remote sensing

MDPI

Article

MSCNet: A Multilevel Stacked Context Network for Oriented Object Detection in Optical Remote Sensing Images

Rui Zhang [1,2], Xinxin Zhang [1,2,*], Yuchao Zheng [1,2], Dahan Wang [1,2] and Lizhong Hua [1,2]

[1] College of Computer and Information Engineering, Xiamen University of Technology, Xiamen 361024, China
[2] Fujian Key Laboratory of Pattern Recognition and Image Understanding, Xiamen University of Technology, Xiamen 361024, China
* Correspondence: zhangxinxin@xmut.edu.cn

Abstract: Oriented object detection has recently become a hot research topic in remote sensing because it provides a better spatial expression of oriented target objects. Although research has made considerable progress in this field, the feature of multiscale and arbitrary directions still poses great challenges for oriented object detection tasks. In this paper, a multilevel stacked context network (MSCNet) is proposed to enhance target detection accuracy by aggregating the semantic relationships between different objects and contexts in remote sensing images. Additionally, to alleviate the impact of the defects of the traditional oriented bounding box representation, the feasibility of using a Gaussian distribution instead of the traditional representation is discussed in this paper. Finally, we verified the performance of our work on two common remote sensing datasets, and the results show that our proposed network improved on the baseline.

Keywords: oriented object detection; multilevel stacked context; remote sensing images

Citation: Zhang, R.; Zhang, X.; Zheng, Y.; Wang, D.; Hua, L. MSCNet: A Multilevel Stacked Context Network for Oriented Object Detection in Optical Remote Sensing Images. *Remote Sens.* **2022**, *14*, 5066. https://doi.org/10.3390/rs14205066

Academic Editors: Qian Du, Yanni Dong and Xiaochen Yang

Received: 9 August 2022
Accepted: 3 October 2022
Published: 11 October 2022

1. Introduction

With the rapid development of remote sensing technology, remote sensing data have gradually decreased acquisition costs, enriched data sources, and improved image resolution and quality. Consequently, remote sensing images have gradually become popular and are used in various industries. As a popular image processing method, convolutional neural networks have also become a powerful tool for processing remote sensing data. Compared with traditional manual feature construction, convolutional neural networks directly generate feature representations from the original image pixels, and the deep stacked structure of convolutional neural networks helps extract more abstract semantic features; these results are beyond the reach of the traditional method. For example, in object detection, Ren et al. [1] proposed a classical two-stage object detection network.

To address previous studies on natural images, a series of convolutional neural network object detection methods suitable for remote sensing images are proposed. For example, Li et al. [2] designed a multiangle anchor for a region proposal network (RPN), and proposed a two-channel feature fusion network to learn local and context features. Liu et al. [3] used dilated convolution to extract features and dynamically adjusted the weight of each position in the dilated convolution kernel to consider the detection accuracy of both large and small objects.

However, due to the orthographic characteristics of optical remote sensing images, object detection networks suitable for natural images have difficulty handling objects in arbitrary directions in remote sensing images. Because most of the objects in natural images are usually perpendicular to the ground due to gravity, the influence of direction is smaller than that in the remote sensing image. However, remote sensing images are taken from the upper space of the earth, and objects on the ground usually have different directions. This problem creates new challenges for the object detection task of remote sensing images. For instance, due to the dense distribution of some targets in remote sensing images, multiple

targets may crowd in the same region of interest (RoI) proposed by the RPN. As a result, overlapping features will make it difficult to train subsequent classifiers.

To address this challenge, people try to use oriented bounding boxes instead of traditional horizontal detection bounding boxes in remote sensing object detection tasks to improve detection performance. Much progress has also been made in this area, and most oriented bounding boxes have been developed by using a Faster R-CNN with a feature pyramid network (FPN) [4] structure, such as DRBox [5] and CAD-Net [6]. Although these studies have verified the excellent performance of these methods on multiple datasets, there is still room for improvement. For example, Zhang et al. combined a global context module and a local pyramid context module in their CAD-Net to enhance both global and local semantic features. Additionally, the attention mechanism of spatial scale is designed in combination with the FPN structure. This combination of global and local context design is the first approach used in remote sensing image target detection and has been proven to be an effective idea. However, in studies using this approach, the relationship between the global and local context features of each layer is not fully utilized.

Many studies have indicated that in deep convolutional networks, the deeper features of images contain mainly semantic information, while lower features show more boundary details of objects. Additionally, remote sensing images contain more positional relations and semantic connections between objects than natural images due to the particularity of orthophotos. Moreover, due to the local limitations of the convolution operation, it is difficult to model the relationship between two distant objects by using the convolution operation. For example, two different regions in an image have the same target, and the semantic relationship between them may be similar. In addition, the spatial relationship also helps improve the detection performance of the network. For example, bridges and harbors have a similar appearance and are usually located in water, but both ends of a bridge are usually connected to the road, and bridges have a spatial relationship with vehicles; in contrast, only one end of a harbor is usually connected to the road, and harbors have a spatial relationship with ships. Due to the local effective receiving field of the convolution operation, these relationships cannot be effectively modeled by the convolution layer alone.

Thus, it is natural to improve the performance of the detection network by capturing and modeling the connections between objects and contexts, and explicitly modeling long-range relations. Following this idea, we designed a multilevel stacked context network (MSCNet) for remote sensing rotation object detection. First, to effectively capture the relationship between long-range objects, a multilevel stacked semantic capture (MSSC) module is embedded in the network. This module obtains different receiving domains by adding multiple parallels dilated convolutions with different dilated rates at the c5 layer of ResNet and overlays the convolutions in a global-to-local manner to take full advantage of the context relations of different distances. In addition, we perform a multichannel weighted fusion of the RoI in the RPN stage to provide better RoI features for further regression and classification operations.

The main contributions are summarized as follows:

1. An effective multilevel stacked semantic acquisition and enhancement module is proposed to enhance the representation of the FPN on remote sensing images.
2. The Gaussian Wasserstein distance loss is used to alleviate the criticality problems caused by the traditional oriented bounding box representation.
3. An improved RoI allocation strategy is used to enhance FPN multilevel information aggregation, which improves the detection performance of multiscale targets by using a multichannel weighted fusion structure instead of a single layer allocation strategy.

The rest of this paper is organized as follows. Section 2 will introduce the main object detection methods and the related studies on remote sensing images for oriented object detection networks in detail. Section 3 introduces the methods we proposed for MSSCNet. Section 4 provides the experimental results and discussions. Finally, Section 5 gives the conclusions.

2. Related Work

2.1. Anchor-Based Object Detectors

Anchor-based object detection networks focus mainly on anchor generation and secondarily on classification and regression. Anchor-based detectors are divided into two main types according to the number of network stages: one-stage detectors and two-stage detectors. As an example of two-stage detectors, R-CNN and its variants [7,8] generate several proposals for an RoI before training, and then the RoI mapped back to the feature map is further classified and regressed. The main one-stage detectors include SSD [9], RetinaNet [10], RefineDet [11], YOLOV2, and variant YOLOV2 networks [12–14]. Unlike two-stage detectors, which generate a series region proposal by using a Selective Search algorithm or an RPN, one-stage detectors predict category and location information directly through the backbone.

Although anchor-based detectors play a dominant role in object detection tasks, these detectors still have the following shortcomings: (1) The anchor design depends on human presets, and the appropriate anchor size and aspect ratio need to be set for different datasets. (2) A fixed-anchor design is not conducive to detecting extreme scale targets. (3) Generating the proposal requires many anchor samplings; however, having many anchors will lead to sample imbalance. (4) To achieve good performance, some hyperparameters need to be carefully adjusted; these include anchor shape and the IoU threshold of positive and negative samples.

2.2. Anchor-Free Object Detectors

Since anchor-based object detectors are subject to various anchor constraints, some people proposed an anchor-free architecture. Anchor-free detectors are also divided into two forms: key-point-based and center-based detectors.

CornerNet [15], borrowing the idea from algorithms such as human gesture estimation, denotes the detection box by combining the upper left corner and lower right corner points of the target. ExtremeNet [16] defines a key point as an extreme point, predicts four multipeak heatmaps for each target to find the extreme point, and predicts the target center through the center heatmap. The geometric center of the extreme point corresponds to the score of the center heatmap as the grouping condition.

FCOS [17] is similar to FCN [18]; FCOS uses a fully convolutional network to regress the distance from each position of the feature map to the four edges of the targets directly and obtains an effect similar to an anchor-based detector. CenterNet [19] transforms the detection problem into a key point problem and uses a heatmap to directly predict the center point and the size of the target. Each target only predicts one positive sample center point, so non-maximum suppression (NMS) is no longer needed to filter positive and negative samples.

2.3. Arbitrarily Oriented Object Detectors

Since most of the objects to be detected are placed horizontally in the natural image object detection task, early studies were based on horizontal bounding boxes. However, with the gradual deepening of the research, when the object detection scene is extended to text detection, aerial image detection, and 3D target detection, the shortcoming of the horizontal bounding box is gradually exposed; that is, the horizontal bounding box cannot provide accurate positions, and it also affects the performance of the detection network. Therefore, it is necessary to propose an effective rotating detector to generate an oriented instead of a horizontal bounding box for the object.

Among the rotating detectors, a significant problem is how to reasonably represent an oriented target. Different detectors give different definitions. In remote sensing image object detectors, SCRDet, R3Det, CADNet [6,20,21], etc., θ is added into the definition of the detection box as an additional parameter. This method of definition is the most concise and intuitive, and Figure 1 shows two different representations of five parameter-oriented bounding boxes: OpenCV representation and long-side representation. Parameter

$\theta \in [-90^\circ, 0^\circ)$ in the OpenCV representation represents the acute angle or right angle between the bounding box and the x-axis; $\theta \in [-90^\circ, 90^\circ]$ in the long-side representation represents the angle between the long edge of the bounding box and the x-axis.

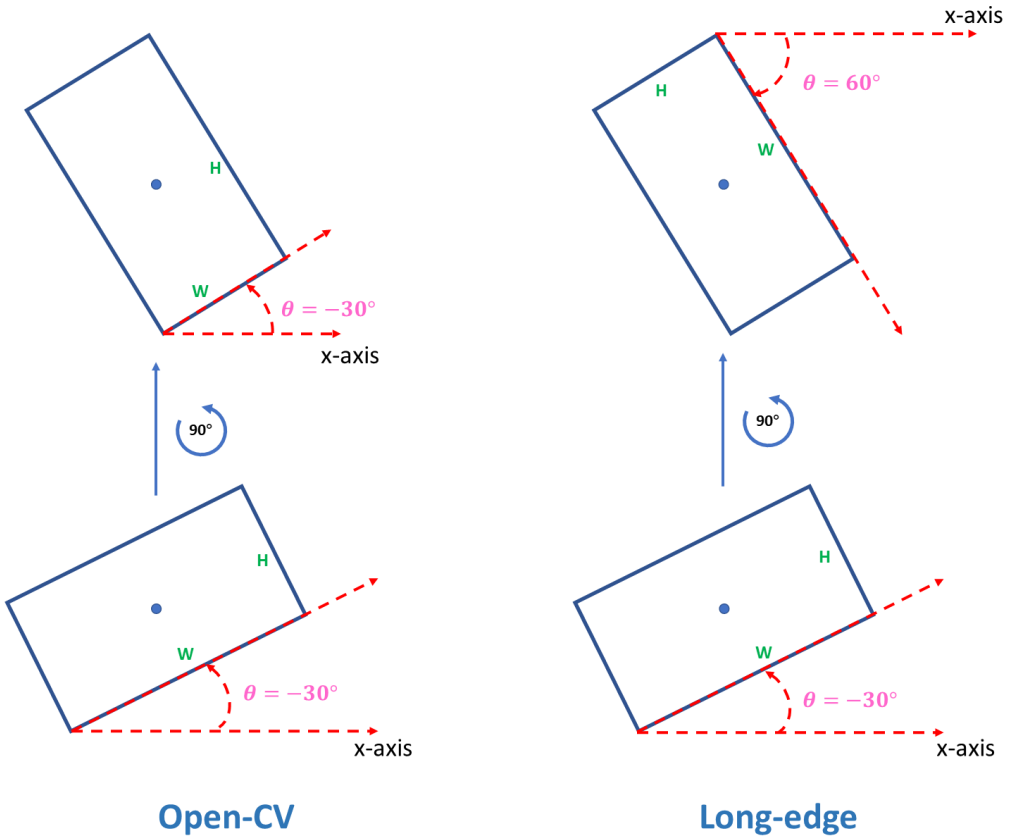

Figure 1. The two representations of oriented bounding boxes. Before rotation occurs, both representations of the bounding boxes are $\{x, y, w, h, \theta = -30^\circ\}$. After the bounding boxes are rotated 90 degrees counter-clockwise, the bounding box is represented in Open-CV as $\{x, y, w' = h, h' = w, \theta = -30^\circ\}$, while the bounding box is represented in Long-edge as $\{x, y, w' = w, h' = h, \theta = 60^\circ\}$.

Unlike the above detectors, Gliding Vertex, Oriented R-CNN [22,23], and others define the oriented bounding box with six parameters $\{x, y, w, h, \theta_1, \theta_2\}$, where $\{x, y, w, h\}$ contains the vertices of the horizontal bounding box of the object; θ_1 and θ_2 are the ratios of the distance to the width and height, respectively, from the vertex of the oriented bounding box to a specific anchor point on the horizontal bounding box; and we usually set the end or midpoint on the width and height of the horizontal bounding box as the anchor point, as shown in Figure 2. Additionally, a few studies still regard the angle of the oriented box as a classification problem; this view is unlike that of most mainstream studies, which regard angle prediction as a regression problem [24].

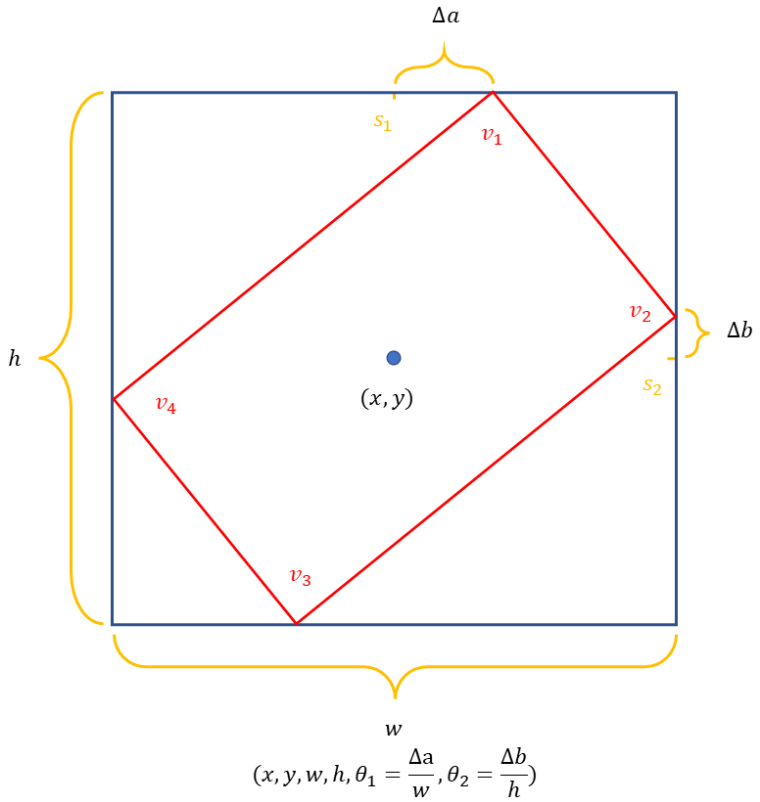

Figure 2. Illustration of six parameters representation for oriented bounding boxes. $s1$ and $s2$ are the midpoints on the width and height of the external horizontal bounding box, Δa and Δb are offsets from the intersecting points to $s1, s2$, or other key points. It adopts $\{x, y, w, h, \theta_1, \theta_2\}$ to represent oriented objects.

3. Methods

3.1. Overview

In this section, we will introduce our proposed network in detail. Figure 3 shows the overall structure of our proposed MSCNet. MSCNet is based on the two-stage detector Faster R-CNN. First, in the feature extraction stage, the long-range relationship capture module is added to enhance the interpretation of the model to the object relationship. For the RPN part, we generate oriented proposals by predicting the anchor offsets and the foreground and background scores. Here, it is necessary to define appropriate rotation parameters when using an oriented bounding box. The simplest way is to add another parameter θ representing the rotation angle, which is expressed as $\{x, y, w, h, \theta\}$. However, this definition will affect detection performance, and we will explain the definition in detail in Section 3.3.1. Thus, to reduce this effect, a Gaussian distribution is used to represent the oriented bounding box, and the Gaussian Wasserstein distance is introduced as a measure of similarity between oriented bounding boxes. Furthermore, in the FPN part, the context information is removed by using the Adaptive RoI Assignment (ARA) module to improve the multiscale expression capability of the FPN. Then, in the R-CNN part, we conduct secondary classification and regression of the proposals in the RPN to obtain the final detection results. Next, we will introduce each part in detail.

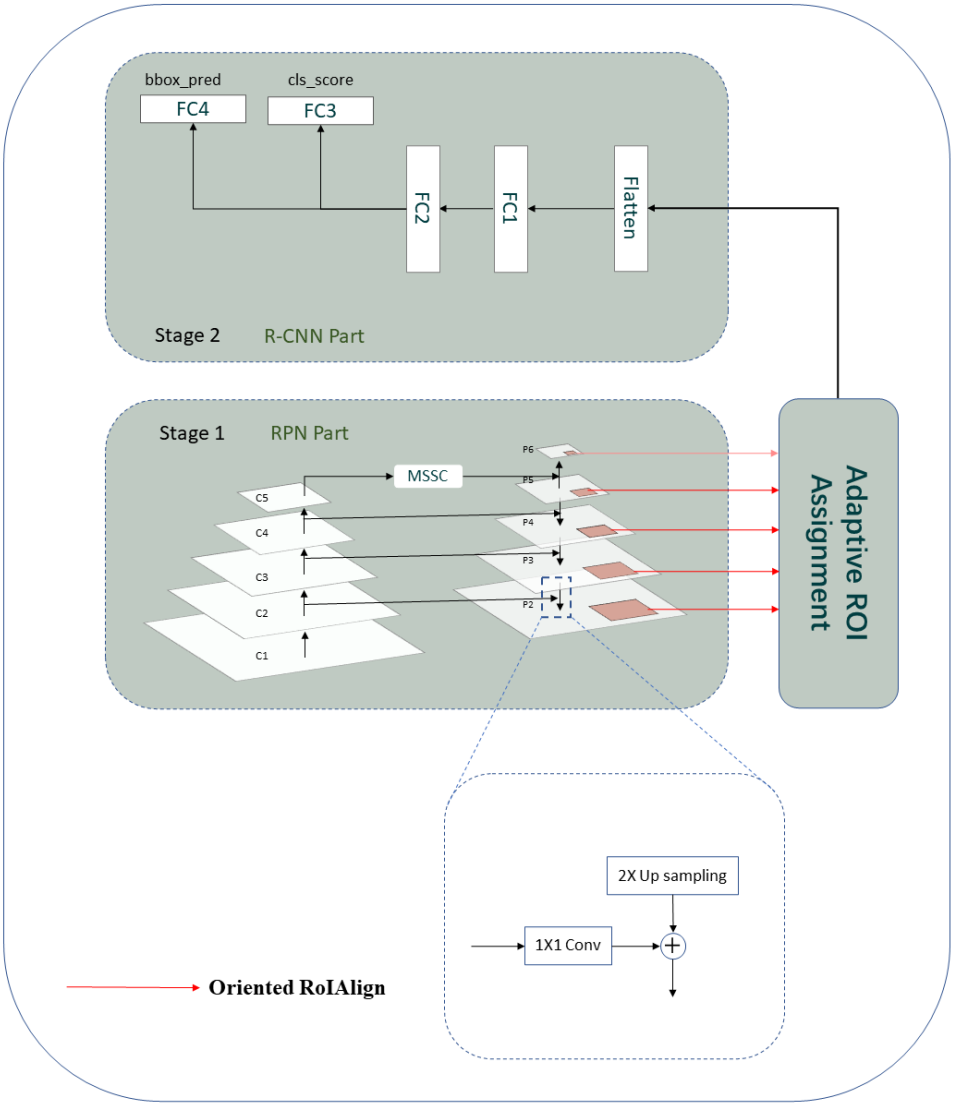

Figure 3. The overall structure of MSCNet.

3.2. MSSC Module

The FPN structure constructs a top-down fusion path to enhance the features of each scale by using a pyramid structure and can improve the detection ability of the network for multiscale targets. Previous studies have shown that the C5 layer in ResNet usually contains rich location and semantic information. Therefore, additional information is added to the $M5$ layer of the FPN by long-range semantic and positional capture of the C5 layer and is added to the bottom layers $\{M4, M3, M2\}$ along with the top-down fusion path. In this paper, an MSSC module is proposed. This module uses a parallel method to obtain receptive fields of different sizes and aggregates them in a stacking way from global receptive fields to local receptive fields. In this way, this module can take full advantage of relationships between multiple contexts and contexts at different levels.

In detail, at the C5 layer, the dilatation convolution with four different dilatation coefficients is used to capture the context information at multiple scales, with $\{3, 6, 12, 18\}$ (see Figure 4).

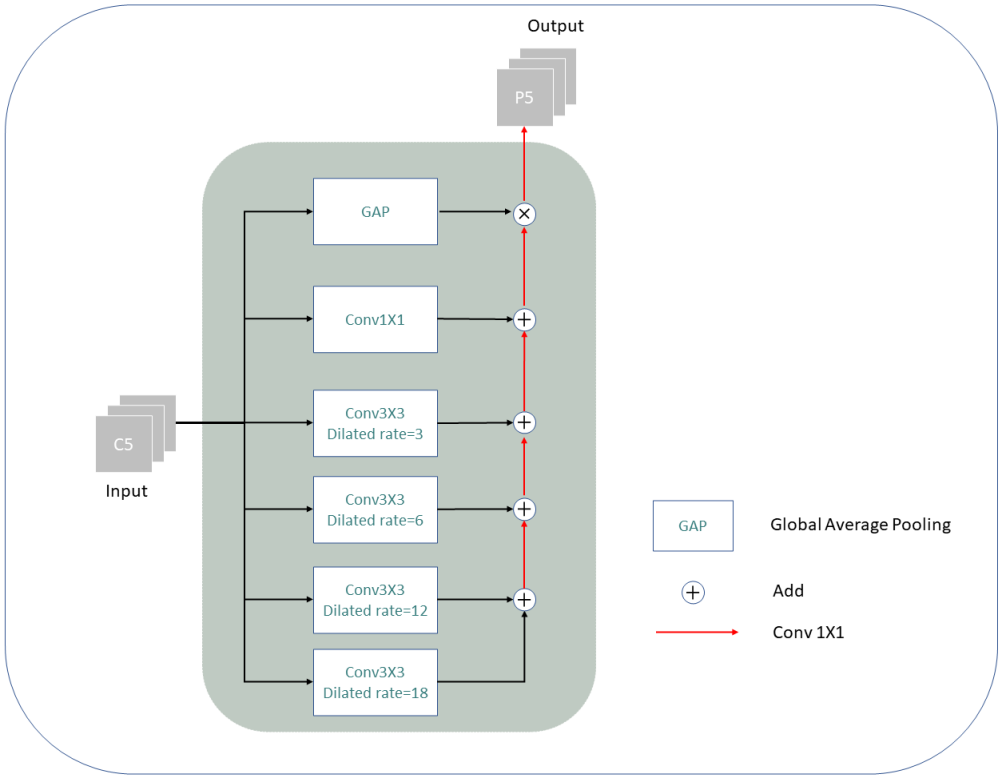

Figure 4. The overall structure of the MSSC module.

3.3. RPN Head with Oriented Bounding Box

To detect oriented objects, we use the RPN with proposing rotating proposals as the first-stage detection head. In detail, we use five levels, P2–P6, which are generated by the FPN with MSSC to generate $a * r$ anchors for each point on each feature map level. Here, $a = \{32^2, 64^2, 128^2, 256^2, 512^2\}$ contains the number of anchor pixel areas at each level, and $r = \{1:2, 1:1, 2:1\}$ represents the number of anchor aspect ratios at each position. Each anchor is represented by a four-dimensional vector $\alpha = (\alpha_x, \alpha_y, \alpha_w, \alpha_h)$, where α_x, α_y is the center point coordinate of the anchor and α_w, α_h are the width and height, respectively, of the anchor.

Each feature output of the FPN is followed by the same detector head, which is used for anchor foreground and background prediction and offset regression. The RPN head consists of a parallel classification branch and regression branch structure. The classification branch proposes an object-ness score for each anchor; this approach is the same as that of Faster RCNN. The regression branch outputs offsets $\Delta = (\Delta x, \Delta y, \Delta w, \Delta h, \Delta\theta)$ based on the anchor for each proposal. To calculate the loss later, it is necessary to decode the offset to obtain the oriented proposal (Figure 5). Here is the decoding formula:

$$\begin{cases} x = \Delta x \cdot \alpha_w + \alpha_x \\ y = \Delta y \cdot \alpha_h + \alpha_y \\ w = \alpha_w \cdot e^{\Delta w} \\ h = \alpha_h \cdot e^{\Delta h} \\ \theta = \Delta\theta \end{cases} \tag{1}$$

where (x, y) is the proposed center point coordinate and w, h are the width and height, respectively, of the proposed oriented bounding box. $\theta \in [-90°, 0°)$ is the proposed deflection angle and indicates an acute or right angle between the bounding box and the x-axis (OpenCV definition).

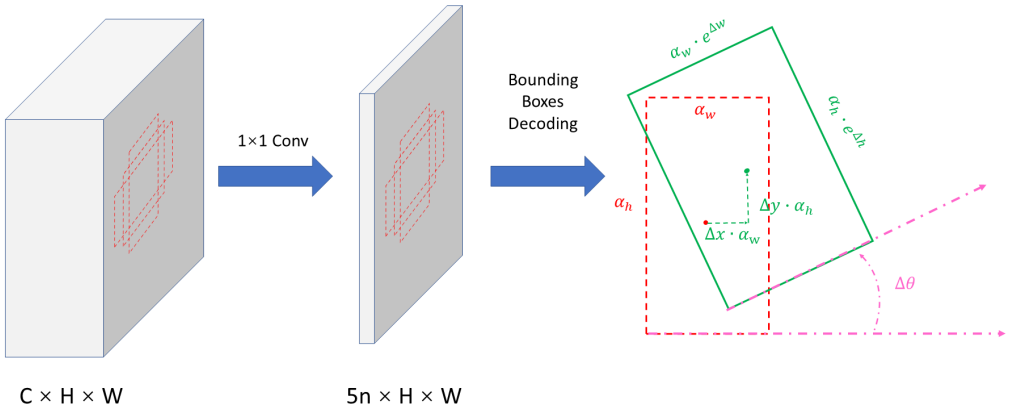

Figure 5. The Region Proposal with Oriented bounding boxes. The prediction of offsets is generated by 1×1 convolution, and n is the number of anchors generated for each position.

3.3.1. RPN Loss Function

To supervise the two prediction branches of the RPN, it is necessary to divide the proposal into positive and negative samples before calculating the loss. The sampling strategy used in this paper is similar to that used by Faster RCNN: (1) an anchor that obtains an IoU overlap higher than 0.7 with any smallest enclosing rectangle of ground-truth is set as a positive sample; (2) an anchor with the maximum IoU is used, and an IoU greater than 0.3 of the smallest ground-truth enclosing rectangle is set as a positive sample; (3) an anchor with an IoU overlap less than 0.3, and any smallest ground-truth enclosing rectangle is set as a negative sample; and (4) an anchor that is neither a positive sample nor a negative sample is set as invalid sampling and does not participate in training.

IoU loss [25] has achieved good results in many object detection tasks, such as those performed by Faster RCNN, FCOS, SCRDet, and YoloV4. However, in the oriented object detection task, IoU loss is not a good measure of the overlap of two rotating bounding boxes. For an object with a large aspect ratio, the angular offset results in a small IoU, as shown in Figure 6. The figure shows two certain angle overlapping bounding boxes. The larger the aspect ratio, the thinner and longer the bounding boxes, and the smaller the IoU, and the decrease rate is particularly obvious when the aspect ratio is in the range of 1 to 4. The larger the overlap angle, the faster the decrease. Additionally, the $L1$ loss also has some shortcomings in the oriented object detection task; they include (1) inconsistency with the measurement and loss function, (2) square-like problems, and (3) boundary discontinuity detection problems [26]. To minimize the influence caused by the above, the Wasserstein distance is adopted as a measure of the similarity between the two oriented bounding boxes, and the Gaussian Wasserstein distance loss (GWD loss) [26] is adopted as the regression loss function of the RPN.

GWD loss, a measurement proposed by Yang et al., reasonably describes the difference between two rotation boxes. GWD loss uses a two-dimensional Gaussian distribution $\mathcal{N}(\mathbf{m}, \mathbf{\Sigma})$ to describe a rotation bounding box and proposes a new regression loss based on the Wasserstein distance to replace the IoU loss. The conversion formula is as follows:

$$\begin{aligned} \mathbf{\Sigma}^{1/2} &= \mathbf{R}\mathbf{S}\mathbf{R}^{\top} \\ &= \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} \frac{w}{2} & 0 \\ 0 & \frac{h}{2} \end{pmatrix} \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix} \\ &= \begin{pmatrix} \frac{w}{2}\cos^2\theta + \frac{h}{2}\sin^2\theta & \frac{w-h}{2}\cos\theta\sin\theta \\ \frac{w-h}{2}\cos\theta\sin\theta & \frac{w}{2}\sin^2\theta + \frac{h}{2}\cos^2\theta \end{pmatrix} \\ \mathbf{m} &= (x, y) \end{aligned} \tag{2}$$

where **R** is the rotation matrix and **S** is the diagonal matrix of the eigenvalues. The Wasserstein distance between Gaussian distributions **W** can be expressed as:

$$\begin{aligned} W_2^2(X,Y) &= \|m_1 - m_2\|_2^2 + \mathrm{tr}\left[\Sigma_1 + \Sigma_2 - 2\left(\Sigma_1^{1/2}\Sigma_2\Sigma_1^{1/2}\right)^{1/2}\right] \\ &= \|m_1 - m_2\|_2^2 + \left\|\Sigma_1^{1/2} - \Sigma_2^{1/2}\right\|_F^2 \\ &= (x_1 - x_2)^2 + (y_1 - y_2)^2 + \frac{(w_1 - w_2)^2 + (h_1 - h_2)^2}{4} \\ &= l_2\text{-norm}\left(\left[x_1, y_1, \frac{w_1}{2}, \frac{h_1}{2}\right]^\top, \left[x_2, y_2, \frac{w_2}{2}, \frac{h_2}{2}\right]^\top\right) \end{aligned} \tag{3}$$

Therefore, the loss function of the RPN is as follows:

$$L1 = \lambda_1 \frac{1}{N}\sum_{i=1}^{N} F_{cls}(p_i, p_{i^*}) + \lambda_2 \frac{1}{N}[\![p_{i^*}]\!]\sum_{i=1}^{N} F_{reg}(t_i, t_{i^*}) \tag{4}$$

$$F_{reg}(b, \hat{b}) = 1 - \frac{1}{\tau + \mathcal{F}(D(b, \hat{b}))}, \tau \geq 1 \tag{5}$$

where λ_1 and λ_2 are the weights of the loss, N is the total number of anchors in a mini-batch, i is the anchor index, p_i is the classification prediction output of the RPN, p_{i^*} is the ground-truth label of the i-th anchor, t_i is the Gaussian distribution by 2D Gaussian transformation of the decoded five-dimensional vector of the i-th positive anchor, t_{i^*} is the Gaussian distribution for the corresponding ground-truth bounding box, F_{cls} is the cross-entropy loss, and F_{reg} is the GWD loss. In Formula (5), τ is a super parameter used to adjust the loss, $D(\cdot)$ is the Wasserstein distance calculation function of two Gaussian distributions, and $\mathcal{F}(\cdot)$ is a nonlinear function used to smooth the Wasserstein distance.

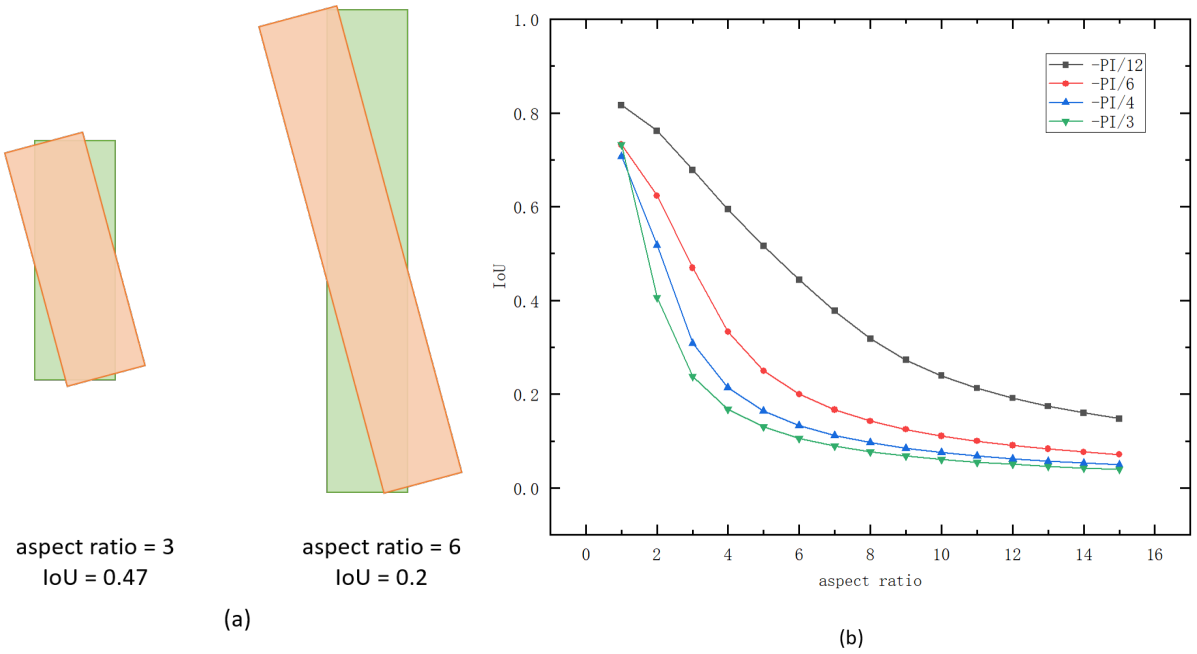

Figure 6. (**a**) indicates the IoU of two bounding boxes with different aspect ratios at an angle of 15°. (**b**) shows the trend of the IoU of the oriented bounding boxes overlapping at different angles as the aspect ratio increases.

3.4. R-CNN Head

For the second stage of the detector, the input consists of $\{P2, P3, P4, P5, P6\}$ provided by the FPN, and a group of RoIs provided by the RPN. For the convenience of the R-CNN head, to facilitate further reasoning and make full use of the FPN information of the different levels, we use oriented RoIAlign to extract fixed-size feature vectors from each layer of the FPN for each RoI, instead of allocating each RoI to a specific layer of the FPN in advance according to the size of the RoI. Then, the feature vectors are fed into the ARA module for adaptive weighted feature fusion. Finally, the fused features are input into two parallel fully connected layers to output the classification score and re-finetune the offset.

3.4.1. Oriented RoIAlign

The oriented proposal generated by the RPN is a five-dimensional vector (x, y, w, h, θ), where $\theta \in [-90^\circ, 0^\circ)$. We mapped the coordinates of the RoIs to the feature map with the stride of $s = [4, 8, 16, 32]$ to extract a fixed-size feature vector. Then, each RoI mapped to the FPN levels was divided into a 7×7 grid, and the feature value in each grid was obtained through bilinear interpolation with a sampling rate of four. This process is the same as RRoI warping [27].

3.4.2. Adaptive RoI Assignment

In the original Faster R-CNN with an FPN, the RoIs generated by the RPN are allocated to different layers of the FPN according to size. Then, fixed-size feature vectors will be extracted by oriented RoIAlign for second-stage reasoning. Typically, larger RoIs are assigned to deeper levels, while smaller RoIs are assigned to shallower levels, but this allocation strategy does not take full advantage of the different levels of information. Therefore, devising a more rational allocation strategy is a natural idea. Specifically, feature vectors from different levels extracted by RoIAlign are first concatenated; then, the weight maps are generated through the ARA module; and the RoI feature vectors of different levels are weighted and fused. The structure of the ARA is shown in Figure 7. The concatenated vectors pass through a global maximum pooling layer, and then two 1×1 convolution layers are used to generate the spatial weight maps.

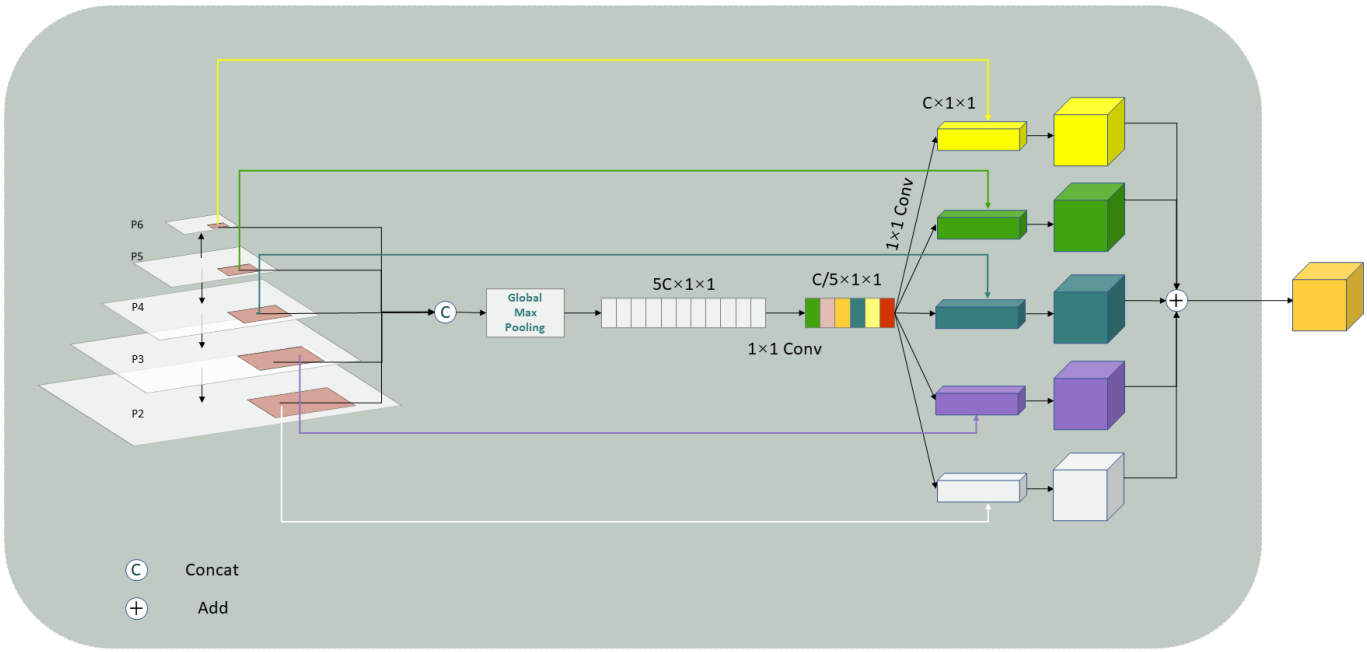

Figure 7. The overall structure of the Adaptive RoI Assignment module. It calculates the weights of each layer of the FPN and fuses the layers to obtain the subsequent features.

4. Experiments

4.1. Datasets and Evaluation Metrics

We test the performance of our proposed method on two common public remote sensing datasets, namely, DOTA [28] and HRSC2016 [29]. All reported results follow standard PASCAL VOC 2007 [30] mean average precision (mAP) metrics.

DOTA is a large-scale public dataset commonly used in remote sensing object detection tasks. It includes 2806 aerial images with sizes ranging from 800 × 800 to 4000 × 4000 pixels. The label contains 188,282 instances of 15 common ground target categories, each marked by its bounding vertices in clockwise order. There are two kinds of detection tasks for the DOTA dataset: detection with oriented bounding boxes (OBB) and detection with horizontal bounding boxes (HBB).

HRSC2016 is a public dataset for oriented ship detection from the Northwest University of Technology. The dataset contains 1061 aerial images with sizes ranging from 300 × 300 to 1500 × 900 pixels. Its annotations are divided into 3 categories and 27 subcategories, with 2976 targets, and use the OBB annotation format.

4.2. Parameter Detail

The experiments of this paper are based on the MMDetection platform [31]. All experiments were finished on a single RTX 3080 with the batch size set to two. We optimize the overall network by using the SGD algorithm with a momentum of 0.9 and a weight decay of 0.0001. We use ResNet50 as our backbone, and it is pretrained on ImageNet [32]. The weight of the RPN loss is set as $\lambda_1 = 1$ and $\lambda_2 = 5$. For DOTA, we used a sliding window with a size of 1024 × 1024 pixels and an overlapping width of 200 pixels for clipping. Twelve epochs were trained in total, the learning rate was set to 0.005, and it was reduced by a ratio of 0.1 after the 8th and 11th epochs. For HRSC2016, we scaled the image to 1333 × 800 before it was sent to the network, and we trained 36 epochs in total. The learning rate was set to 0.005, and it was reduced by a ratio of 0.1 after the 24-th and 33-th epochs.

4.3. Main Results

We compared the proposed method with 17 other oriented object detectors in the DOTA dataset while using the Faster R-CNN network with the rotation bounding boxes' RoI as the baseline, and the results are shown in Table 1. When ResNet-50 is used as the backbone, our method achieves 75.17% mAP, which is 1.38% mAP higher than that obtained by the baseline, and which is superior to that obtained by other detectors. The detection results visualization on DOTA are presented in Figure 8 and the results are presented in Table 1. The table clearly shows that compared with the baseline, the detection accuracies of BD, GTF, SV, RA, and HC have been significantly improved for the following reasons: (1) with the help of the multilevel stacked context module, the recognition accuracies of BD, GTF, and HC with obvious spatial relationship have improved; (2) under the combination of the MSSC and the ARA modules, the features generated by different levels of RPN are fully used to improve the detection accuracy of small targets such as SV; (3) RA shapes are similar to squares, and the introduction of the GWD loss alleviates square-like problems.

We compared the proposed method with 10 other oriented object detectors on HRSC2016, and the results are shown in Table 2. When ResNet-50 is used as the backbone, our method achieves 90.65% AP, which is 0.69% higher than that achieved by the baseline. Some results visualizations are shown in Figure 9, and a comparison of performances is presented in Table 2. This expectation is also consistent with the results of the first experiment; that is, multilevel context stacking helps to improve the network to obtain the characteristics of objects with obvious spatial context relationships. This finding is verified on ships, harbors, and large vehicles, with evidence taken from two datasets.

Table 1. Comparison with state-of-the-art methods on the DOTA dataset. The acronyms for category names are as follows: PL (plane), BD (baseball diamond), BR (Bridge), GTF (Ground field track), SV (Small vehicle), LV (Large vehicle), SH (Ship), TC (Tennis court), BC (Basketball court), ST (Storage tank), SBF (Soccer-ball field), RA (Roundabout), HA (Harbor), SP (Swimming pool), and HC (Helicopter).

	Method	Backbone	Input_size	PL	BD	BR	GTF	SV	LV	SH	TC	BC	ST	SBF	RA	HA	SP	HC	mAP
One-stage	PIoU [33]	DLA-34	512 × 512	80.90	69.70	24.10	60.20	38.30	64.40	64.80	**90.90**	77.20	70.40	46.50	37.10	57.10	61.90	64.00	60.5
	RetinaNet	R-50-FPN	512 × 512	88.67	77.62	41.81	58.17	74.58	71.64	79.11	90.29	82.18	74.32	54.75	60.60	62.57	69.67	60.64	68.43
	DAL [34]	R-50-FPN	800 × 800	88.68	76.55	45.08	66.80	67.00	76.76	79.74	90.84	79.54	78.45	57.71	62.27	69.05	**73.14**	60.11	71.44
	RSDet [35]	R-101-FPN	800 × 800	89.80	82.90	48.60	65.20	69.50	70.10	70.20	90.50	85.60	83.40	62.50	63.90	65.60	67.20	**68.00**	72.2
	P-RSDet [36]	R-101	800 × 800	88.58	77.83	50.44	69.29	71.10	75.79	78.66	90.88	80.10	81.71	57.92	63.03	66.30	69.77	63.13	72.3
	DRN [37]	H-104	1024 × 1024	89.71	82.34	47.22	64.10	76.22	74.43	85.84	90.57	86.18	84.89	57.65	61.93	69.30	69.63	58.48	73.23
	CFC-Net [38]	R-101	800 × 800	89.08	80.41	52.41	70.02	76.28	78.11	87.21	90.89	84.47	85.64	60.51	61.52	67.82	68.02	50.09	73.5
	R3Det	R-101-FPN	800 × 800	88.76	83.09	50.91	67.27	76.23	80.39	86.72	90.78	84.68	83.24	61.98	61.35	66.91	70.63	53.94	73.79
	S^2ANet [39]	R-50-FPN	1024 × 1024	89.11	82.84	48.37	71.11	78.11	78.39	87.25	90.83	84.90	85.64	60.36	62.60	65.26	69.13	57.94	74.12
Two-stage	Faster R-CNN	R-50-FPN	1024 × 1024	88.44	73.06	44.86	59.09	73.25	71.49	77.11	90.84	78.94	83.90	48.59	62.95	62.18	64.91	56.18	69.05
	RoI Transformer	R-101-FPN	1024 × 1024	88.64	78.52	43.44	75.92	68.81	73.68	83.59	90.74	77.27	81.46	58.39	53.54	62.83	58.93	47.67	69.56
	CAD-Net	R-101-FPN	1600 × 1600	87.80	82.40	49.40	73.50	71.10	63.50	76.60	90.90	79.20	73.30	48.40	60.90	62.00	67.00	62.20	69.9
	CenterMap-Net [40]	R-50-FPN	1024 × 1024	88.88	81.24	**53.15**	60.65	**78.62**	66.55	78.10	88.83	77.80	83.61	49.36	66.19	72.10	72.36	58.70	71.74
	SCRDet	R-101-FPN	800 × 800	89.98	80.65	52.09	68.36	68.36	60.32	72.41	90.85	**87.94**	86.86	65.02	66.68	66.25	68.24	65.21	72.61
	FAOD [41]	R-101-FPN	1024 × 1024	**90.21**	79.58	45.49	76.41	73.18	68.27	79.56	90.83	83.40	84.68	53.40	65.42	74.17	69.69	64.86	73.28
	Mask OBB [42]	R-50-FPN	1024 × 1024	89.61	**85.09**	51.85	72.90	75.28	73.23	85.57	90.37	82.08	85.05	55.73	68.39	71.61	69.87	66.33	74.86
	Gliding Vertex	R-101-FPN	1024 × 1024	89.64	85.00	52.26	**77.34**	73.01	73.14	86.82	90.74	79.02	86.81	59.55	**70.91**	72.94	70.86	57.32	75.02
	baseline	R-50-FPN	1024 × 1024	89.95	76.80	50.44	70.84	66.94	**84.40**	**88.74**	90.80	69.87	87.44	**77.42**	67.98	74.89	62.27	48.01	73.79
	MSCNet(ours)	R-50-FPN	1024 × 1024	89.81	79.92	48.62	74.18	68.44	84.28	88.4	90.79	72.49	**87.54**	75.44	69.43	**75.31**	60.07	62.85	**75.17**

Figure 8. Visualization of detection results obtained by MSCNet on the DOTA dataset. The confidence threshold is set to 0.3 in visualization. Our method can perform well despite different targets with different scales and directions.

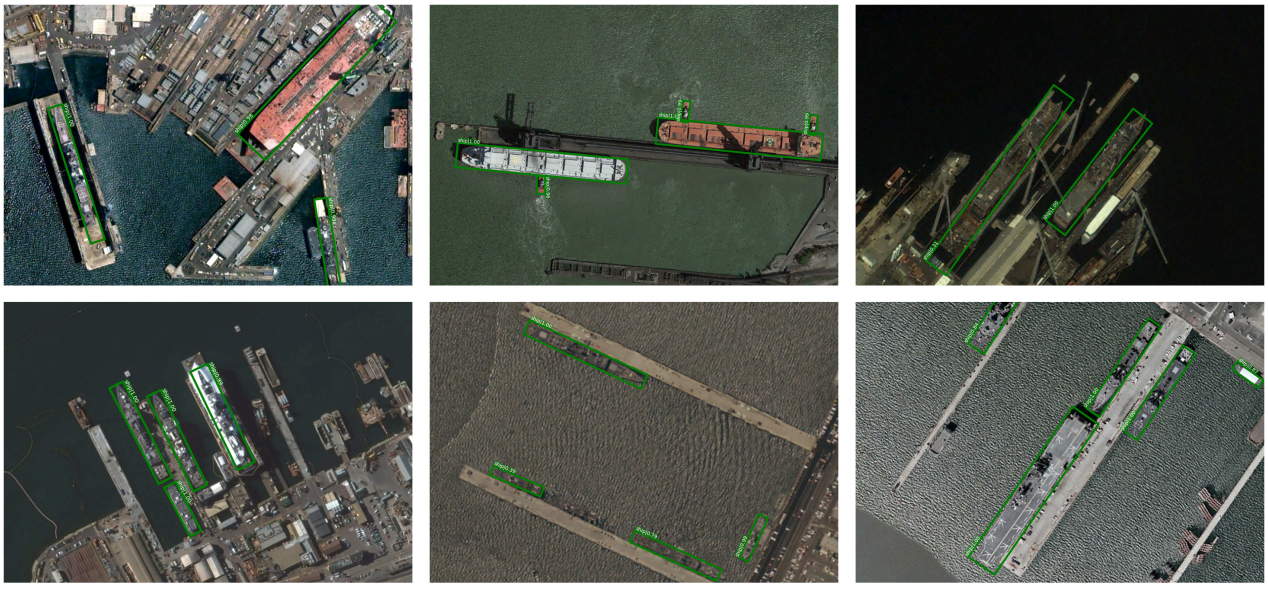

Figure 9. Visualization of detection results obtained by MSCNet on the HRSC2016 dataset.

Table 2. Comparison of network performances on the HRSC2016 dataset.

Model	Backbone	Input_Size	AP_{50}
R2CNN [43]	ResNet101	800 × 800	73.07
RRPN [44]	ResNet101	800 × 800	79.08
RoI-Transformer	ResNet101	512 × 800	86.2
Gliding Vertex	ResNet101	512 × 800	88.2
DAL	ResNet101	416 × 416	88.95
R3Det	ResNet101	800 × 800	89.26
DCL [45]	ResNet101	800 × 800	89.46
CSL	ResNet50	800 × 800	89.62
GWD	ResNet101	800 × 800	89.85
Oriented R-CNN	ResNet101	1333 × 800	90.5
baseline	ResNet50	1333 × 800	89.96
MSCNet(ours)	ResNet50	1333 × 800	**90.65**

4.4. Ablation Study

4.4.1. Baseline Setup

To evaluate the individual effects of the different modules, this section takes Faster R-CNN with oriented RoI head and SmoothL1 loss as the baseline and verifies each module on two datasets. The result of the ablation study is shown in Table 3.

Table 3. The respective contributions of the proposed module to the network. MSSC: multilevel stacked semantic capture, GWD: Gaussian Wasserstein distance loss, and ARA: Adaptive RoI Assignment.

Modules			mAP	AP50
MSSC	GWD	ARA	DOTA	HRSC
			73.79	89.96
✓			74.27	90.36
	✓		74.22	90.21
		✓	74.81	90.18
✓	✓		75.12	90.38
✓		✓	74.4	90.38
	✓	✓	74.94	90.47
✓	✓	✓	75.17	90.65

4.4.2. Effect on Each Module

Effect on MSSC: Due to the complexity of remote sensing images, noise information will be introduced in the feature extraction stage, and it is difficult to extract high-quality target features when the target information is mixed with the noise, leading to a decrease in detection ability. In the MSSC module, a top-down fusion path is added to the FPN structure, and a stack semantic capture structure is used to improve the ability to capture multiscale targets in complex scenes. MSSC module improves the baseline performance by 0.48% mAP on DOTA and 0.4% AP on HRSC2016. Because of the aggregation of stacked multilevel semantics, the FPN makes full use of multilevel semantic relations, and experiments also show that this approach is effective. The response of the network to different targets in the scene with and without the MSSC module is shown in Figure 10. It can be concluded that after using the MSSC module, the response of the network to the background-independent noise is suppressed and the response to the target is enhanced.

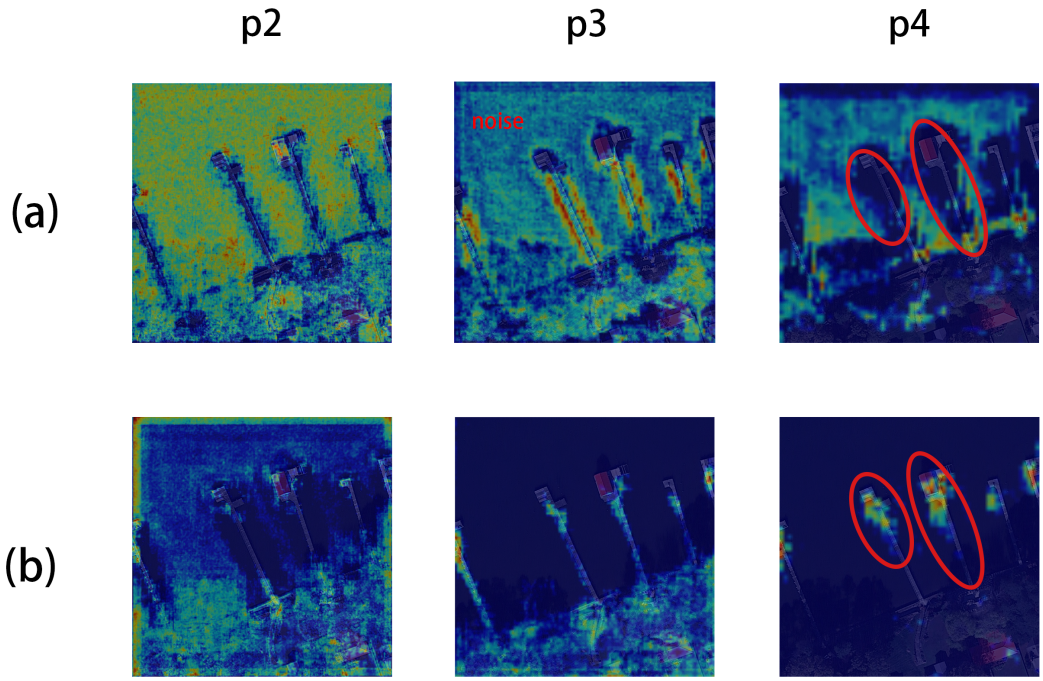

Figure 10. Visualization of the p2-p4 layer outputs of the FPN with and without the MSSC module. (**a**) MSSC is not used. (**b**) MSSC is used. Due to the addition of MSSC, the background noise unrelated to the detection target is suppressed, and the target is enhanced in (**b**).

Effect on GWD: Compared with the traditional five-parameter bounding box representation, the Gaussian distribution representation makes up for some defects of the five-parameter representation. On this basis, a reasonable measurement method is used to describe the similarity between bounding boxes with different aspect ratios and different rotation angles, and the method compensated for the effect between the measurement and reality in oriented object detection tasks. Additionally, the experimental results show that Gaussian distribution representation positively affects the detection task. Using the Gaussian distribution to represent the oriented bounding box and using the Gaussian Wasserstein distance as an overlap metric of two oriented bounding boxes improves the baseline by 0.43% mAP on DOTA and 0.25% AP on HRSC2016. Compared with the performance of HRSC2016, the performance of this method on DOTA is more obvious because DOTA has a variety of objects and many square-like objects; consequently, traditional representation methods are more likely to encounter a bottleneck problem. It can be seen from the ablation experiment in Table 3, after replacing SmoothL1 loss in baseline with GWD loss, the model performance improves by 0.43% mAP on DOTA and by 0.25% AP on HRSC. These experimental results shows that GWD loss has better performance than SmoothL1 loss in oriented object detection tasks to some extent, and the former can better describe the difference between two oriented objects. Figure 11 shows the model convergence ratio and accuracy of GWD loss and SmoothL1 loss when trained on DOTA. It can be seen that at about 750 iterations, the model with SmoothL1 loss showed an oscillation. Compared with the former, the model with GWD loss had smoother convergence and higher accuracy.

Effect on ARA: Adaptive RoI assignment aggregates information from multilevels of the FPN by adaptive weighting, which is especially effective for targets with large-scale variations, especially in DOTA. It improves the baseline performance by 1.02% mAP on DOTA and 0.22% AP on HRSC2016.

Any combination of the two modules can outperform the single module. When all three modules are used in combination, we obtain the highest detection accuracy on both datasets, with improvements of 1.38% mAP and 0.69% AP compared to the mAP and AP obtained by the baseline on DOTA and HRSC2016, respectively. In addition, the inference speed and complexity of MSSCNet are given in Table 4.

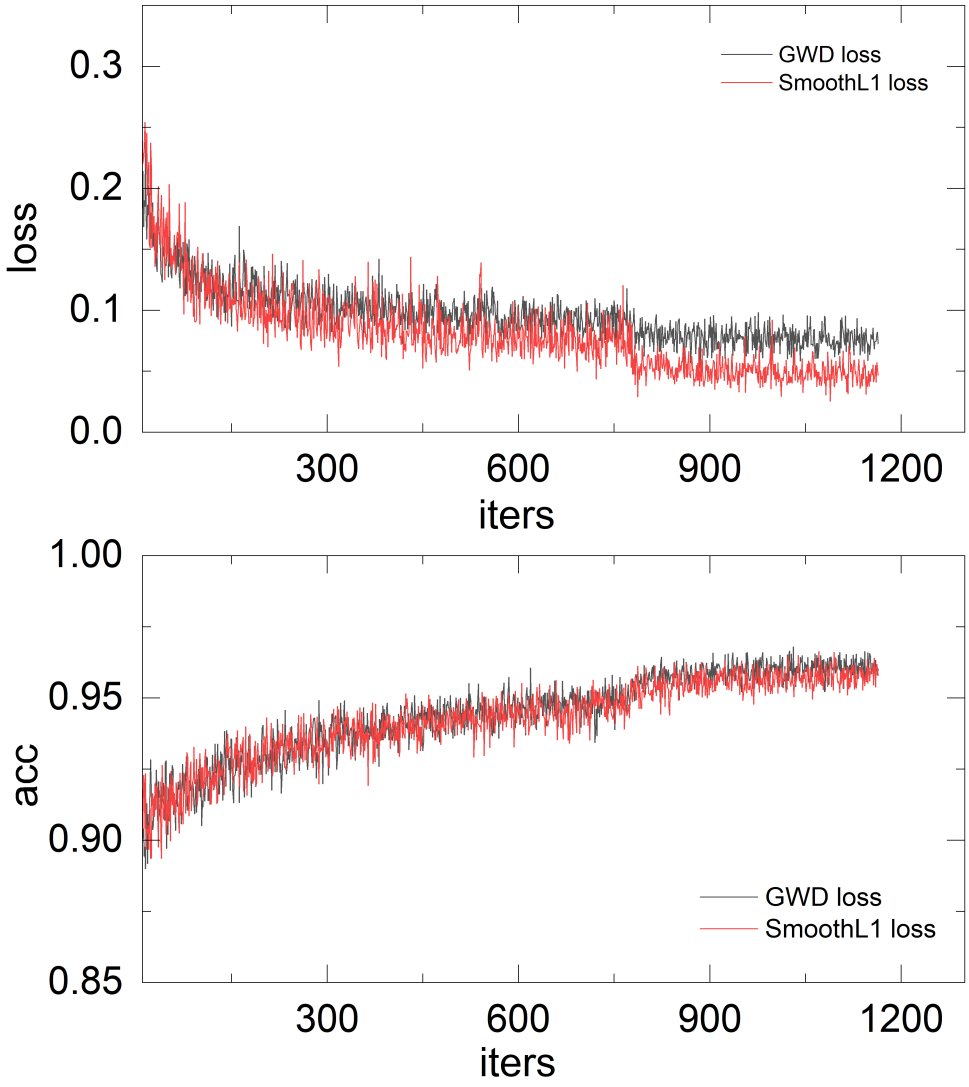

Figure 11. Comparison of model training convergence ratio and accuracy using GWD loss and SmoothL1 loss on DOTA.

Table 4. Comparison of model inference speed and Flops metrics.

	Input Shape	Backbone	Flops	FPS
baseline	3 × 1024 × 1024	ResNet50	211.3 GFLOPs	23 img/s
MSSCNet	3 × 1024 × 1024	ResNet50	503.6 GFLOPs	10.6 img/s

5. Conclusions

In this paper, an arbitrarily oriented object detector is proposed according to the characteristics of remote sensing images. Aiming at the deficiency of traditional object detectors in remote sensing, we enhance the modeling ability of the network for the relationship between the objects at different distances by using multilayer stack structures, make full use of the RoI feature of different levels by using a multichannel weighted fusion structure and use a reasonable representation method to represent the oriented object. Moreover, experimental results show that the proposed method outperforms the baseline network on two remote sensing public datasets, and the method performs the best in the relevant oriented object detection networks. Of course, there is still some room for improvement in this work. For example, in the ablation study, when MSSC and ARA were used together, the result obtained on the DOTA dataset was lower than that obtained when ARA was used alone. Additionally, on the DOTA dataset, the proposed model is ineffective

in some categories, such as BR and SP. In future work, it is necessary to further explore why these problems occur and find a better combination of MSSC and ARA to create a more effective multilevel feature allocation method.

Author Contributions: R.Z. wrote the manuscript, designed the comparative experiments, designed the architecture and performed the comparative experiments; X.Z. and D.W. supervised the study and revised the manuscript; Y.Z. and L.H. revised the manuscript and gave comments and suggestions to the manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China (62103345), the Science and Technology Planning Project of Fujian Province (2020H0023, 2020J02160, 2020J01265), the Science and Technology Planning Project of Quanzhou (2020C074), and the Science and Technology Climbing Program of Xiamen University of Technology (XPDKT18030).

Data Availability Statement: Not applicable.

Acknowledgments: We would like to thank the anonymous reviewers for their constructive and valuable suggestions on the earlier drafts of this manuscript.

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

Abbreviations

ARA	Adaptive RoI Assignment
FPN	Feature Pyramid Network
GWD	Gaussian Wasserstein Distance
IoU	Intersection over Union
mAP	mean Average Precision
MSCNet	Multilevel Stacked Context Network
MSSC	Multilevel Stacked Semantic Capture
NMS	Non-Maximum Suppression
RoI	Region of Interest
RPN	Region Proposal Network
SGD	Stochastic Gradient Descent

References

1. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *arXiv* **2015**, arXiv:1506.01497.
2. Li, K.; Cheng, G.; Bu, S.; You, X. Rotation-insensitive and context-augmented object detection in remote sensing images. *IEEE Trans. Geosci. Remote Sens.* **2017**, *56*, 2337–2348. [CrossRef]
3. Liu, W.; Ma, L.; Wang, J. Detection of multiclass objects in optical remote sensing images. *IEEE Geosci. Remote Sens. Lett.* **2018**, *16*, 791–795. [CrossRef]
4. Lin, T.Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2117–2125.
5. Liu, L.; Pan, Z.; Lei, B. Learning a rotation invariant detector with rotatable bounding box. *arXiv* **2017**, arXiv:1711.09405.
6. Zhang, G.; Lu, S.; Zhang, W. CAD-Net: A context-aware detection network for objects in remote sensing imagery. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 10015–10024. [CrossRef]
7. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
8. Girshick, R. Fast r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
9. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. Ssd: Single shot multibox detector. In Proceedings of the European Conference on Computer Vision, Amesterdam, The Netherlands, 8 October 2016; Springer: Cham, Switzerland, 2016; pp. 21–37.
10. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2980–2988.
11. Lin, G.; Milan, A.; Shen, C.; Reid, I. Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1925–1934.

12. Redmon, J.; Farhadi, A. YOLO9000: better, faster, stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7263–7271.
13. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
14. Bochkovskiy, A.; Wang, C.Y.; Liao, H.Y.M. Yolov4: Optimal speed and accuracy of object detection. *arXiv* **2020**, arXiv:2004.10934.
15. Law, H.; Deng, J. Cornernet: Detecting objects as paired keypoints. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 734–750.
16. Zhou, X.; Zhuo, J.; Krahenbuhl, P. Bottom-up object detection by grouping extreme and center points. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 850–859.
17. Tian, Z.; Shen, C.; Chen, H.; He, T. Fcos: Fully convolutional one-stage object detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 9627–9636.
18. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.
19. Duan, K.; Bai, S.; Xie, L.; Qi, H.; Huang, Q.; Tian, Q. Centernet: Keypoint triplets for object detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 6569–6578.
20. Yang, X.; Yang, J.; Yan, J.; Zhang, Y.; Zhang, T.; Guo, Z.; Sun, X.; Fu, K. Scrdet: Towards more robust detection for small, cluttered and rotated objects. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 8232–8241.
21. Yang, X.; Liu, Q.; Yan, J.; Li, A.; Zhang, Z.; Yu, G. R3det: Refined single-stage detector with feature refinement for rotating object. *arXiv* **2019**, arXiv:1908.05612.
22. Xu, Y.; Fu, M.; Wang, Q.; Wang, Y.; Chen, K.; Xia, G.S.; Bai, X. Gliding vertex on the horizontal bounding box for multi-oriented object detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *43*, 1452–1459. [CrossRef]
23. Xie, X.; Cheng, G.; Wang, J.; Yao, X.; Han, J. Oriented r-cnn for object detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 3520–3529.
24. Yang, X.; Yan, J. Arbitrary-oriented object detection with circular smooth label. In Proceedings of the sEuropean Conference on Computer Vision, Online, 28 August 2020; Springer: Cham, Switzerland, 2020; pp. 677–694.
25. Yu, J.; Jiang, Y.; Wang, Z.; Cao, Z.; Huang, T. Unitbox: An advanced object detection network. In Proceedings of the 24th ACM International Conference on Multimedia, Amsterdam, The Netherlands, 15–19 October 2016; pp. 516–520.
26. Yang, X.; Yan, J.; Ming, Q.; Wang, W.; Zhang, X.; Tian, Q. Rethinking rotated object detection with gaussian wasserstein distance loss. In Proceedings of the International Conference on Machine Learning, Online, 18–24 July 2021; pp. 11830–11841.
27. Ding, J.; Xue, N.; Long, Y.; Xia, G.S.; Lu, Q. Learning roi transformer for oriented object detection in aerial images. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 2849–2858.
28. Xia, G.S.; Bai, X.; Ding, J.; Zhu, Z.; Belongie, S.; Luo, J.; Datcu, M.; Pelillo, M.; Zhang, L. DOTA: A large-scale dataset for object detection in aerial images. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 3974–3983.
29. Liu, Z.; Yuan, L.; Weng, L.; Yang, Y. A high resolution optical satellite image dataset for ship recognition and some new baselines. In International conference on pattern recognition applications and methods. *SciTePress* **2017**, *2*, 324–331.
30. Everingham, M.; Van Gool, L.; Williams, C.K.; Winn, J.; Zisserman, A. The pascal visual object classes (voc) challenge. *Int. J. Comput. Vis.* **2010**, *88*, 303–338. [CrossRef]
31. Chen, K.; Wang, J.; Pang, J.; Cao, Y.; Xiong, Y.; Li, X.; Sun, S.; Feng, W.; Liu, Z.; Xu, J.; et al. MMDetection: Open mmlab detection toolbox and benchmark. *arXiv* **2019**, arXiv:1906.07155.
32. Deng, J. A large-scale hierarchical image database. In Proceedings of the IEEE Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009.
33. Chen, Z.; Chen, K.; Lin, W.; See, J.; Yu, H.; Ke, Y.; Yang, C. Piou loss: Towards accurate oriented object detection in complex environments. In Proceedings of the European Conference on Computer Vision, Online, 28 August 2020; Springer: Cham, Switzerland, 2020; pp. 195–211.
34. Ming, Q.; Zhou, Z.; Miao, L.; Zhang, H.; Li, L. Dynamic anchor learning for arbitrary-oriented object detection. *arXiv* **2020**, arXiv:2012.04150.
35. Qian, W.; Yang, X.; Peng, S.; Guo, Y.; Yan, J. Learning modulated loss for rotated object detection. *arXiv* **2019**, arXiv:1911.08299.
36. Zhou, L.; Wei, H.; Li, H.; Zhao, W.; Zhang, Y. Objects detection for remote sensing images based on polar coordinates. *arXiv* **2020**, arXiv:2001.02988.
37. Pan, X.; Ren, Y.; Sheng, K.; Dong, W.; Yuan, H.; Guo, X.; Ma, C.; Xu, C. Dynamic refinement network for oriented and densely packed object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 11207–11216.
38. Ming, Q.; Miao, L.; Zhou, Z.; Dong, Y. CFC-Net: A critical feature capturing network for arbitrary-oriented object detection in remote-sensing images. *IEEE Trans. Geosci. Remote Sens.* **2021**, *61*, 1–4. [CrossRef]
39. Han, J.; Ding, J.; Li, J.; Xia, G.S. Align deep features for oriented object detection. *IEEE Trans. Geosci. Remote Sens.* **2021**, *60*, 1–11. [CrossRef]
40. Wang, J.; Yang, W.; Li, H.C.; Zhang, H.; Xia, G.S. Learning center probability map for detecting objects in aerial images. *IEEE Trans. Geosci. Remote Sens.* **2020**, *59*, 4307–4323. [CrossRef]

41. Li, C.; Xu, C.; Cui, Z.; Wang, D.; Zhang, T.; Yang, J. Feature-attentioned object detection in remote sensing imagery. In Proceedings of the 2019 IEEE International Conference on Image Processing (ICIP), Taipei, China, 22–25 September 2019; pp. 3886–3890.
42. Wang, J.; Ding, J.; Guo, H.; Cheng, W.; Pan, T.; Yang, W. Mask OBB: A semantic attention-based mask oriented bounding box representation for multi-category object detection in aerial images. *Remote Sens.* **2019**, *11*, 2930. [CrossRef]
43. Jiang, Y.; Zhu, X.; Wang, X.; Yang, S.; Li, W.; Wang, H.; Fu, P.; Luo, Z. R2CNN: Rotational region CNN for orientation robust scene text detection. *arXiv* **2017**, arXiv:1706.09579.
44. Ma, J.; Shao, W.; Ye, H.; Wang, L.; Wang, H.; Zheng, Y.; Xue, X. Arbitrary-oriented scene text detection via rotation proposals. *IEEE Trans. Multimed.* **2018**, *20*, 3111–3122. [CrossRef]
45. Yang, X.; Hou, L.; Zhou, Y.; Wang, W.; Yan, J. Dense label encoding for boundary discontinuity free rotation detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 15819–15829.