



## Article

# Improved Detection Method for Micro-Targets in Remote Sensing Images

Linhua Zhang <sup>1,2</sup> , Ning Xiong <sup>3</sup>, Wuyang Gao <sup>2</sup> and Peng Wu <sup>4,\*</sup> 

<sup>1</sup> Department of Computer Engineering, Taiyuan Institute of Technology, Taiyuan 030008, China; zhanglh@tit.edu.cn

<sup>2</sup> School of Computer Science and Technology, Taiyuan Normal University, Jinzhong 030619, China; gao243361304@163.com

<sup>3</sup> School of Innovation, Design and Engineering, Malardalen University, 72123 Vasteras, Sweden; ning.xiong@mdu.se

<sup>4</sup> School of Computer and Information Technology, Beijing Jiaotong University, Beijing 100044, China

\* Correspondence: 14112078@bjtu.edu.cn

**Abstract:** With the exponential growth of remote sensing images in recent years, there has been a significant increase in demand for micro-target detection. Recently, effective detection methods for small targets have emerged; however, for micro-targets (even fewer pixels than small targets), most existing methods are not fully competent in feature extraction, target positioning, and rapid classification. This study proposes an enhanced detection method, especially for micro-targets, in which a combined loss function (consisting of NWD and CIUO) is used instead of a singular CIUO loss function. In addition, the lightweight Content-Aware Reassembly of Features (CARAFE) replaces the original bilinear interpolation upsampling algorithm, and a spatial pyramid structure is added into the network model's small target layer. The proposed algorithm undergoes training and validation utilizing the benchmark dataset known as AI-TOD. Compared to speed-oriented YOLOv7-tiny, the mAP0.5 and mAP0.5:0.95 of our improved algorithm increased from 42.0% and 16.8% to 48.7% and 18.9%, representing improvements of 6.7% and 2.1%, respectively, while the detection speed was almost equal to that of YOLOv7-tiny. Furthermore, our method was also tested on a dataset of multi-scale targets, which contains small targets, medium targets, and large targets. The results demonstrated that mAP0.5:0.95 increased from "9.8%, 54.8%, and 68.2%" to "12.6%, 55.6%, and 70.1%" for detection across different scales, indicating improvements of 2.8%, 0.8%, and 1.9%, respectively. In summary, the presented method improves detection metrics for micro-targets in various scenarios while satisfying the requirements of detection speed in a real-time system.

**Keywords:** micro-targets; NWD; CARAFE; spatial pyramid; remote sensing images



**Citation:** Zhang, L.; Xiong, N.; Gao, W.; Wu, P. Improved Detection Method for Micro-Targets in Remote Sensing Images. *Information* **2024**, *15*, 108. <https://doi.org/10.3390/info15020108>

Academic Editors: Tao Tang, Canbin Hu and Yuli Sun

Received: 11 January 2024

Revised: 1 February 2024

Accepted: 7 February 2024

Published: 12 February 2024



**Copyright:** © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Remote sensing images, acquired through the detection of ground object data, have become indispensable digital assets, considering the rapid advancement of remote sensing technology [1]. Target detection technology offers precise and valuable data for remote sensing image analysis, making significant contributions to research on natural resource distribution, terrain features, ports, and more. Additionally, the detection technology of small and micro-targets such as airplanes, automobiles, and vessels against complex backgrounds in remote sensing images has also gradually been taken seriously.

In the domain of image detection, the utilization of Deep Learning (DL) algorithms has become indispensable for precise target detection [2]. These DL-based methods leverage intricate neural networks to discern and identify objects within remotely sensed imagery, thereby contributing to enhanced accuracy and efficiency of detection. In general, these DL-based methods can be classified as two- or one-stage approaches. Two-stage methods generate candidate boxes through sampling, utilize Convolutional Neural Networks

(CNNs) for feature extraction and classification, and ultimately achieve accurate target localization through post processing operations. For example, the region-based CNN (R-CNN) series of algorithms [3–5] is a classical two-stage approach. In contrast, one-stage object detection methods do not generate candidate boxes. Instead, they convert the task of localizing the target bounding box into a regression problem and successfully achieve accurate target localization through regression. Representative one-stage algorithms include the Single Shot MultiBox Detector (SSD) [6], Centernet [7], and the You Only Look Once (YOLO) algorithm series [8–18]. Consequently, the former surpasses the latter in target detection accuracy and localization; however, the latter outperforms the former in detection speed. In recent years, transformer-based architectures like Detection Transformer (DETR) [19], have advanced object detection through self-attention mechanisms. Another notable model, the Segment Anything Model (SAM) [20], is an exemplary image segmentation model based on the transformer framework. These advancements have streamlined detection pipelines, eliminating the need for handcrafted components and achieving state-of-the-art results. However, transformer-based models typically exhibit slower speeds, limiting their application in real-time monitoring fields.

Processing speed is a crucial consideration in devices for real-time systems; hence, the one-stage detection method has attracted more scholars' attention, particularly for real-time detection scenarios. Huo et al. [21] proposed SAFF-SSD for small-object detection. Building upon SSD, they achieved enhanced feature extraction capabilities for SAFF-SSD through the incorporation of a local light transformer block. Betti et al. introduced YOLO-S [22], a network akin to YOLO but specifically designed for detecting small objects. This approach demonstrates enhanced performance in detecting small objects. Lai developed a feature extraction component that combines a CNN with multi-head attention to expand the receptive view [23]. The STC-YOLO algorithm performs well for traffic-sign identification. Qu et al.'s approach [24] introduced a feature fusion strategy based on an attention mechanism. By merging target information from different scales, this strategy enhances the semantic expression of shallow features, consequently improving the tiny-object identification capacity of the algorithm. Our team has also proposed the PDWT-YOLO [25] algorithm for target detection in unmanned aerial vehicle images, effectively enhancing its capability to detect small objects. Despite the advancements made through these improvements, several lingering issues continue to persist. SAFF-SSD [21] demonstrates good feature extraction capabilities but lacks a significant speed advantage. YOLO-S [20] employs a detection network with relatively outdated methods, and STC-YOLO [23] is primarily applied to traffic sign detection. The algorithm proposed in Ref. [24] exhibits competent detection performance; however, its detection time is also significantly increased. PDWT-YOLO [25] is primarily designed for detecting small targets and exhibits a fast detection speed; however, it is not suitable for detecting much smaller targets, such as the targets in AI-TOD [26].

In the domain of object detection, small objects typically stand for objects with a pixel area smaller than  $32 \times 32$  pixels [27]. Objects in remote sensing images are often even smaller, such as in AI-TOD, where the average size of targets is approximately 12.8 pixels. In this paper, objects smaller than 16 pixels are defined as "micro-targets". Fewer pixels result in less feature information being extracted from the target, which significantly increases the difficulty of detection. Hence, most small-object detection methods are not suitable for micro-targets, prompting some researchers to focus on their detection [25,26]. Guanlin Lu et al. [28] propose MStrans, a multi-scale transformer-based aerial object detector that effectively tackles the difficulties of detecting micro-instances in aerial images. Shuyan Ni et al. [29] introduce JSDNet, a network designed for detecting micro-targets by leveraging the geometric Jensen–Shannon divergence. JSDNet incorporates the Swin Transformer model to enhance feature extraction for micro-targets and addresses IoU sensitivity through the JSMD module. Nevertheless, MStrans [28] and JSDNet [29] do not meet the real-time application requirements due to their larger model sizes and slower speeds.

This paper introduces a real-time detection method specifically for micro-targets by modifying the YOLOv7-tiny network and loss function. The proposed method incorporates three key innovations, as outlined below:

- (1) Integration of a new loss function: A normalized weighted intersection over union (NWD) [30] is integrated with a CIoU to replace the singular use of CIoU. Through experiments, the optimal fusion factor for the NWD and CIoU is established to mitigate the sensitivity issue related to micro-targets.
- (2) Utilization of lightweight Content-Aware Reassembly of Features (CARAFE): The CARAFE operator takes the place of the initial bilinear interpolation upsampling operator. This lightweight operator effectively reassembles features within predefined regions centered at each position, thereby achieving enhanced feature extraction related to micro-targets through weighted combinations.
- (3) Inclusion of a Spatial Pyramid Structure (SPP) in the high-resolution feature map layer: Contextual Spatial Pyramid Pooling (CSPSPP) is added to the object detection layer, which has a resolution of  $80 \times 80$  pixels. Hence, the algorithm's ability to capture the various scale features of micro-targets has improved.

These improvements collectively address the shortcomings of the existing models, enhancing accuracy and efficiency for the detection of micro-targets. Experimental results validate the efficacy of the suggested approach in identifying micro-targets.

This paper contains six sections that comprehensively address the realm of micro-object detection. Section 2 synthesizes the relevant theoretical frameworks. Section 3 discusses the proposed innovations, details the improved modules, and elucidates the rationale behind each improvement, including the refinements to the Intersection Over Union (IOU) loss function, the integration of CARAFE module, and addition of the CSPSPP module. Section 4 outlines the experiments, describes the chosen datasets and parameter configurations, and provides an in-depth examination of the outcomes. Section 5 presents the discussion of results, provides a comprehensive comparison with classic algorithms, and highlights the current shortcomings and directions for improvement in the research. Section 6 demonstrates the conclusions.

## 2. Related Work

This section provides background information on the detection challenges associated with micro-targets, summarizing relevant technical branches such as network architecture, loss functions, and feature interpolation. We begin by exploring YOLOv7-tiny, a crucial framework in the object detection domain that serves as the foundation for real-time detection, while its steps in feature extraction and fusion significantly impact the accuracy of micro-target detection. Subsequently, we delve into the IOU loss function for improving the precision of object detection predictions. Additionally, we carefully examine feature upsampling, a tailored operation in convolutional network architectures for object detection. Relevant studies suggest that preserving detailed features during upsampling contributes to the detection of micro-targets. These ideas form the basis for the core structure of the proposed method.

### 2.1. YOLOv7-Tiny

The YOLOv7-tiny network is a lightweight framework derived from YOLOv7 [31], which preserves the original cascading-based model-scaling strategy while modifying the Efficient Long-Range Aggregation Network (ELAN). The YOLOv7-tiny algorithm, shown in Figure 1, provides detection accuracy along with relatively low parameters. Hence, it is especially suitable for scenarios requiring real-time system.

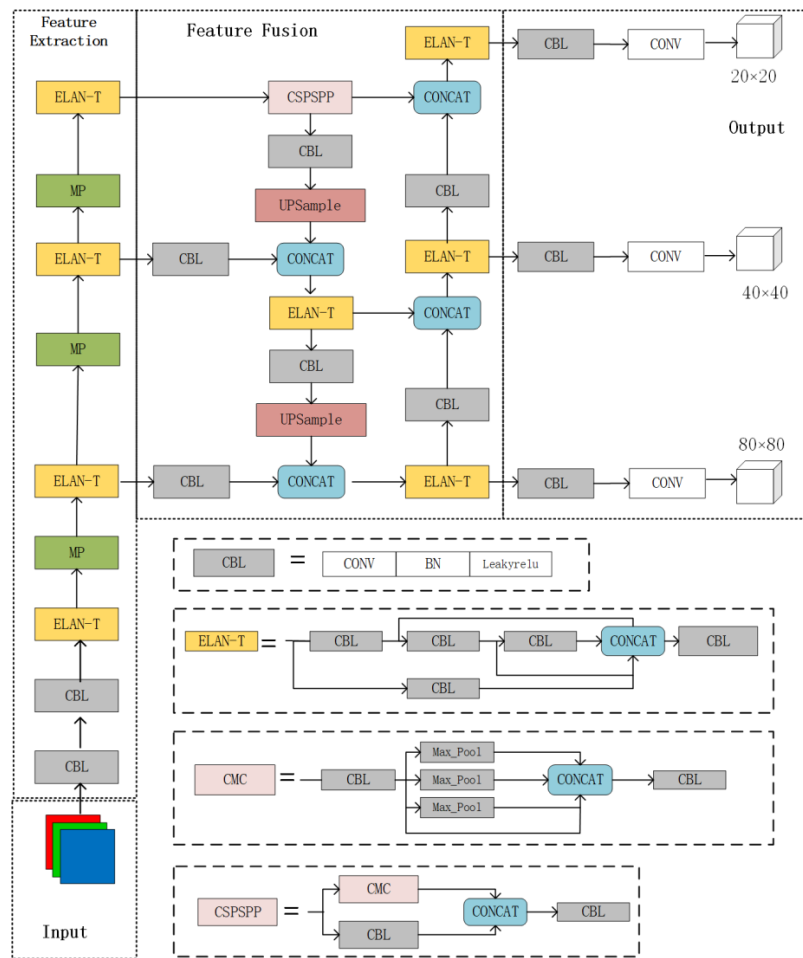


Figure 1. YOLOv7-tiny network architecture.

Innovative techniques are employed in the YOLOv7-tiny algorithm for enhanced training speed and reduced memory consumption. Mosaic technology is utilized at the input stage. Image preprocessing operations, including cropping and scaling, are applied to standardize pixel values.

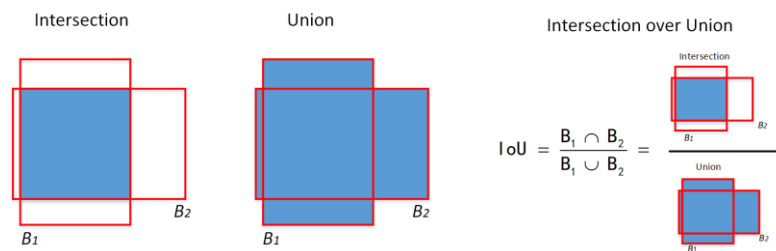
The feature extraction network is composed of blocks known as the Convolutional Block Layer (CBL), an improved ELAN layer called “ELAN-T,” and Mixed-Precision Convolution (MPCConv) layers. The CBL blocks extract raw features, the ELAN-T layer refines these features, and the MPCConv layer concatenates the tensors of the diverse features. The Path Aggregation Feature Pyramid Network (PAFPN) architecture is utilized in YOLOv7-tiny for feature fusion. To achieve multiscale learning, this structure integrates robust semantic data from high-level layers of a Feature Pyramid Network (FPN) [32] and strong localization information tensors from a Path Aggregation Network (PANet) [33]. However, tensor concatenation and nearest-neighbor interpolation upsampling in a fusion network may not fully address the need for comprehensive feature integration between adjacent layers. Additionally, it is difficult to balance the target detection speed and accuracy of nearest-neighbor interpolation methods. As a result, small targets features and information loss may be neglected.

The output step refines the prediction findings by introducing an implicit representation (Implicit) method and using the IDetect [31] detection head, which is comparable to the YoloR model [34]. Simultaneously, this approach categorizes large, medium, and small images based on the associated fusion characteristic values.

## 2.2. IOU Loss Function

IOU serves as a measure to evaluate the accuracy of bounding-box positioning. In this approach, the overlap ratio is represented as the intersection divided by the union. IOU is a straightforward measurement criterion that is applicable to any task for which bounding box prediction appears in the output. The ideal scenario is a complete overlap, where the ratio is 1; the worst-case scenario has no overlap, yielding a ratio of zero.

The IOU calculation is illustrated in Figure 2, where  $B_1$  stands for the ground truth and  $B_2$  represents bounding boxes.



**Figure 2.** IOU calculation.

Although IOU is commonly utilized as a measure for object detection, it is limited to cases where the bounding boxes overlap. In order to overcome this limitation, Rezatofighi et al. [35] presented Generalized IOU (GIOU), which incorporates a penalty term based on the minimum bounding box transformation. However, GIOU degrades to IOU when one bounding box encloses another. Distance IOU (DIOU), proposed by Zheng et al. [36], overcomes the limitations of IOU and GIOU. DIOU enhances the IOU by augmenting the separation between the central coordinates of the predicted and actual boxes, thereby accelerating convergence speed of loss function. Building on DIOU, CIOU incorporates a hyperparameter to adjust the significance of both the distance between center coordinates and aspect ratio, thereby providing a better evaluation in cases with significant aspect ratio differences. However, CIOU emphasizes aspect ratio differences rather than the differences in width and height relative to their confidence.

To address this concern, Zhang et al. [37] proposed Enhanced IOU (EIOU) loss. EIOU decomposes the aspect ratio component by utilizing the minimum bounding rectangles to calculate the intersection and union. This formulation effectively penalizes inaccurate predictions. GIOU, CIOU, and DIOU are primarily employed in non-maximum suppression and loss functions as IOU replacements.

Yang et al. [38] introduced the Gaussian Wasserstein Distance (GWD) loss, focusing on directed object detection. GWD aims to address discontinuities and non-square shapes in directed object detection. A novel NWD metric was proposed by Wang and colleagues [30], specifically designed for the detection of micro-objects utilizing Wasserstein distance. The NWD metric reliably indicates the disparities between distributions, even in cases where they have no overlap. In contrast to IOU, this novel measure demonstrates enhanced evaluation of the resemblance among micro-objects.

## 2.3. Feature Upsampling

The current methods, including nearest-neighbor and bilinear interpolation, utilize the spatial pixel distances as a guiding factor for facilitating the upsampling procedure. Nevertheless, these approaches only consider neighboring pixels at a subpixel level and do not capture the necessary semantic information for densely predicted scenes.

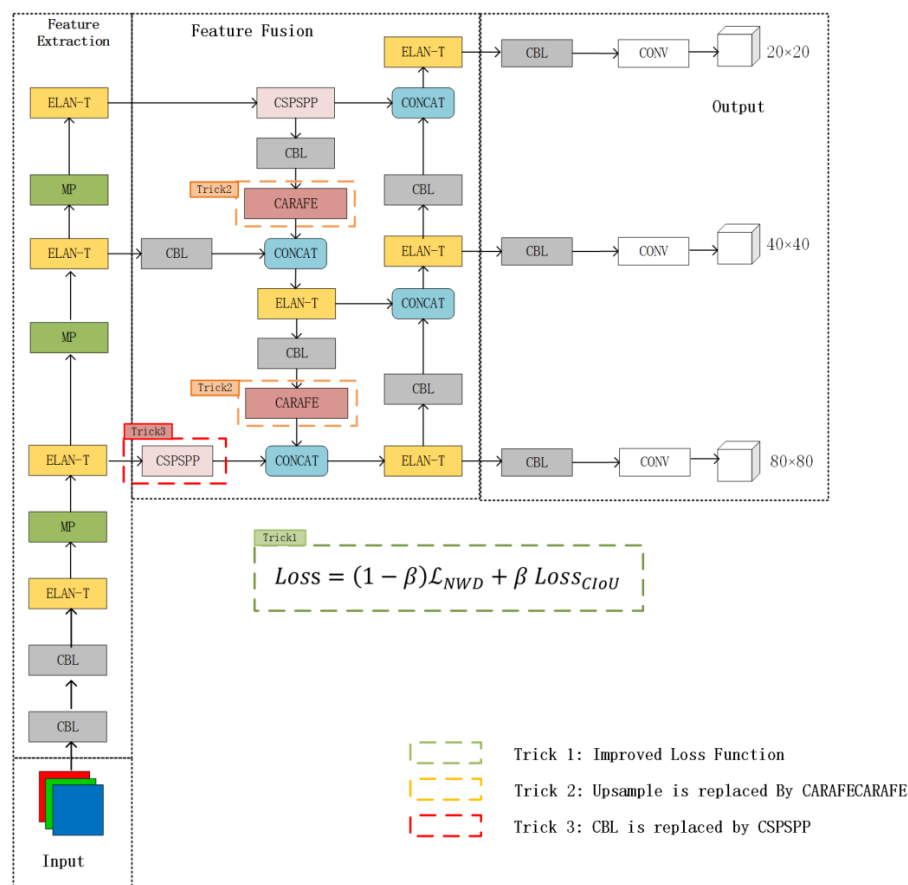
Deconvolution is another important upsampling method [39], which is based on the reverse operation of the convolutional layers. This is achieved by acquiring a set of up-sampling kernels that are unrelated to specific instances. However, deconvolution has two major drawbacks. Initially, a uniform kernel is applied across the entire image, with the underlying content being disregarded; thus, the responsiveness to local variations is con-

strained. Subsequently, owing to the numerous parameters of the deconvolution operator, the computational burden increases significantly when larger kernel sizes are employed. This complexity hinders the effective coverage of expansive regions beyond confined neighborhoods, consequently restricting their expressive capabilities and performance.

In order to overcome these limitations, Wang provided a lightweight yet efficient module known as CARAFE [40], which is characterized by minimal redundancy, robust feature fusion capability, and rapid execution speed. This operator addresses the shortcomings of traditional upsampling methods by efficiently capturing and leveraging content-aware information during the upsampling process; thus, it is particularly well-suited to dense prediction tasks.

### 3. Methodology

In this study, our method was enhanced for micro-targets by improving the loss function, feature upsampling module, and feature fusion for micro-objects. The upgraded network architecture is illustrated in Figure 3. Firstly, the CIOU and NWD were combined to create a novel loss function, which replaced the original CIOU for improved sensitivity towards micro-target positions. Secondly, the initial nearest-neighbor interpolation operator was replaced with the CARAFE upsampling operator. This upgrade enabled the network to leverage the background information around micro-targets and accurately extract the target characteristics. Thirdly, in order to tackle the challenges presented by variations in pixel sizes and scales of micro-targets, we replaced the CBL before the feature fusion of the small target layer with CSPSPP.



**Figure 3.** Improved network architecture. The green dotted box represents Trick 1, in which the new loss function is used. The orange dotted box is Trick 2, in which Upsample in the structure is replaced by CARAFE. The red dotted box represents Trick 3, in which CBL in the structure is replaced by CSPSPP.

### 3.1. Loss Function Optimization

#### 3.1.1. NWD Loss Function

Regarding target identification in remote sensing imagery, micro-targets pose a significant challenge because of their heightened sensitivity to IOU. To address this concern, we used a position regression loss function incorporating the NWD to mitigate the IOU sensitivity to micro-object position variations. NWD offers a unique advantage as it assesses distribution similarity even when the bounding boxes do not overlap or fully contain each other. The NWD's insensitivity to varying scales makes it particularly suitable for comparing micro-targets.

For two-dimensional Gaussian distributions represented by  $\mu_1 = N(m_1, \Sigma_1)$  and  $\mu_2 = N(m_2, \Sigma_2)$ , we define the second-order Wasserstein distance as

$$W_2^2(\mu_1, \mu_2) = \| m_1 - m_2 \|_2^2 + \text{Tr}(\Sigma_1 + \Sigma_2 - 2 \left( \Sigma_1^{\frac{1}{2}} \Sigma_2 \Sigma_1^{\frac{1}{2}} \right)^{\frac{1}{2}}) \quad (1)$$

Simplifying Equation (1) yields

$$W_2^2(\mu_1, \mu_2) = \| m_1 - m_2 \|_2^2 + \| \Sigma_1^{1/2} - \Sigma_2^{1/2} \|_F^2 \quad (2)$$

Using the Frobenius norm ( $\| \cdot \|_F$ ), Equation (2), derived from Gaussian distributions  $\mathcal{N}_a$  and  $\mathcal{N}_b$  representing bounding boxes  $A = (cx_a, cy_a, w_a, h_a)$  and  $B = (cx_b, cy_b, w_b, h_b)$ , can be simplified to Equation (3):

$$W_2^2(\mathcal{N}_a, \mathcal{N}_b) = \left\| \left( \begin{bmatrix} cx_a, cy_a, \frac{w_a}{2}, \frac{h_a}{2} \end{bmatrix}^T, \begin{bmatrix} cx_b, cy_b, \frac{w_b}{2}, \frac{h_b}{2} \end{bmatrix}^T \right) \right\|_2^2 \quad (3)$$

In Equation (3),  $(cx, cy)$ ,  $w$ , and  $h$ , respectively, stand for center coordinates, width, and height.

The corresponding NWD is constructed using the normalized index as Equation (4):

$$NWD(\mathcal{N}_a, \mathcal{N}_b) = \exp \left( - \frac{\sqrt{W_2^2(\mathcal{N}_a, \mathcal{N}_b)}}{C} \right) \quad (4)$$

In this case,  $C$  is a constant. The NWD-based loss is shown in Equation (5):

$$\mathcal{L}_{NWD} = 1 - NWD(\mathcal{N}_a, \mathcal{N}_b) \quad (5)$$

#### 3.1.2. Improved Loss Function

After using NWD, we have successfully addressed the shortcomings of target detection networks in detecting micro-objects. However, the singular use of NWD ignores the detection of large and medium objects. Therefore, we provide an approach that combines NWD and CIOU. Through this approach, the model can comprehensively consider targets of various sizes and improve regression accuracy.

The YOLOV7-tiny coordinate loss is computed using  $CIOU$  as Equation (6):

$$Loss_{CIOU} = 1 - IoU + \frac{\rho^2(b, b^{st})}{c^2} + \alpha v \quad (6)$$

where  $v = \frac{4}{\pi^2} \left( \arctan \frac{w^{st}}{h^{st}} - \arctan \frac{w}{h} \right)^2$  and the equilibrium parameter  $\alpha = \frac{v}{1 - IOU + v}$ .  $IoU$  denote the intersection of the real boxes and predicted boxes, respectively. The predicted box's center point and the true box's Euclidean distance are separated by  $\rho^2(b, b^{st})$ . The bounding box with the smallest diagonal length for both the truth and predicted boxes is represented by  $c$ ;  $v$  is used to evaluate how consistently the aspect ratio changes.

In order to harness the synergy between the two loss functions and capitalize on their individual strengths, we meticulously designed Equation (7):

$$Loss = (1 - \beta)\mathcal{L}_{NWD} + \beta Loss_{CIoU} \quad (7)$$

where  $\beta$  serves as the fusion factor, indicating the *NWD* and *CIoU* proportions. The fusion-factor selection approach is described in the discussion of Section 4.2.4 Sub-Experiment 1.

Equation (7) combines the qualities of both loss functions while overcoming their respective shortcomings to some extent. It results in a more robust, versatile, and efficient loss function, with the potential to improve overall performance in diverse applications.

### 3.2. Replacement of Upsampling Operator with CARAFE

The initial feature fusion network utilizes nearest-neighbor element interpolation for sampling, a process that frequently results in discontinuous gray values and a degradation of image quality, thus affecting the detection ability of micro-objects.

In order to tackle this issue, we enhanced the feature fusion module by replacing conventional nearest-neighbor interpolation with CARAFE, which is a lightweight upsampling operator. Unlike traditional methods, CARAFE employs a content-aware algorithm to derive weighted combinations to reassemble features within a predefined zone centered on each location. The resulting features are rearranged into a spatial block, which enhances the detection capacity and improves the detection ability of micro-targets in the feature map.

Within the feature pyramid structure, CARAFE facilitates feature map sampling at a multiple of two, seamlessly integrating into the PAFPN by replacing the nearest-neighbor interpolation. CARAFE is characterized by minimal redundancy, robust feature fusion capabilities, and efficient operation. Its smooth integration into existing structures eliminates the need for additional modifications. Moreover, when employed across all feature layers, the CARAFE operator ensures a smooth transition, allowing for contextual data combination within a larger receptive field. The CARAFE enhancement to the algorithm is illustrated in trick 2 of Figure 3, and its internal network is detailed in Figure 4.

The CARAFE network has two main modules: kernel prediction and content-aware reconstruction units. The former generates reconstructed kernels, which are then used by the content-aware reconstruction module to rebuild features.

The kernel prediction unit consists of three crucial components: a channel compressor, a content encoder, and a kernel normalizer. The first component is the input feature channel from  $C$  to  $C_m$  (usually set to 64) using a  $1 \times 1$  convolutional layer, effectively reducing the parameters and computing costs for the subsequent stages. This reduction enhances the overall efficiency of the CARAFE module.

The content encoder generates a reassembly kernel of size  $k_{up}$  by employing a convolution layer with  $k_{encoder}$  as the convolution kernel size. The encoder parameter is denoted as  $k_{encoder} \times k_{encoder} \times C_m \times C_{up}$ , where  $C_{up} = \sigma^2 k_{up}^2$ . The content encoder produces reassembly kernels with a size of  $C_{up} \times H \times W$ , while  $H$  stands for height and  $W$  stands for width.

The kernel normalizer standardizes each restructured kernel of size  $k_{up} \times k_{up}$  using the softmax function before its application to the input feature map. This process ensures uniformity and prepares the recombination kernel for an effective integration into the feature extraction module.



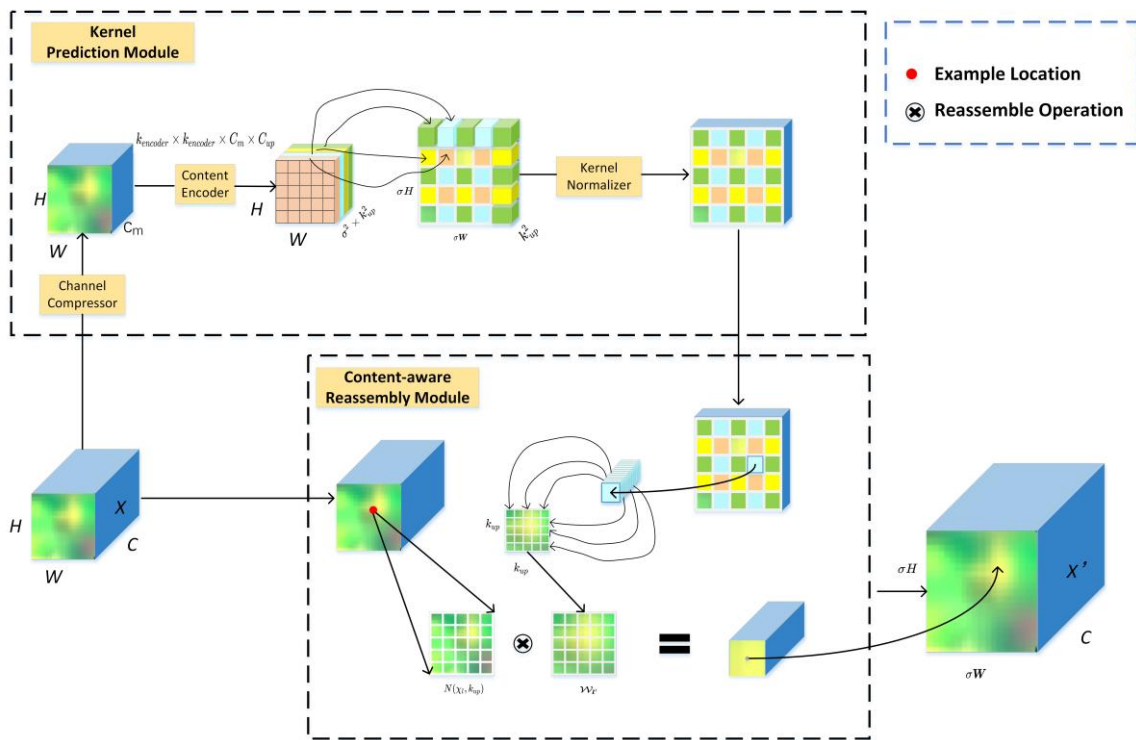


Figure 4. CARAFE structure.

The choice of  $k_{up}$  and  $k_{encoder}$  significantly impacts the detection ability [34]. Increasing the value of  $k_{encoder}$  extends the receptive field. As a result, contextual features over a broader region can be incorporated, which is essential for the recombination kernel computation. A larger  $k_{up}$  increases the computational complexity, but a larger kernel does not necessarily yield a proportionate performance gain. The empirical equation  $k_{encoder} = k_{up} - 2$  strikes a suitable balance between effectiveness and performance. To improve the detection result of micro-targets, this study conducted hyperparameter comparison experiments to select appropriate  $k_{encoder}$  and  $k_{up}$  values, as detailed in Section 4.2.4 Sub-Experiment 2.

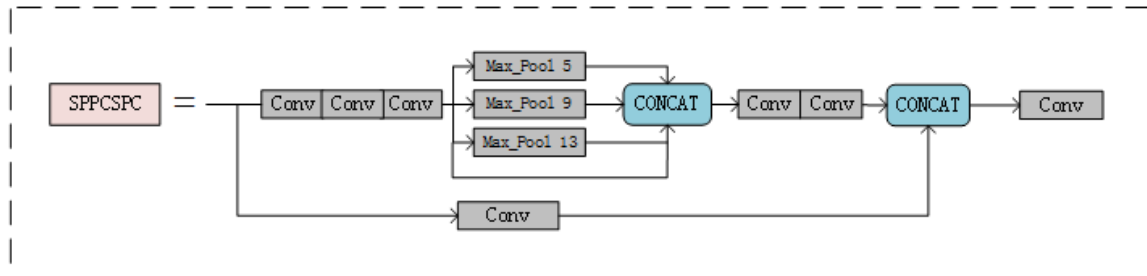
### 3.3. Integration of CSPSPP in Small Object Layer

YOLOv7-tiny incorporates an SPP structure called the CSPSPP module, which is essential for improving the feature extraction performance of different-sized networks. However, due to its location in the large target layer with a resolution of  $20 \times 20$  pixels (as illustrated by CSPSPP in Figure 1), the existing fusion module in YOLOv7-tiny tends to discard information related to micro-objects, leading to limited effectiveness in detecting micro-targets.

Thus, in order to accommodate the characteristics of remote sensing images in which the target sizes vary significantly, the targets are micro, and low-resolution images are prevalent; the network structure was adjusted accordingly in this study. Specifically, a convolutional layer in the feature fusion part of the small target layer was replaced with CSPSPP, as depicted in trick 3 of Figure 3. This CSPSPP layer captures more diverse multiscale features of micro-targets, effectively covering targets with different resolutions and enhancing the overall detection performance.

The SPP structure was first introduced by He in 2015 [41] to address the challenges related to image distortion during processes such as cropping and scaling, and to accelerate the candidate box generation, reduce the computation costs, and eliminate the redundancy in the graph-related feature extraction tasks performed using CNNs. In 2022, an improved SPP (CSPSPP) was applied in YOLOv7. The primary aim of the CSPSPP module is to widen the receptive fields and adjust to different image resolutions. This is achieved

by employing various receptive fields via maximum pooling. In the CSPSPP structure illustrated in Figure 5, the three branches undergo max pooling with kernels of 5, 9, and 13, respectively. These varied max-pooling representations extract object features at different scales, providing three receptive fields for distinguishing between small and large targets.



**Figure 5.** CSPSPP structure.

To assess the effectiveness of the CSPSPP, this study carried out a comparative experiment on a spatial pyramid by adding in different locations and quantities, as detailed in Section 4.2.4 Sub-Experiment 3.

## 4. Experiments

### 4.1. Experiment Setup

#### 4.1.1. Experimental Dataset

This study primarily focused on detecting micro-objects in remote sensing images. The performance of our approach was examined by the images in AI-TOD, which is a high-altitude remote sensing image dataset; its images have sizes of  $800 \times 800$  pixels. In AI-TOD, objects have an average size of approximately 12.8 pixels. Thus, this dataset is well-suited for studies focusing on micro-targets.

The AI-TOD dataset comprises 28,036 aerial images featuring eight distinct object classes: aircraft, bridges, tanks, ships, swimming pools, cars, pedestrians, and windmills. The dataset contains 700,621 instances of these object classes. In order to ensure a comprehensive evaluation, we divided the dataset into training, validation, and testing parts with the ratio of 4:1:5. Figure 6 presents a representative sample of photographs from the dataset. Note that zero padding was applied to some images in the dataset and, therefore, black borders may be present on the right and bottom.



**Figure 6.** Sample images from AI-TOD dataset.

#### 4.1.2. Experiment Environment

A server equipped with an Intel Xeon(R) Silver 4210R CPU and four NVIDIA 3090Ti graphics cards was used for the experiment. The server system was Ubuntu 18.04.4 LTS. A virtual environment was created using Python 3.8.0; Pytorch 1.8.0 and CUDA 11.1 were configured as necessary. Table 1 summarizes the experiment parameters. We maintained the default values for the other hyperparameters. All codes of our model can be accessed at <https://github.com/snufkin-young/micro-targets-detection> (accessed on 23 December 2023).

**Table 1.** Experiment parameter configuration.

Name	Value
epochs	800
batch_size	32
lr0	0.05
lrf	0.1
momentum	0.937
img_size	640

#### 4.1.3. Evaluation Criteria

The mean average precision (mAP) was used to evaluate the accuracy. In addition, frames per second (FPS) were used to gauge the detection speed, which indicates the quantity of processed images in a single second. Parameters (params) were used to evaluate the model complexity. The Giga Floating-Point Operations Per Second (GFLOPs) were used to demonstrate the calculating workload during inference. Finally, the Model Size indicator was used to describe the model dimensions and reveal its complexity.

The *Precision* and *Recall* indexes were computed as Equations (8) and (9), respectively.

$$\text{Precision} = \frac{TP}{TP + FP} \quad (8)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (9)$$

where *TP*, *FP*, and *FN* denote the number of true positives, false positives, and false negatives, respectively.

The Average Precision (AP) index denotes the average detection accuracy across different recall levels and quantifies the precision/recall performance of the network over various confidence thresholds, as shown in Equation (10). The *mAP* is determined by averaging the *AP* values across all categories, as depicted in Equation (11).

$$AP = \int_0^1 P(R) dR \quad (10)$$

$$mAP = \frac{\sum_{i=1}^N AP_i}{N} \quad (11)$$

where *N* is the count of picture classifications, *i* denotes the serial number of detection instances, and *AP<sub>i</sub>* is the average precision for a single classification.

The “mAP0.5” is a metric indicating the mAP at an IoU threshold of 0.5, and “mAP0.5:0.95” represents *mAP* with IoU values between 0.5 and 0.95.

## 4.2. Experiment Results

### 4.2.1. Training Process

The training curves of ten significant indicators used to train our model are described in Figure 7. These curves reveal the developmental changes in the model during training. In this figure, “Box” indicates the mean IoU loss in the training set. A lower box loss

signifies a higher degree of accuracy in the bounding box predictions, which is indicative of the model’s accuracy for target localization. “Objectness” indicates the mean object detection loss on the training set, where a smaller objectness loss signifies more accurate object detection. “Classification” represents the mean classification loss on the training set, where a smaller classification value suggests a more accurate category prediction. Note that “val Box”, “val Objectness”, and “val Classification” represent the means of the various losses on the validation set. The other indicators are introduced in Section 4.1. The horizontal axis of each sub-figure represents the epochs in Figure 7.

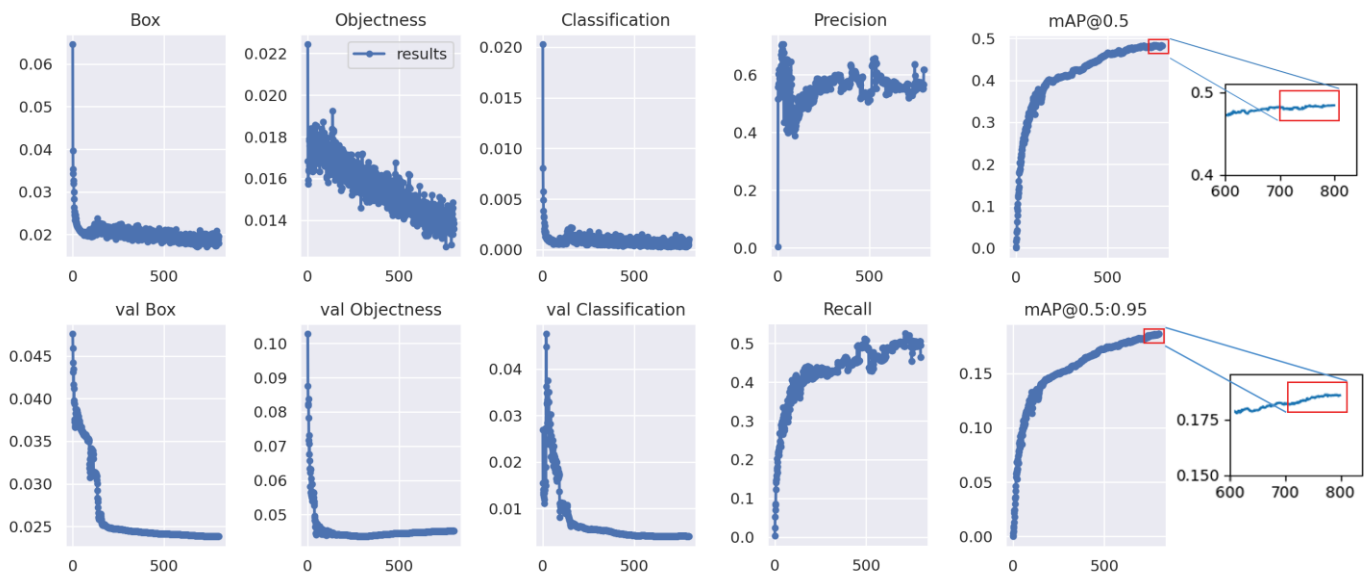


Figure 7. Model training process.

As the training progressed, the loss values continued to decrease gradually. After approximately 500 epochs, various loss types reached relatively low levels. The network model achieved convergence at approximately 700 epochs, with the box, classification, and objectness losses converging normally for both the training and validation sets; beyond 700 epochs,  $mAP_{0.5}$  and  $mAP_{0.5:0.95}$  also stabilized and eventually reached a converged level. Figure 7 demonstrates the convergence of our model during training. This observation indicates that our network is stable and effective, and demonstrates its suitability for the application scenarios.

#### 4.2.2. Ablation Experiment

Several ablation tests verify the effectiveness of each upgraded approach. Table 2 summarizes the results of these experiments; a checkmark (✓) indicates inclusion of the corresponding module.

Table 2. Improvement point ablation experiment. The best results are shown in bold.

Method	Loss	CARAFE	CSPSPP	$mAP_{0.5}/\%$	$mAP_{0.5:0.95}/\%$	FPS	Params/M	GFLOPs
YOLOv7-tiny				42.0	16.8	<b>161</b>	<b>6.02</b>	<b>13.1</b>
Method 1	✓			46.7	17.9	156	6.03	13.1
Method 2		✓		44.5	17.5	156	6.04	13.3
Method 3			✓	45.5	17.9	149	6.47	18.7
Method 4	✓	✓		47.5	18.1	147	6.05	13.3
Method 5		✓	✓	46.4	18.2	142	6.49	18.8
Method 6	✓	✓	✓	<b>48.7</b>	<b>18.9</b>	139	6.49	18.8

Methods 1–3 in Table 2 correspond to inclusion of the designed improvements to the loss function, CARAFE, and CSPSPP modules, respectively. The results in the table indicate that each of these three individual enhancement methods contributed to improved detection performance, with the most significant improvement being observed for the loss function modification.

Loss Function Improvement (Method 1): The largest increase in detection performance was achieved for the following method:  $mAP@0.5$  increased by 4.7%.

CARAFE Module Improvement (Method 2): This approach also enhanced the detection performance, as  $mAP@0.5$  increased by 2.5%.

CSPSPP Module Improvement (Method 3): The method yielded a notable improvement in detection performance, as  $mAP@0.5$  increased by 3.5%.

Method 4 combined the improvements from the loss function and CARAFE module. Compared to Method 1, this approach yielded 0.8% and 0.2% increases, respectively, in  $mAP@0.5$  and  $mAP@0.5:0.95$ .

Method 5 combined the improvements from the CARAFE module and CSPSPP module. Compared to Method 2, this approach yielded 1.9% and 0.7% increases, respectively, in  $mAP@0.5$  and  $mAP@0.5:0.95$ .

Method 6, which corresponded to the final proposed method, including all three improvements, showed further advancements compared to Method 4 and Method 5.

The experiment results affirm that each of the three proposed enhancements can independently or collaboratively enhance the algorithm's effectiveness for remote sensing object detection in aerial imagery, even notwithstanding a slight expansion in model size and computational load. These findings underscore the versatility and efficacy of individual and combined improvements.

#### 4.2.3. Comparative Experiment

##### Training Curve Comparison

In this study, both baseline and progressively enhanced algorithms were trained under consistent experimental conditions. Figure 8 presents training curves to facilitate comparison of YOLOv7-tiny with the improved algorithm for two key indicators:  $mAP@0.5$  and  $mAP@0.5:0.95$ . Specifically, both indicators demonstrated consistent increases with the progression of epochs, reaching a plateau around 700 epochs.

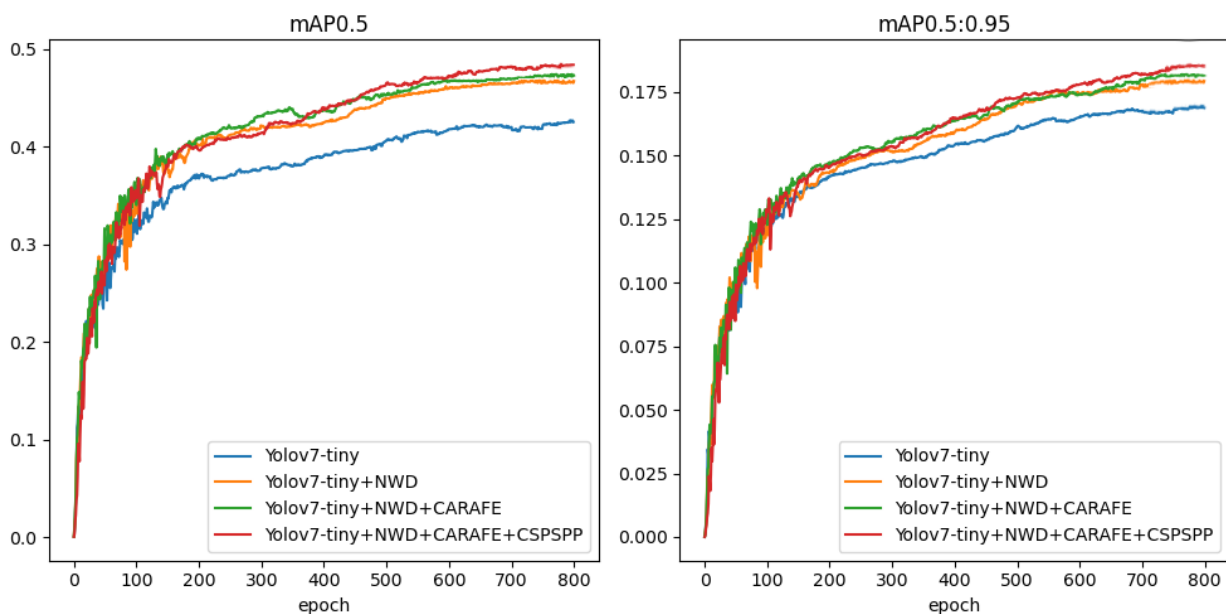
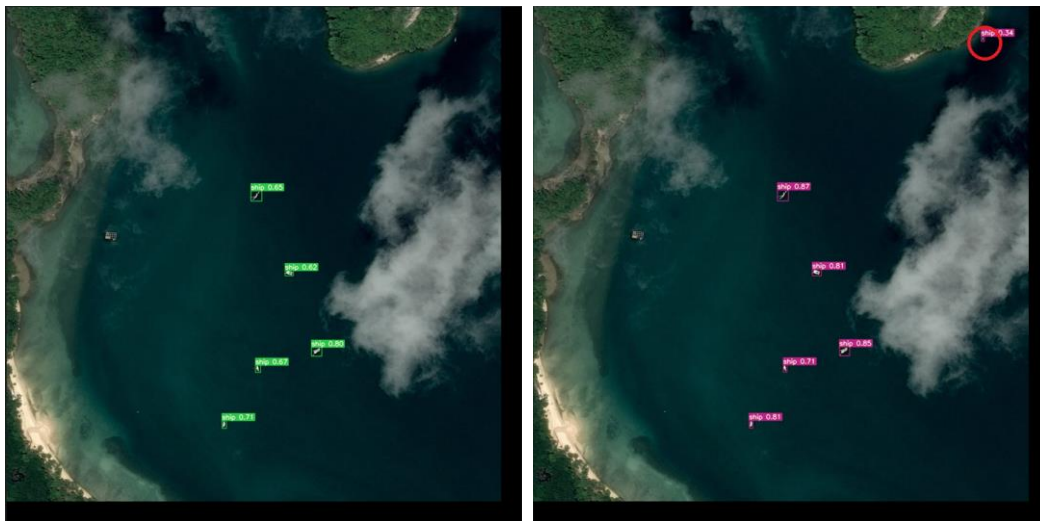


Figure 8. Comparison of training process curves for baseline and enhanced method.

In addition, gradual improvement was observed when the enhanced modules were applied. In detail, the algorithm incorporating incremental improvements exhibited gradient-like behavior after convergence. This suggests that each enhancement contributes to improved detection performance and that these enhancements are mutually compatible.

### Visual Comparison of Detection Results

To evaluate the performance of our algorithm, a contrast analysis between the results of YOLOv7-tiny and our method are provided below. This experiment spanned the detection of micro- and dense objects, as well as micro-object detection on complex backgrounds. In Figures 9–11, parts (a) and (b) depict the results for YOLOv7-tiny and our algorithm, respectively. The key differences are highlighted by red circles in (b).



(a) YOLOv7-tiny

(b) Proposed

Figure 9. Comparison of YOLOv7-tiny and the proposed method for micro-target detection.



(a) YOLOv7-tiny

(b) Proposed

Figure 10. Comparison of YOLOv7-tiny and the proposed method for dense-target detection.



(a) YOLOv7-tiny

(b) Proposed

**Figure 11.** Comparison of YOLOv7-tiny and the proposed method for micro-target detection against complex backgrounds.

- Scenario 1: Micro-Target Detection (Figure 9)

For the micro-object detection scenario, YOLOv7-tiny demonstrated good performance but tended to miss micro-targets. In detail, YOLOv7-tiny detected 6 targets; however, the improved algorithm detected 11 targets with higher confidence scores.

- Scenario 2: Dense-Target Detection (Figure 10)

For dense micro-target detection, YOLOv7-tiny made significant omissions, including both small and micro-targets. However, the improved algorithm is better at detecting micro- and other-scale storage tanks in dense scenarios.

- Scenario 3: Micro-Target Detection Against Complex Backgrounds (Figure 11)

In scenarios with complex backgrounds, YOLOv7-tiny missed micro-targets with less prominent features. In contrast, our algorithm successfully detected many targets that were missed by YOLOv7-tiny in these complex background scenarios.

The results indicate an obvious detection performance enhancement for our method, particularly in scenarios featuring micro- and dense objects and complex backgrounds.

#### Comparison with Other Models

To confirm the enhanced algorithm performance for remote sensing target detection, we performed comparative experiments using the AI-TOD dataset. The enhanced model was compared to mainstream algorithms. The comparative results are shown in Table 3.

The results reveal that our algorithm achieved  $mAP_{0.5}$ ,  $mAP_{0.5:0.95}$ , and  $FPS$  metrics of 48.7%, 18.9%, and 139, respectively. Compared with the YOLOv7-tiny algorithm, improvements in both  $mAP_{0.5}$  and  $mAP_{0.5:0.95}$  were obtained for a limited delay in FPS. Compared to the other models, the proposed model demonstrated enhanced detection accuracy (achieving the second highest  $mAP_{0.5}$  and  $mAP_{0.5:0.95}$ ) coupled with a notable advantage in detection speed (the third-highest FPS and much higher than other models).

**Table 3.** Comparative experiment results for different networks. The best results are shown in bold.

Method	mAP0.5/%	mAP0.5:0.95/%	Model Size/MB	FPS
SSD-512 [6]	21.7	7.0	-	
RetinaNet [42]	13.6	4.7	-	
CenterNet [7]	39.2	13.4	-	
Faster R-cnn [5]	26.3	11.1	236.33	16
ATSS [43]	30.6	12.8	244.56	13
Cascade R-CNN [44]	30.8	13.8	319.45	11
JSDNet [29]	<b>52.5</b>	<b>21.4</b>	88	62
YOLOv5s [13]	42.2	18.6	29.97	102
PDWT-YOLO [25]	45.6	18.2	12.9	141
YOLOv7-tiny	42.0	16.8	<b>12.3</b>	<b>161</b>
Proposed method	48.7	18.9	13.3	139

#### 4.2.4. Hyperparameter Comparative Experiment

Three sub-experiments were performed to investigate the optimal hyperparameters for the loss function, CARAFE upsampling operator, and CSPSPP improvement positions and, thus, to enhance the micro-target detection performance.

- Sub-Experiment 1: Loss function hyperparameter comparison experiment

To determine the optimal ratio between NWD and CIOU, we conducted experiments with several typical coefficients as listed in Table 4.

**Table 4.** Comparison of different CIOU/NWD ratios. The best results are shown in bold.

CIOU	NWD	mAP0.5/%	mAP0.5:0.95/%
1	0	42.0	16.8
0.75	0.25	44.9	17.7
0.5	0.5	46.1	17.8
0.25	0.75	<b>46.7</b>	<b>17.9</b>
0	1	45.6	16.6

The experimental findings showed that the model's performance was improved by combining CIOU and NWD. From Table 4, the best values were obtained for a CIOU/NWD ratio of 0.25:0.75. Therefore, we ultimately choose CIOU and NWD coefficients of 0.25 and 0.75, respectively, to improve the loss function.

- Sub-Experiment 2: CARAFE hyperparameter comparison experiment

Experiments were conducted by incorporating CARAFE into the YOLOv7-tiny network and by comparing the following three settings:  $k_{encoder} = 1, k_{up} = 3$ ;  $k_{encoder} = 3, k_{up} = 5$ ; and  $k_{encoder} = 5, k_{up} = 7$ . Table 5 lists the results.

**Table 5.** Comparison of CARAFE hyperparameter experiment results. The best results are shown in bold.

$k_{encoder}$	$k_{up}$	mAP0.5/%	mAP0.5:0.95/%	GFLOPs
1	3	44.3	<b>17.5</b>	<b>13.1</b>
3	5	<b>44.5</b>	17.4	13.3
5	7	42.8	17.5	14.6

When  $k_{encoder} = 1$  and  $k_{up} = 3$ , the mAP0.5:0.95 and GFLOP scores were improved. However, when  $k_{encoder} = 3$  and  $k_{up} = 5$ , the mAP0.5 score improved. Considering the trade-off between mAP0.5, mAP0.5:0.95, and GFLOPs, we selected  $k_{encoder} = 1$  and  $k_{up} = 3$  as the hyperparameters for the proposed algorithm. This set of hyperparameters exhibited superior performance in various ways while maintaining computational efficiency.



- Sub-Experiment 3: CSPSPP position comparative experiment

In this part, we performed a comparative experiment by three different positions where CSPSPP can be replaced according to our improvement scheme based on CSPSPP. These positions corresponded to CBL feature fusion layers at resolutions of  $40 \times 40$  pixels (CBL\_1 shown in Figure 1) and  $80 \times 80$  pixels (CBL\_2 shown in Figure 1) and the simultaneous replacement of CBL layers at resolutions of  $40 \times 40$  and  $80 \times 80$  pixels (CBL\_1 and CBL\_2 shown in Figure 1). For clarity, these positions are denoted as Positions 1–3, respectively. Table 6 lists the experiment results.

**Table 6.** Comparison of results for various CSPSPP positions. The best results are shown in bold.

Position	mAP0.5/%	mAP0.5:0.95/%	GFLOPs
1	45.3	17.7	18.7
2	<b>45.5</b>	<b>17.9</b>	<b>18.7</b>
3	43.6	17.4	24.3

The results indicate that Position 2, that is, replacement of the CBL feature-fusion layer at  $80 \times 80$  resolution with an SPP structure, achieves the optimal detection-performance improvement.

#### 4.2.5. Generalization Comparative Experiment

The Satellite Imagery Multi-Vehicles Dataset (SIMD) [45] was used to evaluate our method. The SIMD dataset was randomly split into training and validation sets with 4000 and 1000 photographs, respectively. To accommodate the significant scale variations among the target objects in this dataset and to comprehensively evaluate the improvement in multi-scale target detection, three additional metrics— $AP_S$ ,  $AP_M$ , and  $AP_L$ —were proposed to measure the  $mAP$  for small, medium, and large targets, respectively. We also compared our method with YOLOv7-tiny. Table 7 illustrates that the proposed algorithm outperformed the YOLOv7-tiny algorithm, with improvements of 2.8%, 0.8%, and 1.9% in the  $AP_S$ ,  $AP_M$ , and  $AP_L$  scores, respectively.  $AP_S$  exhibited the most substantial improvement, demonstrating a clear enhancement in the small-target detection performance. Furthermore, the improvements in the  $AP_M$  and  $AP_L$  scores indicate a positive impact on medium- and large-target detection accuracy. This evidence underscores the significant enhancement in the small-target detection performance of our proposed algorithm and confirms that these accuracy improvements are maintained for medium and large targets.

**Table 7.** Comparison of YOLOv7-tiny and proposed algorithm results based on SIMD dataset.

Method	mAP0.5/%	mAP0.5:0.95/%	$AP_S$ /%	$AP_M$ /%	$AP_L$ /%
YOLOv7-tiny	80.2	63.3	9.8	54.8	68.2
Proposed	81.7	64.1	12.6	55.6	70.1

#### 4.3. Analysis of Experiment Results

All experimental outcomes underscored the validity of the proposed algorithm in the detection of remote sensing photos, particularly for micro-scale targets. The enhancements included a refined loss function, CARAFE for feature upsampling, and an integrated CSPSPP module, which collectively contributed to the superior detection performance.

The ablation studies conducted on the various proposed enhancements emphasized the remarkable and undeniable positive impacts of each of these improvements. These studies meticulously analyzed the performance of the original system before the implementation of each enhancement and compared it with the performance after the enhancement was introduced. The results were remarkable, as each proposed enhancement contributed significantly to the overall improvement of the system's efficiency, stability, and functionality.

Additionally, our proposed model demonstrated superior detection accuracy compared to mainstream algorithms while maintaining competitive computational efficiency. The experiments conducted on the AI-TOD dataset, designed for micro-target detection in high-altitude remote sensing images, had promising results. Finally, the generalizability and robustness of the algorithm were further confirmed through experiments using the SIMD dataset, which revealed consistent performance gains across various target scales.

## 5. Discussion

The complexity of backgrounds, the abundance of micro-objects, and the insufficient pixel information in remote sensing images pose greater challenges for target detection. Therefore, we proposed a series of improvement strategies based on the YOLOv7-tiny algorithm to address specific difficulties associated with detecting micro-targets in remote sensing.

From the perspective of detection accuracy, the proposed algorithm exhibits significant advantages in the detection of micro-targets, second only to JSDNet, as shown in Table 3, compared to traditional existing methods. Firstly, it achieves a remarkable mAP0.5 of 48.7%, surpassing all other methods, including CenterNet, YOLOv5s, and YOLOv7-tiny, showcasing its robustness in accurately detecting objects of varying sizes. Meanwhile, the elevated mAP0.5:0.95 at 18.9% further reflects its proficiency in precisely locating detected objects. Particularly when compared to our team's PDWT-YOLO algorithm tailored for small objects and considering a similar model size, the proposed algorithm in this paper demonstrates commendable performance in detecting micro-targets.

From the perspective of detection speed, the detection performance of JSDNet marginally surpasses the method proposed in this study; however, its model size is more than six times larger due to the utilization of the Transformer architecture, which makes its detection speed slower than our method. Hence, our method's strength lies in its adept balance between detection accuracy and model size. Despite achieving top-tier mAP0.5, the proposed algorithm maintains a relatively compact model size of 13.3 MB. This equilibrium positions it as a superior choice, particularly in scenarios with constrained computational resources, such as real-time systems. Hence, our algorithm is a convincing choice when considering various factors between detection accuracy, speed, and model size. It is adaptable to real-world scenarios, particularly those with micro- and dense objects.

From the perspective of future research directions, while proficient in detecting micro- and dense objects, its performance might be susceptible to challenging contextual factors like varying illumination. Additionally, the implemented enhancements increase certain algorithmic complexities, even if our algorithm's detection speed remains suitable for real-time systems. These aspects should be explored in future research to refine and extend the algorithm's applicability.

In summary, while our algorithm exhibits significant strengths, future research directions could explore a further refinement of the algorithm to enhance its applicability across a broader spectrum of scenarios.

## 6. Conclusions

This research proposes an enhanced network framework for detecting micro-targets to tackle the challenges associated with complex backgrounds, high micro-target densities, and insufficient features in remote sensing images. The proposed work enhances the performance of micro-target detection performance through loss function refinement, an upsampling module upgrade, and the incorporation of CSPSPP.

The first improvement involves replacing the IOU loss function with a combination of NWD and CIOU to enhance the algorithm's sensitivity towards micro-targets of varying scales. Next, we incorporated the CARAFE upsampling operator, which is a lightweight and efficient alternative, in place of the original operator. This modification significantly improves the performance of multiscale feature extraction for micro-targets. Lastly, we

integrated a spatial pyramid structure into the network, thereby enhancing its suitability for detecting low-pixel micro-targets.

The experimental results demonstrate the efficacy of our model in real-time scenarios, exhibiting a significant improvement in accuracy for detecting micro-targets. By effectively addressing the challenges posed by complex backgrounds and micro-targets, our approach contributes to the advancement of detection accuracy and speed for remote sensing targets. The algorithm presents a compelling solution for applications requiring a fast identification of micro-targets in high-altitude imagery. With regard to future directions, the coming research could explore further optimizations for diverse environmental conditions and the integration of additional modalities to enhance the algorithm's robustness and applicability.

**Author Contributions:** Conceptualization, L.Z. and N.X.; methodology, L.Z.; software, W.G.; validation, W.G.; resources; data curation, L.Z. and N.X.; writing—original draft preparation, L.Z. and W.G.; writing—review and editing, N.X. and P.W.; visualization, P.W. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the general project of Key R & D Plan of Shanxi Province, high-technology field (grant number 201903D121171) and the National Natural Science Foundation of China (serial number 61976134).

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The datasets presented in this study are available at <https://github.com/jwwangchn/AI-TOD> and <https://github.com/ihians/simd> (accessed on 20 June 2023).

**Acknowledgments:** We would like to express our gratitude to Xiaodong Yue (Shanghai University) for providing computational resources and support.

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

## Abbreviations

The following abbreviations are used in this manuscript:

AI-TOD	Tiny Object Detection in Aerial Images Dataset
CARAFE	Content-Aware Reassembly of Features
CIoU	Complete Intersection Over Union
CNNs	Convolutional Neural Networks
CSPSP	Contextual Spatial Pyramid Spatial Pyramid Pooling
DETR	Detection Transformer
DL	Deep Learning
DIoU	Distance Intersection Over Union
EIoU	Enhanced Intersection Over Union
ELAN	Efficient Long-Range Aggregation Network
Fast R-CNN	Fast Region-Based Convolutional Network
FPN	Feature Pyramid Network
CBL	Convolutional Block Layer
GFLOPs	Giga Floating-Point Operations Per Second
GIoU	Generalized Intersection Over Union
GWD	Gaussian Wasserstein Distance
IOU	Intersection Over Union
MPCConv	Compact Convolution Module
NMS	Non-Maximum Suppression
NWD	Normalized Wasserstein Distance
PAFPN	Path Aggregation Feature Pyramid Network
PANet	Path Aggregation Network
R-CNN	Region with CNN Features
SAM	Segment Anything Model

SIMD	Satellite Imagery Multi-Vehicles Dataset
SPP	Spatial Pyramid Structure
SSD	Single Shot Multibox Detector
YOLO	You Only Look Once

## References

1. Tong, K.; Wu, Y.; Zhou, F. Recent Advances in Small Object Detection Based on Deep Learning: A Review. *Image Vis. Comput.* **2020**, *97*, 103910. [[CrossRef](#)]
2. Ahmed, M.; Hashmi, K.A.; Pagani, A.; Liwicki, M.; Stricker, D.; Afzal, M.Z. Survey and performance analysis of deep learning based object detection in challenging environments. *Sensors* **2021**, *21*, 5116. [[CrossRef](#)]
3. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587. [[CrossRef](#)]
4. Girshick, R. Fast R-CNN. In Proceedings of the 2015 IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1440–1448. [[CrossRef](#)]
5. 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](#)]
6. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. SSD: Single shot multibox detector. In Proceedings of the 14th European Conference on Computer Vision 2016, Amsterdam, The Netherlands, 11–14 October 2016; Volume 9905, pp. 21–37. [[CrossRef](#)]
7. Zhou, X.; Wang, D.; Krähenbühl, P. Objects as points. *arXiv* **2019**, arXiv:1904.07850. [[CrossRef](#)]
8. Redmon, J.; Divvala, S.K.; Girshick, R.B.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788. [[CrossRef](#)]
9. 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. [[CrossRef](#)]
10. Redmon, J.; Farhadi, A. YOLOv3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767. [[CrossRef](#)]
11. Bochkovskiy, A.; Wang, C.Y.; Liao, H.Y.M. YOLOv4: Optimal speed and accuracy of object detection. *arXiv* **2020**, arXiv:2004.10934. [[CrossRef](#)]
12. Zhu, X.; Lyu, S.; Wang, X.; Zhao, Q. TPH-YOLOv5: Improved YOLOv5 based on transformer prediction head for object detection on drone-captured scenarios. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops, Montreal, BC, Canada, 11–17 October 2021; pp. 2778–2788. [[CrossRef](#)]
13. Ultralytics: Yolov5. [EB/OL]. Available online: <https://github.com/ultralytics/yolov5> (accessed on 1 November 2021).
14. Chen, Z.; Zhang, F.; Liu, H.; Wang, L.; Zhang, Q.; Guo, L. Real-time detection algorithm of helmet and reflective vest based on improved YOLOv5. *J. Real-Time Image Process.* **2023**, *20*, 4. [[CrossRef](#)]
15. Wu, D.; Jiang, S.; Zhao, E.; Liu, Y.; Zhu, H.; Wang, W.; Wang, R. Detection of *Camellia oleifera* fruit in complex scenes by using YOLOv7 and data augmentation. *Appl. Sci.* **2022**, *12*, 11318. [[CrossRef](#)]
16. Jiang, K.; Xie, T.; Yan, R.; Yan, R.; Wen, X.; Li, D.; Jiang, H.; Jiang, N.; Feng, L.; Duan, X.; et al. An attention mechanism-improved YOLOv7 object detection algorithm for hemp duck count estimation. *Agriculture* **2022**, *12*, 1659. [[CrossRef](#)]
17. Li, B.; Chen, Y.; Xu, H.; Fei, Z. Fast vehicle detection algorithm on lightweight YOLOv7-tiny. *arXiv* **2023**, arXiv:2304.06002. [[CrossRef](#)]
18. Kulyukin, V.A.; Kulyukin, A.V. Accuracy vs. energy: An assessment of bee object inference in videos from on-hive video loggers with YOLOv3, YOLOv4-Tiny, and YOLOv7-Tiny. *Sensors* **2023**, *23*, 6791. [[CrossRef](#)]
19. Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N. End-to-end object detection with transformers. In *European conference on computer vision*; Glasgow, UK, 23–28 August 2020, Springer International Publishing: Cham, Switzerland, 2020; pp. 213–229. [[CrossRef](#)]
20. Kirillov, A.; Mintun, E.; Mintun, E. Segment anything. *arXiv preprint* **2023**, arXiv:2304.02643. [[CrossRef](#)]
21. Huo, B.; Li, C.; Zhang, J.; Xue, Y.; Lin, J. SAFF-SSD: Self-attention combined feature fusion-based SSD for small object detection in remote sensing. *Remote Sens.* **2023**, *15*, 3027. [[CrossRef](#)]
22. Betti, A.; Tucci, M. YOLO-S: A lightweight and accurate YOLO-like network for small target detection in aerial imagery. *Sensors* **2023**, *23*, 1865. [[CrossRef](#)]
23. Lai, H.; Chen, L.; Liu, W.; Yan, Z.; Ye, S. STC-YOLO: S mall object detection network for traffic signs in complex environments. *Sensors* **2023**, *23*, 5307. [[CrossRef](#)]
24. Qu, J.; Tang, Z.; Zhang, L.; Zhang, Y.; Zhang, Z. Remote sensing small object detection network based on attention mechanism and multi-scale feature fusion. *Remote Sens.* **2023**, *15*, 2728. [[CrossRef](#)]
25. Zhang, L.; Xiong, N.; Pan, X.; Yue, X.; Wu, P.; Guo, C. Improved Object Detection Method Utilizing YOLOv7-Tiny for Unmanned Aerial Vehicle Photographic Imagery. *Algorithms* **2023**, *16*, 520. [[CrossRef](#)]
26. Wang, J.; Yang, W.; Guo, H.; Zhang, R.; Xia, G.S. Tiny object detection in aerial images. In Proceedings of the 25th International Conference on Pattern Recognition, Milan, Italy, 10–15 January 2021; pp. 3791–3798. [[CrossRef](#)]

27. Chen, X.; Fang, H.; Lin, T.Y. Microsoft coco captions: Data collection and evaluation server. *arXiv preprint* **2015**, arXiv:1504.00325. [[CrossRef](#)]
28. Lu, G.; He, X.; Wang, Q.; Shao, F.; Wang, J.; Hao, L. MStrans: Multiscale Vision Transformer for Aerial Objects Detection. *IEEE Access* **2022**, *10*, 75971–75985. [[CrossRef](#)]
29. Ni, S.; Lin, C.; Wang, H.; Li, Y.; Liao, Y.; Li, N. Learning geometric Jensen-Shannon divergence for tiny object detection in remote sensing images. *Front. Neurobot.* **2023**, *17*, 1273251. [[CrossRef](#)]
30. Wang, J.; Xu, C.; Yang, W.; Yu, L. A normalized Gaussian Wasserstein distance for tiny object detection. *arXiv* **2021**, arXiv:2110.13389. [[CrossRef](#)]
31. Wang, C.Y.; Bochkovskiy, A.; Liao, H.Y.M. 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, Vancouver, BC, Canada, 17–24 June 2023; pp. 7464–7475. [[CrossRef](#)]
32. Lin, T.Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2117–2125. [[CrossRef](#)]
33. Liu, S.; Qi, L.; Qin, H.; Shi, J.; Jia, J. Path aggregation network for instance segmentation. In Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 8759–8768. [[CrossRef](#)]
34. Wang, C.Y.; Yeh, I.; Liao, H.Y.M. You only learn one representation: Unified network for multiple tasks. *arXiv* **2021**, arXiv:2105.04206. [[CrossRef](#)]
35. Rezatofighi, H.; Tsoi, N.; Gwak, J.Y.; Sadeghian, A.; Reid, I.; Savarese, S. Generalized intersection over union: A metric and a loss for bounding box regression. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 658–666. [[CrossRef](#)]
36. Zheng, Z.; Wang, P.; Liu, W.; Li, J.; Ye, R.; Ren, D. Distance-IoU loss: Faster and better learning for bounding box regression. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; Volume 34, pp. 12993–13000. [[CrossRef](#)]
37. Zhang, Y.F.; Ren, W.; Zhang, Z.; Jia, Z.; Wang, L.; Tan, T. Focal and efficient IOU loss for accurate bounding box regression. *Neurocomputing* **2022**, *506*, 146–157. [[CrossRef](#)]
38. Yang, X.; Zhang, G.; Yang, X.; Zhou, Y.; Wang, W.; Tang, J.; He, T.; Yan, J. Detecting rotated objects as Gaussian distributions and its 3-D generalization. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *45*, 4335–4354. [[CrossRef](#)] [[PubMed](#)]
39. Noh, H.; Hong, S.; Han, B. Learning deconvolution network for semantic segmentation. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1520–1528. [[CrossRef](#)]
40. Wang, J.; Chen, K.; Xu, R.; Liu, Z.; Loy, C.C.; Lin, D. CARAFE: Content-aware reassembly of features. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 3007–3016. [[CrossRef](#)]
41. He, K.; Zhang, X.; Ren, S.; Sun, J. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 1904–1916. [[CrossRef](#)] [[PubMed](#)]
42. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal loss for dense object detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *42*, 318–327. [[CrossRef](#)] [[PubMed](#)]
43. Zhang, S.; Chi, C.; Yao, Y.; Lei, Z.; Li, S.Z. Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 9759–9768. [[CrossRef](#)]
44. Cai, Z.; Vasconcelos, N. Cascade R-CNN: Delving into high quality object detection. In Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 13–23 June 2018; pp. 6154–6162. [[CrossRef](#)]
45. Haroon, M.; Shahzad, M.; Fraz, M.M. Multisized object detection using spaceborne optical imagery. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2020**, *13*, 3032–3046. [[CrossRef](#)]

**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.