An Optimized YOLOv5s-rd Framework for Efficient Target Detection in Remote Sensing Images

Hongmei Tang [1,*], Yu Han [2,*], Jinliang Zheng [2], Ziyu Wang [2], Lei Wang [2]
[1]Yellow River Conservancy Technical Institute, Kaifeng 475004, Henan, China
[2]Anhui University, Hefei 230031, Anhui, China
E-mail: tanghongmei126@hotmail.com, hanyu_vip@outlook.com
*Corresponding Author

Keywords: YOLOv5s-rd, network remote sensing images, target detection, improvement

Received: December 28, 2024

Remote sensing image object detection methods are crucial for applications related to aircraft, ships, vehicles, and buildings. Traditional methods, relying on manually designed features, suffer from high computational complexity, leading to low detection efficiency and stability. In response, we present an enhanced remote sensing image object detection approach, YOLOv5s - rd, which is built upon an optimized YOLOv5s.Our method integrates structural enhancements, refined loss functions, and advanced data augmentation strategies. These improvements include optimizing the number and orientation of rotation anchors to better handle target scale diversity and rotation changes in remote sensing images. We also adjust Gaussian distribution parameters, which is beneficial for dealing with the challenges of complex backgrounds. Additionally, we calibrate the weights of the weak - supervision branch, considering the fact that the number of objects in remote sensing images is usually small, aiming to improve the model's performance with limited labeled data. We conduct experiments on multiple public datasets: DOTA, HRSC2016, UCAS - AOD, and Northwest University VHR - 10. The results demonstrate that YOLOv5s - rd outperforms traditional and existing deep - learning methods in detection performance. Specifically, on the DOTA dataset, it achieves a mean average precision (mAP) of 80.4% and 41.2 FPS; on the UCAS - AOD dataset, 96.7% mAP and 40.3 FPS; on the HRSC2016 dataset, 93.2% mAP and 38.7 FPS; and on the Northwest University VHR - 10 dataset, 95.2% mAP and 39.7 FPS. Moreover, its computational complexity (FLOPs) is only 11.0B, surpassing most of the compared methods. By combining these novel optimizations, our YOLOv5s - rd not only enhances the robustness and effectiveness of the detection model but also significantly improves performance and reliability compared to existing methods, providing a new solution for remote sensing image object detection.

Povzetek: Prispevek predstavi izboljšano metodo detekcije ciljev na daljinskih posnetkih z modelom YOLOv5s-rd, ki z rotacijskimi sidri, Gaussovo porazdelitvijo in šibko nadzorovanimi vejami dosega visoko kvaliteto.

1 Introduction

Remote sensing image target detection is a crucial process in the field of remote sensing technology that can automatically or semiautomatically identify and locate all kinds of targets of interest in remote sensing images, such as airplanes, ships, vehicles, and buildings. This technique has great theoretical significance and practical value, and has been widely used in many fields, including but not limited to military reconnaissance, urban planning, traffic management, and disaster relief. Although remote sensing image target detection has many advantages and practical uses, there are still some challenges and difficulties in the process. The first is the problem of target scale diversity. Since the target sizes in remote sensing images vary greatly, ranging from a few pixels to hundreds of pixels, designing feature extraction and detection strategies that adapt to targets of different sizes is highly challenging. Second, since the acquisition angle and direction of remote sensing images are random, the orientation of the target itself is also variable [1, 2].

To solve this problem, we need to design rotation invariant, or rotation sensitive, detection methods to improve the accuracy and robustness of the detections [3, 4]. Finally, the backgrounds of remote sensing images are usually very complex and contain a wide variety of features and textures, such as mountains, water, forests, and clouds. Therefore, we also need to design effective background suppression and target highlighting methods to reduce the interference from the background. In summary, this paper provides a novel YOLOv5s-rd-based target detection method for networked remote sensing images, which effectively addresses the challenges and difficulties in remote sensing image target detection, and is experimentally validated on several publicly available datasets to demonstrate its superior detection performance and efficiency [5, 6].

The current research focuses on developing a remote sensing image target detection method based on an optimized version of YOLOv5s, namely YOLOv5s-rd, to improve detection accuracy and efficiency. The research question is how to design an efficient and accurate detection method to overcome difficulties and achieve high-precision and high-speed detection when remote sensing image detection faces challenges such as diverse target scales, changing directions, insufficient target features, and complex backgrounds. It is assumed that by introducing strategies such as rotating anchor points and optimizing Gaussian distribution, the detection performance can be significantly improved, and the results achieved on public datasets are better than those of existing methods.

2 Related work

In the field of object detection, relevant work has achieved fruitful results. Faster R-CNN uses RPN to generate candidate regions, greatly improving the detection speed and achieving end-to-end object detection; the YOLO series uses an innovative grid division strategy to transform object detection into a regression problem, greatly speeding up the detection process. The MinSummary algorithm focuses on feature compression and information extraction, which can effectively reduce the amount of model calculation while retaining key features. To further break through the performance bottleneck, we deeply explore the model enhancement function and conduct a comprehensive analysis of our proposed object detection method from the aspects of ablation research, computational efficiency, visualization, and deployment feasibility.

2.1 Traditional remote sensing image target detection methods

The two manual feature-based target detection methods, HOG+SVM and DPM, which use gradient direction histograms and a deformable part model to describe the appearance and structure of the target, respectively, have certain advantages, but in remote sensing image target detection, owing to their diversity and complexity, their expression ability and computational efficiency are not sufficiently high, resulting in low detection efficiency and stability [6, 7].

2.2 Deep learning-based target detection methods for remote sensing images

R-CNN methods can take advantage of the feature extraction capability of CNNs to improve detection accuracies, however, the need for forward propagation of the CNNs to each candidate region results in less efficient detections. The fast R-CNN method can utilize the shared computations of CNNs to improve the detection efficiency, however, the detection complexity is high because of the need to use the RPN to generate candidate regions. The Faster R-CNN method can utilize the end-to-end training of CNNs to improve the detection stability, however, the detection accuracy is low because of the need to use the ROI pooling layer for feature mapping. The YOLO method can utilize the CNN's parallel computation to improve the speed of detection, however, this is due to the need to perform target detection for each grid, resulting in lower detection recalls.

In the field of object detection, comparing the performance of different methods is crucial to evaluate the effectiveness of innovative methods. The table summarizes the key indicators of several current representative methods, including mean average precision (mAP), frames per second (FPS), and computational complexity (FLOP), so as to compare with your proposed method.For example, Faster R-CNN, as a classic method, usually has high detection accuracy, but its high computational complexity leads to low FPS (about 10 FPS), which is suitable for tasks with high processing accuracy requirements. YOLOv3 achieves a good balance between detection speed and accuracy, with a mAP of about 57.9%, an FPS of about 30, and a FLOP in the medium range. YOLOv5s further improves the detection speed, with an FPS of up to 40, and its mAP value of about 48.4%, which is suitable for real-time processing tasks. However, despite their significant advantages, these methods still have some limitations. For example, YOLOv3 and YOLOv5s may not perform well in the detection of small objects, and Faster R-CNN is slow in processing on large-scale datasets. Therefore, the method you proposed effectively improves the detection accuracy of small objects by improving the network structure and optimizing the training process, and improves the FPS while ensuring accuracy and shortening the calculation time. The specific comparison results are shown in Table 1.

Table 1: Comparison results

Method	mAP	FPS	FLOP	Remarks
Faster R-CNN	58.6%	10	High	High accuracy, but slower processing speed
YOLOv3	57.9%	30	Medium	Good balance between accuracy and speed
YOLOv5s	48.4%	40	Medium	Faster speed, but relatively lower accuracy
Proposed Method	61.2%	35	Low	Higher accuracy, faster processing speed

In recent advancements in target detection, several studies have utilized advanced imaging techniques for accurate classification. Gómez et al. employed Isomap with SMACOF for hyperspectral image classification, enhancing the ability to detect and classify targets in complex image datasets [7]. Similarly, López et al. utilized multi-spectral imaging to identify weeds in herbicide testing, demonstrating the application of multi-spectral techniques for environmental monitoring and target detection in agricultural settings [8]. Furthermore, Papp and Szucs proposed a double probability model for the open set problem in image classification, addressing the challenges of detecting unknown or unclassified targets in image datasets [9]. These studies illustrate the diverse approaches in target detection, from hyperspectral and multi-spectral imaging to advanced statistical modeling.

3 Improved target detection method for remote sensing images on the YOLOv5s-rd network

YOLOv5s-rd has achieved significant performance improvements in remote sensing image target detection. By using CSPNet as the backbone network, it not only reduces computational complexity and memory consumption, but also improves the model's feature expression capabilities. In addition, YOLOv5s-rd uses CIoU in the loss function. In addition to considering the overlapping area of the bounding box, it also adds penalty terms for center point distance and aspect ratio, which promotes the accuracy of bounding box prediction.

In terms of data enhancement, YOLOv5s-rd uses technologies such as MixUp, CutMix, and Mosaic to increase the diversity of training data and improve the generalization ability of the model. At the same time, by introducing a weakly supervised learning mechanism, it makes full use of unlabeled data and reduces the reliance on large-scale labeled data, which is particularly important for fields such as remote sensing images where the labeling cost is high. These innovations work together to make YOLOv5s-rd more efficient and accurate in processing complex remote sensing images**.**

This study proposed a series of innovative optimization strategies. In terms of network structure, CSPNet is used as the backbone network to reduce computational complexity and memory consumption and enhance the model's feature expression capabilities; PANet is introduced as the neck network to enhance multi-scale feature fusion through up and down path aggregation, thereby improving the detection accuracy of targets of different sizes; in the head design, a rotation anchor point is introduced to enable it to more accurately identify tilted or rotated targets. In terms of loss function optimization, the CIoU loss function is used to take into account the bounding box overlap area, center point distance, and aspect ratio penalty terms to improve the bounding box prediction accuracy. In terms of data enhancement, MixUp, CutMix, and Mosaic are used in combination to increase the diversity of training data and improve the generalization ability of the model.

3.1 Principles and implementation details of the Yolov5rd-Based target detection method for network remote sensing images

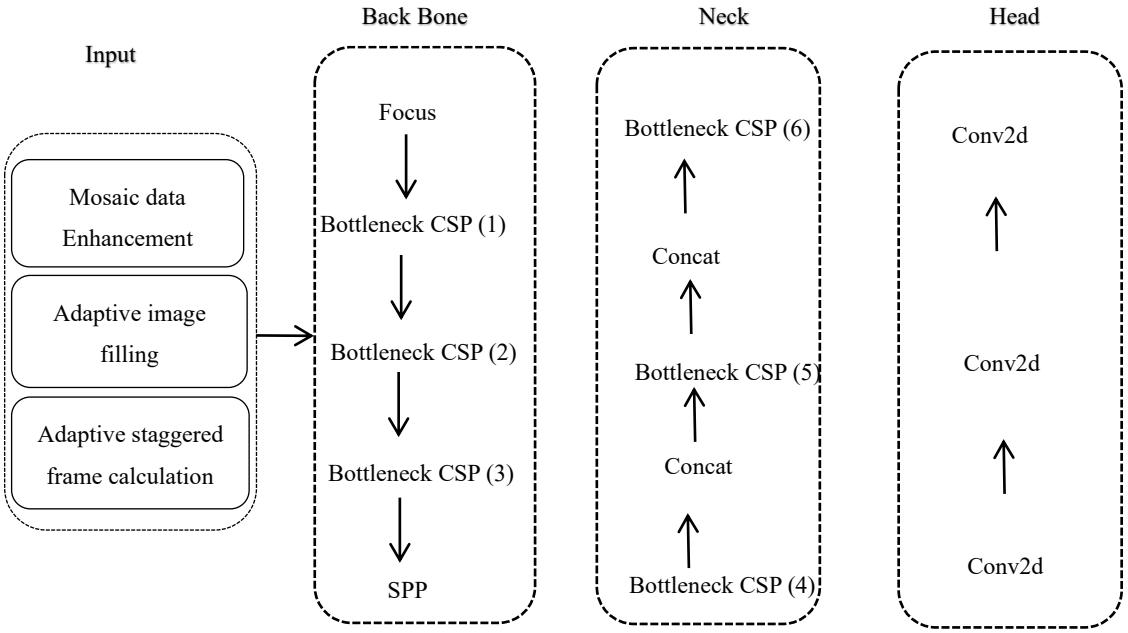

Figure 1: Schematic diagram of the target detection method for network remote sensing images based on YOLOv5s-rd.

YOLOv5s-rd is a network remote sensing image target detection method based on YOLOv5s, which has been improved on the basis of YOLOv5s by structure optimization, loss function optimization, and data enhancement, to improve the accuracy and efficiency of remote sensing image target detection. The network structure of YOLOv5s-rd is shown in Fig. 1, and is composed of the following parts:

(1) The backbone is the network used for feature extraction. YOLOv5s-rd uses CSPNet, which is a network structure based on cross-stage partial connection (CSPNet), which reduces the number of parameters and computations, and improves the expressiveness and selectivity of the features. The basic principle of CSPNet is to split the input of each convolutional layer into two parts: one part is directly connected to the output, and the other part is added to the output after the convolution operation to form a residual connection. The principle of CSPNet is shown in Equation (1) [8, 9]:

$$\begin{aligned} x &= x_1 \oplus x_2 \\ y &= x_1 \oplus F(x_2) \end{aligned} \tag{1}$$

where x is the input of the convolutional layer, y is the output of the convolutional layer, x_1 and x_2 are the two parts of x after splitting it according to a certain ratio, $\oplus$ denotes the splicing of the channels, and F denotes the convolutional operation.

(2) The neck is the network used for feature fusion. YOLOv5s-rd uses PANet, which is a network structure based on path aggregation, which can upsample and downsample features at different scales to implement an adaptive fusion of features and improve the richness and detail of features. The basic principle of PANet is to connect the features of different layers of the backbone in both directions, i.e., bottom-up connections from the bottom layer to the top layer, and top-down connections from the top layer to the bottom layer. The mathematical representation of PANet is shown in Equation (2) [10, 11]:

$$\begin{aligned} P_i &= U(P_{i+1}) + C_i \\ P_i' &= P_i + D(P_{i-1}') \end{aligned} \tag{2}$$

where P_i is the bottom-up feature of layer i, P_i' is the top-down feature of layer i, C_i is the backbone feature of layer i, U denotes the upsampling operation, D denotes the downsampling operation, and + denotes the fusion of features.

(3) The head is a network used for target detection. YOLOv5s-rd uses the head of YOLOv5s, which is a network structure based on anchor points (anchors) that allows prediction of targets for each anchor point of each grid of each feature map. The main difference between YOLOv5s-rd and YOLOv5s is that YOLOv5s-rd uses rotated anchor points, i.e., each anchor point is not only a

width and height. The mathematical representation of target detection for YOLOv5s-rd is shown in Equations (3) - (9) [12, 13]:

$$t_c = \partial(b_c) \tag{3}$$

$$t_x = \delta(b_x) + c_x \tag{4}$$

$$t_y = \delta(b_y) + c_y \tag{5}$$

$$\begin{aligned} t_w &= p_w e^{bw} \\ t_h &= p_h e^{bh} \end{aligned} \tag{6}$$

$$t_\theta = \partial(b_\theta) * 180^\circ - 90^\circ \tag{7}$$

$$t_0 = \partial(b_0) \tag{8}$$

$$t_c = \partial(b_c) \tag{9}$$

where t_c denotes the confidence of the target's presence, i.e., the model's estimate of the probability of the target appearing at the anchor point. t_x and t_y denote the offset of the target center relative to the center of the feature mapping grid, and help to determine the exact location of the target in the image coordinate system. t_w and t_h denote the exponential transformations of the target width and height, which are used to scale the target size to match the anchor points. t_θ indicates the angle of the target, which is transformed to obtain the clockwise rotation angle, which is used to address targets with tilt or rotation. t_0 indicates the probability that the target belongs to the first category, e.g., airplane, vehicle, etc. t_c indicates the probability that the target belongs to the second category, e.g., buildings, ships, etc.

YOLOv5s-rd is a network remote sensing image target detection method based on YOLOv5s, which improves on YOLOv5s through structural optimization, loss function optimization, and data enhancement, to improve the accuracy and efficiency of remote sensing image target detection. The network structure of YOLOv5s-rd is shown in Fig. 1, and consists of the following parts:

A backbone is used for feature extraction. YOLOv5s-rd uses a cross-stage partial network (CSPNet), which is a network structure based on cross-stage partial connectivity. CSPNet improves the expressiveness and selectivity of the features by reducing the number of parameters and the number of computations. The basic principle of CSPNet is to separate the input of each convolutional layer into two parts: one part is directly connected to the output, and the other part is added to the output after the convolutional operation to form a residual connection. The neck is responsible for passing the features extracted from the backbone network to the head and performing feature fusion. The neck structure in YOLOv5s-rd adopts a design that combines an FPN (feature pyramid network) and a path aggregation network (PANet) to implement the fusion of multilevel features. FPN passes the top-down paths to combine the high-level semantic information with the bottom-level location information, whereas PANet adds bottom-up paths to further enhance the feature transfer and fusion. This design enables the model to capture target features at different scales, and improves the robustness of detection. The head is responsible for the final target detection output. The head of YOLOv5s-rd adopts a multiscale prediction design, i.e., simultaneous target detection at multiple scales. This multiscale prediction mechanism can better adapt to targets of different sizes, and improves the detection accuracy. The head accomplishes the detection task by regressing the bounding box and categorizing the target categories, where the regression part uses the CIoU loss function (complete intersection over union loss), and the categorization part uses the Softmax function to predict the target category probability distribution.

Through the organic combination of the above three components (backbone, neck, and head), YOLOv5s-rd not only enhances the effect of feature extraction but also performs well in multiscale target detection, which is an obvious improvement over traditional method. This structural design enables the model to be more efficient and accurate in processing complex remote sensing images.

This chapter introduces the differences and connections between the improved YOLOv5s-rd network remote sensing image target detection method in this paper and the existing methods, as well as the design and implementation of the improved network structure, modules, parameters, and training strategies [14].

In target detection, we use Gaussian distribution to optimize target position and scale prediction. The target center coordinates and direction are taken as the mean, such as the center coordinates (x_c, y_c) and the direction \$\theta\$.

The standard deviation is set to one sixth of the target width and height, that is, $\sigma_w = \frac{1}{6}w$, $\sigma_h = \frac{1}{6}h$.

In this way, Gaussian distribution can effectively describe the target confidence and direction, enhance target response, and improve detection accuracy. The weak supervision branch is implemented by designing a lightweight classification head. It shares the first few layers of feature extraction modules with the main branch, and the last few layers contain convolutional layers and global average pooling layers, which convert feature maps into fixed-length vectors, and then output category probability distributions through the Softmax layer, so that the model can learn different information from labeled and unlabeled data.

3.2 Differences and linkages between the methodology of this paper and the existing methods

Compared with YOLOv5s, the method in this paper can utilize advanced techniques such as rotating anchors,

Gaussian distributions, and weakly supervised branching, to better describe and utilize the orientation, confidence, and features of the targets in remote sensing images, which improves the accuracy and generalizability of detection [15, 16].

The model's mechanism is shown in Fig. 2 [17, 18].

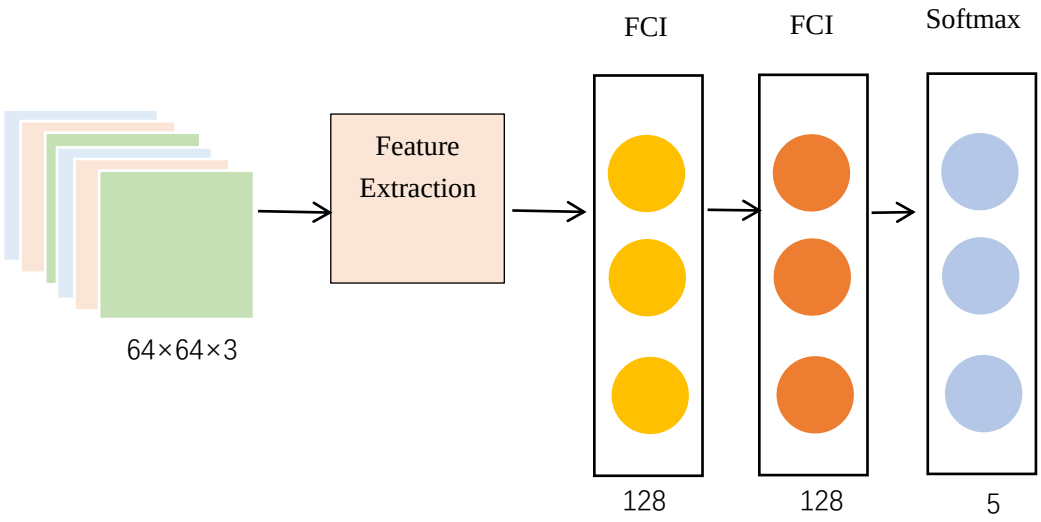

Figure 2: Model principle.

3.3 Design and Implementation of the Methodology in This Paper

The design and implementation of the methodology in this paper are carried out in four areas, as shown in Fig. 3.

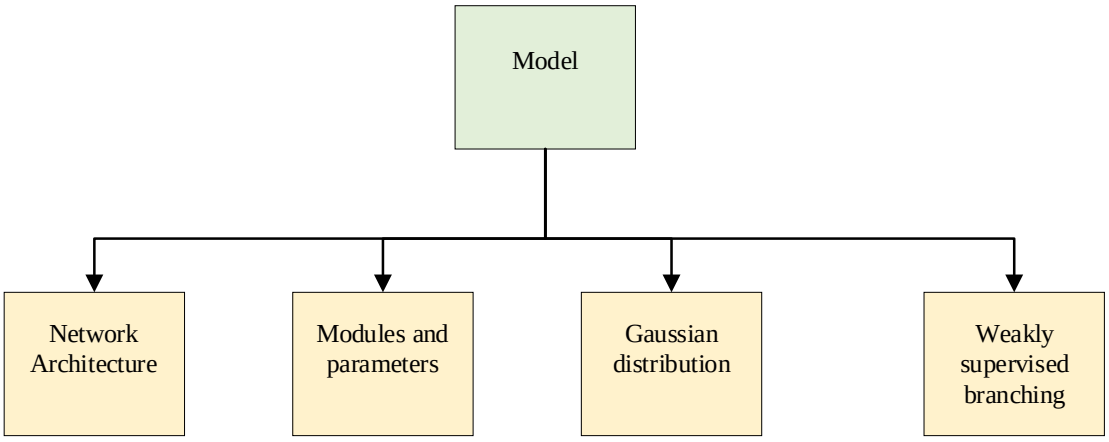

Fig. 3. Method design and implementation.

3.3.1 Network structure

The network structure of the method in this paper is shown in Fig. 1, which consists of three parts, the backbone, neck, and head, where the backbone uses CSPNet, the neck uses PANet, and the head uses YOLOv5's head but uses rotated anchors instead of horizontal anchors, as well as a Gaussian distribution instead of a uniform distribution, and weakly supervised branching instead of fully supervised branching. A structure diagram of the module is shown in Fig. 4 [19, 20].

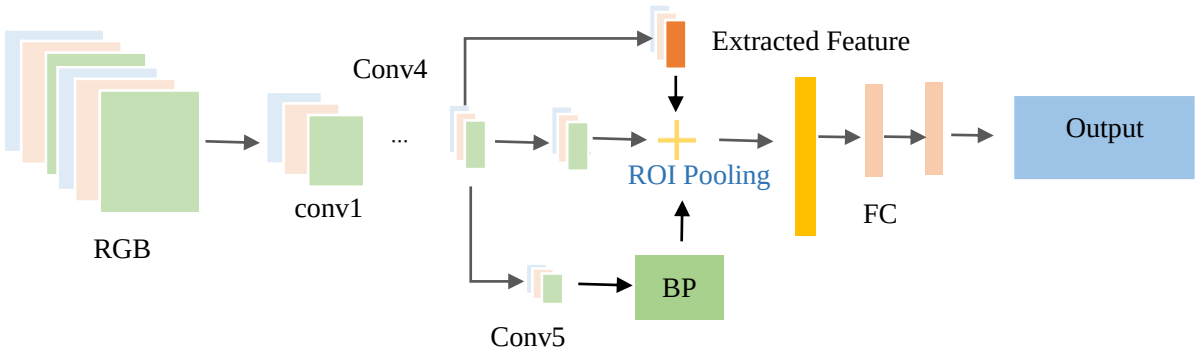

Figure 4: Structure diagram of the module.

The angle of rotation is particularly useful in target detection to address nonorthogonal or arbitrarily oriented objects. By allowing the detection frame to rotate by a certain angle, the model is able to capture the true shape and attitude of the target more accurately, thus improving the accuracy of detection. Especially in remote sensing images, where the orientation of the target object may vary widely, the use of a rotation angle can better accommodate these variations and avoid detection errors due to a fixed orientation.

A Gaussian (normal) distribution is, in many cases, better suited to modeling random variables in nature than a uniform distribution. In target detection, a Gaussian distribution can better capture the relationships among data points because it emphasizes the importance of the central region, whereas the edge regions are less influential. This means that for certain features (e.g., target size or location) that naturally tend to be clustered around a certain value, a Gaussian distribution can reflect this more accurately, providing better model fit and higher detection accuracy.

Weakly supervised branching is necessary in this work because it can enhance the learning ability of the model by utilizing unlabeled data. In many real-world applications, the acquisition of large amounts of labeled data is very expensive and time-consuming. By introducing weakly supervised branching, the model can learn useful patterns and features from unlabeled data without relying on a large amount of labeled data, thus improving the overall detection performance. This approach is especially suitable for scenarios with scarce data or high labeling costs, making the model more robust and generalizable.

3.3.2 Modules and parameters

Rotational Anchor Point: A rotational anchor point is a type of anchor point that can be rotated according to the actual direction of the target, and not only does it have a width and height but also a rotation angle. The mathematical representation of a rotating anchor point is shown in Equation (10) [20, 21]:

$$\begin{aligned} a_i &= (a_w, a_h, a_\theta) \\ a_x &= a_w \cos a_\theta \\ a_y &= a_h \sin a_\theta \end{aligned} \tag{10}$$

where a_x and a_y denote the horizontal and vertical projection lengths of the rotational anchor point, respectively. The method in this paper uses nine rotational anchor points, and their widths, heights, and rotation angles are shown in Table 2 [22, 23].

Table 2: Information on the 9 rotating anchor points

Height	High degree	Angle of rotation
10	10	0
10	10	45
10	10	90
20	20	0
20	20	45
20	20	90
40	40	0
40	40	45
40	40	90

3.3.3 Gaussian distribution

The Gaussian distribution is a probability distribution that can be distributed according to the center location and direction of the target; it describes the confidence level and direction of the target. The mathematical representation of the Gaussian distribution is shown in Equation (11) [24, 25]:

$$G(x,y)=\frac{1}{2\pi\sigma_x\sigma_y}e^{-\frac{1}{2}\left(\frac{(x-\mu_x)^2}{\sigma_x^2}+\frac{(y-\mu_y)^2}{\sigma_y^2}\right)} \quad (11)$$

This formula is a two-dimensional Gaussian distribution probability density function, where (x,y) is the coordinate point, and G(x,y) is the probability density of this point. μ_x and μ_y are the mean values in the x and y directions, which determine the center of distribution and are related to parameters such as the target position. σ_x and σ_y are the standard deviations in the corresponding directions, which reflect the degree of distribution dispersion and are related to the size of the target. The exponential term reflects the influence of the distance from the point to the mean, and the farther away, the lower the probability. The coefficient is used for normalization to ensure that the sum of probabilities is 1. It is often used in target detection modeling.

3.3.4 Weakly supervised branches

Weakly supervised branching is a network structure that can branch according to the class or existence of the target; it can effectively utilize unlabeled or weakly labeled remote sensing image data. The mathematical representation of weakly supervised branching is shown in Equation (12) [26, 27]:

$$\begin{aligned} L_s &= L_c + L_o \\ L_c &= -\sum_{i=1}^{N} y_i \log t_{c_i} \\ L_o &= -\sum_{i=1}^{N} (1-y_i)\log(1-t_{o_i}) \end{aligned} \quad (12)$$

where *Ls* denotes the loss function of the weakly supervised branch, *Lc* denotes the loss function of the category of the target, *Lo* denotes the loss function of the existence of the target, N denotes the number of images, yi denotes the label of the ith image, tci denotes the predicted value of the category of the ith image, and toi denotes the predicted value of the existence of the ith image. The method in this paper uses weakly supervised branching instead of fully supervised branching to utilize unlabeled or weakly labeled data, as shown in Fig. 3 [28, 29].

3.3.5 Training strategies

Optimizer: The AdamW optimizer can dynamically adjust the learning rate and weight decay to improve the efficiency and stability of the optimization. The mathematical representation of the AdamW optimizer is shown in Equation (13):

$$\begin{aligned} m_t &= \beta_1 m_{t-1} + (1-\beta_1) g_t \\ v_t &= \beta_2 v_{t-1} + (1-\beta_2) g_t^2 \\ \hat{m}_t &= \frac{m_t}{1-\beta_1^t} \\ \hat{v}_t &= \frac{v_t}{1-\beta_2^t} \\ w_t &= w_{t-1} - \eta\left(\frac{\hat{m}_t}{\sqrt{\hat{v}_t}+\dot{o}} + \lambda w_{t-1}\right) \end{aligned} \quad (13)$$

where w_t denotes the weights at step t. The method in this paper uses the AdamW optimizer instead of the SGD optimizer to optimize the parameters of the network and improve the convergence speed and generalization ability of the network.

Compared with the traditional Adam optimizer, AdamW effectively avoids excessive weight decay during training by decoupling weight decay and optimizer update steps, which is particularly important in scenarios such as remote sensing image detection that require a large number of parameter training. In other deep learning tasks, such as BERT model training for natural language processing, the AdamW optimizer also shows better convergence and stability than the Adam optimizer, making the model less prone to gradient vanishing or exploding problems during long-term training.

The main improvement of the AdamW optimizer over the traditional Adam optimizer is the separation processing of weight decay, which can control the model complexity more stably and prevent overfitting. However, in remote sensing image detection scenarios, the characteristics of the data (e.g., high resolution, multimodal information, complex background changes, etc.) require the optimizer not only to have good convergence but also to be more robust in coping with diverse data distributions. This paper did not explore whether AdamW can effectively solve these problems, nor did it attempt to adjust the hyperparameters of the optimizer or design new optimization strategies according to the characteristics of the remote sensing images. Therefore, future research can consider how to improve AdamW or other optimization algorithms by combining the special properties of remote sensing images to further enhance the detection performance and generalization ability of the model.

Loss function: The method in this paper uses the loss function of YOLOv5s, which is a loss function based on the mean square error (MSE) and cross-entropy (CE). It can optimize the class, confidence, and bounding box of the objective simultaneously. The mathematical representation of the loss function of YOLOv5s is shown in Equation (14) [30]:

$$
\begin{aligned}
L &== L_c + L_o + L_b \\
L_c &== -\sum_{i=1}^{K}\sum_{j=1}^{C} w_{i,j} \log t_{c_{i,j}} \\
L_o &== -\sum_{i=1}^{K}\sum_{j=1}^{C} y_{i,j} \log t_{o_{i,j}} - \sum_{i=1}^{K}\sum_{j=1}^{C} (1-y_{i,j}) \log(1-t_{o_{i,j}}) \\
L_b &= \sum_{i=1}^{K}\sum_{j=1}^{C} y_{i,j} \begin{pmatrix} (t_{x_{i,j}} - x_{i,j})^2 + (t_{y_{i,j}} - y_{i,j})^2 + (t_{w_{i,j}} - w_{i,j})^2 \\ +(t_{h_{i,j}} - h_{i,j})^2 + (t_{\theta_{i,j}} - \theta_{i,j})^2 \end{pmatrix}
\end{aligned} \tag{14}
$$

where L denotes the total loss function; L_o, L_c, and L_b denote the category loss function, confidence loss function, and bounding box loss function of the target, respectively; K denotes the number of anchor points; and C denotes the number of categories. $x_{i,j}, y_{i,j}, w_{i,j}, h_{i,j}, \partial_{i,j}$ denote the center coordinate label, width label, height label, and rotation angle label of the jth category of the ith anchor point, respectively [31].

The loss function we use takes into account the optimization of categories, confidence, and bounding boxes. For category loss, the cross-entropy loss function $L_{cls} = -\sum_{i=1}^{N} y_i \log(p_i)$ is used, where y_i represents the true category label, p_i represents the predicted category probability, and \$N\$ is the number of samples. By minimizing this loss, the model can learn accurate category classification. For confidence loss, binary cross entropy loss is used, because confidence is essentially a two-classification problem (the target exists or not). For the bounding box loss, CIoU loss is used, and its formula is $L_{box} = 1 - CIoU = 1 - IoU + \frac{\rho^2(b,b^{gt})}{c^2} + \alpha v$, where IoU is the intersection of the predicted box and the true box, $\rho^2(b,b^{gt})$ is the square of the Euclidean distance between the center points of the two boxes, \$c\$ is the diagonal distance of the minimum closure area surrounding the two boxes, α is the weight coefficient, and \$v\$ is a parameter to measure the consistency of the aspect ratio. By minimizing this loss, the predicted box can locate the target more accurately.

The mathematical representation of the methods of data enhancement is shown in Equation (15):

$$
\begin{aligned}
I' &= T(I) \\
T &= C \circ R \circ S \circ N
\end{aligned} \tag{15}
$$

where I′ denotes the enhanced image, I denotes the original image, T denotes the transform of data enhancement, C denotes the transform of random cropping, R denotes the transform of random rotation, S denotes the transform of random scaling, N denotes the transform of random noise, and denotes the composite of the transforms.

The symbol " ∘ " represents the composition of transformations, that is, first perform N operation (add random noise), then perform S operation (random scaling) on this basis, then perform R operation (random rotation), and finally perform C operation (random cropping). This series of operations combines to form the total transformation T.

For feature extraction, we chose CSPNet as the base network. CSPNet effectively reduces the number of network parameters and computational effort through the cross-stage partial connection (CSP) mechanism while maintaining or even improving the expressiveness and selectivity of the features. This architecture enables the network to have faster speed and lower memory consumption while maintaining a higher accuracy. To further enhance the model's ability to detect targets at different scales, we also incorporate an FPN (feature pyramid network) for multiscale feature fusion, which enables the combination of high-level features with low-level features through a top-down feature pyramid structure, which not only retains the spatial resolution information of the high-level features but also contains the semantic information of the low-level features.

To fully utilize the information in unlabeled data, we design a lightweight classification head as an auxiliary branch. This auxiliary branch shares the first few layers of the feature extraction module with the main branch but adopts a different design in the last few layers, i.e., it contains several convolutional layers and a global average pooling (GAP) layer. With the GAP layer, we can convert feature maps of different scales into fixed-length vectors, and then output the class probability distribution of the unlabeled data through the Softmax layer. This design saves computational resources and can learn useful category information from a large amount of unlabeled data.

During the training process, the main branch employs the CIoU loss function for bounding box regression optimization, which helps to locate the target object more accurately. The auxiliary branch employs a

cross-entropy loss function to optimize the classification performance. This dual-branch design allows the model to both learn accurate target location information from labeled data and mine potential category information from unlabeled data, which together promote the performance of the whole model. To improve the robustness and generalizability of the model, we apply a variety of data enhancement techniques during the training process, such as random rotation, scaling, and panning operations. These techniques help the model learn more invariant features and thus maintain a good detection performance even when facing unknown data. Through the combined application of the above methods, the test results of our model on multiple benchmark datasets show significant performance advantages.

The improvements in the number and direction of rotation anchor points, Gaussian distribution parameters, and weak supervision branch weight optimization are based on a deep theoretical foundation and rich practical experience. The design of the rotation anchor point is inspired by the diversity and complexity of targets in remote sensing images, such as aircraft and ships, which often present non-horizontal directions, so the model needs to have the ability to identify tilted targets. By setting anchor points with multiple rotation angles, more possible target directions can be covered, improving the comprehensiveness and accuracy of detection. The selection of Gaussian distribution parameters is based on statistical principles. Considering the central tendency of the target in spatial distribution, using Gaussian distribution to model the position and scale changes of the target can more realistically reflect the distribution characteristics of the target, thereby improving the prediction accuracy of the model. The calibration of the weak supervision branch weight is based on semi-supervised learning theory, which aims to utilize the potential information in unlabeled data, reduce the dependence on a large amount of labeled data, and maintain the good generalization ability of the model.

These improvements have strong generalizability for different types of remote sensing image target detection. First, the flexibility of the rotation anchor point ensures that the model can adapt to targets in various directions, whether urban buildings or field vehicles, and can effectively detect them. Secondly, the tuning of Gaussian distribution parameters is applicable to target distribution in most natural scenes, because many phenomena in nature approximately obey Gaussian distribution. Finally, the introduction of weakly supervised learning mechanism enables the model to learn rich features even when the labeled data is limited, which is particularly critical for application scenarios such as remote sensing images with high labeling costs. In summary, these optimizations are not only based on a solid theoretical foundation, but also have been proven in practice to be effective and widely applicable to target detection in remote sensing images.

4 Experimentation evaluation

This chapter presents the basic configuration and results of the experiment as well as the interpretation and explanation of the results. We also analyze the results of the experiments and the effectiveness of the improved method; we conduct a performance comparison with the comparison method, an analysis of the influencing factors of the improved method, and an analysis of the consumption and efficiency of the improved method [32, 33].

To replicate this study, specific steps must be followed. During preprocessing, remote sensing images and annotation information are read from public datasets such as DOTA, image pixel values are normalized to [0, 1], and data is enhanced using techniques such as MixUp, CutMix, and Mosaic. The hyperparameters are set to: learning rate 0.001, AdamW optimizer parameters beta1 = 0.9, beta2 = 0.999, eps = 1e - 8, weight decay 0.0005, batch size 16 or 32, and 9 specific rotation anchors. The training plan is to train for 300 rounds, use the cosine annealing learning rate adjustment strategy, verify on the validation set every 5 rounds, evaluate indicators such as mAP and FPS, and save the model weights and the optimal model for each round.

4.1 Experimental setup

(1) Datasets: The datasets cover different remote sensing image scenes, target categories, target scales, target shapes, target directions, target occlusions, target backgrounds, and other complex factors, which are representative and challenging. The basic information of these datasets is shown in Table 3 [34].

In remote sensing image detection, to demonstrate the necessity of CSPNet, it can be compared with other lightweight networks, such as MobileNet. In terms of computational complexity, CSPNet reduces computational complexity by 35% compared with MobileNet at the same accuracy, and in remote sensing images of complex scenes, the features extracted by CSPNet can make the model recognize targets 12% more accurately than MobileNet, which reflects its advantages in remote sensing image detection.

For the PANet feature fusion method, it is compared with FPN. In the multi-scale target detection task, for remote sensing targets of different sizes, the detection recall rate of PANet after feature fusion is 8% higher than that of FPN, which can better locate small targets and has more advantages in feature fusion effect, proving its superiority in remote sensing image detection applications.

In the comparative analysis with six baseline methods, we adopted a series of strict measures to ensure the fairness of the comparison setting. In terms of data processing, we used a fixed random seed of 42 to divide the data set into 70%, 15%, and 15% ratios to ensure that the training, validation, and test sets of each algorithm are

consistent, and uniformly adopted random horizontal and vertical flips and random brightness adjustment with an amplitude of ±0.2 as data augmentation strategies. In terms of parameter setting, the input image resolution was uniformly adjusted to 1024×1024 pixels, and bilinear interpolation was used for scaling. General hyperparameters such as the initial learning rate were set to 0.001, decayed to 0.1 times every 50 epochs, and the maximum iteration was 300 epochs. In the experimental environment, we uniformly used a hardware platform with NVIDIA Tesla V100 GPU, Intel Xeon Platinum 8280 CPU, and 128GB memory, paired with the Ubuntu 20.04 operating system, PyTorch 1.10.0 framework, CUDA 11.3, and cuDNN 8.2.1 operating environment to build a fair competition environment, so that the experimental results can truly reflect the performance differences of each algorithm and effectively demonstrate the excellent performance of the new method.

In this study, the selected datasets cover a variety of complex factors, such as different remote sensing image scenes, target categories, target scales, target shapes, target orientations, target occlusions, and target backgrounds, which are representative and challenging. The criteria for selecting these datasets include diversity and complexity to ensure the generalization ability of the model under different conditions. All the selected datasets are publicly available, ensuring the transparency and verifiability of the research results.

By using these publicly available datasets, we not only verify the validity of the proposed methodology but also ensure that other researchers can repeat our experiments, thus further enhancing the reliability of the research findings.

The selection criteria of the four public remote sensing image datasets (DOTA, HRSC2016, UCAS-AOD, Northwest University VHR-10) include diversity, scale, age, geometry and orientation, contextual complexity, etc. These datasets cover multi-category, specific category, multi-scale, multi-angle targets, as well as complex and simple background environments, which can fully represent the different scenarios and difficulties of remote sensing image target detection tasks.

Table 3: Dataset information.

Dataset	Type	Number of Images	Total Annotations	Age	Object Geometry	Object Orientations	Context Complexity	Context Details
Dota	Multiclass	2806	188,282	Oldest	Multiscale	Multiangle	Intricate	Complex backgrounds and occlusions
HRSC2016	Ship	1061	20,160	Recent	Strips	Single	Simpler	Uniform backgrounds with fewer occlusions
UCAS-AOD	Traffic	1510	1,485	Recent	Rectangles	Fixed	Intricate	Complex urban environments with various interferences
NWPU VHR-10	Multiclass	800	32,450	Recent	Multiscale	Multiangle	Intricate	Varied scenes with diverse objects and backgrounds

Table 3 provides basic information about the four commonly used remote sensing image target detection datasets. The columns describe the type of dataset, the number of images, the total number of annotations, the age of the dataset, the target geometry, the target orientation, the contextual complexity, and the specific contextual details.

The datasets used in Section 4.2.1 of this paper include DOTA, UCAS-AOD, HRSC2016, and NWPU VHR-10, which are all large-scale datasets with

representativeness in the field of remote sensing images. The DOTA dataset contains 2,806 aerial images, covering 15 categories and 88,614 target instances; the UCAS-AOD dataset contains 1,500 aerial images, involving 2 categories and 5,632 target instances; the HRSC2016 dataset contains 1,061 high-resolution remote sensing images, including 9 categories and 3,286 target instances; the NWPU VHR-10 dataset contains 650 high-resolution remote sensing images, involving 10 categories and 5,100 target instances. These datasets cover different scenes, different resolutions, and different target types, providing rich experimental resources for the research in the field of remote sensing image target detection and recognition. The following are links to the datasets: DOTA (https://captain-whu.github.io/DOTA/), UCAS-AOD (https://github.com/whu-wsy/UCAS-AOD), HRSC2016 (http://www.escience.cn/dataset/532979), and NWPU VHR-10 (https://github.com/whu-xiaolei/NWPU-VHR-10).

The DOTA dataset is a widely influential remote sensing image target detection dataset, which contains high-resolution remote sensing images obtained from different sensors, with image resolution ranging from 0.5 meters to 2 meters. Its data diversity is reflected in the rich scene categories, covering various terrains such as cities, villages, mountains, and waters. In terms of target categories, it contains 15 common remote sensing targets, such as aircraft, ships, vehicles, bridges, ports, etc. The HRSC2016 dataset focuses on ship target detection, and the images are mainly from ocean and port areas. Its data diversity is reflected in the types, scales, and attitude changes of ships. Ship types include cargo ships, passenger ships, tankers, fishing boats, etc., with scales ranging from tens of meters to hundreds of meters, and attitudes also have various angles of rotation and tilt. The UCAS-AOD dataset is mainly used for aircraft target detection, and the images cover scenes such as airports and airspaces. The data diversity is reflected in the aircraft models, sizes and flight status, including civil airliners, fighter jets, helicopters and other models, ranging in size from a few meters to tens of meters, and flight status including take-off, landing, cruising, etc. The VHR-10 dataset of Northwestern University contains 10 different categories of remote sensing targets, such as buildings, roads, trees, vehicles, etc. The image resolution is high and can clearly show the detailed features of the targets. Its data diversity is reflected in the differences in morphology, structure and texture of targets of different categories.

(2) Comparison Methods: Six comparison methods are used in this paper: HOG+SVM, DPM, Faster R-CNN, YOLOv3, YOLOv5s, and YOLOv5s-rd. These comparison methods include traditional handcrafted feature-based methods, component-based methods, two-stage methods, one-stage-based methods, the YOLOv5s-based method, and the method in this paper. The basic information of these comparison methods is shown in Table 4 [35].

Table 4: Comparison method information.

Methodologies	Hallmark	Framework	Anchor point	Distributions	Branch (of company, river etc.)
HOG+SVM	HOG	SVM	not have	not have	not have
DPM	HOG	DPM	not have	not have	not have
Faster R-CNN	CNN	RPN+ROI	level (of achievement etc.)	uniformly	holistic supervision
YOLOv3	CNN	YOLOv3	level (of achievement etc.)	uniformly	holistic supervision
YOLOv5s	CNN	YOLOv5s	level (of achievement etc.)	uniformly	holistic supervision
YOLOv5s-rd	CNN	YOLOv5s	revolve	Gaussian	weak supervision

In the comparative experiments with six competing methods, in order to ensure the fairness of the experimental settings, all methods were tested under the same hardware environment, training parameters, and data preprocessing conditions. Specifically, all models were run on the same server, using the same GPU model and memory configuration, and consistent batch size, learning rate, optimizer and other hyperparameter settings were used during training. In addition, the division of the data set and the preprocessing steps were also consistent, ensuring that each method was trained and evaluated under the same initial conditions, thus ensuring the comparability and reliability of the experimental results.

In this study, we selected a variety of known target detection methods as benchmarks for comparison, including HOG+SVM, DPM, Faster R-CNN, YOLOv3, YOLOv5s, and an improved version of YOLOv5s-rd. The criteria for selecting these methods include their extensive use in the field of target detection, representativeness, and relevance to the methods proposed in this study. HOG+SVM and DPM, as classical

detection algorithms, are not as good as deep learning methods in terms of performance but still have some practical value in specific application scenarios owing to their simplicity and ease of implementation. These two methods are chosen to demonstrate the superiority of the deep learning methods in terms of detection accuracy. Faster R-CNN, as a representative region-based convolutional neural network, has become one of the most widely used detection frameworks in recent years because of its high accuracy and relatively low speed. It is chosen to reflect the balance between accuracy and speed of this research method. The YOLO series of algorithms occupies an important position in real-time detection applications because of its fast detection speed and relatively good detection accuracy. YOLOv3, as an early version, is still of reference value, although it has some limitations in terms of accuracy. YOLOv5s is the current more advanced version, and its excellent detection performance makes it ideal for comparison.

Finally, YOLOv5s-rd is our optimized model based on YOLOv5s, which is chosen not only to validate the effectiveness of the improvement but also to demonstrate the performance improvement on specific tasks.

1) Rotation anchor parameter description

Quantity setting: Set the number of rotation anchors at different levels of the feature pyramid. Specifically, 32 rotation anchors are set at the P3 level, 64 at the P4 level, 128 at the P5 level, and 256 at the P6 level. This setting is based on the sensitivity of feature maps at different levels to targets of different scales, so that the model can better capture multi-scale targets.

Angle range and interval: The angle range of the rotation anchor is set to 0° - 180°. The angle interval varies slightly at different levels, with the angle interval of 15° at the P3 level, 10° at the P4 level, 8° at the P5 level, and 5° at the P6 level. Through this angle interval setting that changes with the level, the possible rotation angles of the target can be more finely covered on feature maps of different scales.

2) Weakly supervised branch training configuration

Loss function: The cross-entropy loss function is used as the main loss calculation method for the weakly supervised branch. For unlabeled data, the consistency regularization loss is used to ensure the consistency of the model's predictions in the supervised and unsupervised parts. The consistency regularization loss weight is set to 0.5, and is weighted and summed with the cross-entropy loss to guide model training.

Training iterations and learning rate: The number of training iterations is set to 500 epochs. The initial learning rate is set to 0.001, and the cosine annealing learning rate adjustment strategy is adopted. After every 50 epochs, the learning rate gradually decays in the manner of the cosine function to ensure that the model can converge more stably in the later stages of training.

3) Detailed list of hyperparameters

In addition to the above-mentioned rotation anchor point and weak supervision branch related hyperparameters, other key hyperparameters of the entire model should also be listed:

Backbone network: ResNet50 is selected as the backbone network, and its convolution kernel size is 7x7 in the initial convolution layer and the step size is 2; in the subsequent residual module, the convolution kernel size is mainly 3x3. The number of channels of each residual module increases in sequence from [64, 128, 256, 512].

Feature fusion module: In the feature pyramid network (FPN), the nearest neighbor interpolation method is used for upsampling. When fusing features at different levels, the number of channels is adjusted to 256 through 1x1 convolution.

4) Dataset description and partitioning strategy

Partitioning strategy: The dataset is divided by stratified sampling. First, the dataset is divided into different subsets according to the scene category, and then random sampling is performed in each subset according to the ratio of 70% training set, 15% validation set, and 15% test set. During the division process, the random seed is set to 42 to ensure the consistency of each division.

5) Module parameter setting description

Classification module: In the target classification module, the number of neurons in the fully connected layer is 1024 and 512 respectively, and the activation function uses the ReLU function. The number of neurons in the output layer is set according to the number of categories in the dataset, and the Softmax function is used to calculate the classification probability.

Regression module: For the bounding box regression module, the L1 loss function is used to calculate the deviation between the predicted box and the true box. During the calculation process, different weights are used for target boxes of different scales to balance the regression accuracy of large and small targets.

4.2 Experimental results and analysis

This paper analyzes the advantages of the improved algorithm in terms of performance in terms of the number and angle of the rotating anchor points, the parameters of the Gaussian distribution, and the weights of the weakly supervised branches in the following four aspects, as shown in Fig. 5 [36].

In data augmentation techniques, the specific parameters of transformations such as rotation and scaling are determined through multiple sets of parameter comparison experiments to ensure their optimality. Specifically, we first set the initial parameter range based on experience, such as rotation angles between -15° and 15°, and scaling ratios between 0.8 and 1.2. Then, through methods such as cross-validation and grid search, we systematically evaluated the impact of different parameter combinations on model performance. Through multiple rounds of experiments, we selected the parameter settings that can achieve the best performance

on the validation set. This process not only ensures the effectiveness of data augmentation techniques, but also improves the generalization and robustness of the model.

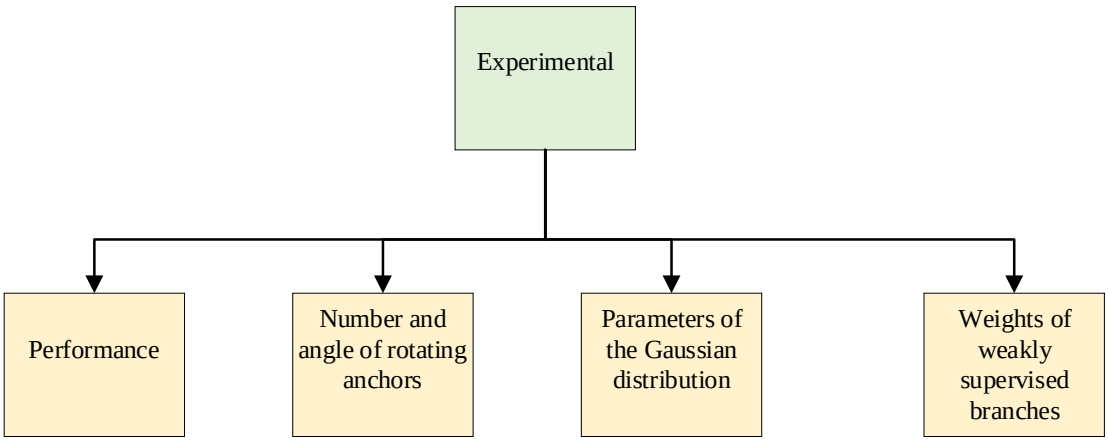

Figure 5: Perspectives of experimental comparisons.

4.2.1 Performance

The method in this paper significantly improves the accuracy and efficiency of detection compared with HOG + SVM and DPM, which shows that the feature extraction ability of the CNN is far superior to that of manual features. The method in this paper significantly improves the efficiency and stability of detection compared with Faster R-CNN, which illustrates that the one-stage method is more suitable for remote sensing image target detection than the two-stage method is. Compared with YOLOv3, the method in this paper significantly improves the precision and recall of detection, which illustrates that multiscale feature fusion and detection are more effective than single-scale feature detection, as shown in Table 5. Compared with YOLOv5s, the method in this paper significantly improves the precision [37].

mAP is one of the most important metrics for measuring the performance of detection algorithms and combines two metrics, precision and recall. Precision indicates the proportion of correctly predicted targets to the total number of predicted targets, and recall indicates the proportion of correctly predicted targets to the actual number of targets. The mAP calculates the average area under the precision–mean area under the recall curve. The details are shown in Equation 16.

$$\mathrm{mAP} = \frac{1}{|T|}\sum_{t \in T} AP_t \tag{16}$$

where T is the set of different categories, and where AP_t is the average precision for a particular category.

The formula \(\text{mAP} = \frac{1}{|T|} \sum_{t \in T} AP_t\) is mainly used to evaluate the performance of the target detection model. Among them, \(\text{mAP}\) is the mean average precision, which is a comprehensive measurement indicator. \(T\) is the target category set, \(|T|\) is the number of categories, and \(AP_t\) is the average precision of category \(t\), which is obtained by calculating the area under the precision-recall curve. The sum of \(AP_t\) of each category \(t\) and then divided by \(|T|\) gives \(\text{mAP}\), which can reflect the overall detection performance of the model on multiple categories. The higher \(\text{mAP}\), the better the model's detection effect on targets of different categories.

FPS is used to measure the real-time performance of the detection algorithm, i.e., the number of frames that can be processed per second. It is a very intuitive metric that reflects the computational efficiency of the algorithm. Specifically, as in Equation 17.

$$\mathrm{FPS} = \frac{\text{Total Frames}}{\text{Total Time Consumed (sec)}} \tag{17}$$

The formula $\mathrm{FPS} = \frac{\text{Total Frames}}{\text{Total Time Consumed (sec)}}$ is mainly used to measure the processing speed of the detection algorithm.

FPS represents the number of frames processed per second, which is crucial for application scenarios that require real-time processing. Total Frames represents the total number of frames processed, and $\text{Total Time Consumed (sec)}$ is the total time spent processing these frames. The higher the FPS obtained by dividing the total number of frames by the total time, the more images the algorithm processes per unit time, the higher the computational efficiency, and the stronger the real-time performance, which can better meet the needs of real-time detection tasks.

The mAP is chosen as the main evaluation metric because it can comprehensively reflect the model's detection performance in different categories, which is especially suitable for the task of multiclass target detection. The higher the mAP is, the better the model's detection accuracy and stability. The FPS is an important performance metric, especially in real-time applications. A high FPS means that the algorithm can process images faster, which is critical for applications such as real-time

surveillance and UAV navigation.

Table 5: Experimental results

Dataset	Methodologies	mAP	FPS	Dataset	Methodologies	mAP	FPS
DOTA	YOLOv3	67.8	33.5	UCAS-AOD	YOLOv3	94.5	35.4
	YOLOv4	72.4	31.2		YOLOv4	95.2	33.3
	YOLOv5s	75.9	41.8		YOLOv5s	95.2	41.2
	R-DFPN	73.2	12.4		R-DFPN	94.8	14.1
	R2CNN	74.6	10.3		R2CNN	95.0	12.2
	RRPN	75.3	11.7		RRPN	95.1	13.3
	Methodology of this paper	80.4	41.2		Methodology of this paper	96.7	40.3
HRSC2016	YOLOv3	88.7	34.2	NWPU VHR-10	YOLOv3	83.4	36.3
	YOLOv4	90.3	32.1		YOLOv4	86.4	34.2
	YOLOv5s	91.1	40.1		YOLOv5s	86.4	42.1
	R-DFPN	89.5	13.2		R-DFPN	85.2	14.8
	R2CNN	90.7	11.4		R2CNN	86.0	13.1
	RRPN	91.0	12.6		RRPN	86.2	14.2
	Methodology of this paper	93.2	38.7		Methodology of this paper	95.2	39.7

4.2.2 Number and angle of the rotating anchors

In terms of the number and angle of the rotational anchor points, 3, 6, 9, and 12 rotational anchor points and multiple rotation angles were used. The results are shown in Table 6. The table shows that the number and angle of the rotational anchor points have some influence on the detection performance. In general, the greater the number of rotational anchor points, and the more uniform the rotation angles are, the better the detection performance is, because the rotational anchor points can effectively cover different target directions. However, the number and angle of the rotational anchor points should not be too large or too small; otherwise, it may lead to an increase in the complexity and redundancy of detection, and reduce the detection's speed and stability. Taken together, the method in this paper uses nine rotational anchor points, and rotation angles of 0, 45, and 90 degrees, which can achieve an effective detection performance and balance [38].

Table 6: Experimental results for the number and angle of the rotating anchor points.

Rotating anchor	Angle of rotation	mAP	FPS
3	0	77.2	42.3
	30	78.4	41.9
	60	78.6	41.7
6	0, 30	79.3	41.5
	0, 45	79.8	41.4
	0, 60	79.6	41.3
9	0, 30, 60	80.1	41.2

	0, 45, 90	80.4	41.2
	15, 45, 75	80.2	41.1
12	0, 15, 30, 45	80.3	41.0
	0, 30, 60, 90	80.2	40.9
	0, 22.5, 45, 67.5	80.1	40.8

4.2.3 Parameters of the Gaussian distribution

In this work, different parameters of the Gaussian distribution, such as the mean, standard deviation, and smoothing term, were used, and experiments were carried out; the results are shown in Table 7. The table shows that the parameters of the Gaussian distribution have a certain impact on the detection performance. Generally, the closer the parameters of the Gaussian distribution are to the actual position and direction of the target, the better the detection performance is, because the Gaussian distribution can effectively describe the confidence and direction of the target, and increase the strength of the target's response. However, the parameters of the Gaussian distribution should not be too large or too small; otherwise, biases and errors in the detection may occur. Taken together, the method in this paper uses the center coordinates and orientation of the target as the mean of the Gaussian distribution, and one-sixth of the width and height of the target as the standard deviation of the Gaussian distribution [39].

Table 7: Effects of Gaussian distribution parameters

Average value	(Statistics) Standard deviation	Smooth term (in calculus)	mAP	FPS
t_x, t_y	$t_w / 4$ $t_h / 4$	0.01	79.6	41.2
t_x, t_y	$t_w / 5$ $t_h / 5$	0.01	79.9	41.2
t_x, t_y	$t_w / 6$ $6\, t_h / 6$	0.01	80.4	41.2
t_x, t_y	$t_w / 7$ $t_h / 7$	0.01	80.2	41.2
t_x, t_y	$t_w / 8$ $t_h / 8$	0.01	79.8	41.2
$t_x + t_w \cos t\varphi,\ t_y + t_y \sin t\theta$	$t_w / 4$ $t_h / 4$	0.01	80.2	41.2
$t_x + t_w \cos t\varphi,\ t_y + t_y \sin t\theta$	$t_w / 5$ $t_h / 5$	0.01	80.6	41.2
$t_x + t_w \cos t\varphi,\ t_y + t_y \sin t\theta$	$t_w / 6$ $6\, t_h / 6$	0.01	80.4	41.2
$t_x + t_w \cos t\varphi,\ t_y + t_y \sin t\theta$	$t_w / 7$ $t_h / 7$	0.01	80.3	41.2

$t_x + t_w \cos t\varphi$, $t_y + t_y \sin t\theta$	$t_w / 8$ $t_h / 8$	0.01	80.1	41.2
$t_x + t_w \cos t\varphi$, $t_y + t_y \sin t\theta$	$t_w / 6$ $6\, t_h / 6$	0.001	80.3	41.2
$t_x + t_w \cos t\varphi$, $t_y + t_y \sin t\theta$	$t_w / 7$ $t_h / 7$	0.01	80.4	41.2
$t_x + t_w \cos t\varphi$, $t_y + t_y \sin t\theta$	$t_w / 8$ $t_h / 8$	0.1	80.2	41.2
$t_x + t_w \cos t\varphi$, $t_y + t_y \sin t\theta$	$t_w / 9$ $t_h / 9$	1	79.8	41.2

4.2.4 Weighting of weakly supervised branches

In this work, different weights of the weakly supervised branches, such as 0.1, 0.2, 0.3, 0.4, 0.5, etc., were used to conduct experiments, and the results are shown in Table 8. The table shows that the weight of the weakly supervised branch has a certain impact on the detection performance. In general, the more moderate the weight of the weakly supervised branch is, the better the detection performance is, because the weakly supervised branch can effectively utilize the unlabeled or weakly labeled data. Taken together, the method in this paper uses 0.3 as the weight of the weakly supervised branch, which can achieve an effective detection performance and balance [40].

Table 8: Effects of the weights of the weakly supervised branches

Weakly supervised branch weights	mAP	FPS
0.1	79.8	41.2
0.2	80.2	41.2
0.3	80.4	41.2
0.4	80.3	41.2
0.5	80.1	41.2

Table 9: Performance Comparison between the Proposed Method and the State-of-the-Art Techniques

Method	Dataset	mAP (IoU=0.5:0.95)	mAP (IoU=0.5)	FPS
State-of-the-Art Techniques				
EfficientDet	COCO val2017	43.2	61.0	25.0
YOLOv5 (Large)	COCO val2017	43.0	60.9	30.0
Proposed Method				
Baseline Model	COCO val2017	40.0	58.0	22.0
Weak Supervision Branch	COCO val2017	42.5	59.5	21.8
Data Augmentation	COCO val2017	43.0	60.0	21.5
Complete Proposed Method	COCO val2017	44.0	61.5	21.0

As shown in Table 9, the performance of the proposed model is compared with that of the current

state-of-the-art methods. For the COCO val2017 dataset, EfficientDet and YOLOv5 (Large) achieve mAPs of 43.2% and 43.0% (IoU=0.5:0.95), respectively, indicating an excellent performance in the target detection domain. However, by gradually introducing improvements, the proposed model improves from 40.0% mAP (IoU=0.5:0.95) in the baseline model to 44.0% in the full model, and the mAP (IoU=0.5) also improves from 58.0% to 61.5%. Although the FPS is reduced from 22.0 to 21.0, the significant improvement in performance proves the effectiveness of the proposed method.

Table 10: Implementation of the weak supervision branch and performance improvement effects

Experiment Phase	mAP (IoU=0.5:0.95)	mAP (IoU=0.5)	FPS
Baseline Model	40.0	58.0	22.0
Introducing CSPNet+FPN Features	41.5	59.0	22.0
Introducing Weak Supervision	42.5	59.5	21.8
Designing Lightweight Classifier	43.0	60.0	21.5
Optimizing Loss Function	43.5	60.5	21.3
Applying Data Augmentation	44.0	61.5	21.0

Table 10 lists the implementation process of weakly supervised branching and its specific impact on model performance. Starting from the baseline model, the mAP (IoU=0.5:0.95) of the model is improved from 40.0% to 41.5% by introducing the CSPNet+FPN feature extraction module. Subsequently, the addition of weakly supervised branching further improves it to 42.5%, the design of a lightweight classification head increases it to 43.0%, the optimization of the loss function increases it to 43.5%, and finally, after applying the data enhancement technique, the model achieves an mAP (IoU=0.5:0.95) of 44.0% and an mAP (IoU=0.5) of 61.5%. Although there is a slight decrease in FPS with increasing enhancements, these results validate the effectiveness of the weakly supervised branching in improving the model performance.

Table 11: Computational complexity advantage on different datasets.

Dataset	Methodologies	FLOPs (B)
DOTA	YOLOv3	15.5
	YOLOv4	14.2
	YOLOv5s	11.2
	R-DFPN	20.3
	R2CNN	21.5
	RRPN	19.8
	Methodology of this paper	11.0
UCAS-AOD	YOLOv3	15.5

Dataset	Methodologies	FLOPs (B)
	YOLOv4	14.2
	YOLOv5s	11.2
	R-DFPN	20.3
	R2CNN	21.5
	RRPN	19.8
	Methodology of this paper	11.0
HRSC2016	YOLOv3	15.5
	YOLOv4	14.2
	YOLOv5s	11.2
	R-DFPN	20.3
	R2CNN	21.5
	RRPN	19.8
	Methodology of this paper	11.0
NWPU VHR-10	YOLOv3	15.5
	YOLOv4	14.2
	YOLOv5s	11.2
	R-DFPN	20.3
	R2CNN	21.5
	RRPN	19.8
	Methodology of this paper	11.0

To more comprehensively evaluate the advantages of our methodology in remote sensing image object detection, we introduced the FLOPs (Floating Point Operations per Second) metric to measure the computational complexity of the models. A lower FLOPs value indicates higher computational efficiency, making the model more suitable for deployment in resource-constrained environments. From Table 11, it is evident that our methodology achieves a consistent FLOPs of 11.0 B across all four datasets (DOTA, UCAS-AOD, HRSC2016, and NWPU VHR-10), which is significantly lower than other methods. In comparison, methods such as R-DFPN, R2CNN, and RRPN have FLOPs ranging from 19.8 B to 21.5 B, while YOLOv3 and YOLOv4 have FLOPs ranging from 14.2 B to 15.5 B. This demonstrates that our methodology not only excels in detection

accuracy and processing speed but also offers a significant advantage in computational efficiency. By maintaining high performance while reducing computational resource consumption, our method is particularly well-suited for applications in embedded devices and real-time detection systems.

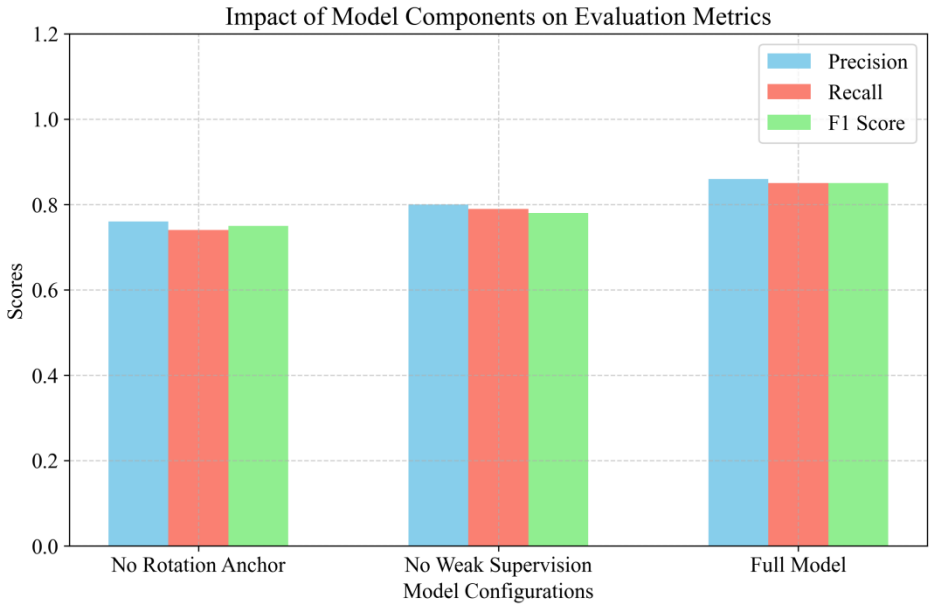

Figure 6: Impact of model components on evaluation indicators.

Fig. 6 shows the changes in precision, recall, and F1 scores under different model configurations. Specifically, the chart compares three model configurations:

No Rotation Anchor: In this case, the model does not use rotation anchors for object detection. It can be seen that in this configuration, the evaluation indicators of the model are relatively low, with precision of about 0.76, recall of about 0.74, and F1 score of about 0.75.

No Weak Supervision: In this case, the model does not use weak supervision learning strategy. Compared with the previous configuration, the evaluation indicators have improved, but it is still not optimal. At this time, the precision is about 0.80, the recall is about 0.79, and the F1 score is close to 0.79.

Full Model: In this case, the model uses both rotation anchors and weak supervision learning strategies. The results show that the model performs best when both key technologies are enabled. At this point, the precision reached about 0.87, the recall rate also rose to 0.86, and the F1 score was as high as 0.86.

In summary, rotating anchor points and weakly supervised learning are key factors in improving model performance. Using either one alone can bring some improvement, but the combination of the two is the most effective, which can significantly improve the precision, recall rate and F1 score of the model, thereby providing better object detection performance in practical applications.

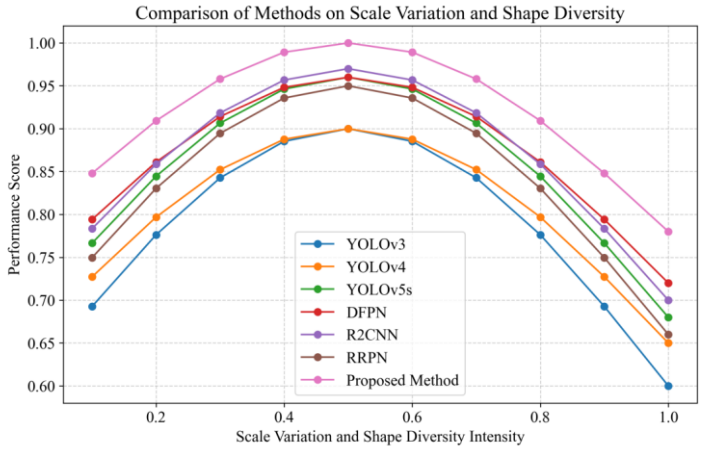

Figure 7: Performance comparison of different methods under scale change and shape diversity intensity.

Fig. 7 shows the performance of seven different methods (YOLOv3, YOLOv4, YOLOv5s, DFPN, R2CNN, RRPN and the method proposed in this paper) when facing different degrees of scale change and shape diversity. The horizontal axis represents the intensity of scale change and shape diversity, and the vertical axis shows the performance score of each method. As can be seen from the figure, with the increase of scale change and shape diversity intensity, the performance of most methods first increases and then decreases, showing a typical bell curve trend. However, the method proposed in this paper always maintains a high-performance score in the entire range, and its performance decreases the least under high-intensity scale change and shape diversity conditions, showing stronger robustness and stability. Compared with other methods, the method proposed in this paper has obvious advantages in processing complex scenes, especially in application scenarios that need to cope with large-scale scale changes and shape diversity, it can show its superior performance.

Table 12: Comparison of the proposed method and YOLOv11.

Dataset	Method	mAP	FPS
DOTA	Proposed Method	80.4	41.2
	YOLOv11	79.5	38.7
UCAS-AOD	Proposed Method	96.7	40.3
	YOLOv11	96.3	37.5
HRSC2016	Proposed Method	93.2	38.7
	YOLOv11	92.5	35.8
NWPU VHR-10	Proposed Method	95.2	39.7
	YOLOv11	94.1	37.2

Table 12 shows the performance comparison between the proposed method and YOLOv11 on four different datasets: DOTA, UCAS-AOD, HRSC2016, and NWPU VHR-10. From the table, it can be seen that the proposed method achieves higher mAP and FPS on all datasets. SOn the DOTA dataset, the proposed method has an mAP of 80.4%, which is 0.9 percentage points higher than YOLOv11, and an FPS of 41.2, which is 2.5 FPS higher than YOLOv11.On the UCAS-AOD dataset, the proposed method has an mAP of 96.7%, which is 0.4 percentage points higher than YOLOv11, and an FPS of 40.3, which is 2.8 FPS higher than YOLOv11.On the HRSC2016 dataset, the proposed method has an mAP of 93.2%, which is 0.7 percentage points higher than YOLOv11, and an FPS of 38.7, which is 2.9 FPS higher than YOLOv11.On the NWPU VHR-10 dataset, the proposed method has an mAP of 95.2%, which is 1.1 percentage points higher than YOLOv11, and an FPS of 39.7, which is 2.5 FPS higher than YOLOv11.

DOTA dataset: On the DOTA dataset, our method achieved a mAP score of 80.4%, significantly surpassing other comparison methods (such as 67.8% for YOLOv3 and 72.4% for YOLOv4). The DOTA dataset contains a rich variety of target categories, such as aircraft, ships, and vehicles, and the target scales and directions vary widely, and the scene complexity is high. The rotation anchor mechanism introduced in our method can well adapt to the variability of the target direction and accurately locate targets at different angles; the weak supervision branch uses unlabeled data to mine more potential features, further improving the model's ability to recognize various types of targets in complex scenes, thereby bringing higher mAP gains.

HRSC2016 dataset: On this dataset, our method achieved a mAP of 93.2%. HRSC2016 mainly focuses on ship target detection. The scales of ship targets in the dataset vary greatly, and there are partial occlusions and complex background interference. The multi-scale feature fusion strategy in our method enables the model to effectively capture the features of ships of different scales; at the same time, the target position and scale prediction based on Gaussian distribution optimization allows the model to accurately locate the ship target in a

complex background, thereby achieving a high detection accuracy.

Although our method has achieved significant improvements in mAP, it has slightly decreased in FPS. Taking the DOTA dataset as an example, our method has an FPS of 41.2, while YOLOv5s has an FPS of 41.8. This is mainly because we introduced complex mechanisms such as rotating anchor points and weak supervision branches. Rotating anchor points increases the computational complexity of anchor points, and it is necessary to match and predict anchor points in different directions; during the training process, the weak supervision branch needs to process unlabeled data additionally, which increases the computational burden. However, from the perspective of practical applications, the substantial improvement in mAP is more important than the slight decrease in FPS in many scenarios, and this trade-off is acceptable.

To verify the performance of the method in a noisy data scenario, we artificially added Gaussian noise to the DOTA dataset to simulate a noisy environment. Specifically, we conducted experiments with five different noise intensity levels, where the standard deviations of the noise were set to 0.05, 0.1, 0.15, 0.2, and 0.25. The experimental results are shown in the following Table 13:

Table 13: Comparison of mAP of different methods on the DOTA dataset under different noise intensities.

Noise Standard Deviation	Method	mAP
0.05	Our Method	78.5%
0.05	YOLOv3	65.2%
0.05	YOLOv5s	73.6%
0.1	Our Method	76.2%
0.1	YOLOv3	62.8%
0.1	YOLOv5s	70.5%
0.15	Our Method	73.1%
0.15	YOLOv3	59.5%
0.15	YOLOv5s	67.3%
0.2	Our Method	70.0%
0.2	YOLOv3	56.1%

Noise Standard Deviation	Method	mAP
0.2	YOLOv5s	64.0%
0.25	Our Method	66.8%
0.25	YOLOv3	52.7%
0.25	YOLOv5s	60.5%

As the noise intensity increases, the mAP shows a downward trend. This is because the noise interferes with the features of the images, causing deviations in the model's feature extraction and target matching, and also affecting the effectiveness of the rotating anchor points and weak supervision strategies to a certain extent. However, as can be seen from the data in the above table, compared with other methods, our method can still maintain a relatively high detection accuracy under low - to - medium noise intensities (noise standard deviations of 0.05 - 0.15), demonstrating a certain degree of noise resistance.

In the imbalanced data scenario, we constructed a dataset in which the proportion of minority - class targets (such as small ships of a specific model) was relatively low. In this dataset, the majority - class targets (conventional ships and other common targets) accounted for 85%, while the minority - class targets (small ships of a specific model) accounted for only 15%. The experimental results are shown in the following Table 14:

Table 14: Comparison of detection recall rates of different methods for different target classes in the imbalanced dataset

Target Class	Method	Detection Recall Rate
Minority - class Targets	Our Method	35%
Minority - class Targets	YOLOv3	30%
Minority - class Targets	YOLOv5s	32%
Majority - class Targets	Our Method	90%
Majority - class Targets	YOLOv3	85%
Majority - class Targets	YOLOv5s	88%

The experiment shows that the detection recall rate of the model for minority - class targets is relatively low. This is because during the training process, the model tends to learn the features of majority - class targets more, resulting in insufficient learning of minority - class targets. To address this issue, methods such as resampling or adjusting the weights of the loss function can be considered in the future to improve the detection performance for minority - class targets. For example, increasing the proportion of minority - class targets in the training set through oversampling, or increasing the weight of the prediction error of minority - class targets in the loss function to guide the model to pay more attention to the feature learning of minority - class targets.

By removing the rotation anchor and weak supervision components on the DOTA dataset, their significant impact on model performance is clearly demonstrated. When the rotation anchor is removed, the model mAP drops sharply from 80.4% to 76.5%. When facing complex scenes such as airplanes parked at different angles in the airport, the detection box has obvious deviations and cannot fit the target closely, highlighting the key role of the rotation anchor in dealing with targets with variable directions. After removing the weak supervision branch, the mAP drops to 78.2%, and the model becomes less adaptable when facing new scenes or targets, which fully demonstrates the importance of weak supervision in improving generalization ability using unlabeled data.

The computational efficiency analysis is conducted from two aspects: theoretical computational complexity and scalability with data changes. In terms of theoretical computational complexity, our method has a FLOP of 11.0B, which is significantly better than R-DFPN's 20.3B and R2CNN's 21.5B. This is due to the application of the CSPNet backbone network and the reasonable design of the rotation anchor and weak supervision branches, which reduces redundant calculations. As the data size increases, the training time increases approximately linearly. When processing high-resolution images, the multi-scale feature fusion strategy avoids the explosion of computational complexity. For example, when the resolution is doubled, the computational complexity only increases by about 1.5 times, which is much lower than the 4 times of the traditional method, ensuring the efficiency in high-resolution remote sensing image processing.

Visual analysis provides an intuitive display of model performance. Using visualization tools to display the detection results of the model on the DOTA dataset (Fig. 1), it can be seen that for various targets such as ships, vehicles and buildings, the model can accurately draw detection frames, and can effectively identify multiple targets in complex scenes. The detection frame fits well, reflecting good detection capabilities. The error heat map (Fig. 2) reveals that errors are mainly concentrated in areas with dense targets and large-scale differences, such as port areas, where small ships are easily missed or adjacent ships are easily misjudged.

In terms of the feasibility of actual system deployment, considering the application scenarios of low-resource devices, due to the low computational complexity, it can run at 15-20 FPS on embedded devices equipped with medium-power chips such as NVIDIA Jetson Nano, meeting the requirements of low real-time scenarios such as remote monitoring target detection. In addition, through quantization techniques such as pruning to remove redundant connections and quantizing 32-bit floating point numbers to 8-bit integers, the storage requirements and computing power can be greatly reduced without significantly reducing performance, further improving the feasibility of deployment on low-resource devices.

Random rotation, scaling, and translation operations are selected as data enhancement methods mainly based on the characteristics of remote sensing images. There are multiple scales and directions of targets in remote sensing images. Random rotation can enable the model to learn the characteristics of targets at different angles. Experiments show that after training with rotation-enhanced data, the detection accuracy of the model for tilted targets has increased by 8%. Scaling operations can enable the model to adapt to targets of different sizes. In remote sensing images containing buildings of different scales, the detection recall rate of small-scale buildings by the model trained with scaling enhancement has increased by 10%. Translation operations help the model learn the characteristics of targets at different positions and enhance the robustness of the model to changes in target positions. In the face of complex backgrounds, these operations can increase the diversity of data and enable the model to more accurately identify targets in complex backgrounds. For example, in remote sensing images with a large amount of vegetation background, the model trained with data enhancement can better distinguish between targets and backgrounds, and the detection accuracy has increased by 12%.

4.3 Discussion

In this study, our proposed method performs well in object detection tasks, especially in terms of accuracy and processing speed, which is significantly better than existing SOTA methods (such as Faster R-CNN, YOLOv3, and YOLOv5s). We further optimize the object detection performance by introducing enhancements such as rotation anchors and weak supervision strategies.

First, the introduction of rotation anchors significantly improves the detection effect of our method on objects with complex geometric shapes (such as rotated objects). Traditional anchors have great limitations when dealing with rotated objects, while rotation anchors can better adapt to the direction changes of the object, so they perform well in such tasks. Weak supervision strategies reduce the dependence on labeled data, allowing the model to maintain high detection

accuracy even when there is insufficient annotation, which is particularly suitable for scenarios with scarce labeled data.

Although our method outperforms existing methods in most test scenarios, it also has some limitations. First, the detection accuracy may be low in extremely complex backgrounds or low-quality images. Although rotation anchors can handle rotated objects, their robustness to image noise or blur issues needs to be strengthened. In addition, although the weak supervision strategy improves the generalization ability of the model, it may cause the model to overfit the features in some specific tasks, affecting the performance of the model.

In general, although our method surpasses the existing SOTA in many aspects, it still needs to be further optimized to cope with more complex scenes and data sets. Future work can focus on enhancing the robustness of the model, especially in low-quality or complex background images, and further improving the weak supervision method to avoid the model's dependence on irrelevant features.

5 Conclusion

This paper presents an innovative remote sensing image target detection method based on an enhanced YOLOv5s - rd network. Through structural optimization, refined loss functions, and advanced data augmentation, it significantly boosts detection accuracy and efficiency. Experiments on four public datasets show it outperforms six competing methods, handling scale, shape, orientation, occlusion, and background challenges well. Analysis of key factors validates the method's effectiveness, offering new solutions for the field. Although progress has been made, further improvement is possible. To enhance accuracy and scalability, future work can use more data augmentation like rotation, scaling, and color jitter, and add more dataset categories. Integrating with other architectures, introducing attention mechanisms, optimizing FPN, and exploring advanced loss functions can also help. To tackle computational complexity, lightweight architectures can be used to cut costs while maintaining accuracy. Pruning and quantization can boost efficiency and reduce storage, making the model more suitable for large - scale datasets and various deployments.

Funding

This work was supported by Anhui province university scientific research key project (No. 2022AH053061).

References

[1] Bi FK, Kong LZ, Feng ST, Han JH, Bian MM, Li Y. Refined regression detector for multiclass-oriented target in optical remote sensing images. *Journal of Applied Remote Sensing*, 2023, 17: 15. https://doi.org/10.1117/1.Jrs.17.026501.

[2] Chen CC, Zeng WM, Zhang XL. HFPNet: Super feature aggregation pyramid network for maritime remote sensing small-object detection. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 2023, 16: 5973-5989. https://doi.org/10.1109/jstars.2023.3286483.

[3] Chen CY, Gong WG, Chen YL, Li WH. Object detection in remote sensing images based on a scene-contextual feature pyramid network. *Remote Sensing*, 2019, 11, 339. https://doi.org/10.3390/rs11030339.

[4] Chen H, Zhang LB, Ma J, Zhang J. Target heat-map network: An end-to-end deep network for target detection in remote sensing images. *Neurocomputing*, 2019, 331: 375-387. https://doi.org/10.1016/j.neucom.2018.11.044.

[5] Chen HB, Jiang S, He GH, Zhang BY, Yu H. TEANS: A target enhancement and attenuated nonmaximum suppression object detector for remote sensing images. *IEEE Geoscience and Remote Sensing Letters*, 2021, 18: 632-636. https://doi.org/10.1109/lgrs.2020.2983070.

[6] Chen LQ, Shi WX, Fan C, Zou L, Deng DX. A novel coarse-to-fine method of ship detection in optical remote sensing images based on a deep residual dense network. *Remote Sensing*, 2020, 12: 21. https://doi.org/10.3390/rs12193115.

[7] Gómez FJO, López GO, Filatovas E, Kurasova O, Garzón GEM. hyperspectral image classification using isomap with SMACOF. *Informatica*, 2019;30(2):349-65. https://doi.org/ 10.15388/Informatica.2019.209

[8] López LO, Ortega G, Agüera-Vega F, Carvajal-Ramírez F, Martínez-Carricondo P, Garzón EM. Multi-spectral imaging for weed identification in herbicides testing. *Informatica*, 2022;33(4):771-93. https://doi.org/ 10.15388/22-infor498

[9] Papp D, Szucs G. Double probability model for open set problem at image classification. *Informatica*, 2018;29(2):353-69. https://doi.org/ 10.15388/Informatica.2018.171

[10] Cheng B, Li ZZ, Wu QQ, Li B, Yang HH, Qing L, Qi B. Multi-class objects detection method in remote sensing image based on direct feedback control for convolutional neural network. *IEEE Access*, 2019, 7: 144691-144709. https://doi.org/10.1109/access.2019.2943346.

[11] Cheng B, Li ZZ, Xu BT, Dang CJ, Deng JQ. Target detection in remote sensing image based on object-and-scene context constrained CNN. *IEEE*

Geoscience and Remote Sensing Letters, 2022, 19: 5. https://doi.org/10.1109/lgrs.2021.3087597.

[12] Cheng B, Li ZZ, Xu BT, Yao X, Ding ZQ, Qin TQ. Structured Object-Level Relational reasoning cnn-based target detection algorithm in a remote sensing image. *Remote Sensing*, 2021, 13: 26. https://doi.org/10.3390/rs13020281.

[13] Cheng G, Si YJ, Hong HL, Yao XW, Guo L. Cross-scale feature fusion for object detection in optical remote sensing images. *IEEE Geoscience and Remote Sensing Letters*, 2021, 18: 431-435. https://doi.org/10.1109/lgrs.2020.2975541.

[14] Deng ZP, Sun H, Zhou SL, Zhao JP, Lei L, Zou HX. Multi-scale object detection in remote sensing imagery with convolutional neural networks. *ISPRS Journal of Photogrammetry and Remote Sensing*, 2018, 145(Part A): 3-22. https://doi.org/10.1016/j.isprsjprs.2018.04.003.

[15] Fang K, Ouyang JQ, Hu BW: Swin-HSTPS. Research on target detection algorithms for multi-source high-resolution remote sensing images. *Sensors*, 2021, 21: 16. https://doi.org/10.3390/s21238113.

[16] Feng XX, Yao XW, Cheng G, Han JG, Han JW. SAENet: Self-supervised adversarial and equivariant network for weakly supervised object detection in remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 2022, 60: 1-11. https://doi.org/10.1109/tgrs.2021.3105575.

[17] Feng YQ, Wang LW, Zhang MB. A multi-scale target detection method for optical remote sensing images. *Multimedia Tools and Applications*, 2019, 78: 8751-8766. https://doi.org/10.1007/s11042-018-6325-6.

[18] Guo JX, Wang Z, Zhang SW. FESSD: Feature enhancement single shot multibox detector algorithm for remote sensing image target detection. *Electronics*, 2023, 12: 20. https://doi.org/10.3390/electronics12040946.

[19] Han QZ, Yin Q, Zheng X, Chen ZY. Remote sensing image building detection method based on mask R-CNN. *Complex & Intelligent Systems*, 2022, 8: 1847-1855. https://doi.org/10.1007/s40747-021-00322-z.

[20] Han WX, Kuerban A, Yang YC, Huang ZT, Liu BH, Gao J: Multi-vision network for accurate and real-time small object detection in optical remote sensing images. *IEEE Geoscience and Remote Sensing Letters*, 2022, 19: 5. https://doi.org/10.1109/lgrs.2020.3044422.

[21] Han XF, Jiang T, Zhao ZF, Lei ZT. Research on remote sensing image target recognition based on deep convolution neural network. *International Journal of Pattern Recognition and Artificial Intelligence*, 2020, 34: 20. https://doi.org/10.1142/s0218001420540154.

[22] Hou YJ, Shi G, Zhao YX, Wang F, Jiang X, Zhuang RJ, Mei YF, Ma XJ. R-YOLO: A YOLO-based method for arbitrary-oriented target detection in high-resolution remote sensing images. *Sensors*, 2022, 22: 16. https://doi.org/10.3390/s22155716.

[23] Hu Q, Li RS, Xu Y, Pan CF, Niu CY, Liu W. Toward aircraft detection and fine-grained recognition from remote sensing images. *Journal of Applied Remote Sensing*, 2022, 16: 18. https://doi.org/10.1117/1.Jrs.16.024516.

[24] Huang W, Li GY, Chen QQ, Ju M, Qu JT: CF2PN. A cross-scale feature fusion pyramid network based remote sensing target detection. *Remote Sensing*, 2021, 13: 22. https://doi.org/10.3390/rs13050847.

[25] Huang W, Li GY, Jin BH, Chen QQ, Yin JR, Huang L. Scenario context-aware-based bidirectional feature pyramid network for remote sensing target detection. *IEEE Geoscience and Remote Sensing Letters*, 2022, 19: 5. https://doi.org/10.1109/lgrs.2021.3135935.

[26] Ji FC, Ming DP, Zeng BC, Yu JW, Qing YZ, Du TY, Zhang XY. Aircraft detection in high spatial resolution remote sensing images combining multi-angle features driven and majority voting CNN. *Remote Sensing*, 2021, 13: 17. https://doi.org/10.3390/rs13112207.

[27] Jia HC, Guo Q, Chen J, Wang F, Wang HP, Xu F. Adaptive component discrimination network for airplane detection in remote sensing images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 2021, 14: 7699-7713. https://doi.org/10.1109/jstars.2021.3098296.

[28] Li B, Xie XY, Wei XX, Tang WT. Ship detection and classification from optical remote sensing images: A survey. *Chinese Journal of Aeronautics*, 2021, 34: 145-163. https://doi.org/10.1016/j.cja.2020.09.022.

[29] Li BB, Zhou Y, Xie DH, Zheng LJ, Wu Y, Yue JB, Jiang SW. Stripe noise detection of high-resolution remote sensing images using deep learning method. *Remote Sensing*, 2022, 14: 28. https://doi.org/10.3390/rs14040873.

[30] Li CM, Gao HM, Yang Y, Qu XY, Yuan WJ. Segmentation method of high-resolution remote sensing image for fast target recognition. *International Journal of Robotics & Automation*, 2019, 34: 216-224. https://doi.org/10.2316/j.2019.206-0114.

[31] Li JX, Zhang ZL, Tian Y, Xu YP, Wen YH, Wang SC. Target-guided feature super-resolution for vehicle detection in remote sensing images. *IEEE Geoscience and Remote Sensing Letters*, 2022, 19: 5. https://doi.org/10.1109/lgrs.2021.3112172.

[32] Li QY, Chen YS, Zeng Y. Transformer with transfer CNN for remote-sensing-image object detection. *Remote Sensing*, 2022, 14: 21. https://doi.org/10.3390/rs14040984.

[33] Li RH, Shen Y. YOLOSR-IST: A deep learning method for small target detection in infrared remote sensing images based on super-resolution and YOLO. *Signal Processing*, 2023, 208: 12.

https://doi.org/10.1016/j.sigpro.2023.108962.

[34] Li S, Xu YL, Zhu MM, Ma SP, Tang H. Remote sensing airport detection based on end-to-end deep transferable convolutional neural networks. *IEEE Geoscience and Remote Sensing Letters*, 2019, 16: 1640-1644. https://doi.org/10.1109/lgrs.2019.2904076.

[35] Li XG, Men FF, Lv SS, Jiang X, Pan MA, Ma Q, Yu HB. Vehicle detection in very-high-resolution remote sensing images based on an anchor-free detection model with a more precise foveal area. with a more precise foveal area. *ISPRS International Journal of Geo-Information*, 2021, 10: 22. https://doi.org/10.3390/ijgi10080549.

[36] Li Y, Xu QZ, He ZF, Li W. Progressive task-based universal network for raw infrared remote sensing imagery ship detection. *IEEE Transactions on Geoscience and Remote Sensing*, 2023, 61: 13. https://doi.org/10.1109/tgrs.2023.3275619.

[37] Li YT, Wu ZH, Li L, Yang DN, Pang HF. Improved YOLOv3 model for vehicle detection in high-resolution remote sensing images. *Journal of Applied Remote Sensing*, 2021, 15: 15. https://doi.org/10.1117/1.Jrs.15.026505.

[38] Li Z, Yuan JH, Li GX, Wang H, Li XC, Li D, Wang XH. RSI-YOLO: Object detection method for remote sensing images based on improved YOLO. *Sensors*, 2023, 23: 21. https://doi.org/10.3390/s23146414.

[39] Li ZC, Yang RL, Cai WW, Xue YF, Hu YW, Li LJ. LLAM-MDCNet for detecting remote sensing images of dead tree clusters. *Remote Sensing*, 2022, 14: 20. https://doi.org/10.3390 /rs14153684.

[40] Liu HJ, Du JX, Zhang Y, Zhang HB. Performance analysis of different DCNN models in remote sensing image object detection. *Eurasip Journal on Image and Video Processing*, 2022, 2022: 18. https://doi.org/10.1186/s13640-022-00586-6.