remote sensing

MDPI

Article

Lightweight Design for Infrared Dim and Small Target Detection in Complex Environments

Yan Chang [1], Decao Ma [1], Yao Ding [1,*], Kefu Chen [1] and Daming Zhou [2]

[1] PLA Rocket Force University of Engineering, Xi'an 710025, China; cyan313@outlook.com (Y.C.); madecaoedu@163.com (D.M.)
[2] School of Astronautics, Northwestern Polytechnical University, Xi'an 710072, China
* Correspondence: dingyao.88@outlook.com; Tel.: +86-176-7915-3192

Abstract: In the intricate and dynamic infrared imaging environment, the detection of infrared dim and small targets becomes notably challenging due to their feeble radiation intensity, intricate background noise, and high interference characteristics. To tackle this issue, this paper introduces a lightweight detection and recognition algorithm, named YOLOv5-IR, and further presents an even more lightweight version, YOLOv5-IRL. Firstly, a lightweight network structure incorporating spatial and channel attention mechanisms is proposed. Secondly, a detection head equipped with an attention mechanism is designed to intensify focus on small target information. Lastly, an adaptive weighted loss function is devised to improve detection performance for low-quality samples. Building upon these advancements, the network size can be further compressed to create the more lightweight YOLOv5-IRL version, which is better suited for deployment on resource-constrained mobile platforms. Experimental results on infrared dim and small target detection datasets with complex backgrounds indicate that, compared to the baseline model YOLOv5, the proposed YOLOv5-IR and YOLOv5-IRL detection algorithms reduce model parameter counts by 42.9% and 45.6%, shorten detection time by 13.6% and 16.9%, and enhance mAP0.5 by 2.4% and 1.8%, respectively. These findings demonstrate that the proposed algorithms effectively elevate detection efficiency, meeting future demands for infrared dim and small target detection.

Keywords: object detection; lightweight; small infrared targets; attention mechanism

check for updates

Citation: Chang, Y.; Ma, D.; Ding, Y.; Chen, K.; Zhou, D. Lightweight Design for Infrared Dim and Small Target Detection in Complex Environments. *Remote Sens.* **2024**, *16*, 3761. https://doi.org/10.3390/rs16203761

Academic Editor: Salah Bourennane

Received: 12 August 2024
Revised: 13 September 2024
Accepted: 30 September 2024
Published: 10 October 2024

1. Introduction

Infrared detection imaging possesses robust anti-interference capabilities, enabling it to cater to the all-weather reconnaissance and monitoring requirements in the military field by providing stable and reliable target images. In recent years, it has garnered significant attention from numerous scholars [1–3]. However, due to the small size of the targets themselves or their great distance from the infrared sensor, coupled with the fact that imaging is in grayscale, the actual targets in infrared source images appear relatively small (occupying no more than 0.12% of the image) [1], lacking shape, color, and texture information, and exhibiting low signal-to-noise ratios. Furthermore, complex backgrounds and clutter interference pose significant challenges to the detection of infrared dim and small targets. This paper addresses the issue of detecting infrared dim and small targets in complex scenarios by proposing a lightweight detection algorithm based on deep learning, with a focus on enhancing detection accuracy while also considering real-time performance.

Currently, the prevalent methods for detecting infrared dim and small targets encompass traditional algorithms and deep learning algorithms. Traditional algorithms primarily employ single-frame detection approaches, which can be categorized into three types: filter-based detection algorithms [4–6], detection algorithms inspired by the human visual system [7–10], and detection algorithms based on image data structures [11,12]. The core principle of these methods involves extracting features such as grayscale and contrast of small targets within a single infrared frame and then precisely detecting the targets by effectively suppressing background information while enhancing target features. Despite their low computational complexity and high detection efficiency, single-frame detection algorithms heavily rely on human prior knowledge, requiring manual adjustments to parameters like segmentation thresholds and target scales. This limitation hinders their adaptability to diverse application scenarios and target characteristics, resulting in poor generalization capabilities. Moreover, due to their sensitivity to background variations, traditional algorithms exhibit unsatisfactory accuracy and stability in infrared images with complex backgrounds and low contrast.

On the other hand, deep learning algorithms, by leveraging deep networks to extract typical target features, have achieved significantly higher accuracy than traditional detection methods, resulting in rapid progress in the detection of infrared dim and small targets. Key examples include attention mechanism algorithms represented by Transformer [13] and Swin-Transformer [14], two-stage detection algorithms such as R-CNN [15] and its variants Fast R-CNN [16], Faster R-CNN [17], and Mask R-CNN [18], as well as single-stage detection algorithms exemplified by SSD [19] and the YOLO [20–26] series. Furthermore, strategies like data augmentation [27], multi-scale feature learning [28], and generative adversarial networks [29] have been introduced to enhance the detection precision of dim and small targets. Thanks to the powerful nonlinear fitting capabilities of neural networks, these methods have significantly improved the feature extraction abilities for infrared dim and small targets. Nevertheless, when confronted with scenarios with low signal-to-noise ratios, the detection algorithms face significant challenges in balancing the false alarm rate and miss detection rate, and their real-time performance remains inadequate. The main reasons are as follows:

(1) Deep learning networks currently used for object detection perform well in general scenes where objects are dispersed and do not exhibit overlap or occlusion. However, for infrared small targets, which are excessively small and lack texture and structural features, the average detection accuracy is poor.

(2) To obtain a larger receptive field, downsampling operations are often employed in neural networks; however, excessive downsampling operations are prone to causing information loss of small targets in deep-level features, thereby making it difficult for the detector to extract effective features.

(3) As the number of layers and parameters in neural networks increases, deep learning detection algorithms generally suffer from issues of high computational complexity and large model sizes, posing challenges for deployment on resource-constrained mobile platforms.

To address the above issues, this paper proposes a full-process lightweight target detection algorithm that maintains high accuracy, based on the single-stage object detection network YOLOv5. Firstly, focusing on the extraction of infrared dim and small target feature information, we design separate modules for network feature extraction and fusion, utilizing fewer parameters to achieve the extraction of key features. Secondly, an enhanced head for small target detection is devised, placing greater emphasis on micro-targets within the field of view. Thirdly, an adaptive weight loss function is developed to balance the loss calculation between positive and negative samples and in overlapping regions, thereby enhancing the model's generalization ability and accuracy. Additionally, a pruning architecture for the backbone network is introduced to further reduce the model parameter count and shorten detection time. The optimized target detection network retains the advantage of automatic feature extraction by deep learning methods while demonstrating stronger adaptability to infrared dim and small targets. It can detect such targets in diverse complex backgrounds, showcasing robustness and versatility. The main contributions of this paper are as follows:

(1) A lightweight infrared target detection algorithm, YOLOv5-IR, specifically designed for detecting infrared dim and small targets, is proposed in this paper. By optimizing the network structure, the backbone network's ability to recognize dim and small

targets is enhanced, making it suitable for extracting features from infrared dim and small targets. This optimization effectively reduces the model parameters and computational cost.

(2) A loss function and a detection head in the head layer are designed, altering the bounding box regression loss function to balance positive and negative samples. This improves the detection accuracy of bounding boxes and enhances the network focus on the infrared characteristics of targets.

(3) A pruning architecture for the backbone network is designed, integrating pruning algorithms with network optimization. This improvement removes redundant channel weight parameters and further results in the lightweight version of the algorithm, named YOLOv5-IRL in this paper.

(4) The detection performance of the proposed algorithm is validated on the dim-small aircraft dataset, featuring a diverse range of target quantities, poses, and complex scenes. Further comparisons with other advanced algorithms under varying signal-to-noise ratio conditions are conducted. Comparison experimental results demonstrate that the algorithm proposed in this paper achieves higher detection accuracy and faster detection speed on the dataset.

2. Related Works

2.1. Deep Learning Algorithms for Infrared Dim and Small Target Detection

Infrared dim and small target detection is predominantly applied in military fields such as early warning reconnaissance, aircraft guidance, and spatial situation awareness, demanding high precision and real-time performance. To address this, researchers have modified deep learning algorithms for infrared dim target detection. Liu et al. [30] introduced an enhancement to YOLOv3 by integrating the Darknet-53 backbone network with SPP (Spatial Pyramid Pooling) for feature extraction, enabling the fusion of local and global features, thereby enhancing the representational capability of feature maps. However, this approach yields lower accuracy and is tailored specifically for infrared ship detection scenarios. Hou et al. [31] proposed ISTDU-Net (Infrared Small-Target Detection U-Net), an infrared dim target detection network, which boosts the weights of small target feature groups and incorporates fully connected layers to suppress backgrounds with similar structures, effectively reducing false alarm rates. Fan et al. [32] leveraged a multi-head self-attention mechanism to accurately capture target location information and replaced the CIoU (Complete Intersection over Union) loss with NWD (Normalized Wasserstein distance) loss, slightly improving target detection performance. Nonetheless, this approach significantly increases the number of parameters, leading to a decrease in detection speed. He [33] proposed the IRI-CNN detection algorithm for airborne infrared dim and small targets, effectively reducing false alarm rates but with compromised real-time performance. Mou [34] presented an improved algorithm based on a feature recombination sampling method. Experimental results demonstrated that this approach outperforms the original model in terms of precision and recall, albeit with a larger number of model parameters and lower computational real-time performance. Yang R [35] modified YOLOv5 for small targets, optimizing anchor boxes and introducing an attention mechanism to improve detection accuracy, but real-time performance remained challenging. While these algorithms have addressed infrared small target detection to some extent, they lack targeted research on infrared dim and small target detection. Hence, to address the real-time issue in infrared dim and small target detection, it is imperative to devise more lightweight network models.

2.2. Model Lightweighting Methods Based on Deep Learning Networks

In order to improve the efficiency of deep learning-based infrared dim and small target detection and facilitate deployment on resource-constrained mobile platforms and embedded devices, researchers have conducted extensive studies on network lightweighting. The core of lightweight networks lies in compressing and optimizing the network structure while maintaining accuracy, thereby improving the computational efficiency of the algorithm. SqueezeNet [36], as one of the earliest proposed lightweight networks, achieves model parameter compression through the use of Fire modules and introduces split convolutions to reduce the computational burden. In 2017, the Google team presented MobileNetV1 [37], a lightweight convolutional neural network that employs depthwise separable convolutions in place of standard 3 × 3 convolutions, significantly decreasing the model parameter count and computational complexity. MobileNetV2 [38] introduced inverted residuals and linear bottlenecks, enhancing the model performance and expressive power. Building upon its predecessors, MobileNetV3 [39] underwent improvements and optimizations, leveraging neural architecture search to determine the network architecture and parameters, thereby reducing computational requirements and latency while maintaining accuracy. ShuffleNet [40] incorporates pointwise group convolutions, dividing convolution operations into multiple groups, which further reduces computation. GhostNet [41] generates a few "intrinsic feature maps" through conventional convolutions and then produces more similar "ghost feature maps" via depthwise separable convolutions, yielding feature maps with the same number of channels as traditional convolutional layers but at a significantly lower computational cost. Among these, Xception-SSD [42] and MobileNet-SSD [43] are exemplary lightweight backbones for detection algorithms. They apply the principles of ShuffleNet and MobileNet, respectively, to reduce SSD network parameters while maintaining detection accuracy, thereby simplifying the model complexity.

Apart from designing network architectures, researchers have also introduced various model compression techniques: (1) Model pruning techniques, which are divided into structured pruning and unstructured pruning [44]. Structured pruning reduces model complexity by decreasing network structural modules, such as channel pruning and layer pruning, while unstructured pruning directly eliminates weight parameters to shrink model size; (2) low-rank factorization [45] employs matrix decomposition to reduce the complexity of convolutional layers or fully connected layers in neural networks; (3) knowledge distillation [46] transfers knowledge from large and complex models to lightweight models, enabling model compression without sacrificing much accuracy.

Based on the aforementioned lightweight methods, Liu et al. [47] proposed the MobileNet-YOLO network, which features low computational costs and can efficiently operate on smart devices with limited power and resources. SHA et al. [48] introduced a reusable residual network, utilizing a backbone network constructed with three-layer reusable connection residual blocks for pedestrian feature extraction. This approach not only reduces the model size but also enhances its feature extraction capabilities. Li et al. [49] adopted the Ghost module in place of standard convolutions for pedestrian recognition on the basis of YOLOv5s, making the model lighter while maintaining accuracy. In terms of attention mechanism research, Zou et al. [50] presented a multi-mask correction attention module to enhance pedestrian contour features. Li et al. [51] embedded an improved channel attention mechanism module into the FairMOT backbone network, effectively reducing the missed detection rate of occluded pedestrians. Hao et al. [52] proposed a cross-scale feature fusion attention mechanism module, significantly improving the detection of small targets. However, while these lightweight networks significantly increase detection speed in end-to-end detection, they are mostly suitable for visible light target detection. Infrared images, due to their unique imaging characteristics such as long wavelengths, high noise levels, poor spatial resolution, and sensitivity to ambient temperature changes, thus limit the generalization ability of algorithms.

3. The Proposed Lightweight Infrared Small Target Detection Algorithm

In order to address the challenges faced by the YOLOv5 network model when detecting small targets like aircraft, including slow inference speed, low accuracy, and missed detections, this study proposes an enhanced network model, YOLOv5-IR, and an even more streamlined version of YOLOv5-IRL. Figure 1 depicts the overall algorithmic framework. Drawing inspiration from MobileNetV3 design principles, this model incorporates a lightweight backbone feature extraction network as a substitute for YOLOv5 backbone component. By designing a feature fusion network and expanding its depth, additional detection layers for small-sized targets are integrated into large-scale feature maps to enhance detection performance. Furthermore, to prioritize poor-quality targets, an adaptive weight loss function is devised, which amplifies the prediction box loss proportion for low-quality target images. Lastly, to facilitate efficient deployment on resource-constrained platforms, network compression techniques are applied, enabling a significant reduction in the number of parameters while maintaining acceptable accuracy levels.

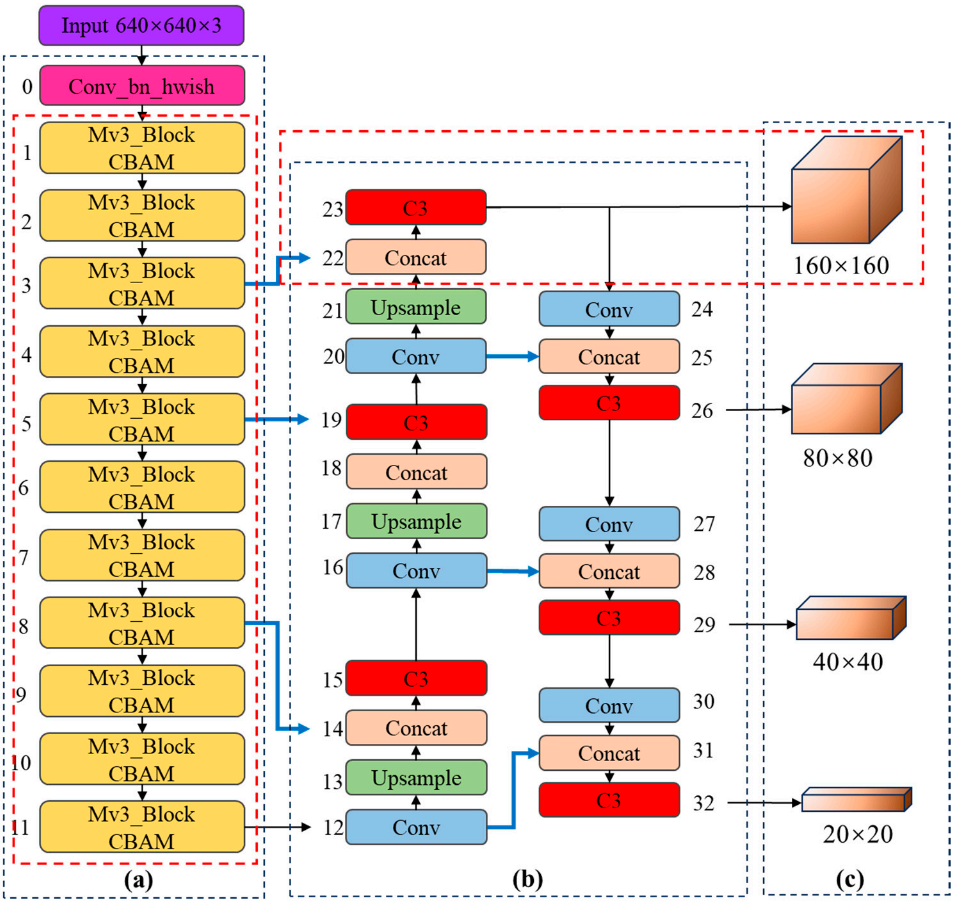


Figure 1. The schematic diagram of the proposed YOLOv5-IR framework. (**a**) Backbone; (**b**) Neck; (**c**) Head.

3.1. Network Architecture Design and Optimization

By analyzing the structure of the YOLOv5 network, it is observed that the use of standard convolutions leads to increased computational load in the convolution layers and inefficient utilization of parameters. Therefore, the YOLOv5-IR backbone feature extraction network structure opts to replace standard convolutions with depthwise separable convolutions, which have a smaller computational load. It aims to maintain detection accuracy while creating a lightweight network model by combining depthwise separable convolutions with attention mechanisms. Specifically, the input feature map is set to 6406403, and the feature map is processed multiple times using 1 Conv_bn_hwish convolution module and 11 Mv3_block modules in sequence. Additionally, CBAM attention mechanisms are added after the depthwise separable convolutions in the 1st, 2nd, 3rd, 4th, 5th, 6th, 7th, 8th, 9th, 10th, and 11th Mv3_block modules, as shown in Figure 2.

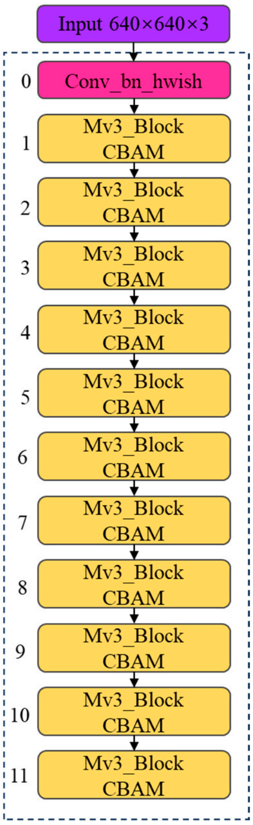


Figure 2. The backbone network architecture.

Based on the integration of channel attention and spatial attention, CBAM enables the model to simultaneously focus on significant channels and spatial locations, thereby enhancing the accuracy of infrared dim and small target feature representation and the decision-making capability of the model. It is divided into two parts: a channel attention block and a spatial attention block. The weighted results of the two blocks and the original feature map are combined to obtain the output information. Its structure is illustrated in Figure 3.

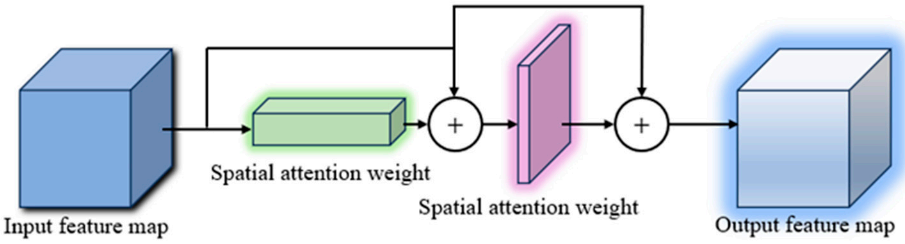


Figure 3. CBAM Attention Mechanism.

Channel attention can identify and enhance the feature channels that are most crucial for the current infrared target detection while filtering out irrelevant information from the features, thereby enhancing the model sensitivity to key information about infrared small targets. The channel attention block processes the input feature map through global maximum pooling and global average pooling to transform it into one-dimensional vectors. These two vectors are then passed through fully connected layers and activated with the sigmoid function, and their sum is fused to obtain the channel attention weights. Its structure is presented in Figure 4.

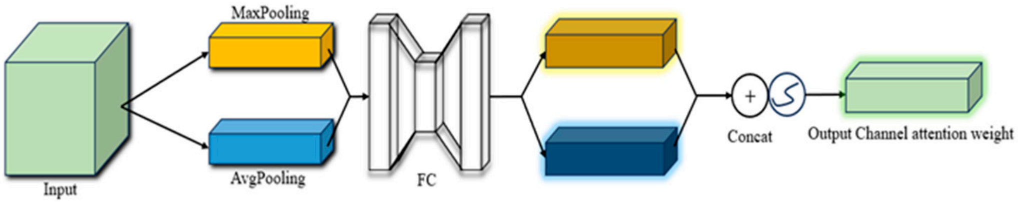


Figure 4. CBAM channel attention chunking.

Spatial attention establishes global connections based on the interrelationships between each element within the target features, enabling feature enhancement within features of the same level through attention mechanisms, thereby strengthening the feature representation of useful information. The spatial attention block applies the channel attention weights to the input information. Different from the channel attention block, it expands the channels through sequential global maximum pooling and average pooling and then uses a standard convolution and sigmoid activation function to obtain two-dimensional spatial features, as shown in Figure 5.

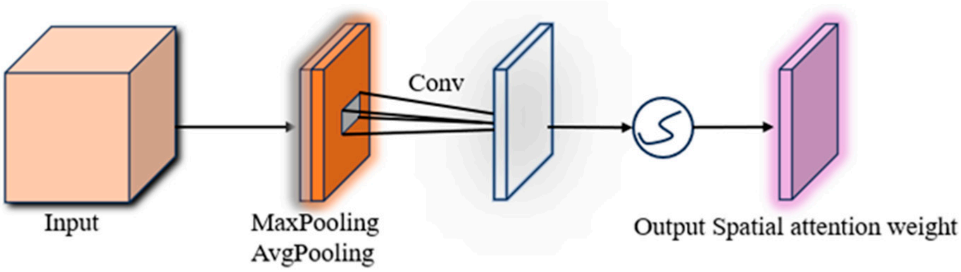


Figure 5. CBAM Spatial Attention Blocking.

To further enhance the model's ability to extract infrared target features, in this paper, the feature fusion network is improved; its structure is illustrated in Figure 1b.

(1) The proposed model incorporates an overall structure with one upsampling and one downsampling process in the feature fusion network. Specifically, three upsampling modules and three downsampling modules are designed.

(2) As the network deepens and learns deeper-level features continuously, it is prone to losing shallow-level information. To retain the features from the backbone feature extraction network, the output values of the 3rd layer from the backbone are fused with the output values of the 22nd layer from the upsampling structure in the feature fusion network. Similarly, the 5th layer is fused with the 19th layer and the 8th layer is fused with the 14th layer. To prevent feature loss during upsampling and downsampling operations in the feature fusion network, the output values of the 20th layer from the upsampling structure are fused with the output values of the 25th layer from the downsampling process, the 16th layer is fused with the 28th layer, and the 12th layer is fused with the 31st layer. This approach aims to fuse shallow and deep features.

(3) Aiming at the challenges posed by the dim and insensitive target representation characteristics of infrared small targets, an additional detection head suitable for small targets is introduced in this improvement. Four detection heads are connected to the 23rd, 26th, 29th, and 32nd layers of the feature fusion network, respectively. The sizes of their output feature maps are 20 × 20, 40 × 40, 80 × 80, and 160 × 160, respectively. A preset anchor box specifically designed for detecting small and dim targets is also added. The sizes of the anchor boxes, from large to small, are [(116, 90), (156, 198), (373, 326)], [(30, 61), (62, 45), (59, 119)], [(10, 13), (16, 30), (33, 23)], and [(5, 6), (8, 14), (15, 11)], respectively.

The feature fusion network proposed in this paper incorporates a crucial module, C3, designed to effectively extract fused network features. It primarily comprises two branches: one utilizing standard convolution and the other featuring a residual module within. The structure is depicted in Figure 6, where "Conv" represents the standard convolution module, encompassing standard convolution, Batch Normalization (BN) for normalization, and SiLU as the activation function, whereas "Bottleneck" denotes the residual module.

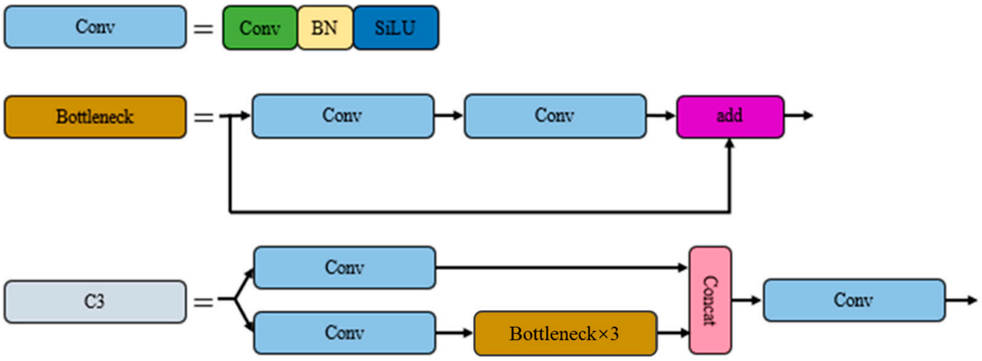


Figure 6. Schematic diagram of the C3 structure.

3.2. Design of a Detection Head Oriented towards Infrared Characteristics

Infrared images are prone to significant interference due to their unique acquisition principles and capture methods. The detection head, as a key structure that directly processes infrared images, plays a crucial role in the overall network model. The YOLOv5 model processes input multi-scale feature maps through its detection head to detect objects of varying sizes. The detection head typically incorporates convolutional layers, which are utilized to extract crucial information from the feature maps and facilitate further analysis.

In the YOLOv5 model, the Detection Head receives as input three feature maps of different sizes, which have undergone feature fusion. These feature maps undergo convolutional operations, with 1 × 1 convolution kernels employed to adjust the number of channels to suit subsequent prediction tasks. The 1 × 1 kernels facilitate dimensionality expansion or reduction of channels without altering the spatial dimensions of the feature maps, enhancing nonlinearity and integrating features through cross-channel information interaction. This design aids the model in extracting richer feature representations while preserving spatial information.

On the other hand, the SimAM attention mechanism is a parameter-free attention module that discovers the importance of each neuron by optimizing an energy function, thereby assigning a unique weight to each neuron in the feature map. This mechanism enables the model to better focus on critical information of targets in infrared images, enhancing detection accuracy. Notably, SimAM utilizes only a single weight to represent the feature importance of an individual neuron. SimAM defines the importance of feature weights through an energy function associated with each neuron, as illustrated in Equation (1):

$$e_t(w_t, b_t, y, x_i) = \frac{1}{M-1}\sum_{i=1}^{M-1}(y_0 - \hat{x}_i)^2 + (y_t - \hat{t})^2 \tag{1}$$

In the equation, $M = H \times W$ represents the number of neurons within a channel of the feature map, where w_t and b_t are the weights and biases for channel transformation, respectively. x_i and t denote other neurons and the focal neuron in the feature map, while y is the output value of the feature map. To identify the linear relationship between neuron t and other neurons within the same channel, the analytical solution of each channel energy function is sought. This reveals that the lower the minimum energy of each channel, the greater the distinction between neuron t and its surrounding neurons, indicating a higher significance of this neuron. The SimAM attention mechanism is illustrated in Figure 7.

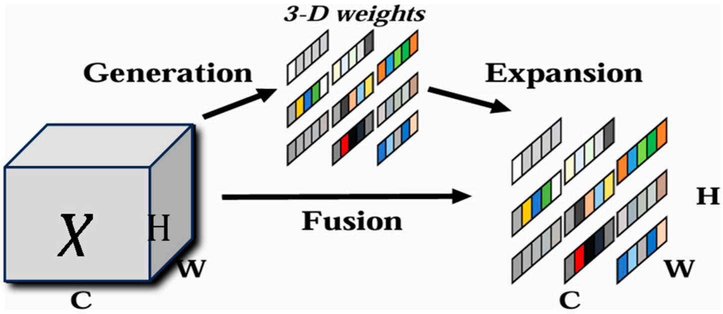


Figure 7. The improved output convolutional architecture.

Based on the designed backbone network, the simplistic 1 × 1 standard convolution in the head layer output process is abandoned. Instead, a convolutional module resembling a residual structure is constructed, and the SimAM attention mechanism is incorporated into this module, as shown in Figure 8.

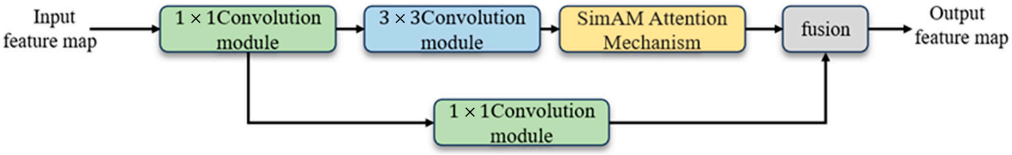


Figure 8. The Improved output convolutional architecture.

3.3. Design of an Adaptive Weighting Loss Function

In the original YOLOv5, there are three types of loss functions: classification loss, object confidence loss, and bounding box regression loss. Infrared images often suffer from high noise or objective shooting conditions such as environmental disturbances during acquisition, and the infrared targets in the images inevitably exhibit poor quality characteristics. Moreover, there are differences between clear and low-quality samples among different targets or even within the same target. Figure 9 illustrates a comparison of high, medium, and low-quality sample images in the dataset.

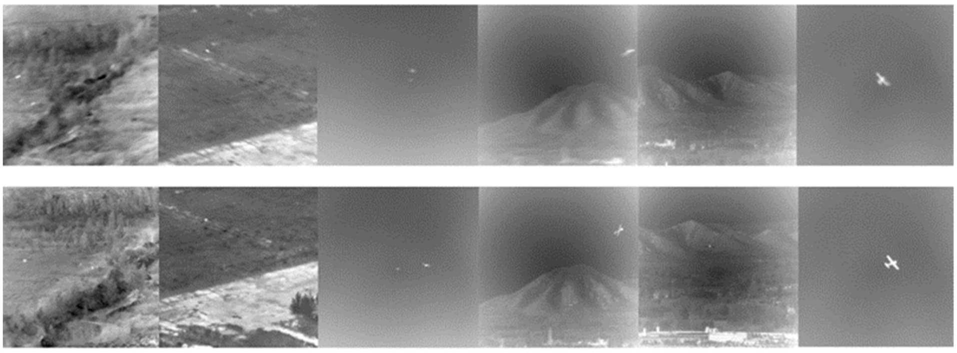


Figure 9. Comparison of different quality sample images.

Addressing this characteristic of infrared images, this paper proposes a Wise_IOU loss, which incorporates an adaptive weighting mechanism into IOU to adjust the loss weights of clearer objects in infrared images. This approach aims to encourage the network model to pay more attention to low-quality targets, thereby enhancing the overall prediction and generalization capabilities of the model. The specific calculation for Wise_IOU is based on Equations (2) and (3):

$$\gamma_{WIOU} = exp\left(\frac{(x_1 - x_2)^2 + (y_1 - y_2)^2}{\left(W_3^2 + H_3^2\right)^*} \right) \tag{2}$$

$$Loss(WIOU) = Loss(IOU) \cdot \gamma_{WIOU} \tag{3}$$

In the formula: γ_{WIOU} represents the adjustment factor for IOU loss, x_1 and y_1 denote the coordinates of the center point of the ground truth box, while x_2 and y_2 represent the coordinates of the center point of the predicted box. W_3 and H_3 signify the length and width of the smallest enclosing box of the ground truth box and the predicted box. The value range of γ_{WIOU} is $[1, e)$, and its role is to enhance the weight proportion of low-quality predicted boxes as much as possible. The step $\left(W_3^2 + H_3^2\right)^*$ separates the smallest enclosing box metrics from the anchor box's own parameters, further accelerating the convergence speed of the loss. The framework and formula descriptions presented above constitute the proposed improved algorithm YOLOv5-IR in this paper.

3.4. Lightweight Design of the Proposed YOLOv5-IR

In order to meet the deployment needs of resource-constrained mobile platforms, this paper proposes an enhanced lightweight model, YOLOv5-IRL (YOLOv5-IR Lightweight), which incorporates channel pruning to compress and accelerate the previously designed YOLOv5-IR network model. The entire process is illustrated in Figure 10.

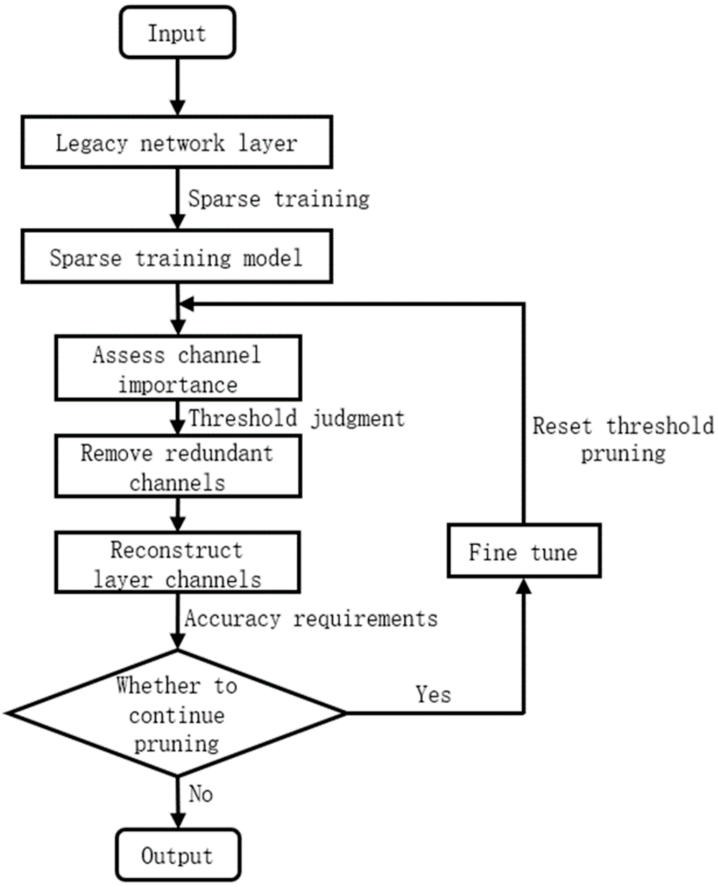


Figure 10. Channel pruning algorithm flow.

(1) Analysis of BN Layer Algorithm

Since the introduction of the Batch Normalization (BN) layer algorithm, due to its powerful integration and convergence capabilities, subsequent deep learning algorithms have nearly all incorporated the BN layer into each convolution operation. The BN layer accelerates the training process by normalizing the layer input, yet it incorporates learnable parameters, including scale and shift parameters.

Therefore, analyzing the algorithm structure of the BN layer can provide a strong basis for judging the importance of channels in the convolution layers. The BN layer algorithm is mainly divided into three steps. First, calculate the mean μ and variance σ^2 of a batch of image data output from the preceding convolution layer. If the number of image data is n, the calculation formulas are as follows:

$$\mu = \frac{1}{n}\sum_{i=1}^{n} X_i \tag{4}$$

$$\sigma^2 = \frac{1}{n}\sum_{i=1}^{n}(X_i - \mu)^2 \tag{5}$$

Then, the input image data is standardized and normalized, transforming the data into a normal distribution with a mean of 0 and a variance of 1. To avoid dividing by zero, ε is used as a small approximate value that is not equal to 0. The calculation formula is as follows:

$$\frac{X_i - \mu}{\sqrt{\sigma^2 + \varepsilon}} = \hat{X}_i \tag{6}$$

(2) Sparse Training Method

Sparse training is a method that introduces sparsity during model training, aiming to reduce model complexity and enhance generalization ability. In deep learning, sparse training is typically achieved through regularization techniques, such as L1 regularization, which encourages the model to learn fewer non-zero parameters during training. For Batch Normalization (BN) layers, sparse training can be implemented by pruning the scaling parameter γ.

In a BN layer, each channel has a scaling parameter γ and a small shift parameter β, which are learned during training to adjust the output of each channel. The goal of sparse training is to identify and retain γ parameters that significantly contribute to model performance while eliminating those with lesser contributions. This approach reduces the number of model parameters, thereby decreasing model complexity.

The purpose of the BN layer is to normalize the input of the layer, ensuring that the output of each channel has a zero mean and unit variance. Due to the loss of spatial distribution information resulting from normalized image data, spatial transformations such as scaling and translation are necessary. These transformations enable the image after normalization to be linearly transformed back to the original image while preserving its features. The transformation formula is as follows:

$$Y_i = \gamma\hat{X}_i + \beta = \gamma\frac{X_i - \mu}{\sqrt{\sigma^2 + \varepsilon}} + \beta \tag{7}$$

where Y_i represents the output value of the original image after passing through the BN layer, γ is the scaling parameter for the linear transformation performed by the BN layer, and β is the translation parameter. The pruning operation on the convolution layer channels can prune the scaling parameters after the BN layer normalizes the input image, removing the γ parameters with less control over channel importance to achieve compression of the number of convolution layer channels.

Before pruning the γ parameters, sparse training is required. Firstly, it is necessary to measure the importance of different channel γ parameters. Therefore, $L1$ regularization is introduced for the γ parameters, and the obtained $L1$ regularization norm sr can serve as a penalty parameter for the network training loss function. The regularization loss is defined as Equations (5) and (6):

$$Z(\gamma) = |\gamma| \tag{8}$$

$$L = \sum_{(x,y)} l(f(x,w),y) + \alpha\sum_{\gamma\in\tau} Z(\gamma) \tag{9}$$

In the formula, w represents the trained weights, and $l(f(x,w),y)$ represents the loss obtained from network training. Since the $L1$ regularization norm sr of the γ parameters may reduce the prediction accuracy of the weights obtained after network training, it is necessary to introduce an appropriately sized balancing factor α to control the weight between the regularization term and the original loss term, ensuring that the model pursues sparsity without excessively compromising accuracy. After sparse training, channels corresponding to γ parameters close to zero can be considered unimportant and pruned, while those with larger γ values are deemed important and should be retained.

(3) Pruning and Fine-tuning

After the completion of sparse training, the importance of each channel in the model was evaluated, allowing for pruning to be performed. By setting a pruning threshold, redundant channels corresponding to the γ parameters after regularization in the BN layer of the convolution module are pruned. The specific operation process is shown in Figure 11.

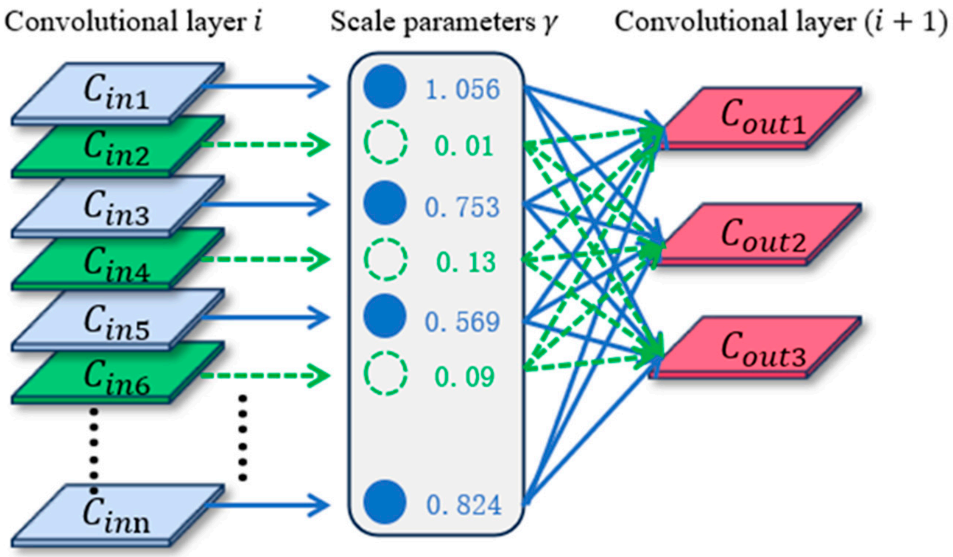


Figure 11. Sparse training combined with channel pre-pruning. The green dashed line indicates the channel layer that can be pruned. The blue solid line indicates the channel layer that cannot be pruned.

In the above figure, each of the n channels in convolution layer i has a corresponding γ parameter value. If the pruning threshold is set to 0.15, channels such as C_{in2}, C_{in4}, and C_{in6}, which have γ values of 0.01, 0.13, and 0.09, will be removed. Channels like C_{in1}, C_{in3}, and C_{in5}, with evaluation values greater than the threshold, will be retained and passed to the next convolution layer $i + 1$. Therefore, the number of channels will be reduced in subsequent convolution operations, resulting in a decrease in the network parameter count. The pruned network is shown in Figure 12 below.

Since channels with small γ values have relatively low importance, pruning them does not have a significant impact on the convolution operation and the overall prediction accuracy of the network model. However, after pruning, the reduced number of model parameters may affect its performance, necessitating fine-tuning to restore accuracy. Fine-tuning is an iterative process that involves adjusting the learning rate and repeatedly training to optimize the remaining parameters. During each iteration, the pruning threshold is adjusted based on the model performance feedback to strike the optimal balance between increasing the pruning rate and maintaining predictive accuracy.

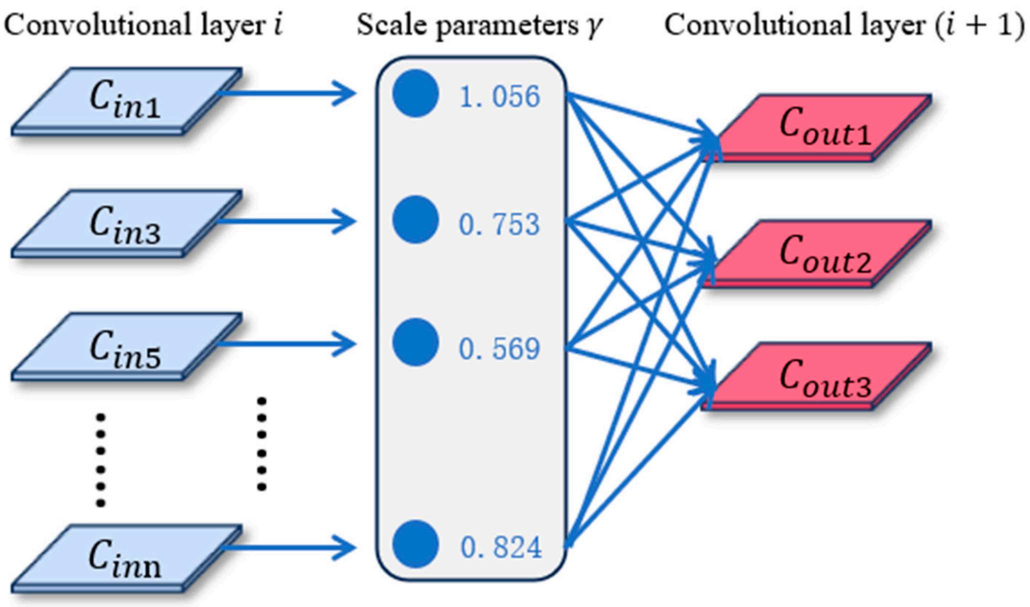


Figure 12. Channel pruning network.

4. Experimental Verification

This paper describes the metrics and datasets used in this study and presents the results. The experiments were conducted using Python 3.8, PyTorch 1.10.0, and Cuda 11.3 on an NVIDIA GeForce RTX 3090 with an Intel Core i9-10980XE CPU @ 3 GHz.

4.1. Dataset

Due to the scarcity of publicly available infrared small target image datasets, the limited availability of measured data samples, and the majority of them being unsuitable for missile-borne guidance systems, this paper has created a custom infrared small target detection dataset based on the dim and small aircraft series [53] for detecting and tracking small aircraft targets in ground/air backgrounds. This image series focuses on detecting one or multiple fixed-wing unmanned aerial vehicle (UAV) targets by simulating tracking and detection of low-altitude flying small aircraft targets. It serves as the foundation for providing precise identification data of infrared dim and small targets. The infrared data has a wavelength of 3~5 µm, and the image size is 256 × 256 pixels. The acquisition scenarios include various backgrounds, such as sky and ground, totaling 22 data segments, 30 trajectories, 16,177 image frames, and 16,944 targets. This dataset is characterized by a rich number and variety of targets, low target brightness and contrast, complex and extensive scenes, and the loss of texture and color information, making it representative of real infrared small targets. Sample images are shown in Figure 13. The data collection targets were aerial fixed-wing UAVs (fuel-powered), and the basic parameters of the UAV targets used in the supplementary experiments are presented in Table 1.

Table 1. Basic parameters of unmanned aerial vehicles (UAVs).

Indicator Items	Parameter
Fuselage length	2.0 m
Wingspan length	2.6 m
Flight speed	Airspeed: 30 km/h
Turning radius	130 m
Near and far ends of the flight route	Near end: 500 m, Far end: 5000 m
Flight altitude	Near end: 50 m, Far end: 500 m

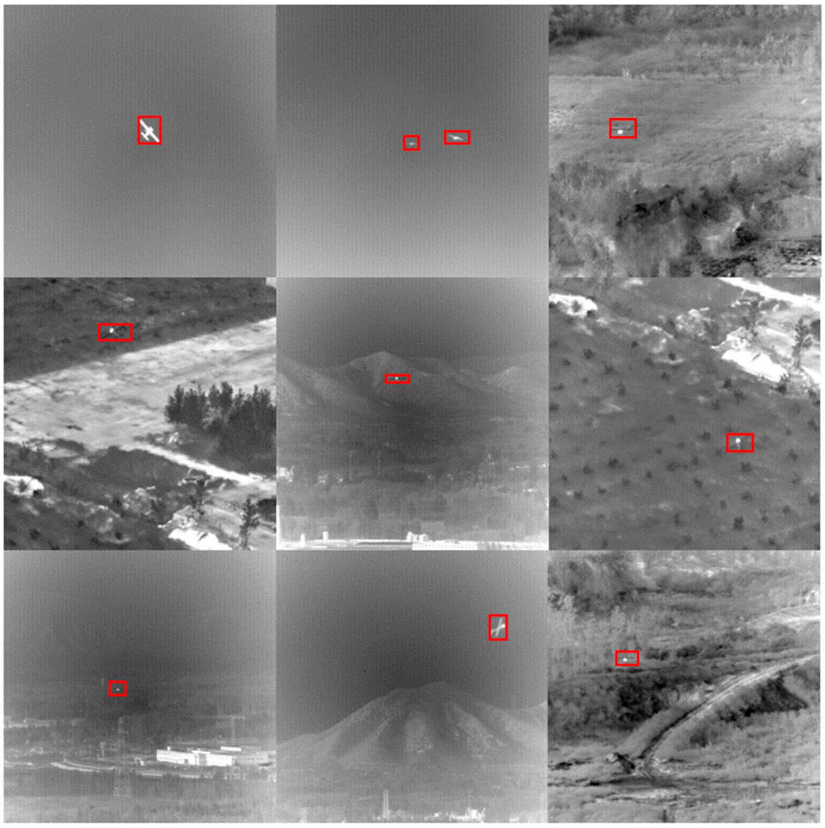


Figure 13. Examples of images from the dim and small aircraft section.

To verify the effectiveness of the improvements made to the network model in this paper, six data segments (Data1, Data2, Data3, Data7, Data16, and Data20) from the dim and small aircraft image series representing six types of targets were selected as experimental data for dataset creation. The dataset includes 399 images from Data1, 599 images from Data2, 100 images from Data3, 399 images from Data7, 499 images from Data16, and 400 images from Data2, totaling 2396 images and 2995 targets. The dataset was divided into training, validation, and testing sets in a ratio of 6:3:1. The backgrounds consist of three scenarios: noisy sky, mountain peaks meeting the sky, and complex ground, with target states all affected by complex backgrounds or interactions between targets. The main information of the dataset is shown in Table 2.

Table 2. Key information for each segment in the self-made dataset.

Data Segments	Number of Images	Number of Targets	Training Set	Validation Set	Test Set	Background	State
Data1	399	399	240	120	39	Noise sky	The target moves in close proximity
Data2	599	1198	360	180	59	Noise sky	The two targets gradually intersect
Data3	100	100	60	30	10	The mountain peaks meet the sky	The target moves over long distances
Data7	399	399	240	120	39	Complex ground	The target is from near to far
Data16	499	499	300	150	49	Complex ground	The target is from far to near
Data20	400	400	240	120	40	The mountain peaks meet the sky	The target is from near to far

4.2. Training Design

The parameter settings for the training process are as follows: The input infrared image size was set to 640 × 640, scale transformation and data augmentation were performed before training, the number of training epochs was 200, the batch size for each training iteration was 16, the initial learning rate (Lr0) was set to 0.01, the SGD momentum was set to 0.937, the network model depth (Depth_multiple) was 0.33, and the network model width (Width_multiple) was 0.50.

4.3. Evaluation Metrics

In different deep learning tasks and application scenarios, appropriate evaluation metrics and the comprehensive performance of multiple metrics can effectively demonstrate the effectiveness of deep learning network models. In this paper, three evaluation metrics are used to assess the model performance: the number of model parameters (parameters), the time taken to detect each image (ms/img), and the mean Average Precision (mAP).

(1) Model Parameters

Model parameters refer to the total number of parameters that need to be learned during model training, including various weight and bias parameters. They can be used to evaluate the size and expressive power of the model. For each convolutional layer, C_{in} represents the number of input image channels, C_{out} represents the number of output channels, k_w and k_h respectively represent the width and height of the convolution kernel, and w represents the bias. The weight parameters obtained from one convolution operation are $C_{in} \times k_w \times k_h$. Therefore, the number of model parameters for each convolutional layer is calculated as Equation (9):

$$Parameters = (C_{in} \times k_w \times k_h + w) \times C_{out} \tag{10}$$

(2) Time taken to detect each image (ms/img)

The time taken to detect each image, also known as the inference time, refers to the duration required by the network model to process and analyze a single infrared image or a frame of video. This metric is used to evaluate the detection and inference speed of the network model.

(3) Mean Average Precision (mAP)

To measure the detection effectiveness of a network model for different predetermined targets in detection tasks, mean average precision (mAP) is a classic evaluation metric. It refers to the average level of detection accuracy for classified targets under different Intersection over Union (IOU) threshold settings, and can effectively express the quality of the network model detection performance. The calculation of mAP is closely related to precision, recall, and average precision (AP).

If TP represents true positive samples, FP represents false positive samples, TN represents true negative samples, and FN represents false negative samples, then precision indicates the proportion of true positive samples among the positive samples predicted by the network model. The calculation formula for precision is as follows:

$$Precision = \frac{TP}{TP + FP} \tag{11}$$

Recall refers to the proportion of true positive samples predicted by the network model out of the total true positive samples. The calculation formula for recall is as follows:

$$Recall = \frac{TP}{TP + FN} \tag{12}$$

Average Precision (AP) is the area integral between the P(R) curve, which relates precision and recall, and the coordinate axes. The calculation formula is as follows:

$$AP = \int_0^1 P(R)D(R) \tag{13}$$

Therefore, the mean Average Precision (mAP) is the arithmetic mean of the Average Precision (AP) across multiple target categories.

4.4. Detection Results and Quality Evaluation

To demonstrate the superiority of the proposed algorithm in this paper, it is compared with classical object detection algorithms, including SSD, YOLOv3, YOLOv5, and YOLOv7. Additionally, comparisons are also made with two typical lightweight object detection algorithms: Xception-SSD and MobileNet-SSD. The SSD employs an anchor strategy, presupposing anchors with different aspect ratios, and each output feature layer predicts multiple detection boxes based on these anchors. SSD is capable of multi-scale detection, with shallow layers used for detecting small objects and deeper layers for large objects. Xception-SSD and MobileNet-SSD replace the original backbone network in SSD with Xception and MobileNet, respectively, achieving higher detection accuracy while reducing the number of parameters.

YOLOv3 divides the input image into grids, with each grid responsible for predicting objects whose centers fall within it. YOLOv3 uses Darknet-53 as its backbone, adopting a residual structure to improve training depth and stability. By predicting multiple bounding boxes and class probabilities within each grid, YOLOv3 can quickly and accurately detect objects in images. YOLOv5 utilizes CSPDarknet53 as its efficient backbone network and introduces a Path Aggregation Network (PANet) to enhance feature fusion. It also employs Mosaic data augmentation to improve the model's ability to detect small objects. YOLOv7 incorporates techniques such as model reparameterization, dynamic label assignment, and compound scaling, enhancing detection accuracy while maintaining real-time processing speed. These algorithms exhibit certain advantages in detecting infrared dim and small targets, making them suitable as comparison algorithms for the proposed algorithm in this paper. Table 3 presents a comparison of model sizes and performance among different algorithms, with the best results highlighted in bold.

Table 3. Comparison of model size and performance.

Model	Precision	Recall	mAP0.5	Parameters/M	ms/img
YOLOv3 [22]	0.647	0.641	0.634	61.53	10.5
YOLOv5m [35]	0.702	0.709	0.699	20.95	7.6
YOLOv5s [32]	0.689	0.697	0.684	7.04	5.9
SSD(VGG) [19]	0.652	0.645	0.657	90.6	15.5
Xception-SSD [42]	0.675	0.672	0.663	56.8	14.92
MoblieNet-SSD [43]	0.679	0.683	0.674	43.6	11.76
YOLOv7 [26]	0.658	0.664	0.652	71.3	16.8
YOLOv5-IR	**0.712**	**0.719**	**0.708**	4.02	5.1
YOLOv5-IRL	0.698	0.705	0.702	**3.83**	**4.9**

The best results highlighted in bold.

From the experimental results presented in Table 3, it can be observed that the detection algorithm model proposed in this paper, specifically tailored for the characteristics of infrared dim and small targets, exhibits a smaller number of parameters, strong real-time performance, and achieves the best detection accuracy. Among all the detection algorithms, YOLOv5-IR stands out with the highest precision of 0.712, recall of 0.719, and mAP0.5 of

0.708, despite having a relatively low number of parameters at 4.02 and a fast processing speed. Furthermore, the proposed lightweight version, YOLOv5-IRL, demonstrates only a slight decrease in detection accuracy, but with an even smaller number of parameters and enhanced real-time performance.

Comparisons of detection effects on some images are shown in Figures 14 and 15. In Data1, the target is in a high-noise sky background and undergoing attitude changes during close-range flight. In Data2, the targets are flying at a long distance, transitioning from parallel to crossing flights, with unremarkable characteristics causing mutual interference. Data3 also features a high-noise sky background, with the target flying in a tilted attitude, providing very limited features for extraction. Data7 is in a rural field and muddy ground boundary area, where the background of the target changes continuously during flight. Data16 features a dense and complex forest background, with low contrast between the target and the background. Data20 is in a boundary area between a town and mountain peaks, with an overall low brightness in the background and the target appearing as a point.

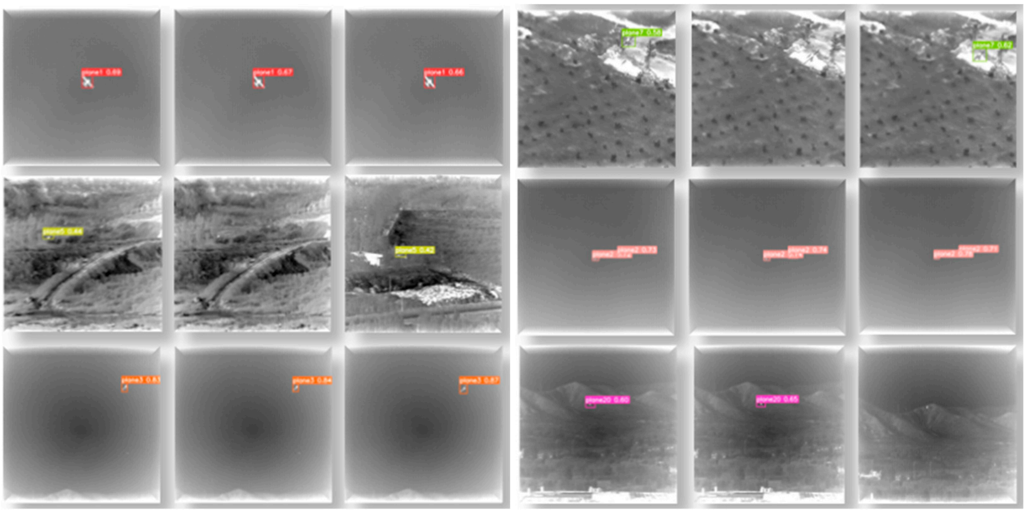


Figure 14. The detection performance of the YOLOv5s model.

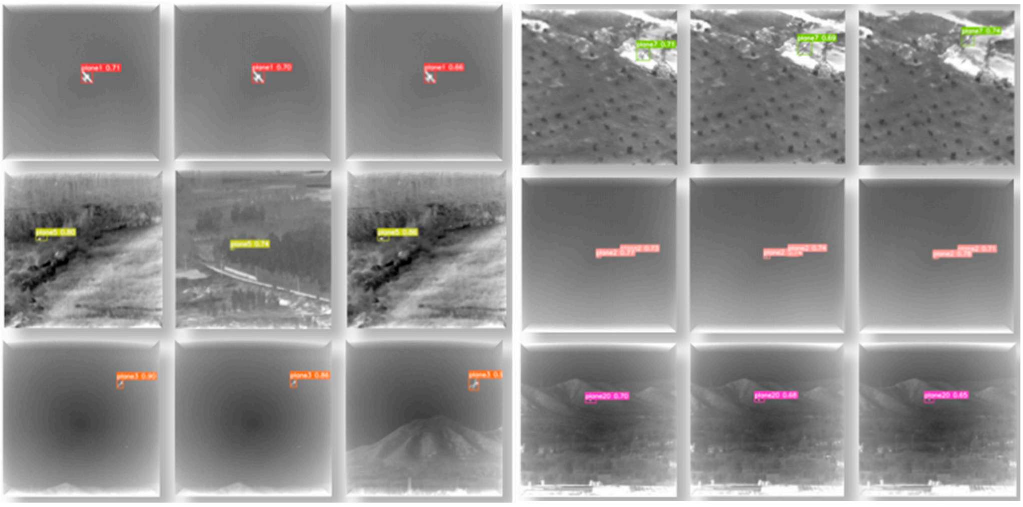


Figure 15. The detection performance of the YOLOv5-IR model.

As demonstrated by Figures 14 and 15, the YOLOv5s network model shows more pronounced detection effects for targets with simple backgrounds like Data1, but its detection accuracy decreases for crossing and tilted flight targets in Data2 and Data3. When dealing with Data7, Data16, and Data20, due to factors such as overall low background brightness, significant background variations, large target attitude changes, and small target feature areas, the detection accuracy is not high, and missed detections may occur. The YOLOv5-IR network model, on the other hand, can better handle these influencing factors, not only slightly improving detection accuracy but also better avoiding missed detections. The results indicate that the algorithm presented in this paper has better recognition outcomes for situations where the target background is difficult to effectively distinguish. Furthermore, the detection results of the algorithm have bounding boxes that fit the targets more closely, providing higher localization accuracy for small targets. Therefore, the algorithm in this paper offers higher detection accuracy for small infrared targets in complex environments.

In summary, the algorithm proposed in this paper shows higher detection accuracy for infrared dim and small targets in complex environments, while requiring a smaller number of model parameters and achieving shorter detection times.

4.5. Ablation Experiment

To illustrate the advantages of the improvements in the proposed YOLOv5-IR algorithm, ablation experiments were conducted in this paper. The experiment consisted of four groups: Group 1 was the original YOLOv5s network model; Group 2 replaced the backbone of the original network with an improved attention convolutional module; Group 3 added a small target detection layer based on Group 2; Group 4 replaced the CIOU loss function of the network model in Group 3 with the WIOU loss function. The experimental results for each group are shown in Table 4. The results indicate that Group 2 did not show significant improvement in metric parameters compared to Group 1, primarily because the improvement focused on reducing convolutional computations for network lightweighting. The addition of the detection layer in Group 3 provided larger feature maps and smaller detection anchor boxes during detection, enhancing the network model sensitivity to small infrared targets, resulting in a 1.4% increase in precision, a 2.5% increase in recall, and a 2.1% increase in mean average precision. Group 4 further adjusted the loss weight configuration using an improved loss function to balance samples of varying quality in the dataset, leading to a 1.1% increase in precision and a 0.2% increase in mean average precision compared to the original network model, showing slight improvements in evaluation metrics. Figure 16 provides a vertical comparison of the ablation experiment metrics.

Table 4. Comparison of ablation experiment indicators.

Groups	Backbone	Head	WIOU	Precision	Recall	mAP0.5
1				68.9%	69.7%	68.4%
2	√			68.7%	69.3%	68.5%
3	√	√		70.1%	72.2%	70.6%
4	√	√	√	71.2%	71.9%	70.8%

A checkmark (√) indicates that this improvement has been added.

In order to validate the effectiveness and superiority of the improvements to the backbone model and feature network of YOLOv5s in this paper for infrared dim and small target detection, the original YOLOv5s was trained using the same dataset under identical training parameters and hardware conditions. The model with the best performance across all batches was then tested to evaluate its detection performance. A comprehensive analysis and comparison of the training convergence and detection performance between the improved and original YOLOv5s models was conducted based on all performance evaluation metrics. During training, the mean Average Precision at 0.5 IoU (mAP0.5), Precision, and Recall are presented in Figure 17, where the YOLOv5s-Mv3-st model refers to the third group of models.

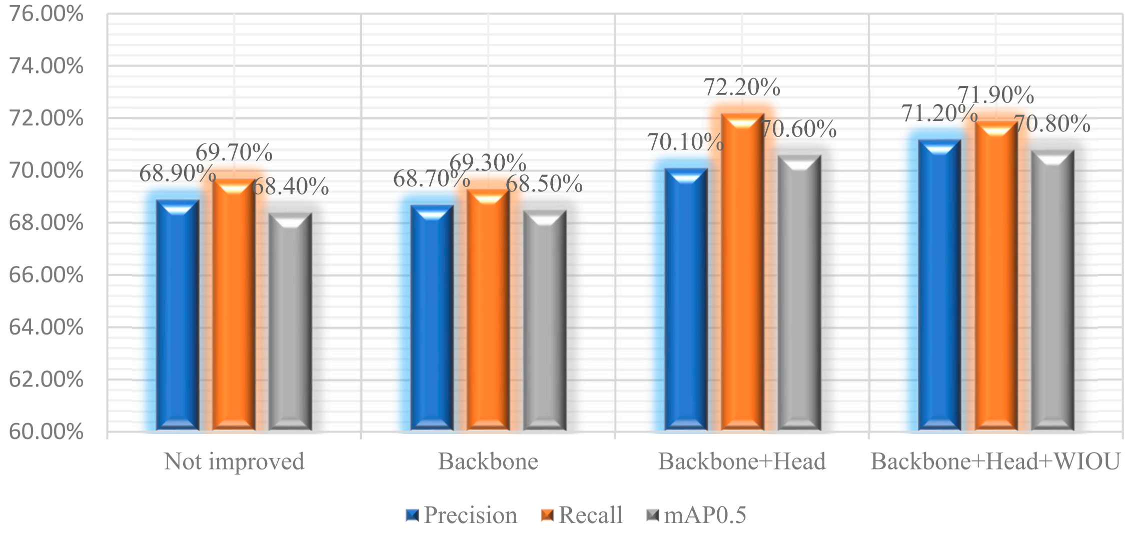
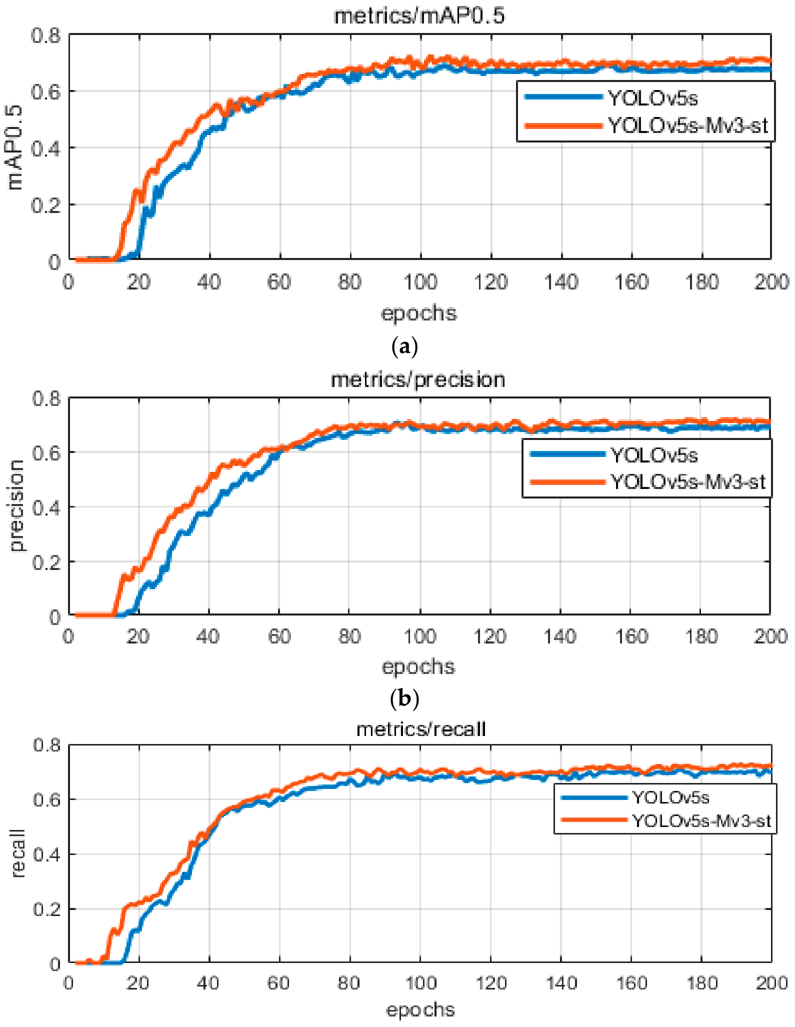


Figure 16. Longitudinal comparison of ablation experiment indicators.



(**c**)

Figure 17. Comparison of model evaluation metrics before and after the improvement. (**a**) mAP0.5; (**b**) precision; (**c**) recall.

The effects of the two network models on various metrics during training are shown in Figure 17. As can be seen from Figure 17, the orange curve represents the changes in the performance of the YOLOv5-Mv3-st network model with training rounds, while the blue curve represents the same for the YOLOv5s network model. As the number of training rounds increases, the changes in the two curves initially fluctuate within a small range. When the training reaches 40 rounds, a turning point is reached, and the training effect becomes significant. When the training proceeds to around 80 rounds, the increase in evaluation metrics slows down, and finally, they gradually stabilize. Furthermore, analysis of the numerical metrics shows that the lightweight model constructed in this chapter has slightly improved overall detection and recognition capabilities for small infrared targets compared to the original YOLOv5s model. The original YOLOv5s model has an mAP0.5 of 68.4%, a Precision of 68.9%, and a Recall of 69.7% for dataset detection. The improved YOLOv5-Mv3-st network model has an mAP0.5 of 70.6%, a Precision of 70.1%, and a Recall of 72.2%, which are increased by 2.2%, 1.2%, and 2.5%, respectively. In addition, the yellow curve shows a significant improvement over the blue curve in a smaller number of training rounds, indicating that YOLOv5-Mv3-st has stronger adaptability and sensitivity to the training and recognition of small infrared targets and can reach the optimal detection effect faster.

For the further lightweight YOLOv5-IRL algorithm, the level of pruning significantly influences the model accuracy. In order to validate the optimization effect and effectiveness of pruning on the network model in this experiment, a comparison was made between the mean Average Precision at 0.5 IoU (mAP0.5) of models under various pruning degrees and that of the YOLOv5-IR model. The experimental results are summarized in Table 5 below. The bold font in Table 5 denotes the proposed YOLOv5-IRL.

Table 5. Comparison of the pruning results.

Model	Pruning Rate	Original Network mAP0.5	After Fine-Tuning mAP0.5	Parameters/M	ms
YOLOv5-IR	0	70.8%	-	4.02	5.1
	0.2	**68.9%**	**70.2%**	**3.83**	**4.9**
	0.3	67.8%	68.3%	3.56	4.6
	0.4	64.7%	65.1%	3.25	4.3
	0.5	59.9%	61.5%	3.09	3.9

The bold font in the Table 5 denotes the proposed YOLOv5-IRL.

As shown in Table 5, with the increase in pruning rate, the number of network model parameters gradually decreases, leading to a slight reduction in the detection time for a single image. However, this is accompanied by a continuous decrease in the mean Average Precision at 0.5 IoU (mAP0.5). When the pruning rate is 20%, the mAP0.5 of the fine-tuned model decreases by only 0.6% compared to the original model, with a reduction of 0.19 MB in model parameters and a 0.2 ms decrease in detection time. At a pruning rate of 30%, the mAP0.5 further drops by 2.5%, accompanied by a 0.27 MB reduction in model parameters and a 0.3 ms decrease in detection time. However, when the pruning level is too high, while the detection time still decreases to some extent, the detection accuracy of the fine-tuned model decreases significantly, with mAP0.5 values of 65.1% and 61.5%, respectively. These results indicate that an excessively high pruning rate can lead to the removal of weights that effectively represent key target features, reducing the network model size but significantly impacting its detection accuracy. Therefore, this paper selects the YOLOv5-IR version with a pruning rate of 20% as the more lightweight version, YOLOv5-IRL, to enhance the feasibility of deploying the network model on resource-constrained platforms and the effectiveness of detection.

5. Discussion

In this paper, YOLOv5-IR and YOLOv5-IRL algorithms are proposed to achieve performance improvements in the field of infrared small and dim flying target detection. By integrating the MobileNetv3 depthwise separable convolution module, the model parameter count was successfully reduced while maintaining high detection accuracy. Experimental results indicate that compared to YOLOv5s, the improved models exhibit increased performance in mAP, Precision, and Recall, demonstrating that the modification enables better detection outcomes with fewer parameters. This improvement stems from several factors: Firstly, the incorporation of numerous attention mechanisms in the backbone feature extraction network effectively retains spatial and channel-wise key features of interest during the deep convolutional process, facilitating the next convolution step. Secondly, the increased upsampling and convolution layers in the feature fusion network, through cross-fusion, further deepen the integration of shallow and deep features, enabling the model to prioritize shape characteristics over semantic features. Additionally, the inclusion of target detection heads at higher levels of the feature fusion network, along with small-sized anchor boxes scanning over larger feature maps rich in detailed features, allows the model to detect even smaller targets. Lastly, a novel pruning structure designed for the YOLOv5-IR backbone network enables the proposed YOLOv5-IRL model to achieve faster detection speeds on resource-constrained devices with minimal precision compromise.

However, despite the advancements in lightweighting achieved by YOLOv5-IR and YOLOv5-IRL, limitations persist. Firstly, while the model parameter count has been reduced, its performance in handling target detection tasks under extremely low contrast and high noise environments still has room for improvement. Secondly, while the modified loss function WIOU enhances the model generalization ability for low-quality samples to some extent, further adjustments may be necessary for complex real-world scenarios to adapt to diverse target characteristics and background conditions. Future works will focus on addressing these two limitations.

6. Conclusions

In this paper, an innovative algorithm YOLOv5-IR is proposed for infrared small and dim target detection in complex environments, along with its further lightweighted version YOLOv5-IRL. Through analysis and systematic experimental validation, the following conclusions are derived:

(1) Network structure optimization: YOLOv5-IR outperforms the original YOLOv5s model in key evaluation metrics such as mean Average Precision (mAP), Precision, and Recall. This demonstrates that despite fewer parameters, the rational design of spatial attention and channel management mechanisms effectively extracts infrared target features. The addition of an enhanced head layer, sensitive to small targets, significantly enhances the focus on infrared small and dim targets. As a result, the proposed model effectively improves the detection of such targets in complex environments.

(2) Loss function optimization: By introducing an adaptive weighting loss function, the model has intensified its focus on targets within low-quality samples, resulting in enhanced detection performance. This demonstrates the paramount importance of loss function design in improving the model adaptability and generalization capabilities in complex scenarios.

(3) Lightweight design: Through the integration of lightweight modules and model pruning techniques, the proposed YOLOv5-IR and YOLOv5-IRL have faster detection speeds while maintaining high accuracy in detecting infrared small and dim targets. This supports the deployment and real-time application of the algorithms on mobile devices.

Compared to YOLOv5, the proposed YOLOv5-IR and YOLOv5-IRL algorithms exhibit a 42.9% and 45.6% reduction in model parameters, respectively, and a 13.6% and 16.9% decrease in detection time. Additionally, they achieve a 2.4% and 1.8% improvement in mAP0.5, respectively. The algorithms presented in this paper demonstrate exceptional performance in infrared small and dim target detection, striking a balance between detection accuracy and real-time performance. They provide an effective solution for edge device deployment in infrared target detection, particularly suitable for domains with stringent requirements on computational resources and real-time capabilities. Future works will aim to further optimize the model structure and enhance its robustness in detecting infrared small and dim targets under extreme conditions.

Author Contributions: Conceptualization, Y.C., Y.D. and K.C.; methodology, Y.C. and D.M.; software, Y.C. and D.M.; validation, K.C., Y.D. and D.Z.; formal analysis, Y.D. and D.Z.; investigation, Y.C., Y.D. and D.Z.; resources, Y.D., K.C. and D.Z.; data curation, Y.C., D.M., K.C. and Y.D.; writing—original draft preparation, Y.C., Y.D. and D.Z.; writing—review and editing, Y.C., Y.D. and D.Z.; visualization, Y.C., D.M. and D.Z.; supervision, Y.D. and D.Z.; project administration, Y.C., Y.D. and D.Z.; funding acquisition, Y.C. and D.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the "Double world-class project" discipline construction fund of Northwestern Polytechnical University (24GH0201265).

Data Availability Statement: Dataset available on request from the authors.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Zhang, W.; Cong, M.; Wang, L. Algorithms for optical weak small targets detection and tracking: Review. *Int. Conf. Neural Netw. Signal Process.* **2003**, *1*, 643–647.
2. Zhang, R.; Zhang, J.; Qi, X.; Zuo, H.; Xu, Z. Infrared target detection and recognition in complex scene. *Opto-Electron. Eng.* **2020**, *47*, 2003–2014.
3. Yang, Y.; Xu, C.; Ma, Y.; Huang, C. Review of research on infrared weak and small target detection algorithms under low signal-to-noise ratio. *Laser Infrared* **2019**, *49*, 643–649.
4. Huang, N.; Li, Z. A new method of infrared small target recognition. In Proceedings of the 2021 7th International Symposium on Mechatronics and Industrial Informatics (ISMII), Zhuhai, China, 22–24 January 2021; pp. 206–210.
5. Gu, Y.; Wang, C.; Liu, B.; Zhang, Y. A kernel-based nonparametric regression method for clutter removal in infrared small-target detection applications. *IEEE Geosci. Remote Sens. Lett.* **2010**, *7*, 469–473. [CrossRef]
6. Wang, X.; Peng, Z.; Kong, D.; Zhang, P.; He, Y. Infrared dim target detection based on total variation regularization and principal component pursuit. *Image Vis. Comput.* **2017**, *63*, 1–9. [CrossRef]
7. Dong, X.; Huang, X.; Zheng, Y.; Shen, L.; Bai, S. Infrared dim and small target detecting and tracking method inspired by human visual system. *Infrared Phys. Technol.* **2014**, *62*, 100–109. [CrossRef]
8. Wang, X.; LÜ, G.F.; Xu, L. Infrared dim target detection based on visual attention. *Infrared Phys. Technol.* **2012**, *55*, 513–521. [CrossRef]
9. Chen, C.; Li, H.; Wei, Y.; Xia, T.; Tang, Y. A local contrast method for small infrared target detection. *IEEE Trans. Geosci. Remote Sens.* **2013**, *52*, 574–581. [CrossRef]
10. Zhang, K.; Yang, K.; Li, S.; Chen, H. A difference-based local contrast method for infrared small target detection under complex background. *IEEE Access* **2019**, *7*, 105503–105513. [CrossRef]
11. Wei, Y.; You, X.; Li, H. Multiscale patch-based contrast measure for small infrared target detection. *Pattern Recognit.* **2016**, *58*, 216–226. [CrossRef]
12. Gao, C.; Meng, D.; Yang, Y.; Wang, Y.; Zhou, X.; Hauptmann, A. Infrared patch-image model for small target detection in a single image. *IEEE Trans. Image Process.* **2013**, *22*, 4996–5009. [CrossRef] [PubMed]
13. Parmar, N.; Vaswani, A.; Uszkoreit, J.; Kaiser, L.; Shazeer, N.; Ku, A.; Tran, D. Image transformer. In Proceedings of the International Conference on Machine Learning, PM-LR 2018, Stockholm, Sweden, 10–15 July 2018; pp. 4055–4064.
14. Liu, Z.; Lin, Y.; Cao, Y.; Han, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2021, Montreal, BC, Canada, 11–17 October 2021; pp. 10012–10022.
15. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the 27th IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2014, Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
16. Girshick, R. Fast R-CNN. In Proceedings of the 2015 IEEE International Conference on Computer Vision(ICCV), Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
17. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [CrossRef] [PubMed]

18. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask R-CNN. In Proceedings of the IEEE International Conference on Computer Vision 2017, Venice, Italy, 22–29 October 2017; pp. 2961–2969.
19. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.; Berg, A. SSD: Single shot multibox detector. In Proceedings of the 14th European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; pp. 21–37.
20. Redmon, J.; Diwala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
21. Redmon, J.; Farhadi, A. YOLO9000: Better, faster, stronger. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 6517–6525.
22. Redmon, J.; Farhadi, A. YOLOv3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
23. Bochkovskiy, A.; Wang, C.; Mao, H. Yolov4: Optimal speed and accuracy of object detection. In Proceedings of the ArXiv Computer Vision and Pattern Recognition 2020, Cornell University, Ithaca, NY, USA, 15–17 March 2020; pp. 10923–10934.
24. Ge, Z.; Liu, S.; Wang, F.; Li, Z.; Sun, J. YOLOX: Exceeding YOLO series in 2021. *arXiv* **2021**, arXiv:2107.08430.
25. Liu, X.; Gong, W.; Shang, L.; Li, X.; Gong, Z. Remote Sensing Image Target Detection and Recognition Based on YOLOv5. *Remote Sens.* **2023**, *15*, 4459. [CrossRef]
26. Wang, C.; Bochkovskiy, A.; Liao, H. YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors. In Proceedings of the 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Vancouver, BC, Canada, 18–22 June 2023; pp. 7464–7475.
27. Kim, J.H.; Hwang, Y. GAN-based synthetic data augmentation for infrared small target detection. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 1–12. [CrossRef]
28. Gu, Y.; Zhang, H.; Sun, S. Infrared small target detection model with multiscale fractal attention. *J. Electron. Inf. Technol.* **2023**, *45*, 3002–3011.
29. Li, J.; Liang, X.; Wei, Y.; Xu, T.; Feng, J.; Yan, S. Perceptual generative adversarial networks for small object detection. In Proceedings of the Conference on Computer Vision and Pattern Recognition, Piscataway, NJ, USA, 18–24 June 2017; pp. 1222–1230.
30. Liu, T.; Yang, D.; Song, J.; Fu, R.; He, J. Air-based down-ward-looking multi-angle infrared target recognition. *Appl. Electron. Tech.* **2022**, *48*, 131–139.
31. Hou, Q.; Zhang, L.; Tan, F.; Xi, Y.; Zheng, H.; Li, N. ISTDU-Net: Infrared small-target detection U-Net. *IEEE Geosci. Remote Sens. Lett.* **2022**, *3*, 1–5. [CrossRef]
32. Fan, X.; Ding, W.; Qin, W.; Xiao, D.; Min, L.; Yan, H. Fusing self-attention and coordconv to improve the YOLOv5s algorithm for infrared weak target detection. *Sensors* **2023**, *23*, 6755. [CrossRef]
33. He, J.; Yang, D.; An, C.; Li, J.; Huang, C. Infrared dim target detection technology based on IRI-CNN. In Proceedings of the Seventh Asia Pacific Conference on Optics Manufacture and 2021 Inter-national Forum of Young Scientists on Advanced Optical Manufacturing (APCOM and YSAOM 2021), Shanghai, China, 28–31 October 2022; pp. 1350–1361.
34. Mou, X.; Lei, S.; Zhou, X. YOLO-FR: A YOLOv5 infrared small target detection algorithm based on feature reassembly sampling method. *Sensors* **2023**, *23*, 2710. [CrossRef] [PubMed]
35. Yang, R.; Li, W.; Shang, X.; Zhu, D.; Man, X. KPE-YOLOv5:an improved small target detection algorithm based on YOLOv5. *Electronics* **2023**, *12*, 817. [CrossRef]
36. Iandola, F.; Han, S.; Moskewicz, M.; Ashraf, K.; Dally, K.; Keutzer, K. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5 MB model size. *arXiv* **2016**, arXiv:1602.07360.
37. Howard, A.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T. MobileNets: Efficient convolutional neural networks for mobile vision applications. *arXiv* **2017**, arXiv:1704.04861.
38. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L. Mobilenetv2: Inverted residuals and linear bottlenecks. *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.* **2018**, *3*, 4510–4520.
39. Howard, A.; Sandler, M.; Chu, G.; Chen, L.; Chen, B.; Tan, M.; Wang, W.; Zhu, Y.; Pang, R.; Vasudevan, V.; et al. Searching for mobilenetv3. In Proceedings of the IEEE/CVF International Conference on Computer Vision 2019, Seoul, South Korea, 27 October–2 November 2019; pp. 1314–1324.
40. Zhang, X.; Zhou, X.; Lin, M.; Sun, J. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2018, Salt Lake City, UT, USA, 18–22 June 2018; pp. 6848–6856.
41. Han, K.; Wang, Y.; Tian, Q.; Guo, J.; Xu, C. Ghostnet: More features from cheap operations. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition 2020, Seattle, WA, USA, 14–19 June 2020; pp. 1580–1589.
42. Gosaye, K.; Moloo, R. A Mobile Application for Fruit Fly Identification Using Deep Transfer Learning: A Case Study for Mauritius. In Proceedings of the 2022 International Conference for Advancement in Technology, Goa, India, 21–23 January 2022; pp. 1–5.
43. Murthy, C.; Hashmi, M.; Keskar, A. Optimized MobileNet+ SSD: A real-time pedestrian detection on a low-end edge device. *Int. J. Multimed. Inf. Retr* **2021**, *10*, 171–184. [CrossRef]
44. Vadera, S.; Ameen, S. Methods for Pruning Deep Neural Networks. *IEEE Access* **2022**, *10*, 63280–63300. [CrossRef]
45. Peng, B.; Tan, W.; Li, Z.; Zhang, S.; Xie, D.; Pu, S. Extreme network compression via filter group approximation. In Proceedings of the European Conference on Computer Vision (ECCV) 2018, Munich, Germany, 8–14 September 2018; pp. 300–316.
46. Gou, J.; Yu, B.; Maybank, S.; Tao, D. Knowledge Distillation: A Survey. *Int. J. Comput. Vis.* **2021**, *129*, 1789–1819. [CrossRef]

47. Liu, L.; Ke, C.; Lin, H.; Xu, H. Research on pedestrian detection algorithm based on MobileNet-YoLo. *Comput. Intell. Neurosci.* **2022**, *5*, 1–12. [CrossRef]
48. Sha, M.; Zeng, K.; Tao, Z.; Wang, Z.; Liu, Q. Lightweight pedestrian detection based on feature multiplexed residual network. *Electronics* **2023**, *12*, 918. [CrossRef]
49. Li, C.; Wang, Y.; Liu, X. A multi-pedestrian tracking algorithm for dense scenes based on an attention mechanism association. *Appl. Sci.* **2022**, *12*, 9597. [CrossRef]
50. Zou, F.; Li, X.; Xu, Q.; Sun, Z.; Zhu, J. Correlation-and-correction fusion attention network for occluded pedestrian detection. *IEEE Sens. J.* **2023**, *23*, 6061–6073. [CrossRef]
51. Li, M.; Sun, G.; Yu, J. A pedestrian detection network model based on improved YOLOv5. *Entropy* **2023**, *25*, 381. [CrossRef] [PubMed]
52. Hao, S.; Gao, S.; Ma, X.; An, B.; He, T. Anchor-free infrared pedestrian detection based on cross-scale feature fusion and hierarchical attention mechanism. *Infrared Phys. Technol.* **2023**, *131*, 104660. [CrossRef]
53. Hui, B.; Song, Z.; Fan, H.; Zong, P.; Hu, W.; Zhang, X.; Lin, J.; Su, H.; Jin, W.; Zhang, Y. Weak and small aircraft target detection and tracking data set in infrared images under ground/air background. *Chin. Sci. Data: Chin. Engl. Online Ed.* **2020**, *5*, 12.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.