IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 9 July 2023, accepted 16 July 2023, date of publication 20 July 2023, date of current version 26 July 2023.

*Digital Object Identifier 10.1109/ACCESS.2023.3297218*

RESEARCH ARTICLE

# YOLOv5s FMG: An Improved Small Target Detection Algorithm Based on YOLOv5 in Low Visibility

**YUNCHANG ZHENG, YUNYUE ZHAN, XIAOYING HUANG, AND GAOQING JI**
Hebei University of Architecture, Zhangjiakou, Hebei 075000, China
Corresponding author: Gaoqing Ji (jgq1941@hebiace.edu.cn)

This work was supported in part by the Basic Scientific Research Business Fund Project of Universities in Hebei Province under Grant 2022QNJS03, and in part by the Project of Zhangjiakou Science and Technology Bureau under Grant 2121029C.

**ABSTRACT** Accurate and real-time detection of small targets of pedestrians and cars in video images is indeed crucial for various applications such as autonomous driving and urban management. Existing detection algorithms face challenges related to small targets and low visibility, resulting in issues such as low accuracy, missed detection and reduced detection efficiency. This paper proposes an improved YOLOv5s FMG (Fine-tuning Slice, Multi-spectral Channel Attention, Ghost Bottleneck) detection method based on YOLOv5, which firstly introduces fine-tuning slicing aided hyper inference (SAHI) to generate small target objects by slicing the pictures into the network. Secondly, the multi-spectral channel attention (MCA) module is integrated into the feature extraction network, which enhances the information dissemination among features and strengthens the network's ability to distinguish between foreground and background. Then, the network uses the convolution network to extract features instead of the full connection layer and uses the lightweight Ghost Bottleneck instead of the bottleneck structure. Finally, the prediction part adopts the complete intersection over union (CIoU) loss function to achieve accurate bounding box regression. Based on the experimental results conducted on the self-made dataset, compared to YOLOv5s, the mAP (0.5) of YOLOv5s FMG on the dataset is improved by 9.3%, and the mAP (0.5:0.95) is improved by 2%. At the same time, the frames per second (FPS) is increased by 41.8%, and the number of parameters has been reduced by 18.5%. The proposed method demonstrates successful detection of small targets of pedestrians and vehicles, ensuring its effective applicability under conditions of low visibility.

**INDEX TERMS** Small target detection, YOLOv5, SAHI, MCA, ghost bottleneck, low visibility.

## I. INTRODUCTION

With the development of cities and technology, managing vehicles and pedestrians through surveillance video has become an important issue for urban management [1]. Meanwhile, as autonomous driving technology advances, the accurate detection of pedestrians and vehicles becomes increasingly crucial in the development of autopilot systems. Especially for surveillance video, longer shooting distances often lead to smaller pedestrian and car targets in the captured video frames, which brings great challenges to target detection. However, current detection technologies have poor performance in detecting small targets, especially in low visibility such as fog or low light conditions.

To refine the detection performance in low visibility, traditional algorithms usually use image processing to classify and locate the corresponding targets, and use the method of manually designing features in the feature extraction stage. The basic process is using the sliding window method to select candidate regions in the image firstly [2], then customizing the design features based on different areas, and finally completing the detection according to the customized underlying features [3]. The manual selection of features in this method is not conducive to acquiring image features. Meanwhile, image

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

enhancement is applied before target detection [4], which is complexity and time-consuming. However, most of the traditional methods are only suitable for some specific scenes. In practical scenarios with the targets complex and changeable, the traditional methods can no longer meet the actual needs of the application.

In recent years, target detection algorithms based on neural networks have experienced growing adoption and can be broadly categorized into two groups: two-stage detectors and single-stage detectors. The two-stage algorithm can generate target candidate frames by various algorithms, and then classify the targets by convolutional neural network. Classic two-stage detection algorithms include R-CNN [5], Faster R-CNN [6], Mask R-CNN [7] and so on. The single-stage detection algorithm does not generate candidate frames, but directly transforms the positioning problem of the object boundary frame into a regression problem for processing. The single-stage detection algorithms mainly include SSD [8], YOLOv2, YOLOv3, YOLOv4, YOLOv5 [9], etc. Among these two kinds of deep learning techniques, the YOLO models are considered a practical target detection algorithm, which can directly create various types of detection frames and confidence levels through appropriate neural networks. The YOLO models have been employed in a wide range of scene detection tasks, and their execution accuracy and speed are very high, leading to the emergence of several small target detection methods based on YOLOv5 in low visibility. In 2022, Image-Adaptive YOLO [10] was proposed for object detection in adverse weather conditions. In 2022, Tan et al. [11] proposed an infrared sensation-based salient target enhancement method. In 2022, mmWave-YOLO [12] was proposed, which enabled accurate object classification and location recognition by applying different detectors to each distance data. In 2023, WilDect-YOLO [13] was proposed, which was an efficient and robust computer vision-based accurate object localization model under various challenging environments. In 2023, EnsembleNet [14] was suggested, leveraging the strengths of both the Faster R-CNN and YOLO models, which was a hybrid approach for vehicle detection. In 2023, Li et al. [15] proposed a visible light small object detection based on YOLOv5, which refined the performance of small target detection. However, existing detection models have faced challenges in achieving high performance while maintaining a balance between accuracy and efficiency.

As shown in FIGURE 1, the current small target detection methods have poor performance in low visibility such as fog and low light conditions due to the following challenges:

(1) Reduced contrast: Low visibility conditions like fog, rain, or snow decrease the contrast between the target and its background. This impedes the ability of detection algorithms to differentiate the target from its surroundings, resulting in decreased accuracy.

(2) Distorted boundaries: Adverse weather conditions cause distortions in the shape and boundaries of objects, rendering them blurry or indistinct. This confusion can lead to false positives or false negatives in target detection algorithms.

(a). Highway on a foggy day.

(b). City in the night.

**FIGURE 1.** Typical low visibility scenarios.

(3) Limited range and resolution: The range and resolution of cameras are significantly curtailed in low visibility scenarios. This limitation affects the ability to detect and precisely locate targets, particularly at longer distances.

In order to solve the mentioned problems and optimize the detection of small targets, this paper proposes YOLOv5s FMG. The main contribution of our work can be summarized as follows:

(1) The proposed method adds the principle of fine-tuning slicing aided hyper inference (SAHI) [16], which realizes the slice operation through the input side.

(2) Multi-spectral channel attention (MCA) [17]module is added into the YOLOv5s network using the multi-scale detection method.

(3) The improved network uses convolutional network to extract features instead of using the fully connected layer, which further improves the network performance and reduces the network weight by using the Ghost module [18].

(4) The complete intersection over union (CIoU) [19] loss function is applied to enhance the original YOLOv5's detection ability for small targets and low visibility scenarios.

The remainder of this paper is structured as follows: Section II reviews the principle of YOLOv5 and describes the improvement details of YOLOv5s FMG. In Section III, the dataset and the results of neural network training are described.

The results and discussions of comparative experiments and ablation experiments are presented. Finally, Section IV illustrates the main conclusions.

## II. PRINCIPLE AND METHOD IMPROVEMENT

### A. PRINCIPLE OF YOLOv5 TARGET DETECTION ALGORITHM

YOLOv5 is the fifth generation of YOLO, which has four structures: YOLOv5s for small size, YOLOv5m for medium size, YOLOv5l for large size and YOLOv5x for extra-large size. The difference between them is the number of architectural parameters. YOLOv5 network structure is divided into four parts: input, Backbone, Neck and Prediction [20], which uses Mosaic data enhancement method in the data input part. This algorithm is improved based on CutMix data enhancement method, with new functions of adaptive anchor box computing and adaptive image scaling. The inputs are spliced by random scaling, random cropping and random arrangement to enhance the dataset. During each training, the best anchor frame values in different training sets are adaptively calculated, the images with different aspect ratios are adaptively scaled, and the minimum black edges are adaptively added to the original images. Focus and cross stage partial (CSP) structure are mainly used in Backbone. Focus structure is introduced into YOLOv5 for the first time to directly process the input pictures.

YOLOv5 adds feature pyramid network (FPN) and path aggregation network (PAN) structure to the Neck [21], which is improved based on YOLOv4. CSP2 structure is designed with reference to CSP net to strengthen the ability of network feature fusion. Prediction improves the loss function of the boundary anchor frame from intersection over union (IoU) loss to generalized IoU loss. In the post-processing process of target detection, YOLOv5 uses weighted non-maximum suppression (NMS) operation to screen multiple target anchor frames [22].

### B. IMPROVEMENT

#### 1) SLICE-ASSISTED FINE-TUNING AND REASONING

Compared with YOLOv4, YOLOv5' s backbone network has a new Focus structure. The important function of the Focus structure is to realize the slicing operation. As shown in FIGURE 2, in the YOLOv5s network model, an ordinary image with a size of $3 \times 608 \times 608$ is input into the network, and the feature image with a size of $12 \times 304 \times 304$ is converted by one focus-slicing operation, and then it is subjected to an ordinary convolution operation of 32 kernels [23].

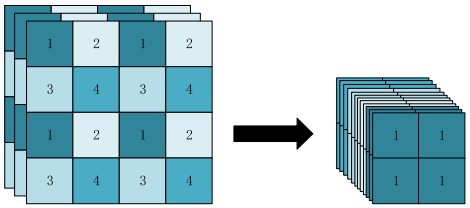



**FIGURE 2.** Focus structure screen cutting operation.

To fully express the features of small targets and enhance the detection accuracy, this paper divides the input image into overlapping slices in the process of fine-tuning. Compared with the images directly input into the network, this method can make the small targets in the images produce a relatively large pixel area, keep the features of small targets as much as possible, avoid the feature loss caused by too few pixels of small targets in the original images, and keep the detection accuracy of large targets in the images.

Fine-tuning is one of the most popular transfer learning strategies in the application of neural networks. In the practice of deep learning, the network is rarely trained from scratch because the data set is not large enough. The common practice is to use the pre-trained network to fine-tune the network parameters. Fine-tuning uses the known network structure and training parameters to adjust the parameters of several layers in front of the output layer to achieve the purpose of initializing the network. This process effectively utilizes the powerful generalization ability of deep neural network and eliminates the need of designing complex models and time-consuming training [16].

Therefore, when training our own network, we only need to build a smaller data set, spend a shorter training period, and fine-tune the network weight based on pre-training, instead of reusing a large data set for training.

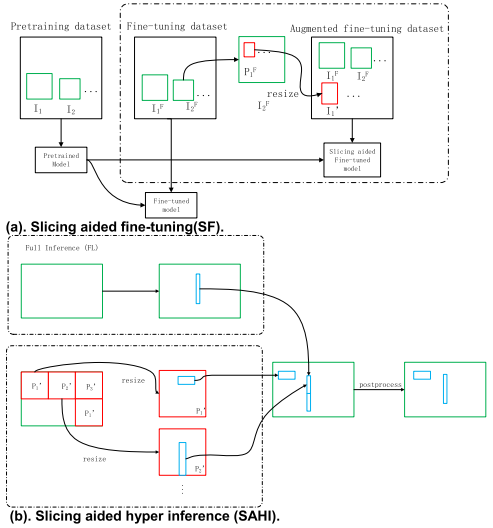



**FIGURE 3.** The SF and SAHI structures.

At the same time, we use slicing to assist this process. As shown in FIGURE 3(a), the small target area is extracted from the picture to form a patch and combined into the original data set, so that the small target is enlarged, thus assisting the fine-tuning of network initialization parameters.

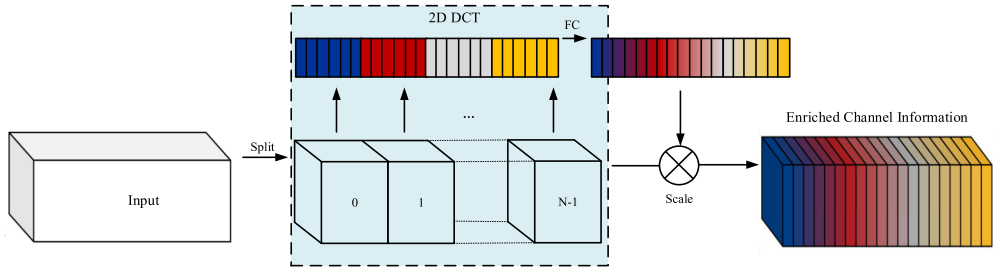



**FIGURE 4.** The brief network structure of MCA.

However, with the decrease of patch size, slice-assisted fine-tuning cannot cover the image area of large objects, which may lead to poor detection performance of large objects. Therefore, the slice-aided reasoning method is used to solve the problems in the fine-tuning process.

As shown in FIGURE 3(b), the original input image $I$ is divided into $L$ overlapping slices of $p \times q$, and the patch size is adjusted based on keeping the aspect ratio unchanged [24]. The forward transmission of target detection is applied to each overlapping slice to reason for small targets, and the optional full inference of the original image is used to detect large targets. Finally, NMS is used to combine the two kinds of prediction results into the original image. In the NMS process, by traversing all the detection frames, the detection frame whose IoU is greater than a certain threshold T or the detection probability is lower than the threshold T with the highest current confidence is eliminated, and finally the target prediction frame is obtained.

### 2) MCA-FUSION

Traditional channel attention modules are dedicated to constructing various channel importance weight functions. SeNET [25] proposes a channel attention mechanism, which performs global average pooling (GAP) on channels, and then uses the full connection layer to adaptively calculate the weight of each channel. ECANet [26] uses one-dimensional convolution layer locally to reduce the redundancy of all connection layers and has achieved remarkable performance improvement. However, there is a lack of feature diversity when dealing with different inputs.

To solve this problem, MCA is used in this paper. The brief network structure of MCA is shown in FIGURE 4. The first step is to calculate the results of each frequency component in the channel attention separately. Following this, the frequency components are combined. Finally, the obtained results are utilized to identify the Top-k frequency components with the best performance, which are then selected and retained. In addition, GAP is the lowest frequency of discrete cosine transform (DCT), and only using GAP is equivalent to discarding other frequency components containing a large amount of information in the feature channel [27]. In this work, GAP is used as a pre-processing method to discard other frequency component information except the lowest frequency component. This method is extended in the frequency domain, and more frequency component information is naturally embedded in MCA framework. Because different frequency components contain different information, more information can be extracted from redundant channels.

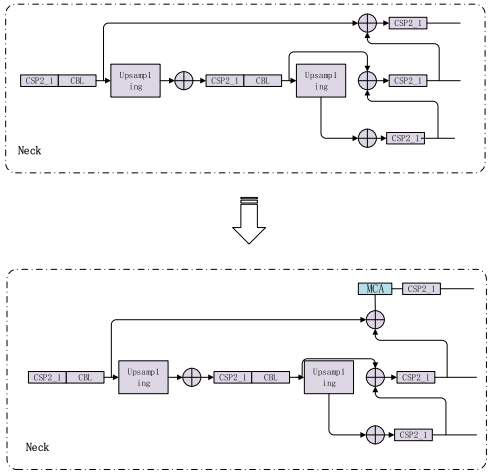



**FIGURE 5.** The YOLOv5s model incorporating MCA module.

As shown in FIGURE 5, the MCA module is integrated into the Neck to enhance the processing and extraction of data. By selectively attending to relevant vectors and disregarding irrelevant ones, the neural network aims to eliminate the interference caused by excessive information and prioritize the calculation of feature vectors. This approach enables effective data information extraction and enhances both the accuracy of calculations and computational efficiency. A channel pruning strategy is proposed to compress it, and the optimal large model is realized as an ultrasmall model for real-time detection.

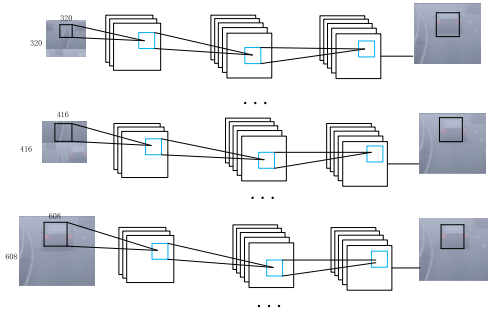
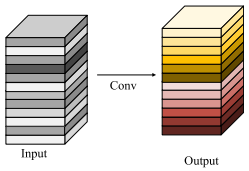
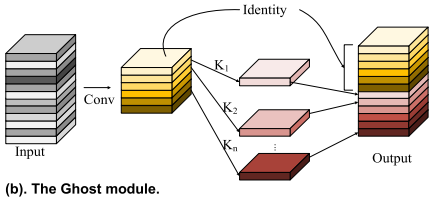



**FIGURE 6.** The schematic diagram of the multi-scale detection process.




(a). The convolutional layer.




(b). The Ghost module.

**FIGURE 7.** The convolutional layer and the Ghost module.

### 3) IMPROVED MULTI-SCALE DETECTION

The improved network uses the convolution network to extract features instead of the full connection layer for small target detection. Therefore, in the process of model training, there is no need to fix the size of the input image [28]. Because the improved network model contains five residual structures, the size of the input image should be a multiple of 32 and the minimum size of the image should be 1/32 of the input image during training. Divide the self-made data set pictures into various sizes, such as 320, 352, 384, . . . 608, etc. During the iterative training of the model, the input size of an image is randomly changed every 10 times, so that the model can adapt to the changes of images of different sizes. The schematic diagram of the multi-scale training process is shown in FIGURE 6. The network model trained by the multi-scale strategy can accept images of any size as input, which is helpful to enhance the generalization ability of the model.

### 4) LIGHTWEIGHT GHOST BOTTLENECK

Compared with the traditional convolution, Ghost Net [29] is divided into two steps. As shown in FIGURE 7, firstly, Ghost Net uses normal convolution calculation to get feature maps with fewer channels, then uses cheap operation to get more feature maps, and then concatenates different feature maps together to form a new output. Compared with the widely used unit of $1 \times 1$ point-wise evolution, the main convolution in Ghost module can customize the kernel size, adopt point wise evolution to process features across channels, and then adopt depth convolution to process spatial information. The Ghost module adopts the splicing method, which further reduces the amount of calculation. Our work uses the lightweight Ghost Bottleneck instead of the bottleneck structure.

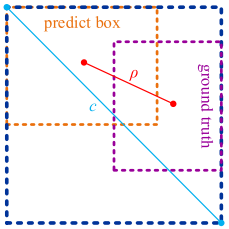



**FIGURE 8.** The CIoU loss function for bounding box regression.

### 5) LOSS FUNCTION

The original YOLOv5 uses generalized intersection over union (GIoU) [30] to process the prediction boundary box, which can effectively address situations where the predicted bounding box and the actual bounding box do not intersect. However, when the two boxes are contained within each other or have different length-to-width ratios, the GIoU fails to accurately determine the relationship between the two boxes, which cannot reflect the intersection position [31].

In order to overcome the shortcomings of GIoU, CIoU [32] is adopted as the regression loss function in our study. CIoU introduces a penalty function to account for the scale of the overlapping area, distance between center points, and aspect ratio. CIoU stabilizes the regression of the target bounding boxes and enhances the accuracy of the prediction. As shown in FIGURE 8, $c$ represents the diagonal distance of the minimum closure area that contains both the prediction box and the ground truth, and$\rho$represents the distance between the two center points of predict box and ground truth. The CIoU loss function can be defined as:

$$IoU = IoU - \frac{\rho^2(b, b^{gt})}{c^2} - \alpha v \tag{1}$$

$$v = \frac{4}{\pi^2}\left(\arctan\frac{w^{gt}}{h^{gt}} - \arctan\frac{w}{h}\right)^2 \tag{2}$$

$$\alpha = \frac{v}{(1 - IoU) + v} \tag{3}$$

$$Loss_{CIoU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v \tag{4}$$

where $b$ and $b^{gt}$ represent the center points of prediction Box B and ground-truth box $B^{gt}$, respectively. The weight

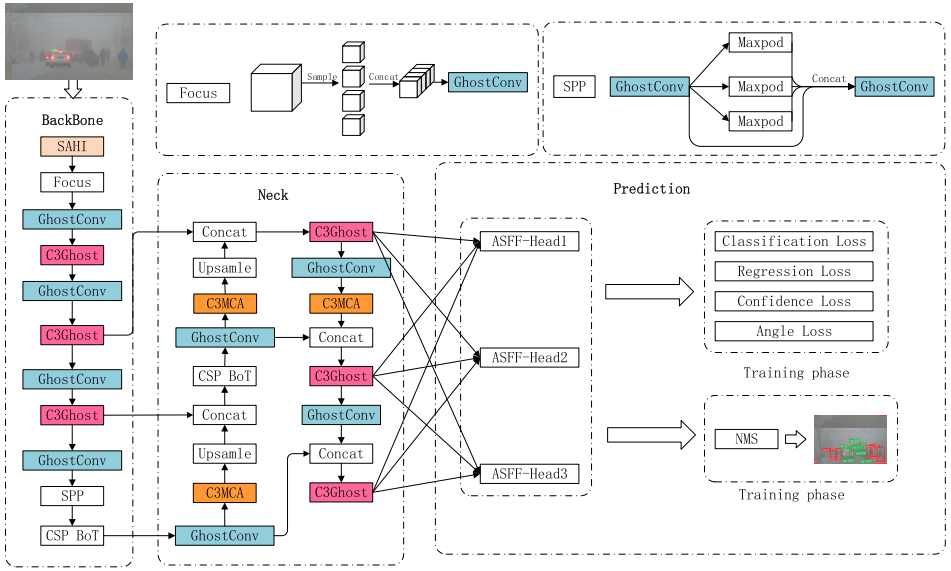



**FIGURE 9.** The network structure of YOLOv5s-FMG.

parameter is denoted as $\alpha$, and the similarity of aspect ratios is measured using a parameter labeled as $v$.

#### 6) YOLOV5S FMG

The overall structure of the algorithm proposed in this article is shown in FIGURE 9. Firstly, we employ SAHI to fine-tune the input image, and the resulting enhanced image is then used as input for further processing. After the image is inputted into the model, the first layer applies a convolution operation with a $6 \times 6$ kernel size, which is equivalent to the original Focus module. Furthermore, all other convolution operations are replaced with GhostConv. The C3 structures are substituted with C3Ghost. For feature fusion using the Neck network, the C3MCA module is employed to enhance the significance of crucial features. At the output stage, GIoU is replaced with CIoU. Weighted NMS is used to eliminate redundant prediction boxes before the final image output.

## III. EXPERIMENTS AND DISCUSSION

The experimental process can be divided into three main parts: dataset construction, model training, and small target detection, as illustrated in FIGURE 10. Firstly, we constructed a self-made dataset by collecting images and performing image pre-processing techniques. Then, by adjusting parameters and model training, we got the model weights of YOLOv5s FMG. Finally, the small target detection was performed by using the trained weights, and the detection results of different methods were compared and analyzed.

### A. DATASET AND PRE-PROCESSING

The images in the dataset of this experiment were mainly taken from daily life and partially collected from the Internet, for a total of 1258 images. The assignment of images to training, validation, and testing images is done randomly, in a ratio of 7:2:1. As shown in FIGURE 11, in order to avoid sample imbalance and make the dataset more in line with practical application scenarios, four image pre-processing methods, including enhancing, mirroring, blurring and Gaussian noising are applied to each group. After the image pre-processing stage, a self-made dataset is constructed with a total of 6290 images, which includes both the original images and the pre-processed images.

The labeling software used for annotating the dataset in YOLO format is LabelImg. To better evaluate the detection performance of small targets in low visibility, the dataset has been labeled with two categories: people and car. The annotations for the dataset are saved as XML files following the PASCAL VOC format.

### B. NETWORK TRAINING

In the network training, Windows 10 (64-bit) is used to build the experimental development platform. The CPU is configured as an 11th Gen Intel (R) Core (TM) i7-13900K CPU @ 5.40 GHz. The GPU is an NVIDIA GeForce RTX 3080 (10 GB). The CUDA version is 11.3. The Python version is 3.8. The deep learning framework used in the experiment is PyTorch.

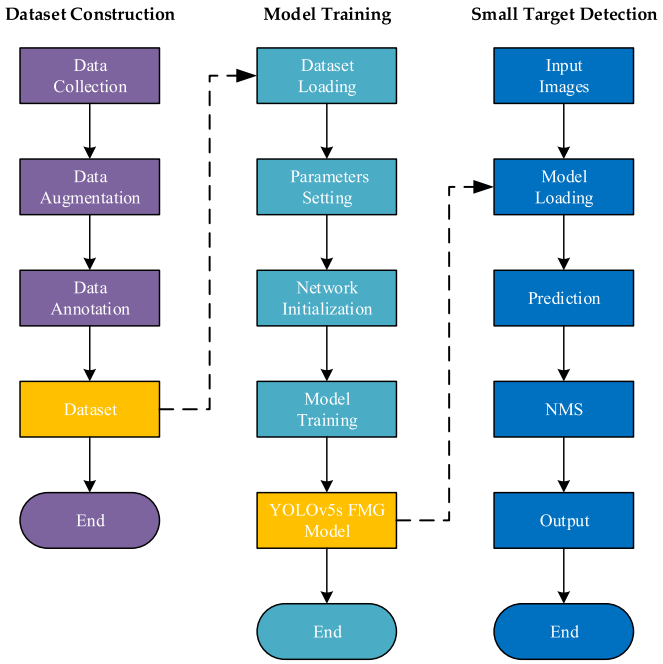



**FIGURE 10.** The flowchart of dataset construction, model training and small target detection.

Before the network training, the hyper-parameters are set to achieve the best performance of the model and prevent the model from overfitting. The batch size is 32, and the learning rate is set at 0.001. We set the number of iterations to 300, and we employ Adam as the optimizer. As shown in TABLE 1, the loss function value drops sharply at 0 to 200 iterations and drops slowly at 200 to 300 iterations. After 300 epochs, the loss value tends to be stable, and the model reaches the best state.

### C. EVALUATION INDICATORS

The evaluation indicators in this paper include precision, recall, average precision (AP), mean average precision (mAP) and frames per second (FPS). Precision refers to the probability that all samples detected as positive by the model are indeed positive samples. Recall represents the probability that the model correctly identifies positive samples from the total number of actual positive samples. The precision and recall are respectively expressed by Equations (5) and (6):

$$Precison = \frac{T_P}{T_P + F_P} \tag{5}$$

$$Recall = \frac{T_P}{T_P + F_N} \tag{6}$$

where $T_P$ (True Positives) represents the number of positive samples correctly predicted by the model. Similarly, $F_P$ (False Positives) represents the number of positive samples incorrectly predicted by the model, and $F_N$ (False Negatives) represents the number of negative samples incorrectly predicted by the model.

AP is a key performance indicator that tries to remove the dependency of selecting one confidence threshold value and is defined by the average precision in the area under the precision-recall curve, which is denoted as Equation (7). Meanwhile, mAP is usually applied to evaluate the results combining precision and recall, which is calculated by taking the average of AP across all the classes under consideration and denoted as Equation (8):

$$AP = \int_0^1 Precision(t)dt \tag{7}$$

$$mAP = \frac{1}{N}\sum_{i=1}^{N} AP_i \tag{8}$$

In addition, mAP (0.5) and mAP (0.5:0.95) are often employed as evaluation metrics in experiments. mAP (0.5) is the mAP with the IoU set to 0.5 and mAP (0.5:0.95)

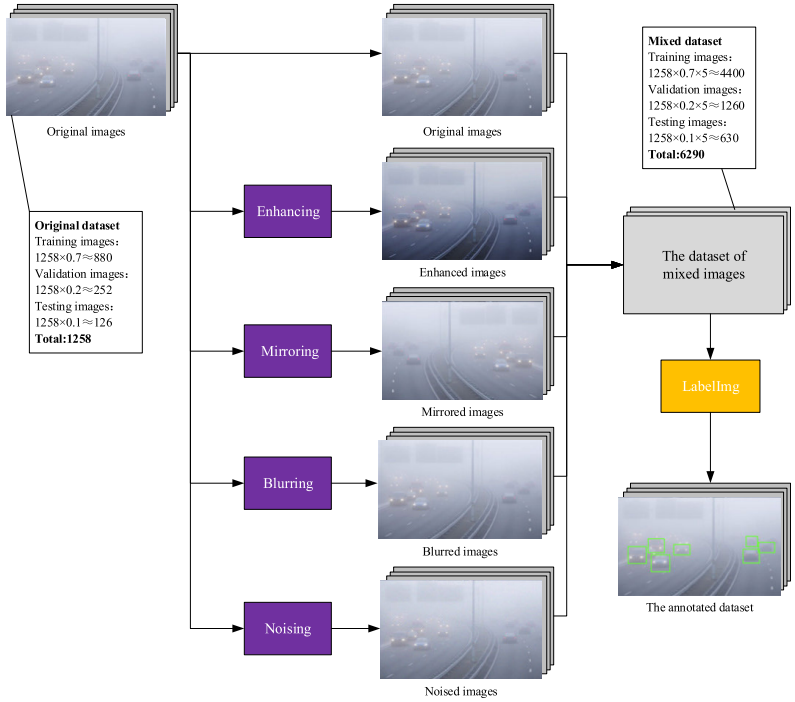



**FIGURE 11.** Image pre-processing and annotation.

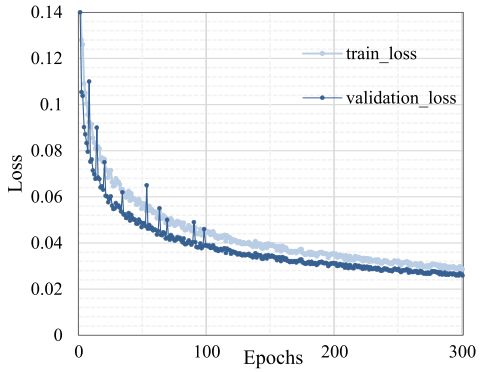



**FIGURE 12.** Training and validation losses.

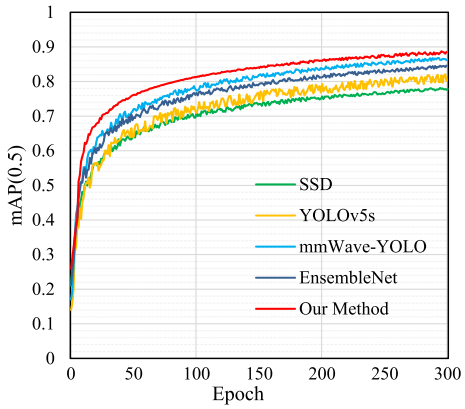



**FIGURE 13.** The mAP (0.5) of related algorithms in training.

represents the mAP computed across a range of IoU thresholds from 0.5 to 0.95.

### D. COMPARATIVE EXPERIMENTS

In order to further verify the detection performance of the proposed YOLOv5s FMG in this article, we compared our method with several one-stage object detection algorithms such as SSD and YOLOv5s, and the methods discussed before, which are mmWave-YOLO [11] and EnsembleNet [14].

**TABLE 1.** The mAP (0.5) comparison of different methods through different epochs.

| Model | Epoch=0 | Epoch=50 | Epoch=100 | Epoch=150 | Epoch=200 | Epoch=250 | Epoch=300 |
|---|---|---|---|---|---|---|---|
| SSD | 0 | 0.63864 | 0.69542 | 0.73302 | 0.75018 | 0.76383 | 0.77613 |
| YOLOv5s | 0 | 0.63946 | 0.71491 | 0.76766 | 0.76818 | 0.78835 | 0.80213 |
| mmWave-YOLO | 0 | 0.71415 | 0.77760 | 0.81577 | 0.83891 | 0.84962 | 0.86236 |
| EnsembleNet | 0 | 0.68820 | 0.76136 | 0.78866 | 0.81205 | 0.83464 | 0.84590 |
| **Our Method** | **0** | **0.75983** | **0.81295** | **0.84084** | **0.86025** | **0.87664** | **0.88502** |

**TABLE 2.** Comparison results of different methods.

| Model | Car-AP | Person-AP | mAP (0.5) | FPS | Parameters ($10^6$) | Model Size (MB) |
|---|---|---|---|---|---|---|
| SSD | 78.7 | 76.2 | 77.6 | 26.1 | 24.01 | 92.8 |
| YOLOv5s | 80.8 | 83.7 | 79.2 | 24.9 | 7.07 | 12.9 |
| mmWave-YOLO | 87.9 | 85.4 | 86.2 | 30.4 | 11.6 | 9.6 |
| EnsembleNet | 87.1 | 81.2 | 84.5 | 27.6 | 7.32 | 11.3 |
| **Our Method** | **90.2** | **84.7** | **88.5** | **35.3** | **5.76** | **7.5** |

**TABLE 3.** Ablation experiments.

| Model | +SAHI | +MCA | +Ghost | +CIoU | Precision | Recall | mAP (0.5) | mAP (0.5:0.95) |
|---|---|---|---|---|---|---|---|---|
| Original YOLOv5s | | | | | 76.2 | 75.3 | 79.2 | 56.4 |
| - | √ | | | | 76.8 | 76.2 | 81.8 | 57.1 |
| - | | √ | | | 78.6 | 87.1 | 83.4 | 57.5 |
| - | | | √ | | 82.5 | 85.3 | 86.3 | 57.9 |
| - | | | | √ | 76.2 | 80.4 | 82.1 | 57.3 |
| - | √ | √ | | | 80.3 | 82.6 | 86.2 | 58.1 |
| - | √ | √ | √ | | 79.8 | 87.2 | 85.4 | 57.9 |
| **Our Method** | √ | √ | √ | √ | **84.5** | **84.7** | **88.5** | **58.4** |

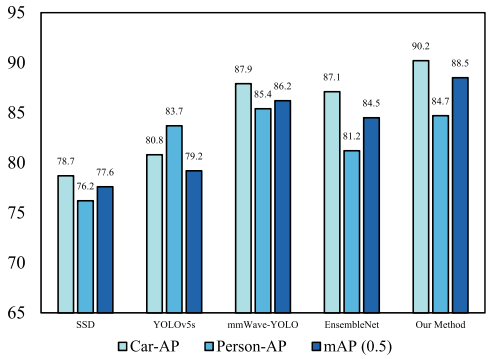



**FIGURE 14.** The Car-AP, Person-AP and mAP (0.5) of different methods.

As shown in FIGURE 13, in the training process, the mAP (0.5) curve of our method is compared with that of other methods for approximately 300 epochs. TABLE 1 shows that the mAP (0.5) for each method experiences a rapid increase in the initial 50 epochs and tends to be stable after the epoch reaches 200.

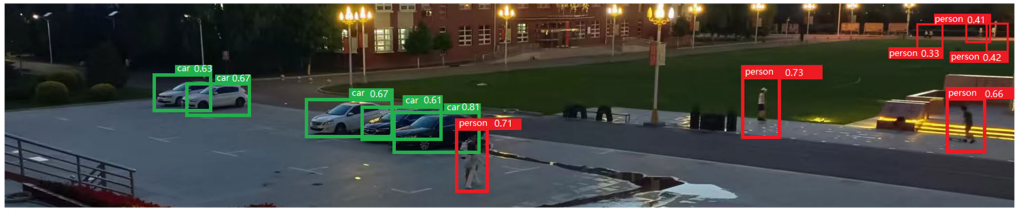
During the training process, the curve of our method consistently surpasses the curves of other methods, indicating that the YOLOv5s FMG network achieves higher detection accuracy. Furthermore, the curve of the improved model exhibits a smoother progression, suggesting improved stability. Finally, the optimal mAP (0.5) of our method is 0.885, which is higher than 0.802 of the original YOLOv5s model and increased by approximately 8.3%, verifying the improvement of the YOLOv5s FMG.

To evaluate the detection performance of the YOLOv5s FMG proposed in this paper, a comparative experiment was conducted on a self-made dataset consisting of various scenes in low visibility. YOLOv5s FMG was compared with SSD, original YOLOv5s, mmWave-YOLO and EnsembleNet. Under the same experimental environment, the weight file with the best training effect is saved as the weight file. The evaluation indicators used for comparative experiments include AP, mAP (0.5), FPS, number of parameters and mode size.

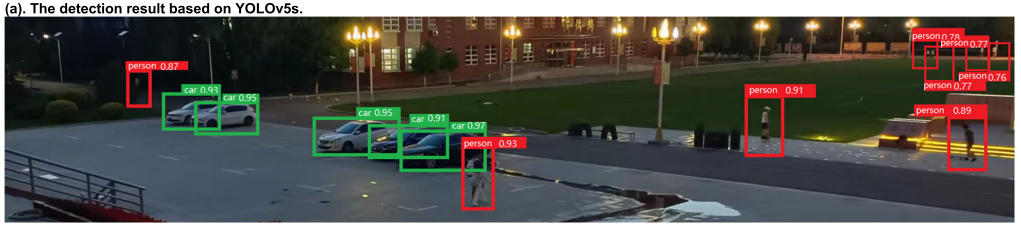
As shown in TABLE 2 and FIGURE 14, our algorithm has been optimized in terms of detection speed, accuracy, model size, and generalization performance.

Compared to SSD, the AP of car and person increase significantly, the mAP (0.5) has improved by 10.9% and the FPS is increased by 35.2%. Meanwhile, the number of parameters has been reduced by 76%, the model size decreases by 91.9%.

Compared to the original YOLOv5s, the Car-AP and Person-AP increase by 9.4% and 1%, the mAP (0.5) has

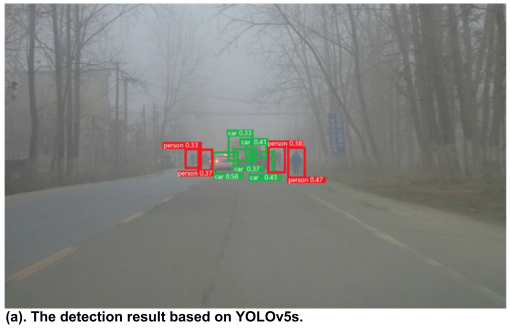



(a). The detection result based on YOLOv5s.

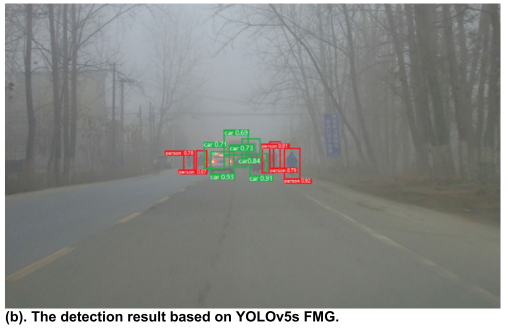



(b). The detection result based on YOLOv5s FMG.

**FIGURE 15.** Comparison of detection results in low light environment.



(a). The detection result based on YOLOv5s.



(b). The detection result based on YOLOv5s FMG.

**FIGURE 16.** Comparison of detection results in foggy environment.

improved by 9.3% and the FPS is increased by 41.8%. Meanwhile, the number of parameters has been reduced by 18.5%, resulting in a substantial decrease in the model size by 41.9%.

Compared to mmWave-YOLO, the Car-AP increases by 2.3%, but the Person-AP decreases by 0.7%. Meanwhile, the mAP (0.5) has improved by 2.3% and the FPS is increased by 16.1%. However, except for a slight decrease in Person-AP, all other evaluation indicators show improvement.

Compared to EnsembleNet, the Car-AP, Person-AP and mAP (0.5) are improved by 3.1%, 3.5% and 4%. The FPS is improved by 27.9% with the number of parameters and the model size decreased by 21.3% and 33.6%.

Considering the complexity of each model and the actual detection results, it can be generally concluded that YOLOv5s FMG performs better compared to the other models.

### *E. ABLATION EXPERIMENTS AND RESULT DISCUSSIONS*

To evaluate the optimization effects of each proposed module in the network, this section employs ablation experiments for verification purposes.

The function of each module is shown in TABLE 3, compared to YOLOv5s, using only SAHI on the input side can increase the mAP (0.5) by 2.6%, and there is a small increase in precision and recall. Adding the MCA module individually to the Neck network can increase the mAP (0.5) by 4.2% compared to YOLOv5s. The addition of Ghost Net to the trunk of YOLOv5s can increase the mAP (0.5) by 7.1%, and the precision and recall increase significantly. The using of the CIoU Loss function can increase the mAP (0.5) by 2.9%.

Additionally, by adding SAHI and MCA to the original YOLOv5s, the mAP (0.5) is enhanced by 7%. Then adding the Ghost Net can increase the mAP (0.5) by 6.2%. When all improvement strategies were applied simultaneously,

mAP (0.5) is improved by 9.3%, mAP (0.5:0.95) is increased by 2%, precision is increased by 8.3%, and recall is increased by 9.4% compared to the YOLOv5s, demonstrating the superiority of the proposedYOLOv5s FMG.

In order to visually evaluate the detection effect of our method, we conducted a performance comparison between the original YOLOv5s and the improved method, and a subset of images was randomly sampled from the dataset for detection. As shown in FIGURE 15, in environments with low light conditions, the YOLOv5s model occasionally resulted in missed detections. For example, in the dim environment on the left in FIGURE 15 (a), the YOLOv5s failed to detect the pedestrian, leaving him undetected. Only one pedestrian with light interference in the upper right corner was successfully detected, and the other was missed. In comparison, our method can detect targets that were missed by the YOLOv5s model, which is shown in FIGURE 15 (b). As shown in FIGURE 16, during foggy weather, the visibility of the road environment is low, resulting in a significant decrease in the overall detection accuracy. As shown in FIGURE 16(a), the YOLOv5s model failed to detect vehicle on the left and the vehicle obstructed by other vehicles in the middle, and even two pedestrians on the right have been mistakenly detected as one pedestrian. In contrast, all the targets are correctly detected by our method in FIGURE 16(b). The proposed YOLOv5s FMG algorithm effectively addresses the challenge of detecting small targets in low visibility conditions. It significantly reduces missed detections and minimizes errors, leading to more accurate and reliable target detection results.

## IV. CONCLUSION

In this paper, to address the challenges of small target detection in low visibility conditions, we propose YOLOv5s FMG to improve the efficiency and accuracy of detection. The main contributions of this paper include: (1) SAHI is incorporated at the input side, allowing for better adaptation to target characteristics and improved detection performance. (2) The MCA module is integrated into the Neck of the YOLOv5s FMG model, enhancing the model's ability to detect small targets in low visibility scenarios. (3) The network employs a convolutional network for feature extraction instead of using fully connected layers, and adopts the lightweight Ghost Bottleneck module as a substitute for the traditional bottleneck structure. (4) The CIoU loss function is adopted to improve the accuracy and localization.

We created a dataset consisting of various low-visibility scenes for model training and performance testing. The experimental results demonstrate that, when compared to the original YOLOv5s, our proposed method exhibits significant improvements. Specifically, the mAP (0.5) has increased by 9.3%, while the FPS has shown a remarkable boost of 41.8%. Additionally, the number of parameters has been reduced by 18.5%, resulting in a substantial decrease in the model size by 41.9%. Meanwhile, compared with other models proposed in recent years, the YOLOv5s FMG has advantages in terms of mAP, detecting speed and model size.

However, the proposed method in this article also has some limitations:

(1) When there is an overlap between tiny light sources and the detection targets, it can lead to a decrease in the detection accuracy of the algorithm. Further research and development are necessary to develop techniques that can effectively differentiate between tiny light sources and actual detection targets.

(2) The images used in this paper were carefully selected and may differ from real-world scenarios. Therefore, it is important to conduct further testing and adjustment of the algorithm in real-world scenes to obtain more accurate and reliable results.

## REFERENCES

[1] L. Xiaomeng, F. Jun, and C. Peng, "Vehicle detection in traffic monitoring scenes based on improved YOLOV5s," in *Proc. Int. Conf. Comput. Eng. Artif. Intell. (ICCEAI)*, Jul. 2022, pp. 467–471.

[2] C. K. Koç, "Analysis of sliding window techniques for exponentiation," *Comput. Math. Appl.*, vol. 30, no. 10, pp. 17–24, Nov. 1995.

[3] X. Guo, Y. Li, and H. Ling, "LIME: Low-light image enhancement via illumination map estimation," *IEEE Trans. Image Process.*, vol. 26, no. 2, pp. 982–993, Feb. 2017.

[4] B. Wu and R. Nevatia, "Tracking of multiple, partially occluded humans based on static body part detection," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, New York, NY, USA, Jun. 2006, pp. 951–958.

[5] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Columbus, OH, USA, Jun. 2014, pp. 580–587.

[6] 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.

[7] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2980–2988.

[8] W. Liu, D. Anguelov, and D. Erhan, "SSD: Single shot multibox detector," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 21–37.

[9] 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)*, Jun. 2016, pp. 779–788.

[10] W. Liu, "Image-adaptive YOLO for object detection in adverse weather conditions," in *Proc. AAAI Conf. Artif. Intell.*, vol. 36, no. 2, 2022, pp. 1–9.

[11] H. Tan, D. Ou, L. Zhang, G. Shen, X. Li, and Y. Ji, "Infrared sensation-based salient targets enhancement methods in low-visibility scenes," *Sensors*, vol. 22, no. 15, p. 5835, Aug. 2022.

[12] A. Kosuge, S. Suehiro, M. Hamada, and T. Kuroda, "MmWave-YOLO: A mmWave imaging radar-based real-time multiclass object recognition system for ADAS applications," *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–10, 2022.

[13] A. M. Roy, J. Bhaduri, T. Kumar, and K. Raj, "WilDect-YOLO: An efficient and robust computer vision-based accurate object localization model for automated endangered wildlife detection," *Ecol. Informat.*, vol. 75, Jul. 2023, Art. no. 101919.

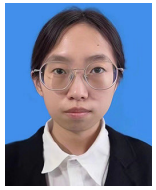
[14] U. Mittal, P. Chawla, and R. Tiwari, "EnsembleNet: A hybrid approach for vehicle detection and estimation of traffic density based on faster R-CNN and YOLO models," *Neural Comput. Appl.*, vol. 35, no. 6, pp. 4755–4774, Feb. 2023.

[15] Y. Li, Y. Liu, S. Hou, Q. Qiu, P. Xie, and Y. Fan, "Visible light small object detection based on YOLOv5," *Proc. SPIE*, vol. 12617, Apr. 2023, Art. no. 126171S.

[16] F. C. Akyon, S. O. Altinuc, and A. Temizel, "Slicing aided hyper inference and fine-tuning for small object detection," 2022, *arXiv:2202.06934*.

[17] Z. Qin, P. Zhang, F. Wu, and X. Li, "FcaNet: Frequency channel attention networks," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 763–772.
[18] K. Han, Y. Wang, Q. Tian, J. Guo, C. Xu, and C. Xu, "GhostNet: More features from cheap operations," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 1577–1586.
[19] C. Zhang, A. Xiong, X. Luo, C. Zhou, and J. Liang, "Electric bicycle detection based on improved YOLOv5," in *Proc. 4th Int. Conf. Adv. Comput. Technol., Inf. Sci. Commun. (CTISC)*, Apr. 2022, pp. 1–5.
[20] H. Zhang, M. Tian, G. Shao, J. Cheng, and J. Liu, "Target detection of forward-looking sonar image based on improved YOLOv5," *IEEE Access*, vol. 10, pp. 18023–18034, 2022.
[21] Y. Fang, Y. Ma, X. Zhang, and Y. Wang, "Enhanced YOLOv5 algorithm for helmet wearing detection via combining bi-directional feature pyramid, attention mechanism and transfer learning," *Multimedia Tools Appl.*, vol. 82, no. 18, pp. 28617–28641, Jul. 2023.
[22] S. Guo, L. Li, T. Guo, Y. Cao, and Y. Li, "Research on mask-wearing detection algorithm based on improved YOLOv5," *Sensors*, vol. 22, no. 13, p. 4933, Jun. 2022.
[23] L. Tang, T. Xie, Y. Yang, and H. Wang, "Classroom behavior detection based on improved YOLOv5 algorithm combining multi-scale feature fusion and attention mechanism," *Appl. Sci.*, vol. 12, no. 13, p. 6790, Jul. 2022.
[24] C. Li, W. Yao, H. Wang, and T. Jiang, "Adaptive momentum variance for attention-guided sparse adversarial attacks," *Pattern Recognit.*, vol. 133, Jan. 2023, Art. no. 108979.
[25] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7132–7141.
[26] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, and Q. Hu, "ECA-Net: Efficient channel attention for deep convolutional neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11531–11539.
[27] C. Sun, S. Zhang, P. Qu, X. Wu, P. Feng, Z. Tao, J. Zhang, and Y. Wang, "MCA-YOLOV5-light: A faster, stronger and lighter algorithm for helmet-wearing detection," *Appl. Sci.*, vol. 12, no. 19, p. 9697, Sep. 2022.
[28] R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi, "Convolutional neural networks: An overview and application in radiology," *Insights Imag.*, vol. 9, no. 4, pp. 611–629, Aug. 2018.
[29] K. Han, Y. Wang, C. Xu, J. Guo, C. Xu, E. Wu, and Q. Tian, "GhostNets on heterogeneous devices via cheap operations," *Int. J. Comput. Vis.*, vol. 130, no. 4, pp. 1050–1069, Apr. 2022.
[30] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese, "Generalized intersection over union: A metric and a loss for bounding box regression," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Long Beach, CA, USA, Jun. 2019, pp. 658–666.
[31] B. Xu, X. Cui, W. Ji, H. Yuan, and J. Wang, "Apple grading method design and implementation for automatic grader based on improved YOLOv5," *Agriculture*, vol. 13, no. 1, p. 124, Jan. 2023.
[32] Z. Zheng, P. Wang, W. Liu, J. Li, R. Ye, and D. Ren, "Distance-IoU loss: Faster and better learning for bounding box regression," *Proc. AAAI Conf. Artif. Intell.*, Apr. 2020, vol. 34, no. 7, pp. 12993–13000.

**YUNCHANG ZHENG** received the B.S. degree in electronic science and technology and the M.S. degree in signal and information processing from the University of Electronic Science and Technology of China, in 2013 and 2016, respectively. Since 2017, he has been a Lecturer with the College of Electrical Engineering, Hebei University of Architecture, Zhangjiakou, China. His research interests include image processing, deep learning, and artificial intelligence.



**YUNYUE ZHAN** is currently pursuing the degree in measurement and control technology and instruments with the Hebei University of Architecture, Zhangjiakou, China. Her research interests include machine learning and artificial intelligence.

**XIAOYING HUANG** received the M.S. degree in control theory and control engineering from North China Electric Power University, in 2016.

She joined the Hebei University of Architecture, in 2016. Her research interests include the research and application of advanced control strategies and data-driven modeling methods and applications.

**GAOQING JI** received the master's degree in signal and information processing from the University of Electronic Science and Technology of China, in 2013. Since 2014, he has been a Lecturer with the College of Electrical Engineering, Hebei University of Architecture, Zhangjiakou, China. His research interests include autopilot, signal processing, and deep learning.