformer-based object detection framework easily learn target information. To extract rich information from occluded and dense targets while keeping the model lightweight, we use iRMB, a lightweight transformer module, to efficiently stack the model into the backbone network to extract features. Fig. 2 shows its structure, which includes a convolution block with normalization and activation functions. To expand the channel, we first apply a channel expansion factor (take $\lambda = 4$). Then, using the functions of the convolutional neural network to extract local features and the transformer to learn global information, we use

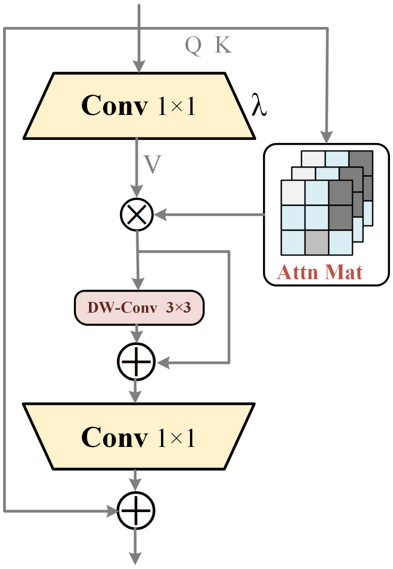


FIGURE 2. The structure of inverted residual mobile block (iRMB).

efficient window-MHSA (multi-head self-attention), namely EW-MHSA, depth-separable convolution (DW-Conv [23]), and skip connections to extract effective information from feature maps. DW-Conv binds to BN [24] + SiLU [25], whereas EW-MHSA attaches to LN [26] + GeLU [27]. By adjusting the step size of DW-Conv, we can accomplish down-sampling without introducing the position embedding module. We then use the channel shrinkage factor as a reciprocal to ensure that the number of output channels matches the number of input channels. Finally, we add the output of the feature extraction module to the parallel branch of the identity mapping to obtain the final output. To reduce the computation, take $Q = K = X(\in R^{C\times H\times W})$, $X_e = V(\in R^{\lambda C\times H\times W})$, and the whole module can be expressed as:

$$F(*) = (DW\text{-}Conv, Skip)(EW\text{-}MHSA(*)) \tag{1}$$

The iRMB architecture incorporates depth-wise convolution and multi-head self-attention (MHSA), which are both highly effective and flexible. Because MHSA mechanism is more suitable for deep semantic feature extraction, we just use it in the 3rd and 4th stages of the backbone network with a reference to [5] when we extract features with the iRMB architecture. The amounts of iRMB stacked in N1, N2, N3, and N4 are 3, 3, 9, and 3, respectively.

B. DBMCSP STRUCTURE

The parallel substructure helps the neural network avoid greater depth while enhancing the variety of semantic information. This paper uses the Diverse Branch Block (DBB) [28], RepVGG [29], and the $C3$ modules of YOLOv5s-7.0 to create a DBMCSP (Diverse Branch Modules are inserted into Cross-Stage Partial network) module. Fig. 3 shows that on one path, the n Diverse Branch Modules (DBM) follow a Conv-BN-SiLU block. The output concatenates with another Conv-BN-SiLU block from the other path. At the end of the DBMCSP, the cascaded information from the two branches passes through another Conv-BN-SiLU block for information integration. The structure of DBM in Fig. 3 differs between the training stage and the inference stage. During the training, DBM consists of four branches: 1 × 1 convolution-batch normalization, 1 × 1 convolution-batch normalization + $k \times k$ convolution-batch normalization, 1 × 1 convolution-batch normalization + average pooling-batch normalization, and $k \times k$ convolution-batch normalization; the output of each branch is added at the end. The number of intermediate channels in 1 × 1 convolution and $k \times k$ convolution is equal to the number of input channels. During training, DBM can extract more diverse information and improve model performance through different branch structures. In the inference stage, however, a single convolution structure, whose parameters come from the average combination of all branches' parameters in DBM, replaces DBM, ensuring processing speed. The neighboring region can easily overwrite the small target's limited information during the convolution operation of the $C3$ module of the original YOLOv5s 7.0. Therefore, we replaced all $C3$ modules of the YOLOv5s 7.0 architecture with the DBMCSP module so that the model could extract more detailed information, thus enhancing its performance to detect small targets.

C. LOSS FUNCTION

The loss function in the YOLOv5 algorithm consists of classification loss, location regression loss, and confidence loss, where location loss determines the positioning performance of detection results. To speed up training convergence, we introduce the latest $WIoU_{v3}$ [6] as the regression loss function of the detection box position, replacing the original $CIoU$. Fig. 4 illustrates the relationship between the ground truth box and the predicted box. The ground truth box is on the upper left, and the prediction box is on the lower right. According to Fig. 4, the relevant calculation equation for $WIoU_{v3}$ can be found below.

The union area of the ground truth box and the predicted box is S_u:

$$S_u = wh + w_{gt}h_{gt} - W_iH_i \tag{2}$$

And the IoU loss function is obtained as follows:

$$L_{IoU} = 1 - IoU = 1 - \frac{W_iH_i}{S_u} \tag{3}$$

Then R_{WIoU} is deduced:

$$R_{WIoU} = \exp\left(\frac{(x - x_{gt})^2 + (y - y_{gt})^2}{W_g^2 + H_g^2}\right) \tag{4}$$

Where $R_{WIoU} \in [1, e)$, $L_{IoU} \in [0, 1]$, then construct the $L_{WIoU_{v1}}$ loss function:

$$L_{WIoU_{v1}} = R_{WIoU}L_{IoU} \tag{5}$$

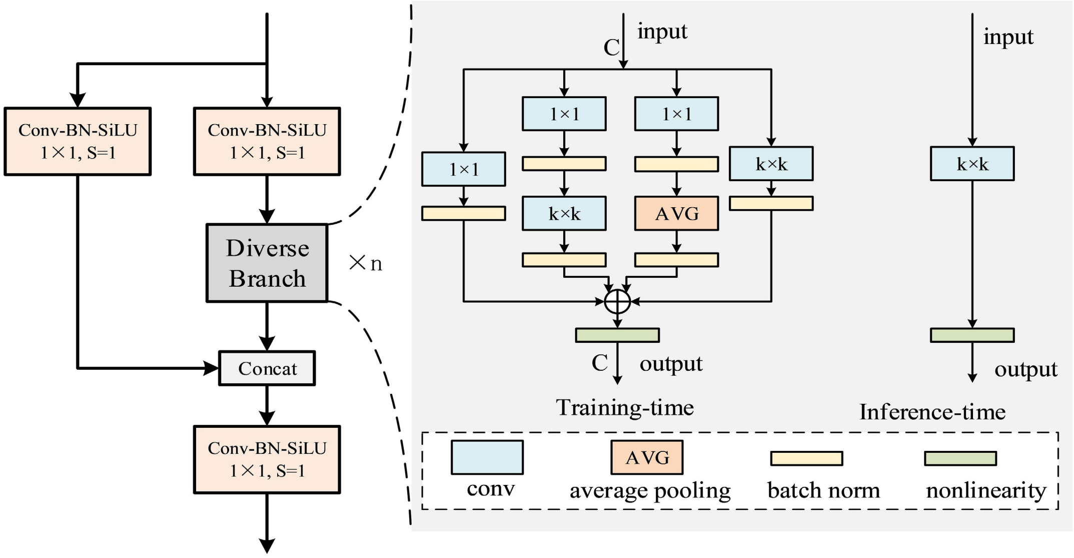


FIGURE 3. The structure of DBMCSP (Diverse Branch Modules are inserted into Cross-Stage Partial network).

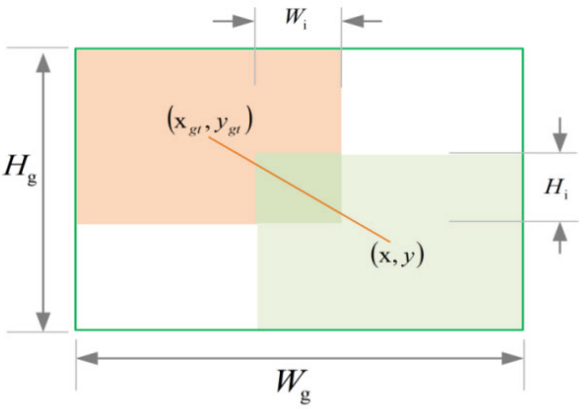


FIGURE 4. Relationship between ground truth box and predicted box.

The outlier degree of the anchor box is:

$$\beta = \frac{L_{IoU}^{*}}{L_{IoU}} \in [0, \infty) \tag{6}$$

Further obtain the $L_{WIoU_{v3}}$:

$$L_{WIoU_{v3}} = \gamma L_W IoU_v 1, \gamma = \frac{\beta}{\delta \alpha^{(\beta-\delta)}} \tag{7}$$

By adjusting the hyper-parameters δ and α, the loss function with good performance can be obtained. In this model, we take $\alpha = 1.9$, $\delta = 3$.

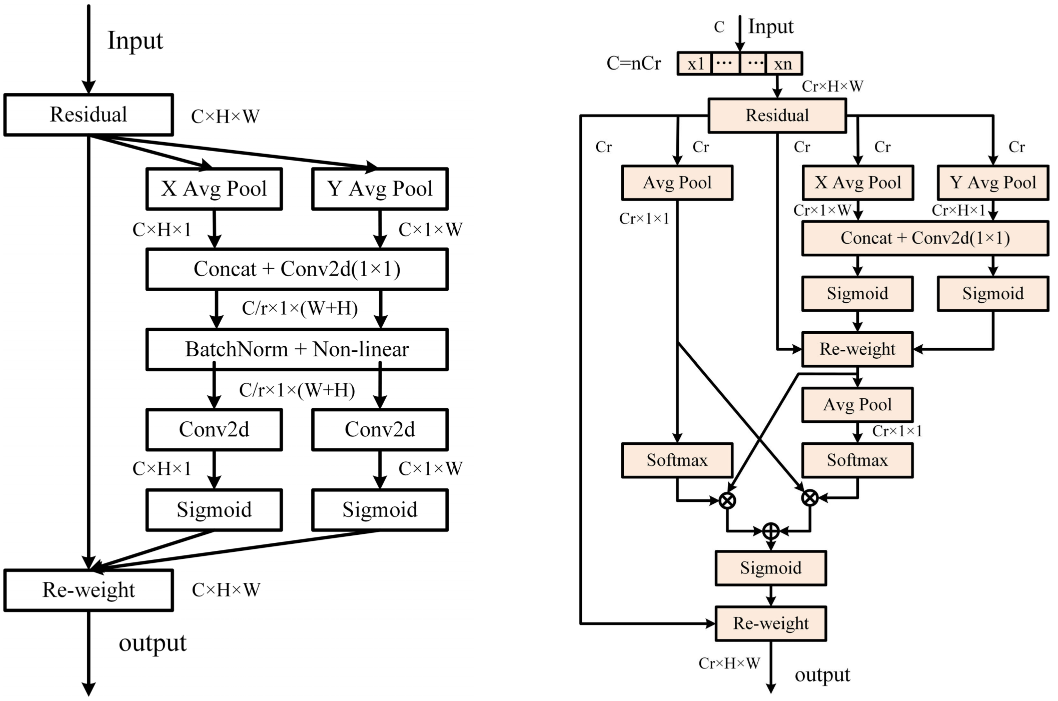
D. ATTENTION MECHANISM

The attention mechanism is a plug-and-play module that can increase focus on critical information and filter out noisy information. In this section, we provide details about the embedded position of the attention mechanism and outline our proposed attention mechanism module. We later conducted an ablation experiment to assess the performance of our proposed CAPA (Coordination and Adaptive Pool Attention module) in comparison to CA (Coordination Attention module) [30]. The result is shown in table 3.

1) COORDINATE AND ADAPTIVE POOLING ATTENTION

As shown in Fig. 5 (a), coordinate attention can embed location information into channel attention. Channel attention is decomposed into two parallel one-dimensional feature coding processes, effectively embedding integrated spatial coordinate information into the generated attention feature map. Two 1D global average pooling operations are used to aggregate input features into two independent direction-aware feature maps along the X and Y directions, respectively. The two feature maps embedded with direction-specific information are encoded as two attention maps, each of which captures the long-range dependencies of the input feature maps along a spatial direction. Therefore, the generated attention map can save location information. The two attention maps are then applied to the input feature maps by multiplication operation to emphasize the representation of interest.

As shown in Fig. 5 (b), we created the Coordination and Adaptive Pooling of Attention (CAPA) module with reference to Coordinate Attention [30] and Efficient Multi-Scale Attention [31]. It has the potential to improve CA in terms of micro-structure layout, reduce convolution operations, and increase channel attention information. To begin with, we employ a multi-branch structure, which is similar to the DBMCSP module since it can extract more information. For the sake of simplicity, all attention mechanisms have the same structure. The channel is divided into n groups, and its structure is similar to the multi-head attention mechanism. After training, the parameters of each group of attention mechanism blocks differ, resulting in richer information. To encode our rightmost coordinate information, we make the Coordinate Attention mechanism of CA simpler by getting



(a) Coordinate Attention (CA) module

(b) Coordinate and Adaptive Pooling Attention (CAPA) module

FIGURE 5. Details of attention mechanism module.

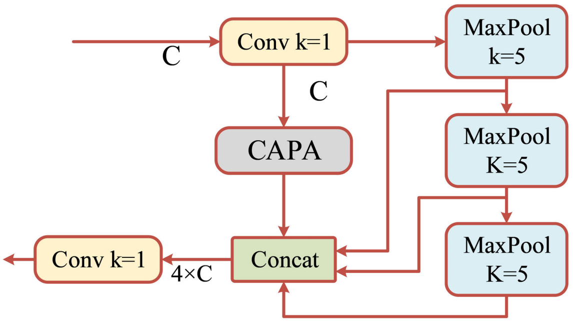


FIGURE 6. Frame diagram of SPPFA.

TABLE 1. Experimental hardware and software system.

Experimental environment	Configuration
Processor	AMD Ryzen 95900X@3.70GHz
Graphics card	NVIDIA GeForce RTX3090(24GB)
Memory	64GB
Executive system	Windows10 64 bit
Python version	Python 3.8
Pytorch version	Pytorch1.8.0(torchvision0.9.0)
CUDA/CUDNN	11.1/18.0.5

rid of its batch normalization, nonlinear processing, and convolution operation of the relevant branches.This reduces the number of parameters and the amount of computation. We call it the C (coordinate) branch. In particular, we use two adaptive average pooling (Avg Pool) layers in the X and Y directions to encode x_n's input coordinate information, then concatenate them together and feed them into a 1×1 convolution. Then, two sigmoid nonlinear operations are performed to obtain coordinate coding information, which is multiplied by the copy branch to optimize the coordinate parameters. After that, an Avg Pool layer is executed, and coordinate coding is integrated into channel coding. In order to avoid mutual interference between coordinate coding and channel coding, another Avg Pool layer is connected in parallel to encode the channel information of x_n separately, which is noted as an AP (Adaptive Pooling) branch. Before the softmax operation, the coordinate coding information of the C branch and the channel coding information of the AP branch are copied, respectively, and multiplied by each other's branches after the softmax operation to realize cross-channel interaction between the two parallel branches. Following the addition of the two results, a sigmoid function operation is performed to capture the data relationship, and the obtained values are multiplied by x_n to update the data.

CAPA not only maps the rich spatial coordinate information to the channel, but also encodes the channel information separately. It then engages in information interaction to adjust the importance of the corresponding information. The designed CAPA has only one 1×1 group convolution, and the rest are pooling operations, softmax functions, and sigmoid functions. Compared to CA, CAPA has fewer computations, a richer micro-structure, and more abstract information encoding.

2) SPATIAL PYRAMID POOLING-FAST ATTENTION

Currently, the object detection model either embeds many attention mechanisms into the backbone network to efficiently extract important information or integrates them

TABLE 2. Ablation experiment on the NWPU VHR-10.

Modified module				mAP@0.5(%)	FPS	Model size(MB)	P	R
iRMB	DBMCSP	$WIoU_{v3}$	CAPA					
				91.8	**212.7**	13.8	0.93	0.88
✓				96.7	178.6	**10.4**	0.96	0.95
	✓(overall)			93.5	208.3	20.0	0.93	0.91
✓	✓(neck)			96.9	153.8	13.2	0.96	0.93
✓	✓(neck)	✓		97.1	181.8	13.2	0.96	0.95
✓	✓(neck)	✓	✓	**97.8**	166.6	13.2	**0.97**	**0.95**

TABLE 3. Experimental comparison between CA and CAPA on NWPU VHR-10.

Comparison of attention mechanisms			mAP@0.5(%)	FPS	Params	Params+	P
YOLOv5s-7.0	CA	CAPA					
✓			91.8	212.7	656896	0	0.93
✓	✓		92.7	200.0	663576	6680	0.91
✓		✓	**93.0**	**212.7**	**657152**	**256**	**0.95**

TABLE 4. Performance of different algorithm models on the NWPU VHR-10.

Algorithm	mAP@0.5(%)	mAP@0.5:0.95(%)	FPS	Model size (MB)
YOLOv5s-7.0	91.8	58.4	**212.7**	13.8
YOLOv6s	87.2	56.0	172.1	38.7
YOLOv7-tiny	89.2	57.1	175.4	**11.7**
YOLOv8s	94.1	**64.8**	175.4	21.4
LTEA-YOLO (ours)	**97.8**	61.9	166.6	13.2

into its neck network to enhance discriminative learning. These mechanisms rely on the insertion of multiple attention mechanism modules. Our method primarily relies on the iRMB module of the detection model for feature extraction, with the CAPA module serving as a minor calibration. Therefore, we opted to incorporate the CAPA module into a branch of the SPPF at the backbone network's end. This enhanced module is known as Spatial Pyramid Pooling-Fast Attention (SPPFA), as illustrated in Fig. 6. It does not make any changes to the three maximum pooling operations; it only embeds our attention mechanism in the identity branch. It will not increase too many parameters to affect the model size, nor will it dominate the network learning, but only give the network a slight and appropriate correction.

IV. EXPERIMENT

A. DATASET AND EXPERIMENTAL ENVIRONMENT

The benchmark datasets we used were NWPU VHR-10 and VisDrone 2019, which are both very challenging small object detection datasets. NWPU VHR-10 [32] belongs to the open remote sensing small target dataset, including 10 categories: airplane, ship, storage tank, baseball diamond, tennis court, basketball court, ground track field, harbor, bridge, and vehicle. To ensure fairness, we randomly selected 75% of the images as the training dataset and the rest as the verification dataset. VisDrone 2019 [33] consists of 10,209 aerial images, including 11 categories and 2.6 million labeled targets. Different unmanned aerial vehicle models collect datasets in diverse scenes with varying weather and lighting conditions, ensuring wide coverage and diverse target scales. To conduct the experiment, we gather the training dataset and verification dataset with labels together, and then re-divide the training dataset and test dataset according to a ratio of 9:1.

The main hardware and software of the experimental environment are shown in Table. 1. We trained all models for 300 epochs, and the training strategy and evaluation index align with the benchmark model YOLOv5.

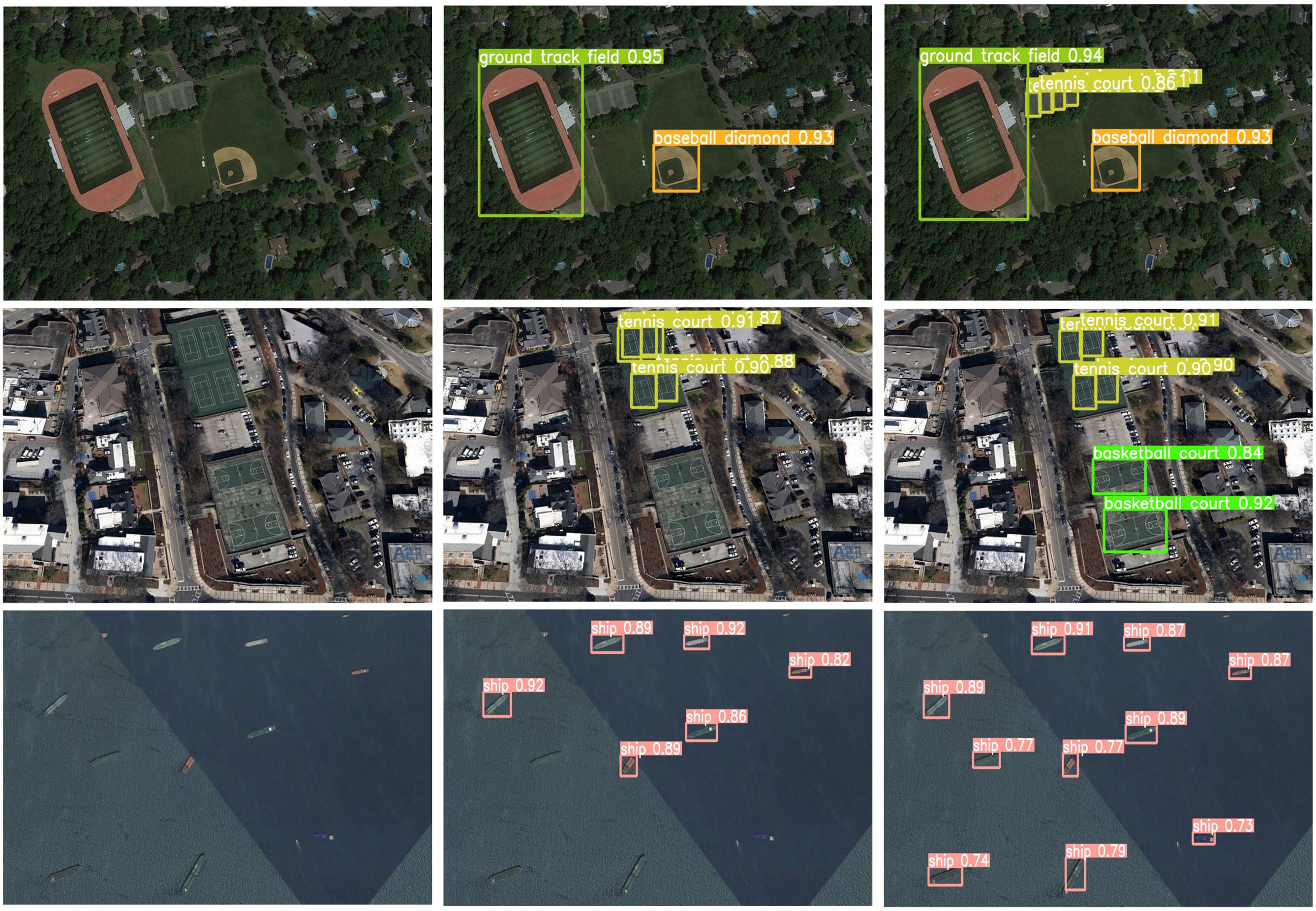
B. EXPERIMENT AND ANALYSIS

1) ABLATION EXPERIMENT

Compared with the original YOLOv5s-7.0, LTEA-YOLO has several improvements, as shown below. To extract features, we use the effective iRMB stack as the backbone network. We construct the DBMCSP module to replace all $C3$ modules in the neck fusion section, thereby enhancing the diversity of information. We replaced the $CIoU$ loss function with the most recent $WIoU_{v3}$ loss function, which can speed up training convergence. We designed a lightweight and efficient attention mechanism called CAPA embedded in SPPF to appropriately allocate the attention range. In order to verify the contribution of these improvements to LTEA-YOLO,

TABLE 5. Performance of different algorithm models on VisDrone 2019.

Algorithm	mAP@0.5(%)	mAP@0.5:0.95(%)	FPS	Model size (MB)
Faster RCNN	17.5	8.2	27	108.0
SDD	10.4	6.0	54	95.7
YOLOv4	18.6	10.8	21	244.0
YOLOv5-5.0	25.9	16.2	60	178.0
YOLOv5s-7.0	34.5	18.5	125	13.7
YOLOv6s	32.4	**18.9**	**196.8**	38.7
YOLOv7-tiny	34.5	18.5	119	**11.7**
LTEA-YOLO (ours)	**35.8**	18.6	111.1	13.2



(a) Input image

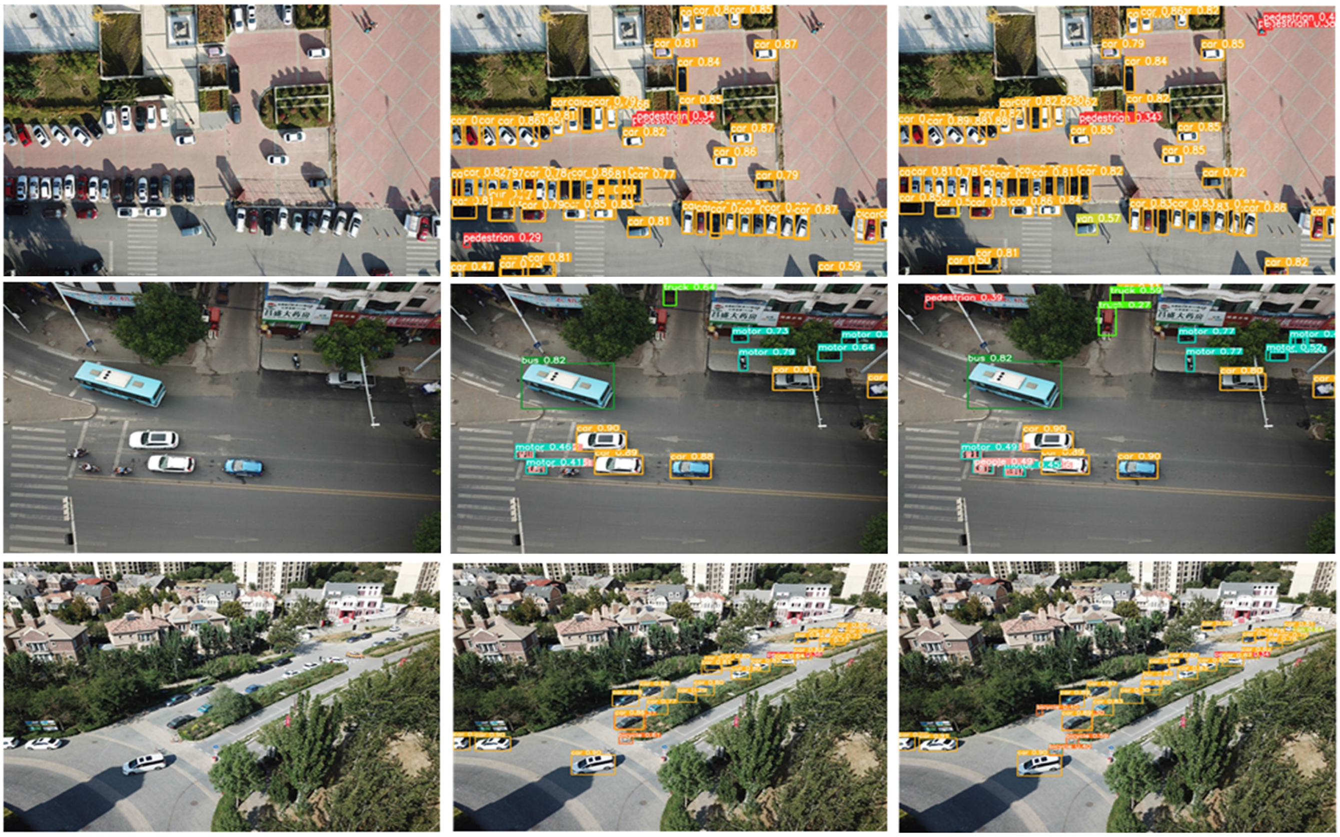
(b) Results of YOLOv5s

(c) Results of LTEA-YOLO (our)

FIGURE 7. Examples of detection results of YOLOv5s and LTEA-YOLO on the NWPU VHR-10.

ablation experiments are carried out on the NWPU VHR-10 dataset, and the experimental results are shown in Table. 2.

In Table. 2, the symbol ✓ refers to the use of the corresponding module. If there is no ✓ for the four improvement measures, the benchmark model, YOLOv5s-7.0, is used. When only iRMB is used as the backbone network, the FPS (frames per second) is reduced from 212.7 to 178.6, the model size is reduced from 13.8 MB to 10.4 MB, and P (precision rate) and R (recall rate) are improved by 3% and 7%, respectively. Surprisingly, mAP@0.5 improved from 91.8% to 96.7%. It shows that iRMB, a lightweight Transformer module, can improve the model's performance and reduce the memory burden. When all $C3$ modules in the overall model (including the backbone and neck) were replaced with DBMCSP modules, the model size increased by 44.9%. It has little effect on FPS and P, whereas increasing R and mAP@0.5 by 3% and 1.7%, respectively, makes a great contribution to the model performance. To avoid making the model larger, we used iRMB as the backbone network and replaced $C3$ with DBMCSP only in the neck



(a) Input image

(b) Results of YOLOv5s

(c) Results of LTEA-YOLO (our)

FIGURE 8. **Examples of detection results of YOLOv5s and LTEA-YOLO on the VisDrone2019.**

fusion section and not in the overall model. As a result, the model size is reduced from 13.8 MB to 13.2 MB, while the mAP@0.5, P, and R increased by 5.1%, 3% and 5% respectively. We further used $WIoU_{v3}$ to replace $CIoU$ as the regression loss function and attained a 0.2% increase in mAP@0.5, and the training convergence is faster. Finally, based on the above three improvements, our designed CAPA is incorporated into SPPF, resulting in a 0.7% increase in mAP@0.5 and an 8% decrease in FPS, which still meets the real-time requirements. These improvements work together to maximize the overall performance of the model and reduce its size.

The experimental results of the comparison between CA and CAPA are shown in Table. 3. The first row is the benchmark model without attention mechanisms. After we inserted CA into the SPPF of YOLOv5s-7.0, mAP@0.5 increased by 0.9% and P decreased by 2%, compared to the benchmark model. As the parameters increased, the FPS decreased slightly. After we replaced CA with CAPA, mAP@0.5 and P increased by 1.2% and 2%, respectively, while FPS remained unchanged due to very few parameter changes compared to the benchmark model. It is obvious that our proposed CAPA performs better than CA, with fewer parameters, faster detection speed, and higher detection accuracy.

2) EXPERIMENTAL COMPARISON ON MULTIPLE MODELS

To evaluate the performance of the lightweight model in small target detection, we used our proposed LTEA-YOLO model and YOLOv5s [4], YOLOv6s [34], YOLOv7-tiny [35], and YOLOv8s [36] to conduct comparative experiments on the NWPU VHR-10 dataset, as shown in Table. 4. Among them, the benchmark model YOLOv5s-7.0 achieves the highest FPS but otherwise has medium performance. YOLOv6s has the worst performance in terms of both model size and detection accuracy because it uses the RepVGG feature extraction network, which is insufficient to extract diverse information and decrease model parameters. YOLOv7-tiny has the smallest model size, but it only slightly outperforms YOLOv6s in other aspects. YOLOv8s has good performance in detection accuracy with the highest mAP@0.5:0.95 because it replaces C3 module with C2f module to extract richer feature information, which is similar to YOLOv7 and YOLOv5. Our proposed LTEA-YOLO model achieves excellent overall performance with the best mAP@0.5 and the second-best mAP@0.5:0.95, and the second-smallest model size, although FPS is 27.7% lower than YOLOv5s-7.0, slightly lower than other models.

To further verify the generalization ability of the LTEA-YOLO model, we carried out comparative experiments on the VisDrone2019 dataset, as shown in Table. 5. YOLOv6s achieves a maximum of mAP@0.5:0.95 and FPS, but its model size is also relatively large. The YOLOv7-tiny achieves excellent performance with the smallest model size. Our LTEA-YOLO model reached the best mAP@0.5, which demonstrated its excellent performance in small target detection. It is worth noting that the light-weight version of YOLOv5, YOLOv5s-7.0, achieves better performance than the larger version, YOLOv5-5.0. It proves that a lightweight model suitable for deployment on mobile terminals can maintain superior performance. All three versions of YOLOv5s-7.0, YOLOv7-tiny, and YOLOv8s use the cross-stage partial network as their primary feature extraction module. This makes the model more accurate by reducing the number of parameters and calculations required, adding more feature information, and making it run faster. Our proposed LTEA-YOLO algorithm further refines the idea of a cross-stage partial network to ensure that the model can extract more diversified information. It combines the advantages of the Transformer and introduces a lightweight and efficient attention mechanism to make learning abilities more powerful and comprehensive, thereby improving the model's overall performance.

3) EXPERIMENTAL VISUALIZATION

Fig. 7 and Fig. 8 show the results of our comparison experiments. These show how well our LTEA-YOLO model and the benchmark model YOLOv5s-7.0 detected small objects on the NWPU VHR-10 validation dataset and the VisDrone2019 validation dataset, respectively. Both datasets contain different kinds of targets, sparse targets, crowded targets, and targets with different sizes, which makes them suitable for evaluating the performance of small targe detection. From Fig. 7, it is clear that all the tennis courts, basketball courts, and some ships were not detected by YOLOv5s, but they can be detected by our LTEA-YOLO. Fig. 8 reveals that LTEA-YOLO accurately and comprehensively detects numerous small targets, including dense ones, that YOLOv5s have missed. Experiments demonstrated the strong advantages of our lightweight model, LTEA-YOLO, in small target detection.

V. CONCLUSION

Small object detection is a very challenging task. Although previous studies have made many achievements in this field, most of them are large-scale models that require large memory and powerful computing ability, which is not conducive to deployment on embedded mobile terminals with insufficient computing resources. We proposed LTEA-YOLO, a lightweight and efficient small object detection model. The lightweight Transformer module iRMB is stacked in the backbone network of LTEA-YOLO, and its sequence-to-sequence characteristics make it more sensitive to dense targets and occluded targets, thus extracting more abundant and useful information. We created a DBMCSP module to replace all C3 modules in the neck fusion part of the original YOLOv5s. During training, DBMCP functions as a multi-branch structure, while in inference, it transforms into a single convolution layer with parameters derived from the multi-branch structure. This allows DBMCP to extract a wider range of feature information without compromising the speed of inference. We use *WIoU* as the loss function for box regression, which further accurately locates the small object and accelerates model training convergence. Finally, we created a lightweight and efficient CAPA module and embedded it into the SPPF module to enhance attention to target, thus further improving detection accuracy. Experiments on the NWPU VHR-10 show that our lightweight LTEA-YOLO model has superior overall performance in small target detection, but there is still much room for improvement in detection on the VisDrone 2019. So, how to further improve the model detection accuracy and speed is still the focus of our follow-up work.

REFERENCES

[1] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.

[2] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 779–788.

[3] W. Liu, D. Anguelov, D. Erhan, and C. Szegedy, "SSD: Single shot multibox detector," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Amsterdam, The, Netherlands, Oct. 2016, pp. 21–37.

[4] G. Jocher, A. Stoken, and J. Borovec. *v7.0-YOLOv5 SOTA Realtime Instance Segmentation*. Accessed: Jun. 9, 2020. [Online]. Available: https://github.com/ultralytics/yolov5/releases/tag/v7.0

[5] J. Zhang, X. Li, J. Li, L. Liu, Z. Xue, B. Zhang, Z. Jiang, T. Huang, Y. Wang, and C. Wang, "Rethinking mobile block for efficient attention-based models," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Paris, France, Oct. 2023, pp. 1389–1400.

[6] Z. Tong, Y. Chen, Z. Xu, and R. Yu, "Wise-IoU: Bounding box regression loss with dynamic focusing mechanism," 2023, *arXiv:2301.10051*.

[7] O. C. Koyun, R. K. Keser, I. B. Akkaya, and B. U. Töreyin, "Focus-and-detect: A small object detection framework for aerial images," *Signal Process., Image Commun.*, vol. 104, May 2022, Art. no. 116675.

[8] L. Zhou, C. Zheng, H. Yan, X. Zuo, Y. Liu, B. Qiao, and Y. Yang, "RepDarkNet: A multi-branched detector for small-target detection in remote sensing images," *ISPRS Int. J. Geo-Inf.*, vol. 11, no. 3, p. 158, Feb. 2022.

[9] S.-H. Kang and J.-S. Park, "Aligned matching: Improving small object detection in SSD," *Sensors*, vol. 23, no. 5, p. 2589, Feb. 2023.

[10] L. Jiao, C. Kang, S. Dong, P. Chen, G. Li, and R. Wang, "An attention-based feature pyramid network for single-stage small object detection," *Multimedia Tools Appl.*, vol. 82, no. 12, pp. 18529–18544, May 2023.

[11] G. Li, X. Hao, L. Zha, and A. Chen, "An outstanding adaptive multi-feature fusion YOLOv3 algorithm for the small target detection in remote sensing images," *Pattern Anal. Appl.*, vol. 25, no. 4, pp. 951–962, Apr. 2022.

[12] Z. Liu, X. Gao, Y. Wan, J. Wang, and H. Lyu, "An improved YOLOv5 method for small object detection in UAV capture scenes," *IEEE Access*, vol. 11, pp. 14365–14374, 2023.

[13] H. Zhang, Q. Du, Q. Qi, J. Zhang, F. Wang, and M. Gao, "A recursive attention-enhanced bidirectional feature pyramid network for small object detection," *Multimedia Tools Appl.*, vol. 82, no. 9, pp. 13999–14018, Apr. 2023.

[14] Y. Liu, G. He, Z. Wang, W. Li, and H. Huang, "NRT-YOLO: Improved YOLOv5 based on nested residual transformer for tiny remote sensing object detection," *Sensors*, vol. 22, no. 13, p. 4953, Jun. 2022.

[15] H. Liang and S. Seo, "UAV low-altitude remote sensing inspection system using a small target detection network for helmet wear detection," *Remote Sens.*, vol. 15, no. 1, p. 196, Dec. 2022.

[16] J. Zhu, X. Wang, Y. Liu, Q. Ji, Z. Zhao, and S. Wang, "UavTinyDet: Tiny object detection in UAV scenes," in *Proc. 7th Int. Conf. Image, Vis. Comput. (ICIVC)*, Jul. 2022, pp. 195–200.

[17] H. Zhao, H. Zhang, and Y. Zhao, "YOLOv7-sea: Object detection of maritime UAV images based on improved YOLOv7," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. Workshops (WACVW)*, Waikoloa, HI, USA, Jan. 2023, pp. 233–238.

[18] Z. Liu, Y. Gao, Q. Du, M. Chen, and W. Lv, "YOLO-extract: Improved YOLOv5 for aircraft object detection in remote sensing images," *IEEE Access*, vol. 11, pp. 1742–1751, 2023.

[19] A. Dosovitskiy, L. Beyer, and A. Kolesnikov, "An image is worth 16 × 16 words: Transformers for image recognition at scale," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, Jan. 2021.

[20] P. Gao, M. Zheng, X. Wang, J. Dai, and H. Li, "Fast convergence of DETR with spatially modulated co-attention," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Montreal, QC, Canada, Oct. 2021, pp. 3601–3610.

[21] 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," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Montreal, QC, Canada, Oct. 2021, pp. 9992–10002.

[22] L. He and S. Todorovic, "DESTR: Object detection with split transformer," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, New Orleans, LA, USA, Jun. 2022, pp. 9367–9376.

[23] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*.

[24] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 448–456.

[25] S. Elfwing, E. Uchibe, and K. Doya, "Sigmoid-weighted linear units for neural network function approximation in reinforcement learning," *Neural Netw.*, vol. 107, pp. 3–11, Nov. 2018.

[26] J. L. Ba, J. Ryan Kiros, and G. E. Hinton, "Layer normalization," 2016, *arXiv:1607.06450*.

[27] D. Hendrycks and K. Gimpel, "Gaussian error linear units (GELUs)," 2016, *arXiv:1606.08415*.

[28] X. Ding, X. Zhang, J. Han, and G. Ding, "Diverse branch block: Building a convolution as an inception-like unit," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Nashville, TN, USA, Jun. 2021, pp. 10881–10890.

[29] X. Ding, X. Zhang, N. Ma, J. Han, G. Ding, and J. Sun, "RepVGG: Making VGG-style ConvNets great again," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Nashville, TN, USA, Jun. 2021, pp. 13728–13737.

[30] Q. Hou, D. Zhou, and J. Feng, "Coordinate attention for efficient mobile network design," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Nashville, TN, USA, Jun. 2021, pp. 13708–13717.

[31] D. Ouyang, S. He, G. Zhang, M. Luo, H. Guo, J. Zhan, and Z. Huang, "Efficient multi-scale attention module with cross-spatial learning," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Jun. 2023, pp. 1–5.

[32] G. Cheng, J. Han, P. Zhou, and L. Guo, "Multi-class geospatial object detection and geographic image classification based on collection of part detectors," *ISPRS J. Photogramm. Remote Sens.*, vol. 98, pp. 119–132, Dec. 2014.

[33] D. Du, "VisDrone-DET2019: The vision meets drone object detection in image challenge results," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshop (ICCVW)*, Oct. 2019, pp. 213–226.

[34] C. Li, L. Li, and H. Jiang, "YOLOv6: A single-stage object detection framework for industrial applications," 2022, *arXiv:2209.02976*.

[35] C.-Y. Wang, A. Bochkovskiy, and H.-Y.-M. Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Vancouver, BC, Canada, Jun. 2023, pp. 7464–7475.

[36] G. Jocher, A. Chaurasia, and J. Qiu. *YOLOv8.2 Released by Ultralytics*. Accessed: Nov. 12, 2023. [Online]. Available: https://github.com/ultralytics/ultralytics

BO LI received the M.S. degree from South China University of Technology, in 2006. He is presently pursuing the Doctorate of Philosophy at Macao Polytechnic University. Since 2015, he has been an Associate Professor with Zhongshan Institute, University of Electronic Science and Technology of China. His recent research interests include machine vision and deep learning.

SHENGBAO HUANG was born in Guangxi, China. He is currently pursuing the master's degree with the University of Electronic Science and Technology of China. From 2018 to 2020, he was an Assistant Engineer in a subsidiary of Guangxi Automobile Group Company for two years. His recent research interests include artificial intelligence and object detection.

GUANGJIN ZHONG was born in Guangxi, China. He received the bachelor's degree in mechatronics electronics engineering from Guangxi University, in 2022. He is currently pursuing the master's degree in mechanical engineering with the School of Mechanical and Electrical Engineering, University of Electronic Science and Technology of China. His current research interests include machine vision and deep learning.