MHNet: Multi-scale Hierarchical Extraction Network for Small Object Detection in UAV Images

Ziyang Xing, Xuebin Xu, Meiling Sun, and Kuihe Yang

***Abstract*—Unmanned aerial vehicle (UAV) images have the characteristics of small object sizes, dense distributions, and complex backgrounds. Existing detection methods perform well under normal circumstances, but perform poorly when processing UAV images. In this paper, we propose MHNet, a small object detection framework for UAV images, which solves the above problems through multi-scale feature processing and more efficient feature fusion. First, we design a multi-scale hierarchical convolution (MHC) module that extracts features at different scales, layer by layer, providing finer-grained feature information and a larger receptive field. Second, we designed the SPPFC module to capture the multi-scale features extracted by the backbone. We introduce contextual anchor attention (CAA) in the SPPFC module to bolster contextual dependency and fortify feature information across various scales, thereby augmenting the semantic information of high-level features. At the same time, this paper uses an auxiliary detection head, combined with a new feature fusion architecture to improve the prediction ability of small objects. The CAA module downsample the input features of the auxiliary detection head to enhance the feature information of the other two detection heads. This design effectively promotes the fusion of high-level and low-level information. Multiple experiments on VisDrone2019 and UAVDT have demonstrated the effectiveness of MHNet. On VisDrone2019, with mAP and mAP50 reaching 28.2% and 45.8%, respectively. Compared with the benchmark, our MHNet improves mAP and mAP50 by 4.8% and 6.7%, respectively.**

Link to graphical and video abstracts, and to code: https://latamt.ieeer9.org/index.php/transactions/article/view/9496

***Index Terms*—Object detection, Unmanned aerial vehicle images, Multiscale feature, Attention mechanism.**

I. Introduction

OBJECT detection plays an important role in UAV image recognition, such as pedestrian detection [1], vehicle detection [2], etc. The purpose is to accurately specify the placement and category of each object in the UAV image by identifying predefined object categories. This allows for the fast and accurate classification and tracking of objects. Drones are small, flexible, and have a wide field of view. They are widely used in many fields such as traffic management [3], meteorological detection and urgent rescue [4].

The associate editor coordinating the review of this manuscript and approving it for publication was Gladston Moreira (*Corresponding author: Kuihe Yang*).

Z. Xing, X. Xu, and Kuihe Yang are with Hebei University of Science and Technology, Shijiazhuang 050018, China (e-mails: xingziyang@stu.hebust.edu.cn, xuxuebin@stu.hebust.edu.cn, and ykh@hebust.edu.cn).

M. Sun is with Shijiazhuang Preschool Teachers College, Shijiazhuang, China (e-mail: sunmeiling2025@gmail.com).

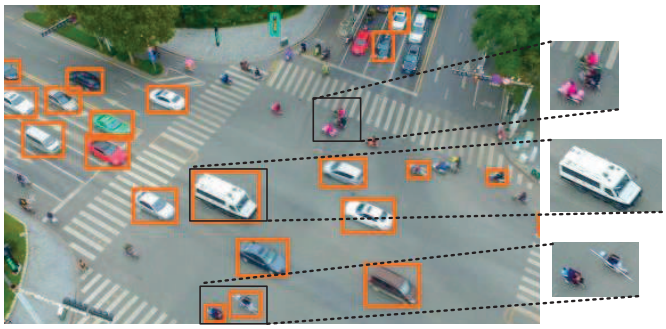

Fig. 1. Results of UAV image detection using YOLOv8. A large number of small objects were not detected in the images, while there were false detections. Two cyclists and a van were incorrectly recognized as cars.

At present, there are two primary classes of object detection algorithms: single-stage and two-stage. Single-stage detection methods, such the YOLO series [5]-[8] and Single Shot MultiBox Detector (SSD) [9], directly forecast the bounding box of the object and its classification in a singular stage. Single-stage detection predicts the object's position and category immediately, bypassing region proposal generation, resulting in enhanced inference speed. Consequently, single-stage algorithms hold increased importance in real-time monitoring contexts. Conversely, two-stage detection techniques, like Fast R-CNN [10], Faster R-CNN [11], Mask R-CNN [12], and Grid R-CNN [13], operate in two phases, allowing the region proposal network to identify a candidate box for item detection and classification within that box. Two-stage detection has superior accuracy owing to enhanced selection of region recommendations; however, its detection speed is rather sluggish. Consequently, in detecting settings necessitating high precision, two-stage detection is more advantageous. Real-time monitoring is an essential component of UAV image detection.

However, these detection networks often do not perform well when detecting UAV images . This is mainly because drone images are very different from normal scenes. For example, drone images have low resolut ion, small objects, uneven brightness distribution, arbitrary object distribution, different scales, and object occlusions. In addition, the complexity and computational speed requirements of deep learning models have hindered the application of drone image detection in real-time monitoring scenarios. Thus, it is essential to explore an object detection system with real-time, accuracy and robustness [14]-[17]. In UAV images object detection, the key to the detection algorithm lies in feature extraction, feature fusion, or a combination of various feature manipulation methods

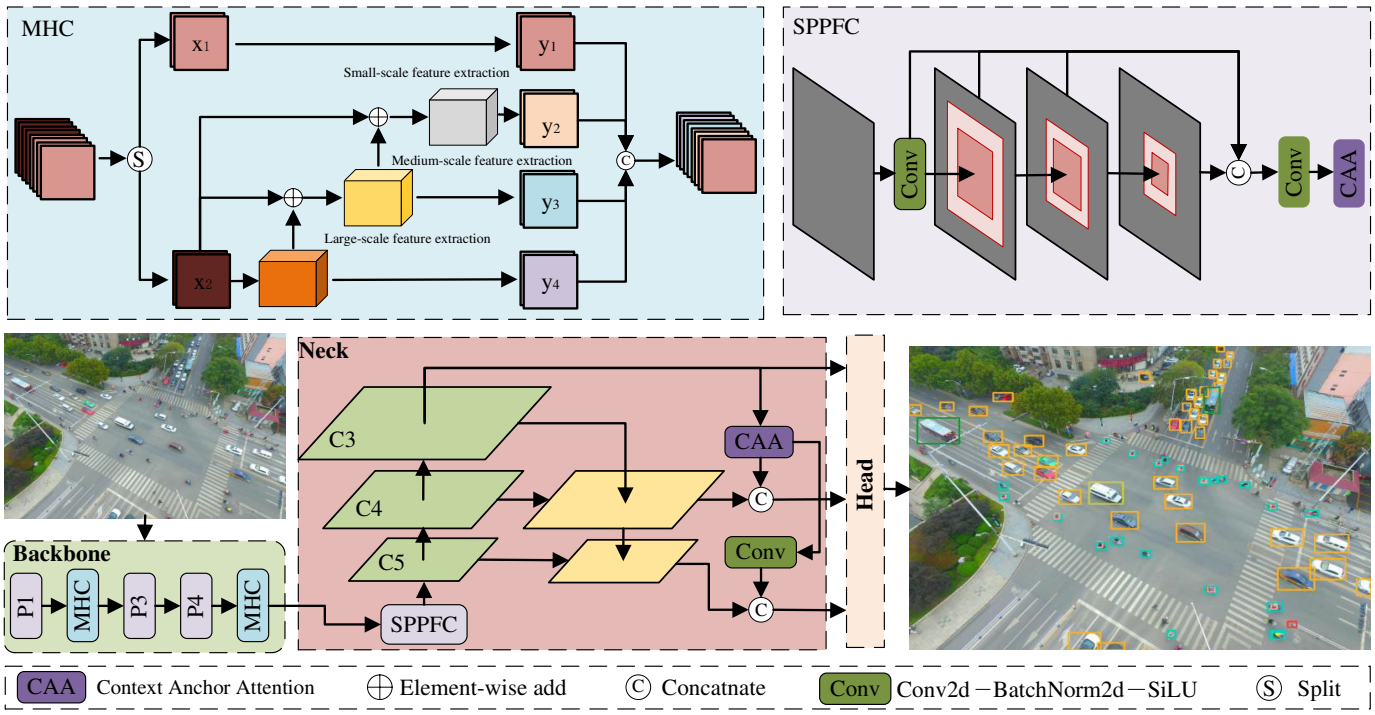

Fig. 2. The overall architecture of MHNet. The backbone network receives the UAV image and uses it to acquire several feature maps. The backbone network includes the first, third, and fourth stages of the YOLOv8 backbone and the MHC module proposed by us. The last three feature maps are used as the input to the neck layer, which are denoted as $\{C3, C4, C5\}$, respectively. The neck layer first uses SPPFC to capture multi-scale information and then performs layer-by-layer fusion of features from different layers. The head ensures the accuracy of predicted categories and localization by using a small object auxiliary detection head.

through the network architecture of the neck to augment ability to object recognition. Due to the small size, low resolution, fuzzy boundaries, and complex background of small objects, there have been many corresponding solutions to these problems. YOLOv8-QSD [18] adds a small object detection head to strengthen the seamless integration of shallow and deep information. This can complement the missing information during downsampling in various modules and effectively retain contextual information. APNet [19] adds a precisely positioned variable convolution to the backbone, which can dynamically change the shape of the kernel to enhance refined features when detecting dense regions. This method uses a fine-grained approach to distinguish multiple objects in dense areas. DC-YOLOv8 [20] uses depth-separable convolution for downsampling, which can replenish the missed information by the module during downsampling and effectively retain feature information. MSFT-YOLO [21] uses convolution block with larger receptive fields in the backbone and neck to provide richer multi-level information for detection. CourNet [22] fuses specific and abstract features by learning shallow and deep ones. It also combines global and local information based on the relationship between individual objects and the local environment and constructs a feature pyramid through different layers of deep feature information.

These methods have improved detection performance, but some problems remain. First, the object distribution in the UAV image is dense and there is object occlusion. When ordinary convolution block extract features, they cannot accurately capture objects of various sizes. In particular, when detecting small objects, a large amount of feature information is lost. Second, after multiple layers of convolution, there is a large semantic difference between objects of different scales, and multi-scale features cannot be effectively retained. In addition, when processing the extracted feature information, the shallow semantic information and the deep location information cannot be effectively fused, which can lead to errors in localization and classification. As shown in Fig. 1, larger objects are normally recognized correctly, while there are more false detections for small objects.

To combat the aforementioned issues, we put forward an UAV image detection method that is MHC-based. First, we design a new multi-scale hierarchical convolution block, which is different from the existing convolution blocks. MHC pays more attention to the extraction of multi-scale information and can successively process images of different scales layer by layer. Additionally, we use a weighted attention feature pyramid SPPFC in the neck to increase the context dependence and receptive field of each scale when fusing multi-scale information. Finally, we use a small object auxiliary detection head and combine it with richer feature fusion of semantic and positional information for detection. Our contributions are as follows:

a) We propose a multi-scale hierarchical convolution (MHC) method that introduces multi-layer residual connections to obtain more fine-grained multi-scale features and larger receptive fields. In the hierarchical structure, characteristics at various sizes are extracted in turn, and the missing information in the upper layer is supplemented. This effectively reduces missed

detection.

b) A weighted feature spatial pyramid pooling module (SPPFC) is designed, in which the context anchor attention (CAA) mechanism is used. This design can capture more context dependencies and enhance central features. It solves the problem of losing small object information during backbone extraction.

c) A small object auxiliary detection head is designed, and this detection head is used to replace the original large object detection head. The auxiliary detection head provides a wealth of small object feature information for other detection heads. This method, combined with a new feature fusion method, brings shallow semantic information and deep location information closer together logically, effectively improving the situation where information fusion is insufficient.

d) We verified our model on the VisDrone2019 dataset, achieving 45.8% and 28.2% in mAP and mAP50, respectively. The experimental results verify the advantages of our model.

II. Methods

A typical network architecture consists of three parts: backbone, neck, and head. YOLOv8 has already shown good real-time monitoring results, but it is not good at detecting small objects. The MHNet proposed in this paper is a model specifically for small objects. Fig. 2 shows the overall structure of MHNet. The backbone network uses the CSPDarknet53 [23] architecture and adds the MHC module. Context anchor attention is introduced in the spatial pyramid pooling module to extract and retain multi-scale features. A more effective feature fusion method is used, combined with a small object auxiliary detection head to obtain richer location and classification information.

A. Multiscale Hierarchical Convolution

Traditional convolution modules. They use a fixed kernel size, which has a fixed receptive field during feature extraction and cannot effectively capture complex scale changes. When dealing with multi-scale features, a common practice is to use convolutions for residual connection and concatenation combinations to meet the needs of multi-scale feature extraction. For example, the Res2Net [24] module and the C2f [25] module have been widely used for feature extraction. The Res2Net module adopts a layered architecture to process features in sequence. This module can effectively increase the receptive field. The C2f module processes features at different scales to improve the model's multi-scale perception capabilities and effectively extract the characteristics of the object. However, since the objects in the drone image are densely distributed and have a plethora of small objects, richer positional and semantic information is required. Although this method can obtain good results in general scenarios, it cannot effectively detect small objects in UAV images. When the Res2Net module processes small objects, the feature transfer between different scale levels cannot be effectively fused together. And in module C2f's feature extraction stage, the features of large-scale objects will cover those of small-scale objects, which cannot obtain good results in UAV image detection.

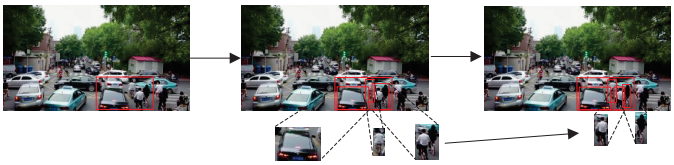

Fig. 3. MHC module feature extraction process. After the features pass through the first layer, only the car is accurately extracted. However, the bicycle and pedestrian are only accurately extracted in subsequent layers due to their small objects and occlusions.

To address the above issues, this study provides a multi-scale hierarchical convolution (MHC) module. We use a three-layer MHC module to replace the second and fifth C2f modules in the CSPDarknet architecture. This design can achieve the best balance between network speed and detection performance. Specifically, the MHC module is divided into three layers, each with a different receptive field, and different layers are used to process features at different scales. We first use a 1×1 convolution to reduce the dimensionality of the input feature map in the channel direction and split it. The two parts of the split are denoted as $\{x_1, x_2\}$. x_1 is used for identity mapping, and x_2 is input to the hierarchical structure. In the hierarchical structure, each set of filters takes the result of the upper set of filters and x_2 as input, and the result of each set of filters is input to a 1×1 convolution for fusion. Each layer of filters uses a 3×3 convolution residual concatenated structure to extract features locally. With the increase of the stratification, the extracted local feature scales become increasingly smaller. The unprocessed x_2 is used as the input to avoid feature overlap and feature loss in the upper filters. The filters can be reused, but considering that more filters will lead to an increase in calculation speed and cost, we have chosen a more balanced three-layer structure.

Each group of filters is denoted as a bottleneck. The bottleneck has the same input and output dimensions. Each layer of filters is denoted as $\{B_{1,\,k},\ B_{2,\,k},\ B_{3,\,k}, \ldots,\ B_{n,\,k}\}$, where n indicates the layer to which bottleneck belongs, and k controls the feature dimension within the combination. Starting from $B_{2,\,k}$, the output of layer $B_{n-1,\,k}$ is part of the input of layer $B_{n,\,k}$. This combination method increases the receptive field range within the layer, extracts finer-grained features, and reduces the occurrence of missed detections. This process can be expressed by Equation (1). Finally, the outputs of each small module are fused using a 1×1 convolution as shown in Equation (2).

$$\begin{cases} B_{n,\,k}\left(x_2\right), & n = 1; \\ B_{n,\,k}\left(x_2 + B_{n-1,\,k}\right), & n > 1; \end{cases} \tag{1}$$

$$y = Conv_{1\times1}\left(Concat\left(x_1 + \sum_{i=1}^{n} B_i\right)\right) \tag{2}$$

Among them, y is the result after MHC processing, and $B_{n,\,k}$ is the result of the nth bottleneck. Furthermore, within the bottleneck, the accuracy and lightweight can be better balanced by changing the feature dimension k within the combination.

MHC module feature extraction process is shown in Fig. 3. After the features go through the first layer, only cars are accurately extracted. While bicycles and pedestrians are not accurately extracted until the subsequent layers due to small objects and occlusion.

B. Weighted Feature Spatial Pyramid Pooling

To capture extracted feature information from the backbone, the general approach is to first process it in the neck using a feature spatial pyramid pooling (SPP) module [26]. The SPP module allows inputs of any size and produces outputs of the same size. However, the tiny size of objects in UAV images, and severe loss of object feature information will occur when pooling operations of different scales are performed. Low-level features have richer positional information and are suitable for accurately locating objects. However, low-level features have less semantic information and may not be able to perform classification tasks well.

To tackle the lack of semantic information, this work is inspired by the Poly Kernel Inception Network [27] and designs the SPPFC module. The context dependence of the direct pixel is captured by CAA to enhance the central features, and the structure of which is shown in Fig. 4. The CAA module has two long and narrow depth separable convolutions. The combination of these two convolutions has a larger receptive field and is much smaller in scale than a large kernel convolution with the same receptive field. This combination can obtain more critical semantic information when performing feature extraction. The input features are enhanced both laterally and longitudinally. The SPPFC module separately enhances input features of any size, fuses the enhanced information, and produces an output of the same size. The small size of pedestrian objects in the UAV image is one of the difficulties in detection. However, the location information of pedestrians is in most cases a narrow rectangular frame, and it is easier to extract the characteristic information of pedestrians using long and narrow convolutions.

After performing maximum pooling on inputs of different scales, the SPPFC module uses average pooling and 1×1 convolution to obtain local features:

$$F^{pool} = Conv_{1\times1}\left(P_{avg}\left(X\right)\right) \tag{3}$$

Where X is the input to the CAA. The two long and narrow depth separable convolutions in the horizontal and longitudinal directions approximate a large kernel convolution, and the feature map is input to the large convolution kernel for feature extraction:

$$F^{w} = DWConv_{1\times k_b}\left(F^{pool}\right) \tag{4}$$

$$F^{h} = DWConv_{k_b\times1}\left(F^{pool}\right) \tag{5}$$

These two convolutions are more lightweight than the $k_b \times k_b$ convolution while obtaining more contextual features to effectively enhance semantic information at different scales. A 1×1 convolution is then used to fuse the extracted feature maps and finally activated with Sigmoid. The SPPFC module uses a large sensory field to detect the surrounding information

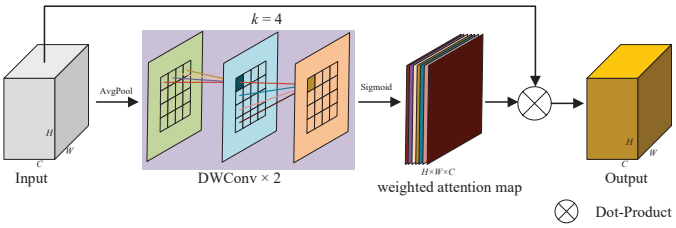

Fig. 4. Architecture of CAA. The input features are first average pooled and then extracted through two long and narrow depthwise separable convolutions. The extracted feature map is activated by Sigmoid to generate a weighted attention map, which is used to enhance the important parts of the input features.

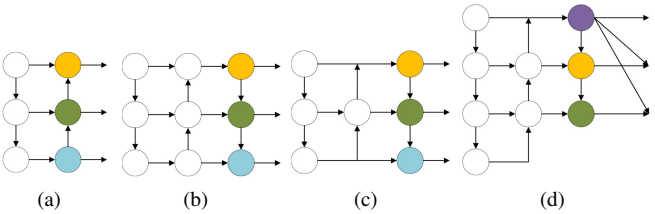

Fig. 5. Neck structure. (a) FPN. (b) PAN. (c) Neck of YOLOv8. (d) Neck of MHNet (Ours)

of tiny objects, reducing the occurrence of situations where small objects cannot be recognized due to complex backgrounds, and effectively and accurately retaining small object information. This method achieves good results with only a small increase in computational cost.

C. Small Object Auxiliary Detection Head and Effective Feature Fusion

The detection head is designed with reference to YOLOv8 and contains three detection layers. The third detection layer is adapted to find objects over 32×32. However, in UAV image detection, there are few objects larger than 32×32. The existing method cannot effectively identify the underlying semantic information and the upper-level positional information, and there is a lot of computational redundancy.

To address the above challenges, this study designed a small object auxiliary detection head and a new feature fusion structure, as shown in Fig. 5(d). We removed the third detection layer and added a detection layer that could detect objects over 4×4. At this time, the detection layers are used to detect objects larger than 4×4, 8×8, and 16×16, respectively. The low-level feature map contains fuller semantic information. The information input to the shallowest detection layer is downsampling and input to the other two detection layers to assist in adjusting the input data of the other two detection layers. When downsampling the feature information of the shallowest layer, the CAA structure in Fig. 4 is used, and replace its second convolution kernel with a 3×3. In the improved network, feature fusion can get richer feature data for small objects from the shallow feature map. This method effectively improves detection performance while reducing computational redundancy.

TABLE I
COMPARISON RESULTS OF CHANGING THE DETECTION HEAD

Method	mAP50(%)	Parameters(M)
New Detection Head	43.6	10.6
Auxiliary Detection Head	43.5	7.7

In Fig. 5, purple represents the minimum object detector, yellow represents the small object detector, green represents the large object detector, and blue represents the maximum object detector. In Fig. 5 (a), FPN [28] connects information at high and low levels in a top-down manner to enrich feature information at different scales. In Fig. 5 (b), PAN [29] After the FPN, a down-to-up pyramid is added to pass low-level localization features to the higher levels. By aggregating feature maps of different levels, the information in each feature map can be fully utilized. In Fig. 5 (c), YOLOv8 combines FPN and PAN to pass features through top-down and bottom-up paths, thereby achieving bidirectional feature fusion.

III. EXPERIMENTS

A. Dataset and Experiment Details

In order to evaluate the MHNet network, the VisDrone2019 dataset [30], the UAVDT dataset [31], and the TinyPerson dataset [38] are selected for verification in this paper.

The VisDrone2019 dataset is a dataset collected by the AISKYEYE team from the Machine Learning and Data Mining Laboratory at Tianjin University. The dataset comprises 261,908 video frames and 10,209 static images, with 6,471 utilized for training, 548 for validation, and 1,610 for testing. In this dataset, the objects are small and densely distributed, with hundreds of objects in each image. These objects cover 10 categories. According to the definition of small objects, the VisDrone dataset contains 68% small objects. The UAVDT dataset, which was presented at ICCV 2018, contains many photos. We used 24,143 of them for training and 16,592 for testing. The photos were extracted from videos taken by drones and include different locations, and the image resolution is 1080×540 pixels. The TinyPerson dataset is designed for human detection in large scenes and contains 1,610 images, of which 794 are used for training and 816 are used for testing. These images were captured using drones and are primarily divided into two categories: sea_person and earth_person.

The hardware environment used in this experiment is an NVIDIA RTX 3090 GPU with 24 GB of graphics memory. The software environment includes: Windows operating system, Python 3.8, PyTorch 1.11.0, and CUDA 11.3. The training parameters are set as follows: batch size 4, 300 training epochs, SGD optimizer, learning rate 0.01, no pre-trained weights, and mosaic enhancement disabled for the last 10 epochs.

B. Evaluation Indicators

The evaluation metrics include mean average precision (mAP), precision (P), recall (R), number of images detected per second (FPS) and number of parameters (Para). Equations (6), (7) and (8) calculate these metrics. Here, mAP50 represents the mean average precision at an IOU of 0.5, while mAP represents the mean mAP for IOU between 0.5 and 0.95.

$$Precision = \frac{TP}{TP + FP} \tag{6}$$

$$\text{Re}call = \frac{TP}{TP + FN} \tag{7}$$

$$mAP = \frac{\sum_{n=1}^{C} AP_i}{C},\ AP = \int_0^1 p\left(r\right) dr \tag{8}$$

TP is true positives, FP is a false positive example, FN is a false negative example, C is the number of categories, AP_i is the average accuracy of i category, and $P\left(r\right)$ is the $P - R$ curve.

TABLE II
COMPARISON RESULTS FOR CHANGING PARAMETER k IN THE MHC MODULE

Method	mAP50(%)	Precision(%)	Parameters(M)
Stage 1	45.8	56.4	13.8
Stage 2	44.8	55.3	10.4
Stage 3	43.6	54.1	11.2
Stage 4	43.0	53.0	8.8

TABLE III
ABLATION EXPERIMENT ON VISDRONE2019

Model	mAP50(%)	R(%)	P(%)	Para(M)
Baseline [25]	39.1	37.9	52.1	11.1
MHC	40.6	40.1	53.2	17.3
SPPFC	39.7	38.4	52.8	14.6
ADH	43.5	41.8	54.1	7.7
ADH+SPPFC	44.2	42.0	55.3	8.7
ADH+SPPFC+MHC	45.8	44.0	56.4	13.8

C. Ablation Experiments

To demonstrate the validity of the method, ablation experiments on the VisDrone2019 dataset were conducted. Only the modified modules were changed, and the other experimental environments were exactly the same. We tested the performance of individual modules and then integrated them. During the integration process, we first replaced the large object detection head with a small object detection head and adopted an efficient feature fusion architecture. Next, the CAA module is combined with SPPF to supplement missing features. Finally, the MHC module is added to the backbone. The results of the ablation experiments are shown in Table III. In Table III, we use ADH to represent Auxiliary Detection Head.

The results of the ablation experiments clearly show the improvement of each module to the model. By continuously adjusting the recall rate and accuracy, the best model effect is obtained. Compared to baseline YOLOv8s, after replacing the detection head and using a new feature fusion method, the recall rate of the model is greatly improved, with a recall rate increase of 3.9% and an mAP50 increase of 4.4%. After the CAA module was added, the accuracy rate increased by 1.2%

(a) (b) (c) (d)

Fig. 6. Visualize the results and the feature heatmap visualization results. (a)(c) show the visualization results and feature heatmap visualization results detected by the baseline model. (b)(d) show the visualization results and feature heatmap visualization results detected by our model.

and the mAP increased by 0.7%. The introduction of MHC further balanced the recall rate and the accuracy rate, resulting in an increase of 1.6% in mAP. Overall, MHNet achieved the best results, with an accuracy rate increase of 4.3%, a recall rate increase of 6.1%, and an mAP increase of 6.7%.

Furthermore, we carried out additional comparative experiments In order to analyze the influence of the detection head situation and the parameters in the MHC module. We increased the number of detection heads in the baseline to 4, which we refer to as new detection head. Auxiliary detection head signifies the replacement of the large detection head with a small one. Table I displays the experimental results of the two models. Auxiliary detection head features 27.3% fewer parameters than new detection head, yet it only enhances the mAP by 0.2%. Therefore, the improvement method of auxiliary detection head was chosen. The parameter k in the MHC module controls the influence of the dimension. The parameter k_1 indicates the number of data splits after the first convolution kernel, while k_2 indicates the adjustment of data in the multiplexing module. Comparative experiments were conducted on different parameters, with k_1=2 and k_2=4 expressed as $(2,\ 4)$. Stage 1, Stage 2, Stage 3, and Stage 4 were used to represent $(2,\ 2)$, $(2,\ 4)$, $(4,\ 2)$, and $(4,\ 4)$, respectively. The comparison results are shown in Table II. Stage 1 achieved the highest mAP. Stage 2 reduced the number of parameters by 24.6% compared to Stage 1, but at the same time the mAP decreased by 1%. The two experiments with k_1=2 performed poorly. After considering the parameter quantity and mAP value, k_1=2 and k_2=2 were selected as the parameters in the MHC module.

D. Comparative Experiments.

To verify the superiority of the MHNet model, we compared it with other mainstream models on VisDrone2019 and UAVDT. The comparison results on VisDrone2019 are shown in Table IV, where mAP and mAP50 reaching 45.8% and 28.2%, respectively. Compared with the benchmark, our MHNet improves mAP and mAP50 by 6.7% and 4.8%, respectively. The comparison results on the UAVDT dataset are shown in Table V. MHNet's mAP50 is 31.2%, which is 0.3% higher than YOLOv8, and its mAP is 18.1%, which exceeds other mainstream models. The comparison results on the TinyPerson dataset are shown in Table VI. MHNet achieves an mAP50 of 21.2%, which is 1.9% higher than YOLOv8, and an mAP of 7.5%. We visualize the detection results of MHNet and the baseline model as shown in Fig. 6. MHNet performs better in small object detection.

IV. Conclusion

In this paper, we propose a novel small object detection network, called MHNet, for object detection in UAV images. First, we propose the MHC module, which performs feature extraction by extracting objects of different sizes one by one in a hierarchical manner to improve the model's multi-scale representation ability. At the same time, the small object auxiliary detection head is used to replace the large object detection head in the detection head part, and a more efficient feature fusion network is combined. This method allows for fuller integration of shallow semantic information and deep positional information, resulting in richer small object features. Secondly, a weighted feature space pyramid pooling module is used in the neck, which incorporates CAA to obtain more contextual dependencies of the object and enhance the central features. Experimental results show that the mAP, recall rate, and accuracy of this algorithm have all been greatly improved compared to the baseline, and a mAP50 of 45.8% was achieved on the VisDrone2019 dataset. However, the research also faces challenges such as the increased complexity of model performance and the feasibility of deployment on drones. Future research will be devoted to improving detection performance while maintaining a lightweight model, exploring the actual performance of deploying drones, and better serving the drone industry.

TABLE IV
COMPARISON RESULTS OF DIFFERENT MODELS ON VISDRONE2019

Model	Backbone	Resolution	mAP(%)	mAP50(%)	Parameters	FPS	FLOPs
Faster-RCNN [11]	ResNet50	1000×600	19.0	31.7	41.2M	37	37.6G
YOLOv5-m [32]	CSPDarknet53	640x640	20.7	36.9	21.2M	69	49G
YOLOv8-m [25]	CSPDarknet53	.640×640	25.9	42.5	25.9M	93	78.9G
DC-YOLOv8 [20]	CSPDarknet53	640×640	24.7	41.5	-	-	-
DSHNet [33]	ResNet50	1000×600	24.6	44.4	-	-	-
RT-DETR [34]	ResNet18	640×640	27.6	45.2	21.3	92	60.0G
VAMYOLOX-m [35]	CSPDarknet53	640×640	27.2	45.1	27.1M	55	54.9G
MHNet(ours)	CSPDarknet53	640×640	28.2	45.8	13.8M	96	64.8G

TABLE V
COMPARISON RESULTS OF DIFFERENT MODELS ON UAVDT

Model	mAP50(%)	mAP(%)
ClusDet [36]	26.5	13.7
DMNet [37]	24.6	14.7
RT-DETR [34]	27.0	16.3
APNet [32]	29.4	18.0
YOLOv8-s [25]	30.9	17.9
MHNet(ours)	31.2	18.1

TABLE VI
COMPARISON RESULTS OF DIFFERENT MODELS ON TINYPERSON

Model	mAP50(%)	mAP(%)
Faster R-CNN	15.1	5.8
YOLOv5	20.7	7.4
YOLOv7-UAV [39]	20.5	6.0
YOLOv8	19.3	7.5
MHNet(ours)	21.2	7.5

REFERENCES

[1] X. Liu, C. Wang, and L. Liu, "Research on Pedestrian Detection Model and Compression Technology for UAV Images," Sensors, vol. 22, no. 23, p. 9171, Nov. 2022, doi: 10.3390/s22239171.

[2] I. Bisio, H. Haleem, C. Garibotto, F. Lavagetto, and A. Sciarrone, "Performance Evaluation and Analysis of Drone-Based Vehicle Detection Techniques From Deep Learning Perspective," IEEE Internet of Things Journal, vol. 9, no. 13, pp. 10920–10935, Jul. 2022, doi: 10.1109/jiot.2021.3128065.

[3] W. Sun, L. Dai, X. Zhang, P. Chang, and X. He, "RSOD: Real-time small object detection algorithm in UAV-based traffic monitoring," Applied Intelligence, vol. 52, no. 8, pp. 8448–8463, Oct. 2021, doi: 10.1007/s10489-021-02893-3.

[4] S. H. Alsamhi et al., "UAV Computing-Assisted Search and Rescue Mission Framework for Disaster and Harsh Environment Mitigation," Drones, vol. 6, no. 7, p. 154, Jun. 2022, doi: 10.3390/drones6070154.

[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 (CVPR), pp. 779–788, Jun. 2016, doi: 10.1109/cvpr.2016.91.

[6] J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6517–6525, Jul. 2017, doi: 10.1109/cvpr.2017.690.

[7] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," arXiv preprint arXiv: 1804.02767, 2018, doi: 10.48550/arXiv.1804.02767.

[8] Z. Ge, S. Liu, F. Wang, Z. Li and J. Sun, "YOLOX: Exceeding YOLO series in 2021," arXiv preprint arXiv:2107.08430, 2021, doi: 10.48550/arXiv.2107.08430.

[9] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, et al., "Ssd: Single shot multibox detector", Computer Vision-ECCV 2016: 14th European Conference, pp. 21-37, October 11–14, 2016, doi: https://doi.org/10.1007/978-3-319-46448-0_2.

[10] R. Girshick, "Fast R-CNN," in Proceedings of the IEEE InternationalConferenceonComputerVision (ICCV), 2015, pp. 1440–1448, doi: 10.1109/ICCV.2015.169.

[11] S. Ren, K. He, R. Girshick, and J. Sun, "FasterR-CNN:Towards realtime object detection with region proposal networks,"IEEE Transactions on Pattern Analysis and Machine Intelligence(TPAMI), vol. 39, no. 6, pp. 1137–1149, 2017, doi: 10.1109/TPAMI.2016.2577031.

[12] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "MaskR-CNN," arXiv preprint arXiv:1703.06870, 2017, doi: 10.48550/arXiv.1703.06870.

[13] X. Lu, B. Li, Y. Yue, Q. Li, and J. Yan, "Grid R-CNN," 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 7355–7364, Jun. 2019, doi: 10.1109/cvpr.2019.00754.

[14] J. Wan, B. Zhang, Y. Zhao, Y. Du, and Z. Tong, "VistrongerDet: Stronger Visual Information for Object Detection in VisDrone Images," 2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW), pp. 2820–2829, Oct. 2021, doi: 10.1109/iccvw54120.2021.00316.

[15] X. Li, W. Diao, Y. Mao, P. Gao, X. Mao, X. Li, and X. Sun, "OGMN: Occlusion-guided multi-task network for object detection in UAV images," ISPRS Journal of Photogrammetry and Remote Sensing, vol. 199, pp. 242–257, May 2023, doi: 10.1016/j.isprsjprs.2023.04.009.

[16] X. Fu, G. Wei, X. Yuan, Y. Liang, and Y. Bo, "Efficient YOLOv7-Drone: An Enhanced Object Detection Approach for Drone Aerial Imagery," Drones, vol. 7, no. 10, p. 616, Oct. 2023, doi: 10.3390/drones7100616.

[17] Z. Song, L. Wang, G. Zhang, C. Jia, J. Bi, H. Wei, Y. Xia, C. Zhang, and L. Zhao, "Fast Detection of Multi-Direction Remote Sensing Ship Object Based on Scale Space Pyramid," 2022 18th International Conference on Mobility, Sensing and Networking (MSN), pp. 1019–1024, Dec. 2022, doi: 10.1109/msn57253.2022.00165.

[18] H. Wang, C. Liu, Y. Cai, L. Chen, and Y. Li, "YOLOv8-QSD: An Improved Small Object Detection Algorithm for Autonomous Vehicles Based on YOLOv8," IEEE Transactions on Instrumentation and Measurement, vol. 73, pp. 1–16, 2024, doi: 10.1109/tim.2024.3379090.

[19] P. Zhang, G. Zhang, and K. Yang, "APNet: Accurate Positioning Deformable Convolution for UAV Image Object Detection," IEEE Latin America Transactions, vol. 22, no. 4, pp. 304–311, Apr. 2024, doi: 10.1109/tla.2024.10472961.

[20] H. Lou, X. Duan, J. Guo, H. Liu, J. Gu, L. Bi, and H. Chen, "DC-YOLOv8: Small Size Object Detection Algorithm Based on Camera Sensor," Apr. 2023, doi: 10.20944/preprints202304.0124.v1.

[21] Z. Guo, C. Wang, G. Yang, Z. Huang, and G. Li, "MSFT-YOLO: Improved YOLOv5 Based on Transformer for Detecting Defects of Steel Surface," Sensors, vol. 22, no. 9, p. 3467, May 2022, doi: 10.3390/s22093467.

[22] Z. Song, Y. Zhang, Y. Liu, K. Yang, and M. Sun, "MSFYOLO: Feature fusion-based detection for small objects," IEEE Latin America Transactions, vol. 20, no. 5, pp. 823–830, May 2022, doi: 10.1109/tla.2022.9693567.

[23] C.-Y. Wang, H.-Y. Mark Liao, Y.-H. Wu, P.-Y. Chen, J.-W. Hsieh, and I.-H. Yeh, "CSPNet: A New Backbone that can Enhance Learning Capability of CNN," 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Jun. 2020, doi: 10.1109/cvprw50498.2020.00203.

[24] S.-H. Gao, M.-M. Cheng, K. Zhao, X.-Y. Zhang, M.-H. Yang, and P. Torr, "Res2Net: A New Multi-Scale Backbone Architecture," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 43, no. 2, pp. 652–662, Feb. 2021, doi: 10.1109/tpami.2019.2938758.

[25] G. Jocher, A. Chaurasia, and J. Qiu, "Ultralytics YOLO," Jan. 2023. [Online]. Available: https://github.com/ultralytics/ultralytics

[26] 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, Sep. 2015, doi: 10.1109/tpami.2015.2389824.
[27] X. Cai, Q. Lai, Y. Wang, W. Wang, Z. Sun, and Y. Yao, "Poly Kernel Inception Network for Remote Sensing Detection," 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 27706–27716, Jun. 2024, doi: 10.1109/cvpr52733.2024.02617.
[28] T.-Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature Pyramid Networks for Object Detection," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Jul. 2017, doi: 10.1109/cvpr.2017.106.
[29] H. Li, P. Xiong, J. An, and L. Wang, "Pyramid Attention Network for Semantic Segmentation," arXiv preprint arXiv:1805.10180, 2018, doi: 10.48550/arXiv.1805.10180.
[30] D. Du, P. Zhu, L. Wen, X. Bian, H. Lin, Q. Hu, T. Peng, J. Zheng, X. Wang, Y. Zhang, et al., "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, doi: 10.1109/ICCVW.2019.00030.
[31] D. Du, et al., "The Unmanned Aerial Vehicle Benchmark: Object Detection and Tracking," Computer Vision – ECCV 2018, pp. 375–391, 2018, doi: 10.1007/978-3-030-01249-6_23.
[32] G. Jocher, "YOLOv5 by ultralytics," May2020. [Online]. Available: https://github.com/ultralytics/yolov5
[33] W. Yu, T. Yang, and C. Chen, "Towards Resolving the Challenge of Long-tail Distribution in UAV Images for Object Detection," 2021 IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 3257–3266, Jan. 2021, doi: 10.1109/wacv48630.2021.00330.
[34] Y. Zhao et al., "DETRs Beat YOLOs on Real-time Object Detection," 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 16965–16974, Jun. 2024, doi: 10.1109/cvpr52733.2024.01605.
[35] Y. Yang, X. Gao, Y. Wang, and S. Song, "VAMYOLOX: An accurate and efficient object detection algorithm based on visual attention mechanism for uav optical sensors," IEEE Sensors Journal, vol. 23, no. 11, pp. 11139–11155, 2023, doi: 10.1109/jsen.2022.3219199.
[36] F. Yang, H. Fan, P. Chu, E. Blasch, and H. Ling, "Clustered Object Detection in Aerial Images," 2019 IEEE/CVF International Conference on Computer Vision (ICCV), pp. 8310–8319, Oct. 2019, doi: 10.1109/iccv.2019.00840.
[37] C. Li, T. Yang, S. Zhu, C. Chen, and S. Guan, "Density Map Guided Object Detection in Aerial Images," 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 737–746, Jun. 2020, doi: 10.1109/cvprw50498.2020.00103.
[38] X. Yu, Y. Gong, N. Jiang, Q. Ye, and Z. Han, "Scale Match for Tiny Person Detection," 2020 IEEE Winter Conference on Applications of Computer Vision (WACV), Mar. 2020, doi: 10.1109/wacv45572.2020.9093394.
[39] Y. Zeng, T. Zhang, W. He, and Z. Zhang, "YOLOv7-UAV: An Unmanned Aerial Vehicle Image Object Detection Algorithm Based on Improved YOLOv7," Electronics, vol. 12, no. 14, p. 3141, Jul. 2023, doi: 10.3390/electronics12143141.

Ziyang Xing was born in 2001 in Handan City, Hebei Province, China. He obtained his bachelor's degree in 2023. Currently, he is a master's student in computer science and technology at Hebei University of Science and Technology (China). His research interests are machine learning and computer vision.

Xuebin Xu was born in 1999 in Zhoukou City, Henan Province, China. He obtained his bachelor's degree in 2023. Currently, he is a master's student in computer science and technology at Hebei University of Science and Technology (China). His research interests are machine learning and computer vision.

Meiling Sun was born in 1998 in Shijiazhuang City, Hebei Province, China. In 2020, she received her bachelor's degree. In 2023, she received her master's degree from Hebei University of Science and Technology (China). Currently, she is a lecturer at Shijiazhuang Preschool Education College. Her research interests are machine learning and computer vision.

Kuihe Yang was born in 1966, in Handan, Hebei Province, China. He received the B.S. degree from Tianjin University (China) in 1988, the M.S. degree from University of Science and Technology Beijing (China) in 1997, and the Ph.D degree in computer application technology from Xidian University (China) in 2004. From 2005 to 2007,he was a Postdoctoral Fellow in Army Engineering University of PLA (China). He went to Manchester University (UK) for short-term training in 2011. Currently, He is professor and master tutor with Hebei University of Science and Technology (China). His research interests include database application technology, artificial intelligence and machine learning.