remote sensing

MDPI

*Article*

# HVConv: Horizontal and Vertical Convolution for Remote Sensing Object Detection

**Jinhui Chen [1], Qifeng Lin [1,*], Haibin Huang [1], Yuanlong Yu [1], Daoye Zhu [1] and Gang Fu [2]**

[1] College of Computer and Data Science, Fuzhou University, Fuzhou 350108, China; 221020023@fzu.edu.cn (J.C.); 221027161@fzu.edu.cn (H.H.); yuyuanlong@fzu.edu.cn (Y.Y.); zhudaoye@fzu.edu.cn (D.Z.)
[2] Department of Computing, The Hong Kong Polytechnic University, Hong Kong 999077, China; gangfu@polyu.edu.hk
[*] Correspondence: linqf@fzu.edu.cn

**Abstract:** Generally, the interesting objects in aerial images are completely different from objects in nature, and the remote sensing objects in particular tend to be more distinctive in aspect ratio. The existing convolutional networks have equal aspect ratios of the receptive fields, which leads to receptive fields either containing non-relevant information or being unable to fully cover the entire object. To this end, we propose Horizontal and Vertical Convolution, which is a plug-and-play module to address different aspect ratio problems. In our method, we introduce horizontal convolution and vertical convolution to expand the receptive fields in the horizontal and vertical directions, respectively, to reduce redundant receptive fields, so that remote sensing objects with different aspect ratios can achieve better receptive fields coverage, thereby achieving more accurate feature representation. In addition, we design an attention module to dynamically aggregate these two sub-modules to achieve more accurate feature coverage. Extensive experimental results on the DOTA and HRSC2016 datasets show that our HVConv achieves accuracy improvements in diverse detection architectures and obtains SOTA accuracy (mAP score of **77.60%** with DOTA single-scale training and mAP score of **81.07%** with DOTA multi-scale training). Various ablation studies were conducted as well, which is enough to verify the effectiveness of our model.

**Keywords:** object detection; irregular aspect ratio; redundancy receptive fields; backbone network

check for updates

**Citation:** Chen, J.; Lin, Q.; Huang, H.; Yu, Y.; Zhu, D.; Fu, G. HVConv: Horizontal and Vertical Convolution for Remote Sensing Object Detection. *Remote Sens.* **2024**, *16*, 1880. https://doi.org/10.3390/rs16111880

Academic Editor: Pedro Melo-Pinto

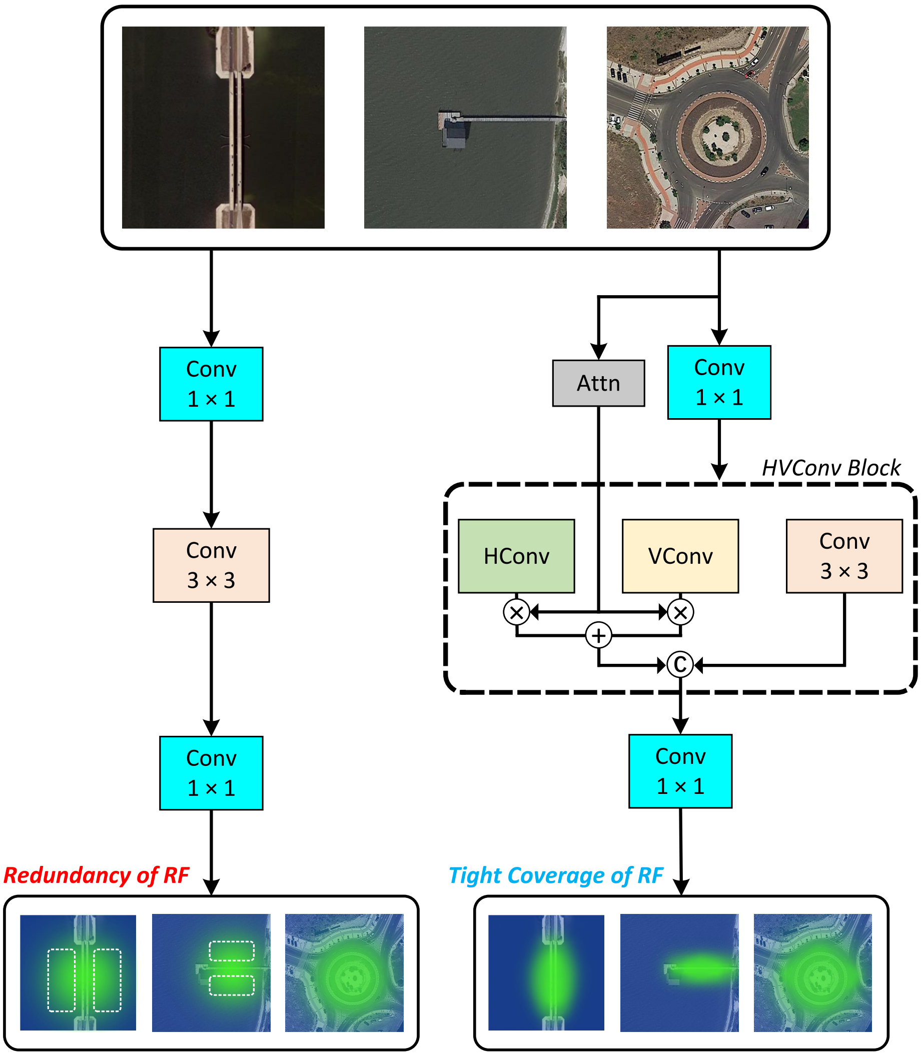
Received: 20 April 2024
Revised: 17 May 2024
Accepted: 20 May 2024
Published: 24 May 2024

## 1. Introduction

Remote sensing object detection, as an advancing field within computer vision, diverges from common object detection. Unlike typical object detection scenarios where images are captured from conventional viewpoints, remote sensing data are gathered from satellites or aerial platforms at elevated altitudes. Objects within remote sensing images, such as vehicles, ships, and planes, exhibit diverse orientations, presenting added complexity to detection tasks. Additionally, the irregular aspect ratios of these objects pose significant challenges. Given these distinct characteristics of remote sensing targets, the pursuit of effective object detection in this domain remains a formidable research endeavor.

Recently, considerable efforts have been dedicated to addressing the challenges presented by remote sensing imagery. Specifically, in tackling the issue of detecting rotated objects, various detection frameworks have emerged. Notable examples include S$^2$ANet [1] and R3Det [2], which align the features between the horizontal receptive fields and rotated anchors. DRN [3] can dynamically select and refine features to detect oriented objects. MEDNet [4] and MPME [5] introduce a multi-model to enhance the semantic ability of model and an LD-kEC strategy for non-labeled datasets training. All these methods further enhance the detection performance of the remote sensing object. However, there is currently almost no research focusing on the irregular aspect ratios problem.

In the backbone domain, ResNet [6], as a popular convolution backbone network, is widely used in object detection, including remote sensing scenarios. However, ResNet [6] relies on fixed 3 × 3 convolutions for feature extraction, resulting in uniform expansion of the receptive fields in both horizontal and vertical directions. Consequently, these receptive fields fail to accurately cover the remote sensing objects with irregular aspect ratios. Although backbone networks like LSKNet [7] extracting context information and Adaptive Rotated Convolution Network (ARCNet) [8] keeping rotation invariance are proposed for remote sensing detection, the irregular aspect ratios issue remains to be solved. Faced with remote sensing objects of varying aspect ratios, the inclusion of non-target regions in the conventional receptive fields coverage area is inevitable, impacting the feature representation for the regions of interest. In conclusion, the equal-ratio convolution results in the redundancy of the receptive fields, as indicated by the white-boxed area in Figure 1a, ultimately leading to the feature map affected by the irrelevant information.



(**a**) Regular Conv in ResNet (**b**) HVConv in ResNet

**Figure 1.** The illustration of receptive fields coverage in two different blocks from the same input. The input images are bridge (BR), harbor (HA), and roundabout (RA). HVConv block as a plug-and-play module in ResNet. **RF** stands for receptive fields. **Attn** is the attention module.

To address this issue, we proposed the *Horizontal and Vertical Convolution* (**HVConv**), as shown in Figure 1b. Compared to the original residual block, our approach reduces the redundancy of the receptive fields by shrinking the convolution kernel size in the horizontal or vertical direction, allowing for more precise object coverage. In our method, we incorporate two distinct convolution paths, *horizontal convolution* (**HConv**) and *vertical convolution* (**VConv**) for expanding the horizontal and vertical receptive fields, respectively.

Given that objects in remote sensing images exhibit varying orientations and significant changes in aspect ratio even with a 90-degree rotation, it is imperative to dynamically adjust the weights of the horizontal and vertical paths. To optimize the utilization of both horizontal and vertical convolution outputs, we draw inspiration from SENet [9] and integrate an attention module. This module facilitates the aggregation of feature maps from both paths by generating two weights for each, enhancing the model's ability to adapt to diverse object orientations and aspect ratios.

Overall, our contributions can be summarized as follows:

- A new convolution fashion, leveraging the horizontal and vertical convolution (HVConv), was proposed to reduce receptive fields redundancy, which accommodates object features with non-uniform aspect ratios of length and width for a higher precision of object detection coverage.
- The attention mechanism is cleverly coordinated with our HVConv to dynamically achieve the fusion of different receptive fields to adapt to various aspect ratios of remote sensing objects.
- HVConv is designed as a plug-and-play module for expanding horizontal and vertical receptive fields. It can be easily applied to different networks to improve detection capabilities.

As far as we know, our method is the first attempt to take the irregular receptive fields aspect ratio problem into consideration and remove redundant receptive fields to reduce the influence of irrelevant information, thereby enhancing the representation ability of the feature map. As a plug-in module, our module can be quickly applied to different networks to improve the model's detection capabilities. We integrated our module into different network structures. Extensive experiments were conducted on the HRSC2016 [10] and DOTA [11] datasets, achieving state-of-the-art results and thereby verifying the effectiveness of our work.

## 2. Related Work

### *2.1. Remote Sensing Object Detection*

Object detection has consistently been a popular and crucial task in computer vision, aiming to achieve high accuracy in recognizing various objects within different images. There are two approaches for object detection: one-stage methods [12,13] and two-stage methods [14,15]. The Faster R-CNN [14] with Feature Pyramid Network [16] is widely used as a two-stage method in common object detection. Nevertheless, remote sensing objects are smaller, denser, and positioned at various angles [17,18]. Moreover, different from the horizontal bounding box in common detection, remote sensing detection usually uses an oriented bounding box to capture objects more tightly. Thus, with the aim of solving the angle problem, more rotated object detection detectors are proposed based on existing general methods (e.g., RetinaNet [12] or Faster R-CNN [14]). Rotated RPN [19] proposed using angles for bounding box regression, resulting in a more precise rotation of the bounding boxes to cover the objects. RoI Transformer [20] introduces RRoI Learner for learning rotated RoIs from the feature map of horizontal RoIs and RRoI Warping to extract rotation-invariant features for detection. $S^2$ANet [1] mainly consists of two modules, the Feature Alignment Module (FAM) and Oriented Detection Module (ODM). By incorporating these two parts, the overall framework is enhanced to be more sensitive to rotation information. The pyramid squeeze has been adopted into $S^2$ANet to improve the network's ability to extract important information as well [21]. LSKNet [7], which is based on SKNet [22], used a larger kernel convolution to obtain the context information of objects. In accordance with context features, the accuracy of detection improved significantly. ARCNet [8] adopted adaptive rotated convolution to replace the normal 3 × 3 convolution. It is a plug-and-play module, and the aim of it is to facilitate feature extraction of the same object across various orientations, rendering it invariant, which enhances the capability of classification. Although these methods address issues from various angles, our strategy provides a whole new perspective and is totally different from theirs. Inception Network [23] introduced replacing $n \times n$ convolution with $1 \times n$ and $n \times n$ convolutions to reduce computational costs and achieve even accuracy. Inspired by this work, our HVConv aims to solve the aspect ratios problem in remote sensing object detection, as this approach can reduce redundancy receptive fields and expand it in an irregular aspect ratio by using narrow and asymmetric convolution, which could improve the network's ability to detect narrow as well as long objects.

### *2.2. Attention Module*

Recently, with the emergence of Vision Transformer (ViT) [24], an increasing number of computer vision tasks have embraced ViT, achieving remarkable success across various applications. The triumph of ViT can be attributed to the core component of the Transformer—the self-attention mechanism module. More variants of the ViT model improve the self-attention mechanism and patch encoding to deal with the tiny scale object detection issues. However, ViT models typically come with a large number of parameters and require extensive pretraining, incurring a substantial computational cost. Furthermore, Eupea [25] calculates Euclidean distances and Pearson coefficients between pixels to assess pixel correlations, making it an attention module. The emergence of SENet [9] has marked a significant advancement in CNNs. It incorporates an attention mechanism applied to CNNs, extracting informative features within local receptive fields by combining spatial and channel-level information. By weighting the channels, effective information is emphasized, and invalid information is suppressed. SKNet [22] further improved the channel-wise receptive fields attention of SENet [9] by using convolution at different scales.

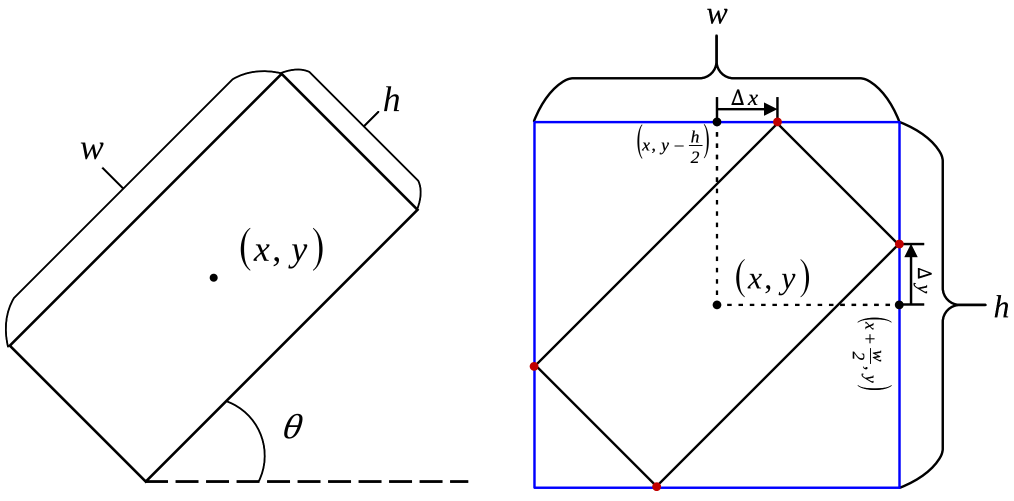
An object could be set in any angle for each image. Therefore, the same object in different scenes may sometimes lean horizontally and sometimes lean vertically. As our approach aims to combine the feature map in horizontal and vertical receptive fields, it is crucial for us to weigh these two parts. We take SENet [9] into account for generating weights for each path. SENet [9] introduced the SEblock to improve the quality of network-generated representations by explicitly modeling the interdependencies between feature channel evolution in the network. By this mechanism, it is able to recalibrate features on a channel-wise level, which means the network could learn the global information to stress feature information and suppress non-relevant information. The distinction between our method and SENet [9] lies in our method's use of it for horizontal and vertical attention rather than channel-wise attention.

## 3. Method

The aspect ratios of objects in remote sensing images are often inconsistent; therefore, we propose conducting multiple computations by using kernels with different aspect ratios. This approach can increase the depth of the network ensuring that the receptive fields expand more quickly in either the horizontal or the vertical direction. Our goal is to reduce redundant receptive fields while minimally increasing computational cost rather than decreasing the model size.

### *3.1. Overall Architecture*

We choose to use Oriented R-CNN [26] and combine it with our method. This is because Oriented R-CNN [26] is a significant method in oriented object detection. As Rotated RPN [19] sets 54 anchors for generating proposals, it increases the computational burden significantly. Although RoI Transformer [20] reduces the number of anchors, it also incurs expensive computational costs. Oriented R-CNN [26] sets anchors with three aspect ratios {1:2, 1:1, 2:1} for feature maps from each level. As a result, it is unnecessary to set anchors with various scales. The midpoint offset method is used in RPN to represent oriented bounding boxes. This method removes the angle parameter and introduces two offset values for the bounding box. It reduced the cost of the generation of proposals from various angles. The comparison of the general represent method and midpoint offset are shown in Figure 2 Additionally, the Oriented R-CNN head contains Rotated RoI Align to extract the rotation-invariant feature. After proposals are generated by RPN, the network converts possible diamond proposals into rectangles based on their longer diagonal. This generates rectangular-oriented RoIs. Consequently, for the next classification and regression computations in the Oriented R-CNN head, the rotated RoIs should be projected onto the horizontal feature maps. This whole procedure is the Rotated RoI Align. The midpoint offset notation for proposals in Oriented RPN and Rotated RoI Align before Oriented R-CNN head for classification and regression significantly improves the accuracy of remote sensing object detection. Accordingly, Oriented R-CNN [26] enables us to seamlessly integrate our modules into the backbone network by replacing the convolution operation, yielding excellent results. Validating the effectiveness of our modules becomes more straightforward. The overall architecture is shown in Figure 3. Our main focus is on the backbone network. We replace the 3 × 3 convolution of ResNet [6] with our own backbone network to achieve high performance and facilitate accessible experimental comparisons. Figure 4 illustrates our HVConv block architecture, and Figure 5 presents the structure of the Attention module. The block comprises three main components. To achieve improved object detection results in aerial images, HVConv serves as the core element, incorporating two distinct subparts, HConv and VConv, designed for expanding receptive fields in two directions and for the separate extraction of horizontal and vertical features. Another crucial component of HVConv is the Attention module, which assigns weights to the two paths of HVConv, ensuring emphasis on one of the paths. This is particularly important when dealing with objects in images captured from different angles. To aggregate additional information such as context background details from feature maps, we incorporate a standard convolution operating at a larger scale than HVConv to cover larger receptive fields. We combine the details from the horizontal and vertical aspects with those from the standard scale. This aids the network in learning features across diverse directions and scales.


$w$
$h$
$(x, y)$
$\theta$
$w$
$\Delta x$
$\left(x, y - \frac{h}{2}\right)$
$(x, y)$
$\Delta y$
$h$
$\left(x + \frac{w}{2}, y\right)$

(**a**) (**b**)

**Figure 2.** Comparison of two different bounding box representations. (**a**) Normal bounding box representation. Using five parameters, $x$, $y$, $w$, $h$ and $\theta$. (**b**) midpoint offset bounding box representation. Using six parameters, $x$, $y$, $w$, $h$ $\Delta x$ and $\Delta y$.

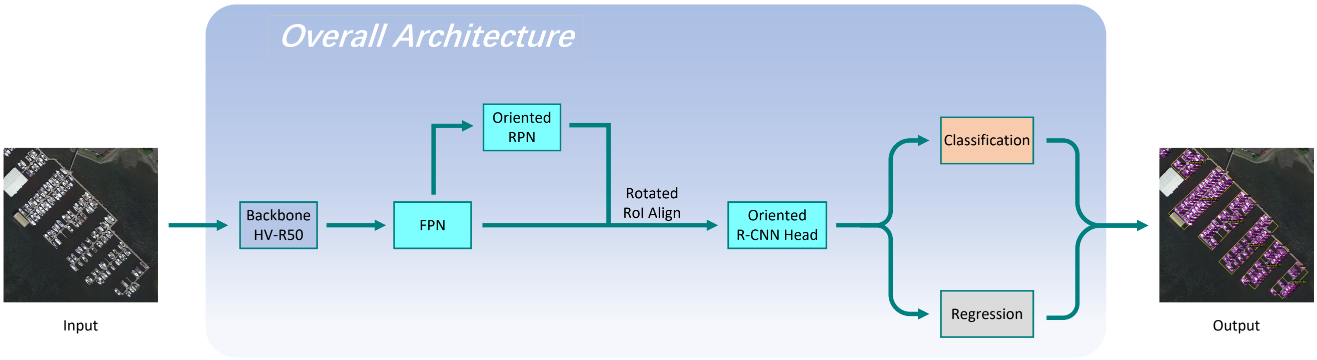


**Figure 3.** The overall network architecture.

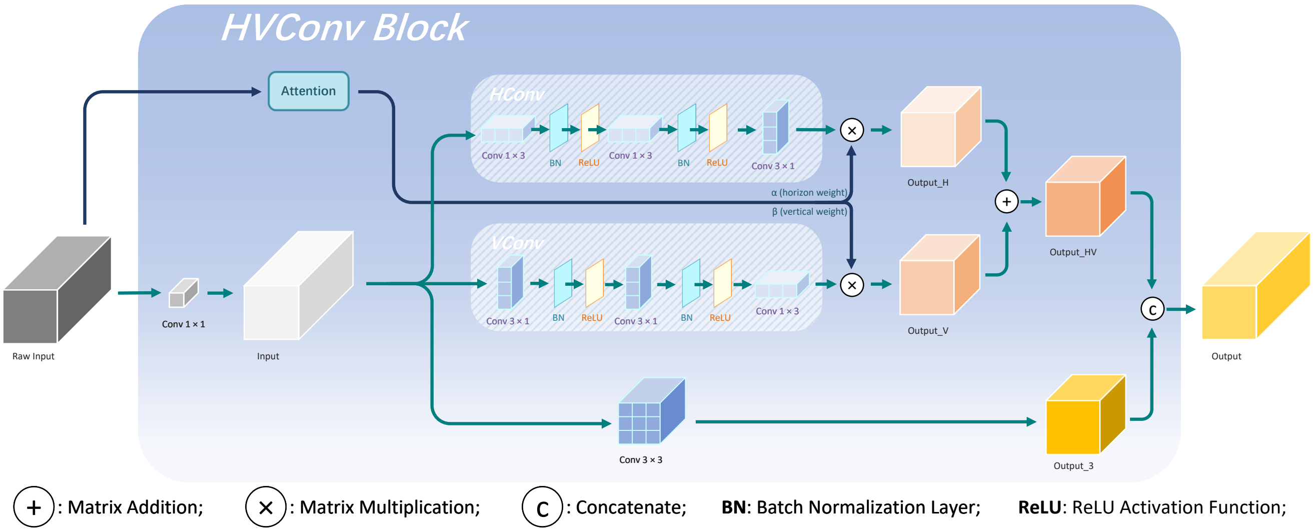


**Figure 4.** The overall framework of HVConv block.

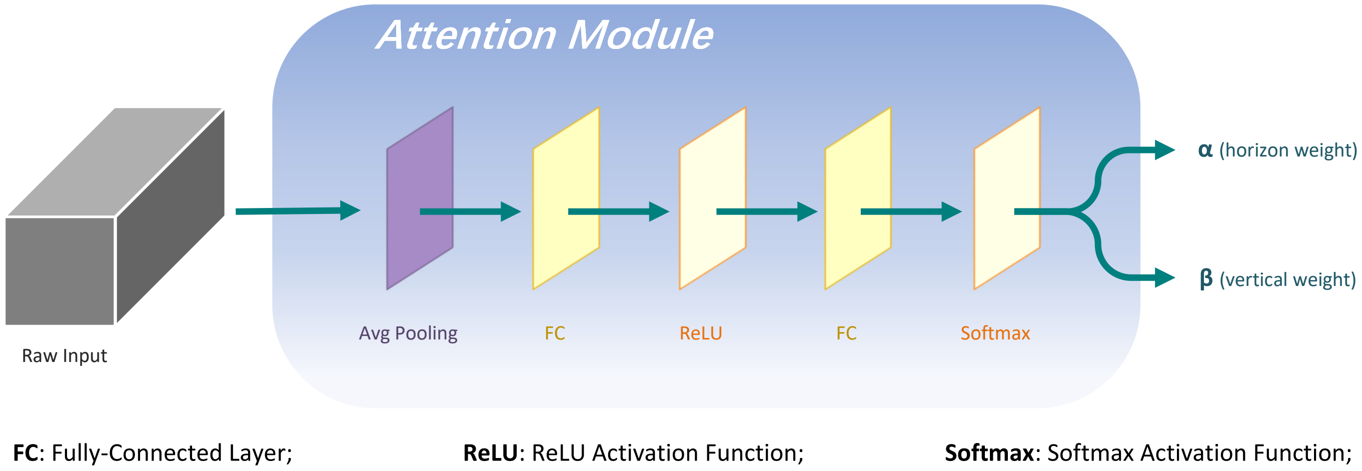

Attention Module
Raw Input
Avg Pooling
FC
ReLU
FC
Softmax
α (horizon weight)
β (vertical weight)
FC: Fully-Connected Layer;
ReLU: ReLU Activation Function;
Softmax: Softmax Activation Function;

**Figure 5.** The structure of Attention module.

### *3.2. Horizontal and Vertical Convolution*

HVConv consists of HConv and VConv. The entire HVConv block is presented as shown in Figure 4. We have established these two different paths to expand the horizontal and vertical receptive fields, respectively. Given an input $x \in \mathbb{R}^{c\times h\times w}$ and the output of $y \in \mathbb{R}^{c\times h\times w}$, the two paths of HVConv are denoted as $\mathbf{H}(x)$ and $\mathbf{V}(x)$. We represent a $1 \times 3$ convolution (horizontal convolution) and a $3 \times 1$ convolution (vertical convolution) as $h(x)$ and $v(x)$, respectively. Batch normalization is applied, using as $BN(\cdot)$, and $\sigma(\cdot)$

represents the ReLU activation function. HConv is basically constructed by two continuous horizontal convolutions and a vertical convolution. VConv is similar to HConv, which is constructed by two continuous vertical convolutions and a horizontal convolution. A single HConv or VConv can be expressed as:

$$h(x) = Conv^{1\times3}(x), \quad v(x) = Conv^{3\times1}(x) \tag{1}$$

$$H(x) = \sigma(BN(h(x))), \quad V(x) = \sigma(BN(v(x))) \tag{2}$$

In the HConv, for the first two $h(x)$ operations, they rapidly expand the receptive fields in the horizontal direction. After the receptive fields expand in the horizontal direction, the last $v(x)$ operation expands the entire receptive fields in the vertical direction to enlarge entire receptive fields. Consequently, the horizontally narrow receptive fields will expand in the vertical direction with a larger scale to better cover objects. The VConv is similar to the HConv but with the opposite effectiveness, which is to expand vertical receptive fields. It is noteworthy that the third HConv or VConv operation removes batch normalization and the activation function for the next step, which is path fusion. The normalization and the activation function have been applied after this whole block. For clarity, HVConv can be represented as shown below:

$$\begin{cases} \mathbf{H}(x) = v(H_{h,2}(H_{h,1}(x))) \\ \mathbf{V}(x) = h(V_{v,2}(V_{v,1}(x))) \end{cases} \tag{3}$$

### 3.3. Attention

We believe that different weights ought to be assigned to HConv and VConv based on the input images given their distinct aspect ratios and angles. Consequently, we design an attention module based on SENet [9] to enhance the network's ability to focus on different types of feature maps—whether they are more horizontal or vertical in nature. The entire structure is shown in Figure 5.

It is generally based on the squeeze and excitation block from SENet [9]. The input of this module is raw feature maps, which is the same input of 1 × 1 convolution. It is first calculated by an average pooling layer and then through a fully connected layer and a relu activation. The next fully connected layer transfers the channels to two dimensions, horizon and vertical. The final output consists of two weights by the softmax activation function, which are used to fuse the results from the HConv and VConv paths. We define $AVG(\cdot)$ as average global pooling, $FC(\cdot)$ as a fully connected layer, and $\mathbf{x}$ as raw feature maps input before 1 × 1 convolution computation. The attention module can be described as shown below:

$$Attn\{\alpha, \beta\} = Softmax(FC(\sigma(FC(AVG(\mathbf{x}))))) \tag{4}$$

By extracting horizon and vertical features from different channels of feature maps, we use a softmax function to obtain the weights $\alpha$ and $\beta$ for HConv and VConv, respectively. The softmax activation function makes sure the sum of two generated parameters is 1. However, in order to simplify the calculation and improve efficiency, these two parameters act on the entire feature maps from two paths, HConv and VConv, by multiplying them directly rather than channel-wise. It is crucial for us to optimize this calculation and reduce costs as much as possible. In this paper, we use $\alpha$ and $\beta$ to multiply the outputs of HConv and VConv and then sum them up. Thus, the output of HVConv can be formulated as shown below:

$$HV(x) = \alpha_{Attn(\mathbf{x})} \times \mathbf{H}(x) + \beta_{Attn(\mathbf{x})} \times \mathbf{V}(x) \tag{5}$$

Note that the fusion of HVConv and 3 × 3 Conv is different. The inner combination of HVConv is an additional operation. As the output of $\mathbf{H}(x)$ and $\mathbf{V}(x)$ multiply the weights of each assigned, it reflects the different weights of the horizontal or vertical information based on the raw feature maps being computed from input images. Consequently, we use the summation to fuse the result of these two paths. The output of $HV(x)$ is combined with 3 × 3 Conv through concatenate operation, as 3 × 3 Conv keeps the information in the original scale. We aggregate the information on a more horizontal or vertical scale with the original scale in channel dimensions. Denoting **Cat** as a concatenate operation, the HVConv operation could be simplified as shown below:

$$y = \mathbf{Cat}[HV(x), Conv^{3\times3}(x)] \tag{6}$$

For computation efficiency, we set the output channels of 3 × 3 Conv and $HV(x)$ as half that of the original output channels, respectively. We concatenate output feature maps from 3 × 3 Conv and HVConv in the channel-wise dimension. It keeps the original ResNet [6] output channel number of each block, which means HVConv could be seamlessly used for various different convolution backbone networks.

### *3.4. Loss Function*

Following the baseline method, we choose two different loss functions for classification and regression. The cross-entropy loss function, presented as Equation (6), is used for classification. $N$ denotes the number of samples and $K$ denotes the number of categories. $t_{i,k}$ is the true value of the $k$ class in sample $i$. If it belongs to a class, the value of $t$ is 1. If not, the value is 0. $p_{i,k}$ is the prediction probability of the $k$ labels in sample $i$.

$$L_{cls}(t, p) = -\frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{K} t_{i,k}\log(p_{i,k}) \tag{7}$$

We use a smooth L1 loss function for bounding box regression. The smooth L1 loss function is shown as Equation (8), and $z$ means the input parameter value. Equation (9) and Equation (10) are the entire bounding box loss functions for RPN and RoI head, respectively, based on the smooth L1 loss function. $u$ is the prediction value and $v$ is the true value. The representation of the bounding box in the RoI head is different from the midpoint offset method of Oriented RPN. Consequently, the loss function of them is different as well. The first difference is the parameter. Parameters in RPN, $x$, $y$, $w$, $h$, $\Delta x$, and $\Delta y$, represent the oriented bounding box and its envelope rectangle bounding box. Parameters in the RoI head, $x$, $y$, $w$, $h$ and $\theta$, represent the oriented bounding box. The other one is the value of $\beta$. In the RPN loss function, as Equation (9), $\beta$ is $0.\dot{1}$. However, in the RoI head loss function, as Equation (10), $\beta$ is set at 1.0 as the default setting.

$$L_{sll}(z) = \begin{cases} 0.5z^2 & \text{if } |z| < \beta \\ |z| - 0.5 & \text{otherwise} \end{cases} \tag{8}$$

$$L_{reg\_rpn}(u_i, v_i) = \sum_{i\in\{x,y,w,h,\Delta x,\Delta y\}} L_{sll}(u_i - v_i) \tag{9}$$

$$L_{reg\_head}(u_i, v_i) = \sum_{i\in\{x,y,w,h,\theta\}} L_{sll}(u_i - v_i) \tag{10}$$

## 4. Experiments

### *4.1. Datasets*

HRSC2016 [10] is a high-resolution remote sensing images dataset that is collected for ship detection. It consists of 1061 images which contain 2976 instances of ships. Some of the images from the HRSC2016 dataset are shown in Figure 6.

DOTA-v1.0 [11] is a large-scale dataset for object detection in aerial images, consisting of 2806 remote sensing images. It contains 188,282 instances of 15 categories: Plane (PL), Baseball diamond (BD), Bridge (BR), Ground track field (GTF), Small vehicle (SV), Large vehicle (LV), Ship (SH), Tennis court (TC), Basketball court (BC), Storage tank (ST), Soccer ball field (SBF), Roundabout (RA), Harbor (HA), Swimming pool (SP), and Helicopter (HC). Some images from the DOTA-v1.0 dataset are shown in Figure 7. The instances are in different scales and shapes.

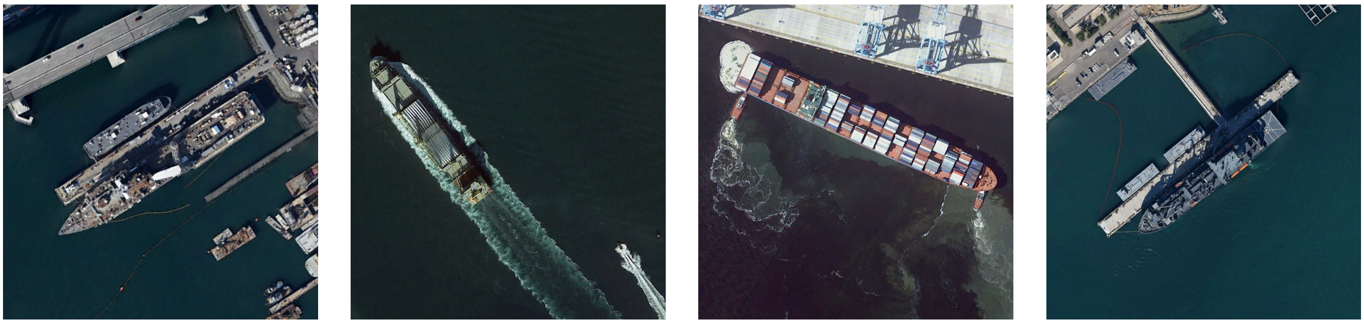


**Figure 6.** Ship images from the HRSC2016 dataset.

**Figure 7.** Remote sensing images from the DOTA-v1.0 dataset.

### *4.2. Implementation Details*

The backbone of our module is firstly pretrained on the ImageNet-1K dataset and then fine-tuned on the remote sensing dataset. In the first two ablation studies, we adopted the 100-epoch backbone pretrain and single-scale dataset for experiment efficiency. In addition, to pursue higher accuracy, we adopted the 300-epoch backbone pretrain on our main results. We conduct extensive experiments on the HRSC2016 and DOTA-v1.0 datasets. On the DOTA-v1.0 dataset, we use two different scales for training and testing. We cropped the images into 1024 × 1024 patches with a stride of 824, which means the pixel overlap between two adjacent patches is 200. As for multi-scale training and testing, we first resize the raw images at three scales (0.5, 1.0, and 1.5) and crop them to 1024 × 1024 patches with a stride of 524. Following the common practice, we use the training set and validation set for training and the testing set for testing. We train the models for 36 epochs on HRSC2016 datasets and 12 epochs on DOTA-v1.0 with the AdamW optimizer. The initial learning rate is set to 0.00005 for the HRSC2016 dataset, 0.0001 for the DOTA-v1.0 datasets with single-scale training, and 0.0002 for multi-scale training. We use four RTX3090 GPUs with a batch size of eight on the DOTA-v1.0 dataset for multi-scale training and a batch size of four for single-scale training. On the HRSC2016 dataset, we conduct our experiments by using one RTX3090 GPU with a batch size of one for equitable assessment. We use one RTX3090 GPU for all the testing. MMRotate [27] is a convenient tool for constructing networks tailored for rotation object detection. Consequently, we conducted all our experiments by using it.

### *4.3. Evaluation Metrics*

We take VOC 2007 [28] metrics into account, as most of the experiments in other studies use it as well. The mean Average Precision (mAP) is calculated by the Average Precision (AP) of each class in the dataset. The AP metric considers the precision (P) and recall (R), and the calculation formula is shown below:

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

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

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

TP stands for the number of samples where the model correctly predicts positive instances; FP stands for the number of samples where the model incorrectly predicts positive instances; FN stands for the number of samples where the model incorrectly predicts negative instances. In different confidence levels, P and R are different, the PR-curve can be drawn, and the AP can be calculated by the curve. The mAP could be expressed as shown below:

$$mAP = \frac{1}{k}\sum_{i=1}^{k} AP_i \tag{14}$$

k is the number of categories in the dataset.

In the DOTA-v1.0 dataset, the Intersection over Union (IoU) of mAP is 0.5. In the HRSC2016 dataset, AP50 signifies an IoU of 0.5 for mAP, AP75 denotes an IoU of 0.75 for mAP, and mAP represents the average mAP across various IoUs ranging from 0.5 to 0.95 with a step of 0.05.

### 4.4. Comparison with State-of-the-Art

We provide specific experiment results on DOTA dataset, including the precision of each category and the mean average precision (mAP) to ensure a fair comparison. Table 1 shows the specific results compared with 18 state-of-the-art methods in single-scale training, including both one-stage methods and two-stage methods. HVConv expedites the convergence of the network in single-scale training. As a result, our approach achieves the mAP score of **77.60%**.

**Table 1.** Experimental results on the DOTA-v1.0 dataset with single-scale training and compared with state-of-the-art methods.

| Method | Backbone | mAP | PL | BD | BR | GTF | SV | LV | SH | TC | BC | ST | SBF | RA | HA | SP | HC |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| *One-stage methods* | | | | | | | | | | | | | | | | | |
| DRN [3] | H104 | 70.70 | 88.91 | 80.22 | 43.52 | 63.35 | 73.48 | 70.69 | 84.94 | 90.14 | 83.85 | 84.11 | 50.12 | 58.41 | 67.62 | 68.60 | 52.50 |
| R3Det [2] | R101 | 73.79 | 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 |
| PIoU [29] | DLA34 | 60.50 | 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 |
| RSDet [30] | R101 | 72.20 | 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** |
| DAL [31] | R50 | 71.44 | 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 |
| G-Rep [32] | R50 | 75.56 | 87.76 | 81.29 | 52.64 | 70.53 | **80.34** | 80.56 | 87.47 | 90.74 | 82.91 | 85.01 | 61.48 | 68.51 | 67.53 | 73.02 | 63.54 |
| MIOUC [33] | ELAN based | 75.80 | 89.30 | 82.10 | 54.70 | 65.60 | 80.10 | 84.40 | 87.70 | 90.80 | 79.00 | 87.10 | 50.40 | 64.40 | 80.30 | 80.50 | 60.10 |
| $S^2$ANet [1] | R50 | 76.11 | 88.70 | 81.41 | 54.28 | 69.75 | 78.04 | 80.54 | 88.04 | 90.69 | 84.75 | 86.22 | 65.03 | 65.81 | **76.16** | **73.37** | 58.86 |
| *Two-stage methods* | | | | | | | | | | | | | | | | | |
| RoI Trans [20] | R101 | 69.56 | 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 |
| SCRDet [34] | R101 | 72.61 | 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 |
| G.Vertex [35] | R101 | 75.02 | 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 |
| FAOD [36] | R101 | 73.28 | **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 |
| Mask OBB [37] | R50 | 74.86 | 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 |
| ReDet [38] | ReR50 | 76.25 | 88.79 | 82.64 | 53.97 | 74.00 | 78.13 | **84.06** | 88.04 | 90.89 | 87.78 | 85.75 | 61.76 | 60.39 | 75.96 | 68.07 | 63.59 |
| AOPG [39] | R101 | 75.39 | 89.14 | 82.74 | 51.87 | 69.28 | 77.65 | 82.42 | 88.08 | 90.89 | 86.26 | 85.13 | 60.60 | 66.30 | 74.05 | 67.76 | 58.77 |
| SASM [40] | R50 | 74.92 | 86.42 | 78.97 | 52.47 | 69.84 | 77.30 | 75.99 | 86.72 | 90.89 | 82.63 | 85.66 | 60.13 | 68.25 | 73.98 | 72.22 | 62.37 |
| AFF-Det [41] | R50 | 75.72 | 88.34 | 83.06 | 53.77 | 72.16 | 79.54 | 78.09 | 87.65 | 90.69 | 87.19 | 84.50 | 57.46 | 64.96 | 74.88 | 70.80 | 61.24 |
| Oriented R-CNN [26] | R50 | 75.87 | 89.46 | 82.12 | 54.78 | 70.86 | 78.93 | 83.00 | **88.20** | 90.90 | 87.50 | 84.68 | 63.97 | 67.69 | 74.94 | 68.84 | 52.28 |
| **HVConv** | **HV-R50** | **77.60** | 89.25 | 84.07 | **55.59** | 75.56 | 78.40 | 83.69 | 87.89 | 90.87 | 86.07 | 85.26 | **68.38** | 68.14 | 75.88 | 70.87 | 64.05 |

In the backbone column, H104 represents the 104-layer hourglass network [42], R50 and R101 mean ResNet-50 and ResNet-101 [6], respectively, DLA34 [43] refers to the 34-layer deep layer aggregation network, ReR50 is proposed in ReDet [38], HV-R50 is our backbone network, which replaced the 3 × 3 convolution of ResNet-50 [6] with horizontal and vertical convolution blocks.

The dataset for multi-scale training is three times larger than that for single-scale training. Apart from the normal-sized images, the images in multi-scaled training are resized to twice their original size as well as half their original size. This training strategy enhances the model generalization and detection ability of small or large objects, which improves the performance of networks. Thus, we also compare against 11 state-of-the-art methods in multi-scale training. Table 2 shows the mAP score of each method, and our work achieves the best mAP score of **81.07%**.

**Table 2.** Experimental results on the DOTA-v1.0 dataset with multi-scale training and compared with state-of-the-art methods.

| Method | Backbone | mAP |
|---|---|---|
| Oriented R-CNN [26] | R50 | 80.87 |
| R3Det-GWD [44] | R152 | 80.19 |
| R3Det-KLD [45] | R152 | 80.63 |
| AFF-Det [41] | R50 | 80.73 |
| KFIoU [46] | Swin-T | 80.93 |
| RVSA [47] | ViT-B | 81.01 |
| $S^2$ANet [1] | R50 | 79.42 |
| ReDet [38] | Re-R50 | 80.10 |
| AOPG [39] | R50 | 80.66 |
| R3Det [2] | R152 | 76.47 |
| G-Rep [32] | Swin-T | 80.16 |
| **HVConv** | **HV-R50** | **81.07** |

Swin-T represents Swin Transformer [48].

### *4.5. Ablation Studies*

We conduct a series of experiments on the DOTA-v1.0 and the HRSC2016 datasets for ablation studies, including the stage for replacement, fusion strategy, and effectiveness validation.

#### 4.5.1. The Stage for Replacement

We set three different conditions for replacing the stages of ResNet-50 [6], covering from Stage1 to Stage4, Stage2 to Stage4, and Stage1 to Stage3. These two conditions represent not using the HVConv module in the first stage or last stage. The receptive fields in feature maps of the first stage tend to be the smallest, and the last stage tends to be the largest. We try to figure out which of the stages is more important for our module. The results are shown in Table 3, indicating that replacing the convolution from Stage1 to Stage4 could achieve the highest mAP score of **77.30%**. By comparing the results from Stage1 to Stage3 and from Stage2 to Stage4, it is easy to find that the earlier stages are more important than the later stages because the earlier stages deal with tiny objects. In remote sensing object detection, tiny objects are numerous. To achieve better performance balance, it is crucial to fit these tiny objects. Although the mAP score of Stage1 to Stage3 dropped 0.02%, the FPS of this model is the highest. Considering computing efficiency, we finally chose to replace Stage1 to Stage 3 as our backbone network for other experiments.

**Table 3.** Ablation results on DOTA-v1.0 with single-scale training for stages replacement in ResNet-50 [6].

| Stage1 | Stage2 | Stage3 | Stage4 | mAP | FPS |
|---|---|---|---|---|---|
| ✓ | ✓ | ✓ | ✓ | **77.30** | 14.9 |
| ✓ | ✓ | ✓ |  | 77.28 | **17.6** |
|  | ✓ | ✓ | ✓ | 77.14 | 16.2 |

FPS represents output images per second.

#### 4.5.2. The Effective of Attention Module

We compare our attention module with the other three strategies. The results are presented in Table 4. First, we only use one of HConv or VConv by setting the weight of each path to 1 or 0. Following that, we combine these two parts with the same weight. Each weight of HConv and VConv is 0.5. Finally, we use our attention module to generate $\alpha$ and $\beta$ for weighting each part, by aggregating them, and achieve the mAP score of **77.28%**, which is the best score among these experiments. These experimental results indicate that applying dynamic weights could further enhance the performance by fitting the object in two directions.

**Table 4.** Ablation results on DOTA-v1.0 with single-scale training for fusion method.

| Fusion Method | | mAP |
|---|---|---|
| HConv | VConv | |
| 0 | 1 | 76.68 |
| 1 | 0 | 76.38 |
| 0.5 | 0.5 | 76.67 |
| $\alpha$ | $\beta$ | **77.28** |

Columns of HConv and VConv mean the weights of HConv and VConv, respectively. $\alpha$ and $\beta$ are the outputs of the attention module.

### 4.5.3. Effectiveness on Different Architecture

To further validate the effectiveness of our works, we performed experiments on the HRSC2016 dataset and DOTA-v1.0 dataset with single-scale training. The results are shown in Figure 8 and Table 5. We compared our backbone method with ResNet [6], in six different network structures, involving Rotated RetinaNet [12], R3Det [2], S$^2$ANet [1], Rotated Faster R-CNN [14], CFA [49] and the baseline method Oriented R-CNN [26]. Our approach significantly improves the mAP score across all architectures, including both one-stage and two-stage methods. This firmly demonstrates the effectiveness of our method.

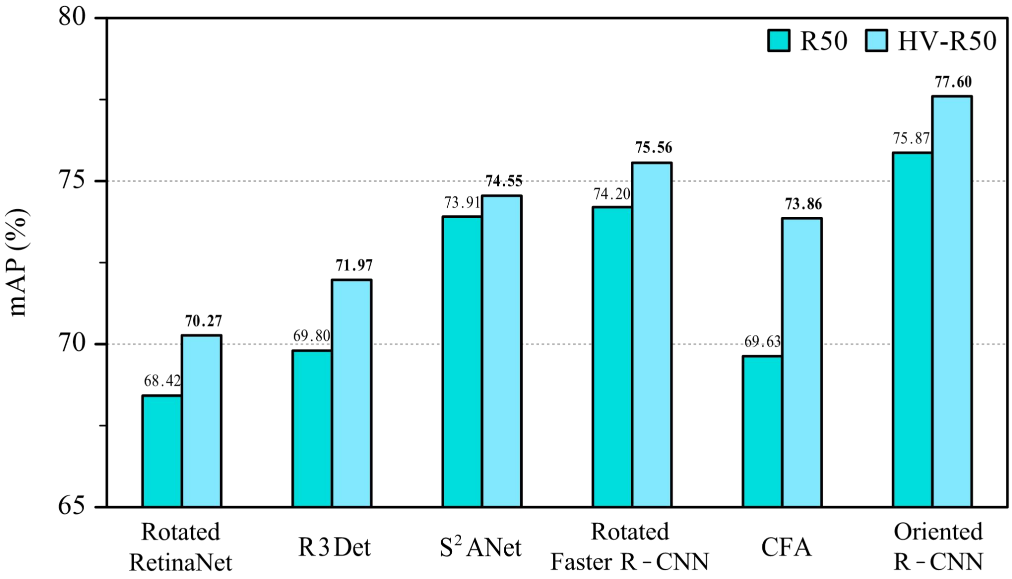


**Figure 8.** Experimental results on the DOTA-v1.0 dataset with single-scale training. Replacing the backbone with HV-R50 to validate the effectiveness on various architectures.

**Table 5.** Experimental results on the HRSC2016 dataset. Replacing the backbone with HV-R50 to validate the effectiveness on various architectures.

| Method | Backbone | AP50 | AP75 | mAP |
|---|---|---|---|---|
| Rotated RetinaNet [12] | R50<br>**HV-R50** | 72.20<br>**81.70** | 36.60<br>**46.80** | 38.53<br>**46.41** |
| Rotated Faster R-CNN [14] | R50<br>**HV-R50** | 78.20<br>**78.80** | 41.10<br>**46.70** | 43.59<br>**44.78** |
| R3Det [2] | R50<br>**HV-R50** | 88.10<br>**89.30** | 46.80<br>**57.30** | 49.07<br>**53.25** |
| RoI Transformer [20] | R50<br>**HV-R50** | 90.10<br>**90.30** | 79.60<br>**80.00** | 63.46<br>**63.63** |
| Oriented R-CNN [26] | R50<br>**HV-R50** | 90.60<br>90.60 | 89.30<br>**89.60** | 70.85<br>**71.98** |

### *4.6. Visualization and Analysis*

We provide some visualized results for a more distinct view of our method. Figure 9 presents some results of the test set from the DOTA-v1.0 dataset, including Small Vehicle, Large Vehicle, Ship, Harbor, Bridge, Plane, Helicopter, and Roundabout. Figure 10 shows the confusion matrix result of the validation set from the DOTA-v1.0 dataset, which reflects the classification ability of the model. It is noted that the accuracy of Storage tank tends to be lower than other labels, which means that the performance of detecting objects in different scales remains to be improved, since the Storage tank usually has various scales in the same photo, as seen in the last image we showed in Figure 7.

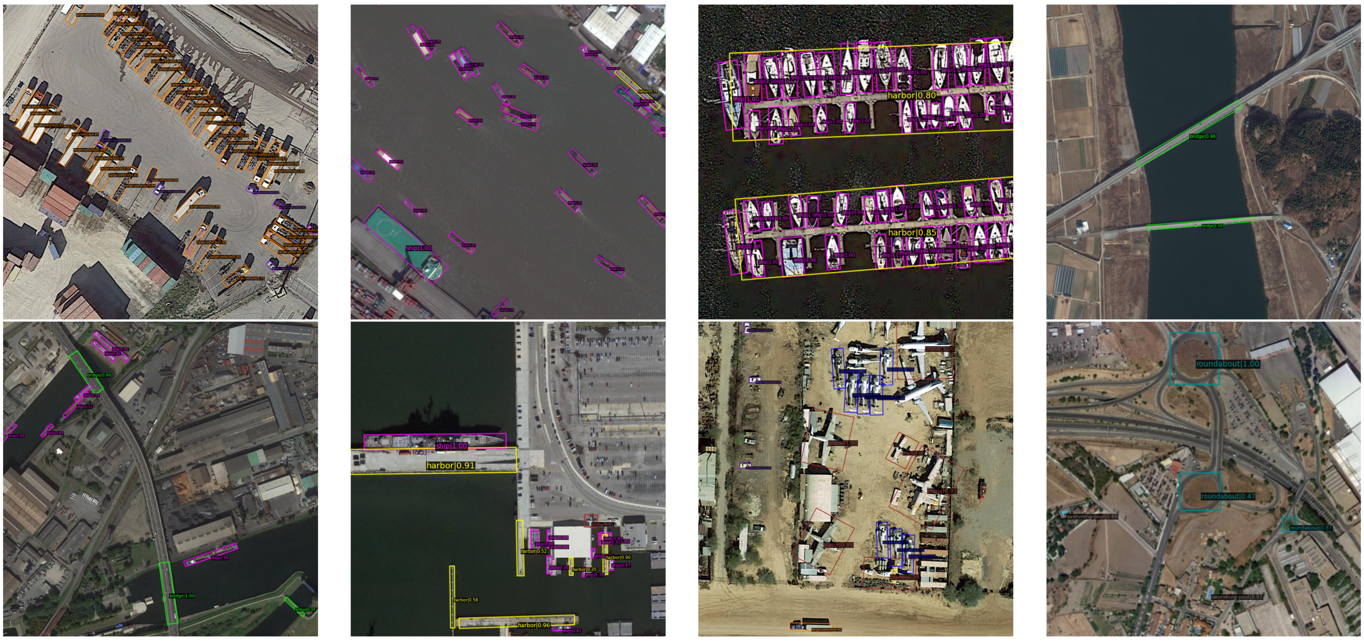


**Figure 9.** Results of the test set from the DOTA-v1.0 dataset.

Figure 11 presents the detection result comparisons of our method with the baseline method. The left images are the baseline method's outputs, and the right images are the outputs of our method. In the comparison between the first two groups, the outputs of the baseline method (Figure 11a,c) missed the detection of narrow objects, Bridge and Harbor. On the contrary, our method detected these two narrow aspect ratio objects accurately (Figure 11b,d). The bounding box of the baseline method in the third group has been cut into two segments (Figure 11e), which indicates the wrong localization. The final comparison is a roundabout on the edge of an image; the baseline method omitted the roundabout (Figure 11g), but ours detected it correctly. This reflects that our method also possesses the ability to detect objects in normal aspect ratios. These visualized results prove that our horizontal and vertical convolution method is skilled in narrow object detection and correctly detecting general objects as well.

We provide comparisons of the heat map results from the baseline method and our method's backbone layers outputs, as shown in Figure 12. There are four groups, and each group consists of three images: the input image, the heat map image from the baseline method's backbone, and the heat map image from ours. The objects in the first group (Figure 12a) and the second group (Figure 12b) have a regular aspect ratio. The heat map results of these groups convincingly demonstrate that our method possesses the capability to detect objects with common aspect ratios as well as normal convolution. Additionally, benefiting from the horizontal and vertical convolution, the receptive field coverage is much tighter than that of the baseline method. The next two groups have different scenarios: the first is a long bridge (Figure 12c), and the other one is a complex harbor environment (Figure 12d). The heat map results clearly reveal the redundant areas of the baseline method and the accuracy of HVConv. It reflects that our method reduces redundant receptive fields, thereby diminishing the overlapping of bounding boxes and the occurrence of bounding box segmentation caused by irregular aspect ratios, ultimately achieving higher performance for remote sensing object detection.

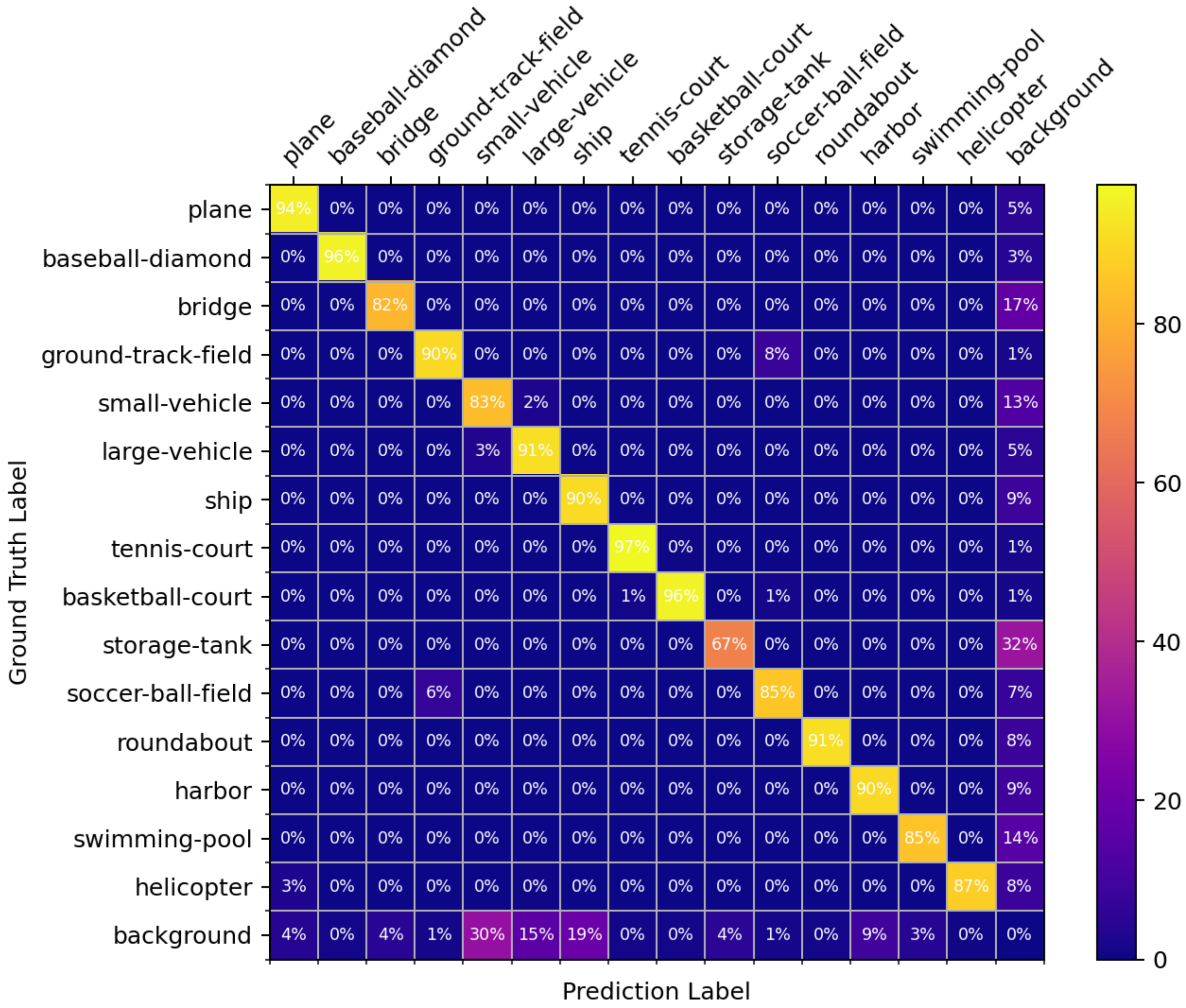
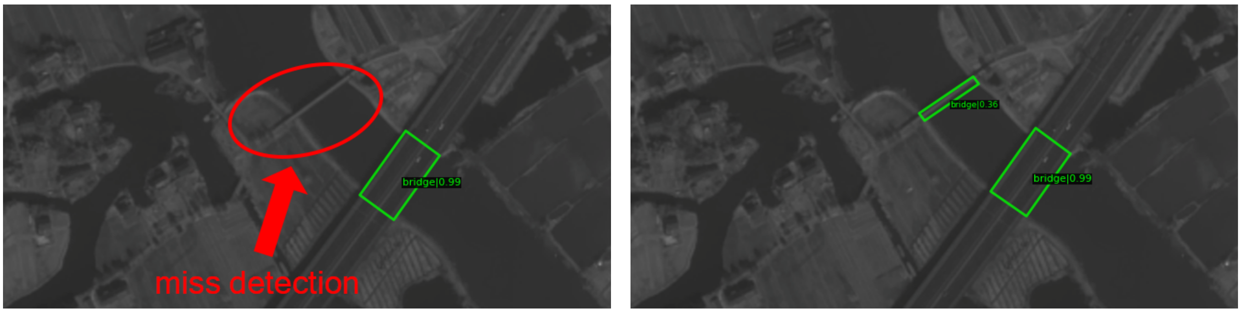
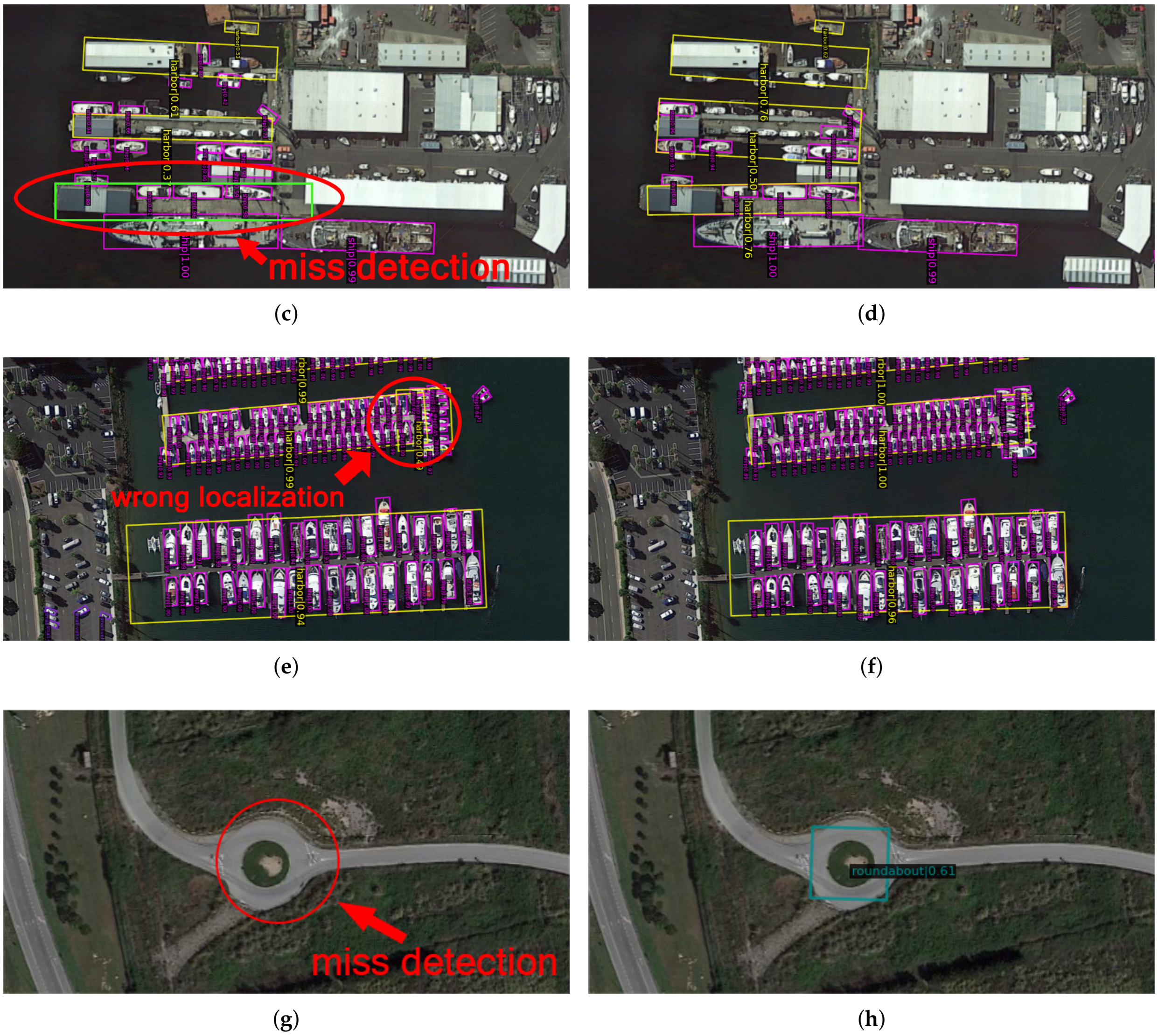


**Figure 10.** The confusion matrix result of multi-scale training on the validation set from the DOTA-v1.0 dataset.



(**a**) (**b**)

**Figure 11.** *Cont.*



(**c**) (**d**)

(**e**) (**f**)

(**g**) (**h**)

**Figure 11.** The output results of baseline and our method. The left column (**a**,**c**,**e**,**g**) shows the baseline detection results and the right column (**b**,**d**,**f**,**h**) shows the detection results of our method.

In Table 1, our method's AP score of BR is the highest among the others. The AP score of HA has been improved as well. This is generally because these objects have a dramatic irregular aspect ratio in remote images. Our method could fit such classes' features well by expanding narrow and long receptive fields and reducing background redundant information. Also, we choose to apply our module to the first three stages of the backbone network, improving the tiny object detection performance, which is because receptive fields will grow at each stage's calculation and incur tiny object detail information loss. These results reflect that it is useful to expand horizontal and vertical receptive fields for long and narrow instances in remote sensing object detection. However, this work can be developed in the future. In our future work, we will concentrate on the convolution calculation sequence and kernel size for a more rapid receptive field expansion. The attention mechanism can be refined for a better combination of two paths, which finally leads to a higher accuracy of detection.

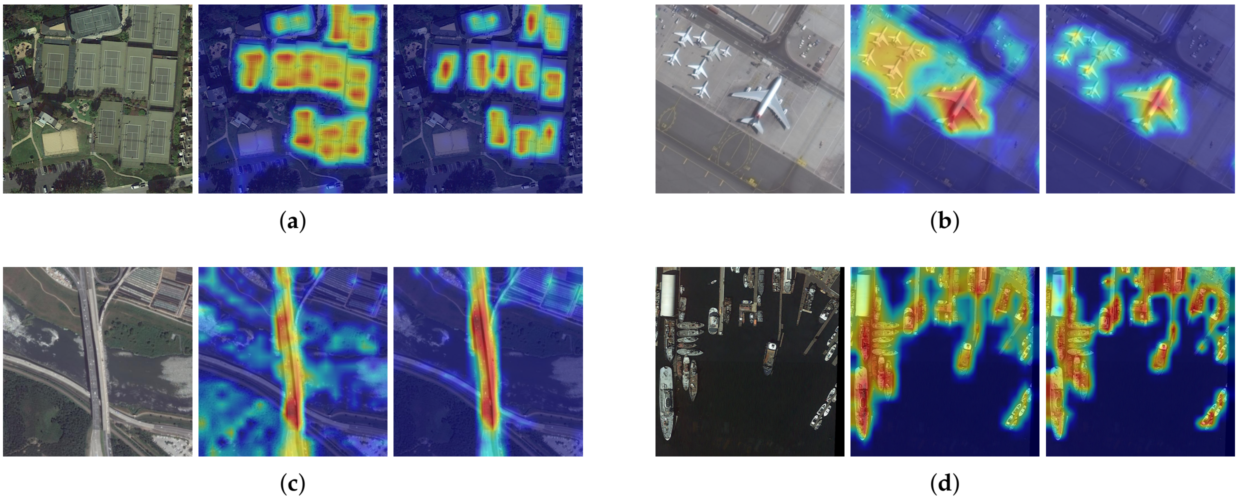


**Figure 12.** The heat map comparison of backbone layers on the DOTA-v1.0 dataset with single-scale training. Images in each group from the left are the input image, heat map of the baseline method, and heat map of ours. (**a**) has several tennis courts. (**b**) has planes in different scales and with a tilt angle. (**c**) is a long bridge. (**d**) is a complex circumstance of the harbor.

## 5. Conclusions

This paper proposed Horizontal and Vertical Convolution, which is a plug-and-play module for remote sensing object detection. Different from common object detection, remote sensing object detection tends to be in tiny, narrow, and arbitrary directions. This convolution approach expands the receptive fields in horizontal and vertical directions, respectively, and a normal convolution is adopted as well for gathering background information, which is in order to acquire elongated feature representations in two different directions with normal feature information. Through an attention mechanism, combining these two directions features and aggregating with the normal features enhanced the ability to recognize narrow objects and maintain a general object detection ability level. Our method proposes a new perspective (i.e., receptive fields optimization) to reduce the impact of irrelevant information around remote sensing objects on model recognition capabilities. It indicates that the use of the convolution method for special computer vision tasks still has room for improvement. Our experimental results on the DOTA-v1.0 and HRSC2016 datasets achieved state-of-the-art accuracy, which fully verified the effectiveness of our module. We hope our method can be used not only in remote sensing areas but also in other domains with irregular aspect ratio object problems. As well, we hope that this paper inspires other researchers to study the receptive fields' redundancy problem. In the future, we will continue this study to optimize the attention module for a more efficient computation.

**Author Contributions:** Conceptualization, J.C. and Q.L.; methodology, J.C.; software, J.C.; validation, J.C.; formal analysis, J.C. and Q.L.; investigation, J.C. and Q.L.; resources, J.C.; writing—original draft preparation, J.C.; writing—review and editing, Q.L. and D.Z.; visualization, J.C. and H.H.; supervision, Q.L. and G.F.; funding acquisition, Q.L. and Y.Y. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was partly supported by the National Natural Science Foundation of China (No. 62371135 and U21A20471), the Natural Science Foundation of Fujian Province (Grant No. 2023J01431), and the Research Program for Young and Middle-Aged Teachers of Fujian Province (No. JAT210036).

**Data Availability Statement:** All data results are contained within the article. The public datasets are used for our research: HRSC2016 dataset: https://sites.google.com/site/hrsc2016/ (accessed on 3 April 2024); DOTA-v1.0 dataset: https://captain-whu.github.io/DOTA/index.html (accessed on 7

April 2024). Our codes and models are provided as https://github.com/chjinhuu/hvconv (accessed on 16 May 2024).

**Acknowledgments:** The authors thank the College of Computer and Data Science, Fuzhou University, for the fundamental and technical support. The authors are also thank the anonymous reviewers for their valuable suggestions.

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

## References

1. Han, J.; Ding, J.; Li, J.; Xia, G.S. Align Deep Features for Oriented Object Detection. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 1–11. [CrossRef]
2. Yang, X.; Yan, J.; Feng, Z.; He, T. R3Det: Refined Single-Stage Detector with Feature Refinement for Rotating Object. In Proceedings of the 35th AAAI Conference on Artificial Intelligence, Virtual, 4–7 February 2021; Volume 4A, pp. 3163–3171.
3. 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 2020 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Virtual, 14–19 June 2020; pp. 11204–11213. [CrossRef]
4. Lin, Q.; Zhao, J.; Du, B.; Fu, G.; Yuan, Z. MEDNet: Multiexpert Detection Network With Unsupervised Clustering of Training Samples. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 1–14. [CrossRef]
5. Feng, L.Q.; Luo Jun, L.; Yuan Long, Y.; Fu, G. A Multiple Prediction Mechanisms Ensemble for Complex Remote Sensing Scenes. In Proceedings of the 31st ACM International Conference on Multimedia, Ottawa, ON, Canada, 29 October–3 November 2023; pp. 8635–8643. [CrossRef]
6. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the 2016 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; Volume 2016, pp. 770–778. [CrossRef]
7. Li, Y.; Hou, Q.; Zheng, Z.; Cheng, M.M.; Yang, J.; Li, X. Large Selective Kernel Network for Remote Sensing Object Detection. In Proceedings of the 2023 IEEE International Conference on Computer Vision, Paris, France, 2–6 October 2023; pp. 16748–16759. [CrossRef]
8. Pu, Y.; Wang, Y.; Xia, Z.; Han, Y.; Wang, Y.; Gan, W.; Wang, Z.; Song, S.; Huang, G. Adaptive Rotated Convolution for Rotated Object Detection. In Proceedings of the 2023 IEEE International Conference on Computer Vision, Paris, France, 2–6 October 2023; pp. 6566–6577. [CrossRef]
9. Hu, J.; Shen, L.; Sun, G. Squeeze-and-Excitation Networks. In Proceedings of the 2018 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 7132–7141. [CrossRef]
10. Liu, Z.; Wang, H.; Weng, L.; Yang, Y. Ship Rotated Bounding Box Space for Ship Extraction from High-Resolution Optical Satellite Images with Complex Backgrounds. *IEEE Geosci. Remote Sens. Lett.* **2016**, *13*, 1074–1078. [CrossRef]
11. 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 2018 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 3974–3983. [CrossRef]
12. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollar, P. Focal Loss for Dense Object Detection. In Proceedings of the 2017 IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; Volume 2017, pp. 2999–3007. [CrossRef]
13. Zhang, G.; Yu, W.; Hou, R. MFIL-FCOS: A Multi-Scale Fusion and Interactive Learning Method for 2D Object Detection and Remote Sensing Image Detection. *Remote Sens.* **2024**, *16*, 936 [CrossRef]
14. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [CrossRef] [PubMed]
15. Lin, Q.; Zhao, J.; Fu, G.; Yuan, Z. CRPN-SFNet: A high-performance object detector on large-scale remote sensing images. *IEEE Trans. Neural Netw. Learn. Syst.* **2020**, *33*, 416–429 [CrossRef] [PubMed]
16. Lin, T.Y.; Dollar, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. In Proceedings of the 2017 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; Volume 2017, pp. 936–944. [CrossRef]
17. Lin, Q.; Zhao, J.; Tong, Q.; Zhang, G.; Yuan, Z.; Fu, G. Cropping Region Proposal Network Based Framework for Efficient Object Detection on Large Scale Remote Sensing Images. In Proceedings of the 2019 IEEE International Conference on Multimedia and Expo, Shanghai, China, 8–12 July 2019; pp. 1534–1539. [CrossRef]
18. Lin, Q.; Long, C.; Zhao, J.; Fu, G.; Yuan, Z. DDBN: Dual detection branch network for semantic diversity predictions. *Pattern Recognit.* **2022**, *122*, 108315. [CrossRef]
19. 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]
20. 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 2019 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; Volume 2019, pp. 2844–2853. [CrossRef]

21. Li, J.; Chen, M.; Hou, S.; Wang, Y.; Luo, Q.; Wang, C. An Improved S2A-Net Algorithm for Ship Object Detection in Optical Remote Sensing Images. *Remote Sens.* **2023**, *15*, 4559. [CrossRef]
22. Li, X.; Wang, W.; Hu, X.; Yang, J. Selective kernel networks. In Proceedings of the 2019 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; Volume 2019, pp. 510–519. [CrossRef]
23. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the Inception Architecture for Computer Vision. In Proceedings of the 2016 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; Volume 2016, pp. 2818–2826. [CrossRef]
24. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An image is worth 16X16 Words: Transformers for image recognition at scale. In Proceedings of the ICLR 2021—9th International Conference on Learning Representations, Virtual, 3–7 May 2021.
25. Guan, X.; Dong, Y.; Tan, W.; Su, Y.; Huang, P. A Parameter-Free Pixel Correlation-Based Attention Module for Remote Sensing Object Detection. *Remote Sens.* **2024**, *16*, 312. [CrossRef]
26. Xie, X.; Cheng, G.; Wang, J.; Yao, X.; Han, J. Oriented R-CNN for Object Detection. In Proceedings of the 2021 IEEE International Conference on Computer Vision, Virtual, 11–17 October 2021; pp. 3500–3509. [CrossRef]
27. Zhou, Y.; Yang, X.; Zhang, G.; Wang, J.; Liu, Y.; Hou, L.; Jiang, X.; Liu, X.; Yan, J.; Lyu, C.; et al. MMRotate: A Rotated Object Detection Benchmark using PyTorch. In Proceedings of the 30th ACM International Conference on Multimedia, Lisboa, Portugal, 10–14 October 2022; pp. 7331–7334. [CrossRef]
28. Everingham, M.; Van Gool, L.; Williams, C.K.I.; Winn, J.; Zisserman, A. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. Available online: http://host.robots.ox.ac.uk/pascal/VOC/voc2007/ (accessed on 21 March 2024).
29. 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 Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Glasgow, UK, 29 October 2020; Volume 12350 LNCS, pp. 195–211. [CrossRef]
30. Qian, W.; Yang, X.; Peng, S.; Yan, J.; Guo, Y. Learning Modulated Loss for Rotated Object Detection. In Proceedings of the 35th AAAI Conference on Artificial Intelligence, Virtual, 4–7 February 2021; Volume 3B, pp. 2458–2466.
31. Ming, Q.; Zhou, Z.; Miao, L.; Zhang, H.; Li, L. Dynamic Anchor Learning for Arbitrary-Oriented Object Detection. In Proceedings of the 35th AAAI Conference on Artificial Intelligence, Virtual, 4–7 February 2021; Volume 3B, pp. 2355–2363.
32. Huo, L.; Hou, J.; Feng, J.; Wang, W.; Liu, J. Global and Multiscale Aggregate Network for Saliency Object Detection in Optical Remote Sensing Images. *Remote Sens.* **2024**, *16*, 624. [CrossRef]
33. Shen, Y.; Liu, D.; Chen, J.; Wang, Z.; Wang, Z.; Zhang, Q. On-Board Multi-Class Geospatial Object Detection Based on Convolutional Neural Network for High Resolution Remote Sensing Images. *Remote Sens.* **2023**, *15*, 3963. [CrossRef]
34. 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 2019 IEEE International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; Volume 2019, pp. 8231–8240. [CrossRef]
35. 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.* **2021**, *43*, 1452–1459. [CrossRef] [PubMed]
36. 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 International Conference on Image Processing, Taipei, Taiwan, 22–25 September 2019; Volume 2019, pp. 3886–3890. [CrossRef]
37. 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]
38. Han, J.; Ding, J.; Xue, N.; Xia, G.S. ReDeT: A Rotation-equivariant Detector for Aerial Object Detection. In Proceedings of the 2021 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 19 June–25 June 2021. [CrossRef]
39. Cheng, G.; Wang, J.; Li, K.; Xie, X.; Lang, C.; Yao, Y.; Han, J. Anchor-Free Oriented Proposal Generator for Object Detection. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 1–11. [CrossRef]
40. Hou, L.; Lu, K.; Xue, J.; Li, Y. Shape-Adaptive Selection and Measurement for Oriented Object Detection. In Proceedings of the 36th AAAI Conference on Artificial Intelligence, Virtual, 22 February–1 March 2022; Volume 36, pp. 753–761.
41. Zhen, P.; Wang, S.; Zhang, S.; Yan, X.; Wang, W.; Ji, Z.; Chen, H.B. Towards Accurate Oriented Object Detection in Aerial Images with Adaptive Multi-level Feature Fusion. *ACM Trans. Multimedia Comput. Commun. Appl.* **2023**, *19*, 6. [CrossRef]
42. Yang, J.; Liu, Q.; Zhang, K. Stacked Hourglass Network for Robust Facial Landmark Localisation. In Proceedings of the 2017 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, Honolulu, HI, USA, 21–26 July 2017; Volume 2017, pp. 2025–2033. [CrossRef]
43. Yu, F.; Wang, D.; Shelhamer, E.; Darrell, T. Deep Layer Aggregation. In Proceedings of the 2018 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 2403–2412. [CrossRef]
44. 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 2021 Machine Learning Research, Virtual, 18–24 July 2021; Volume 139, pp. 11830–11841.
45. Yang, X.; Yang, X.; Yang, J.; Ming, Q.; Wang, W.; Tian, Q.; Yan, J. Learning High-Precision Bounding Box for Rotated Object Detection via Kullback-Leibler Divergence. In Proceedings of the 2021 Advances in Neural Information Processing Systems, Virtual, 6–14 December 2021; Volume 22, pp. 18381–18394.

46. Yang, X.; Zhou, Y.; Zhang, G.; Yang, J.; Wang, W.; Yan, J.; Zhang, X.; Tian, Q. The KFIoU Loss for Rotated Object Detection. *arXiv* **2022**, arXiv:2201.12558. [CrossRef]
47. Wang, D.; Zhang, Q.; Xu, Y.; Zhang, J.; Du, B.; Tao, D.; Zhang, L. Advancing Plain Vision Transformer Toward Remote Sensing Foundation Model. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, 1–15. [CrossRef]
48. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin Transformer: Hierarchical Vision Transformer using Shifted Windows. In Proceedings of the 2021 IEEE International Conference on Computer Vision, Virtual, 11–17 October 2021; pp. 9992–10002. [CrossRef]
49. Guo, Z.; Zhang, X.; Liu, C.; Ji, X.; Jiao, J.; Ye, Q. Convex-Hull Feature Adaptation for Oriented and Densely Packed Object Detection. *IEEE Trans. Circuits Syst. Video Technol.* **2022**, *32*, 5252–5265. [CrossRef]

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