remote sensing

MDPI

Article

RADet: Refine Feature Pyramid Network and Multi-Layer Attention Network for Arbitrary-Oriented Object Detection of Remote Sensing Images

Yangyang Li *, Qin Huang, Xuan Pei, Licheng Jiao and Ronghua Shang

Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education, International Research Center for Intelligent Perception and Computation, Joint International Research Laboratory of Intelligent Perception and Computation, School of Artificial Intelligence, Xidian University, Xi'an 710071, China; qinhuang@stu.xidian.edu.cn (Q.H.); xuanpei@stu.xidian.edu.cn (X.P.); lchjiao@mail.xidian.edu.cn (L.J.); rhshang@mail.xidian.edu.cn (R.S.)

* Correspondence: yyli@xidian.edu.cn

Received: 24 December 2019; Accepted: 21 January 2020; Published: 25 January 2020

Abstract: Object detection has made significant progress in many real-world scenes. Despite this remarkable progress, the common use case of detection in remote sensing images remains challenging even for leading object detectors, due to the complex background, objects with arbitrary orientation, and large difference in scale of objects. In this paper, we propose a novel rotation detector for remote sensing images, mainly inspired by Mask R-CNN, namely RADet. RADet can obtain the rotation bounding box of objects with shape mask predicted by the mask branch, which is a novel, simple and effective way to get the rotation bounding box of objects. Specifically, a refine feature pyramid network is devised with an improved building block constructing top-down feature maps, to solve the problem of large difference in scales. Meanwhile, the position attention network and the channel attention network are jointly explored by modeling the spatial position dependence between global pixels and highlighting the object feature, for detecting small object surrounded by complex background. Extensive experiments on two remote sensing public datasets, DOTA and NWPUVHR -10, show our method to outperform existing leading object detectors in remote sensing field.

Keywords: remote sensing; arbitrary-oriented object detection; feature pyramid network; attention mechanism; mask

1. Introduction

Remote sensing image processing is a hot issue, which includes many types of tasks, such as image segmentation and object detection. Many scholars have proposed many methods, for example, in [1–4], researchers have proposed a series of machine learning-based image segmentation methods to improve SAR remote sensing image segmentation. In this paper, we mainly study the object detection of optical remote sensing image based on deep learning.

With the development of Deep Neural Network, object detection has made great progress in natural images in recent years. The object detection networks based on deep learning can be divided into two types: two-stage object detection networks and single-stage object detection networks. Most of the current two-stage object detectors are developed on the basis of region proposals with CNNs (R-CNN) [5]. In a two-stage framework of object detection, such as Faster R-CNN [6], category-independent region proposals generated from an image in the first stage. Based on the region proposals, features are extracted individually from the feature maps obtained by a CNN backbone for each region of interest (RoI). Then, the features are used to achieve category-specific classification and regression for the corresponding proposals. Finally, the final detection result is obtained through post-processing, such as non-maximum suppression. Faster R-CNN is a classical two-stage object detector, which is composed of Region Proposal Networks (RPN) and a detection network consists of classifiers and regressors, and can detect objects quickly and accurately in an end-to-end manner. Based on Faster R-CNN, more improved two-stage object detectors such as Region-based Fully Convolutional Networks (R-FCN) [7] and Mask R-CNN [8] were proposed. To further improve the efficiency of the object detector, Joseph Redmon et al. proposed a single stage target detector based on regression, called YOLO [9]. For the simple structure, You Only Look Once (YOLO) is extremely fast, but its accuracy is lower than that of the two-stage detector. Based on YOLO, YOLO v3 [10] and YOLO 9000 [11] were proposed successively. To trade off the detection speed and accuracy, Single Shot MultiBox Detector (SSD) [12] was proposed, whose speed and accuracy were between YOLO series algorithm and R-CNN series algorithm.

However, the above-mentioned classical object detectors aforementioned assume that the objects for detection are located along the horizontal line in images, which typically lead to misalignments between the bounding box and the object, especially in the field of remote sensing where the objects can be in any direction and position. To detect objects in any direction, several rotation object detectors have been proposed. In the field of scene text detection, Jiang et al. [13] proposed a Rotational Region Convolutional Neural Network (R2CNN), and achieved excellent results in scene text detection. RRPN [14] used rotation Anchor to get better proposals. Nevertheless, Scene text detection is a single-category object detection task, while there are often many different categories of objects in remote sensing images to be detected. Therefore, in the field of remote sensing, many multi-category arbitrary-oriented object detectors are proposed. Ref. [15] based on Faster R-CNN, detect arbitrary-oriented objects in remote sensing image by adding a rotation branch. Rotation Dense Feature Pyramid Networks (R-DFPN) [16] is an efficient multi-category rotation object detector due to Dense Feature Pyramid Network (DFPN) and Rotational Non maximum Suppression (R-NMS). It is worth noting that the above-mentioned rotation object detectors are realized by designing a rotation Anchor and adding a rotation regression branch. In fact, using rotation Anchors will dramatically increase the amount of computation of the model, while the method of rotation regression will reduce the robustness of the bounding box regression. In this paper, we use the object shape mask predicted by the instance segmentation branch of the network to obtain the rotation bounding box of the object, which is a novel method for detecting targets in any direction in remote sensing images. Although these rotation detectors have achieved target rotation bounding box predictions, the use of rotation anchors or the addition of rotation branches has greatly increased the calculation amount of the model, and has also made the model more complicated. In Table 1, we summarize the advantages and disadvantages of existing object detection methods.

In fact, arbitrary-oriented object detection can be realized more easily based on instance segmentation. Modern Instance segmentation methods, such as Mask R-CNN, are usually develop as a multi-task learning problem, which can effectively separate the object from the background. However, directly transfer Mask R-CNN to the remote sensing image object detection is likely to cause some problems, because Mask R-CNN is designed for natural images and is not good at solving the following three problems existing in remote sensing images: (1) Arbitrary orientations of objects. Due to the special imaging perspective, the objects in the remote sensing image exist in any direction. (2) Scales of objects vary greatly. Remote sensing image is taken from a long distance and wide angle. Therefore, an image will contain objects with large scale differences, such as the baseball diamond and the small vehicle shown in Figure 1a. (3) Complex background. Many objects of interest in large-scale remote sensing images are often surrounded by complex backgrounds, such as the swimming pool shown in Figure 1b. The complex background can seriously interfere with the detection of the object of interest.

Table 1. The advantages and disadvantages of existing object detection methods.

Types	Methods	Advantages	Disadvantages
Classical Detectors	Two-stage Detectors (e.g. Faster R-CNN, R-FCN, FPN, Mask R-CNN, etc.)	High detection precision, Low misdetection rate	Non-real-time detection, Locate objects with horizontal bounding box
	One-stage Detectors (e.g. YOLO, SSD, YOLO v2, YOLO v3, etc.)	Real-time detection, Simple network structure	Low detection precision, Locate objects with horizontal bounding box, Poor results for small and dense objects, Easy to mislocate
Rotation Detectors	e.g. R2CNN, RRPN, RDFPN, FR-O, etc.	Locate objects with rotation bounding box, Using rotaion anchors	Large model calculations, Greatly affected by artificial factors, Non-real-time detection, Complex

For the problem of target scale change, building a multi-layer network is the most effective strategy. As is known to all, the low-level high-resolution feature map of deep neural network can retain the location information of the object, while the high-level low-resolution feature map can provide rich semantic clues of the object. There are many methods to improve object detection and instance segmentation by using multiple scale feature maps. Fully convolutional networks (FCN) [17] improves semantic segmentation result by summing the partial scores for each category over multiple scales. Some other methods, such as HyperNet [18], ParseNet [19] and Inside-Outside Net (ION) [20], concatenate features of multiple layers to make predictions, which is equivalent to summing up features from different scale feature maps. Both SSD and MS-CNN [21] detect targets on multi-scale feature maps, without combining features or scores. Feature pyramid networks (FPN) [22] is a network that merges the lower-layer feature map with the higher-level feature map to get the multi-scale feature maps. It consists of a Bottom-up pathway and a Top-Down pathway. Based on SSD, [23] proposed RefineDet, which fuses the higher-level feature map of the backbone network in the SSD with the lower-layer feature map to obtain multi-scale feature maps for object detection.

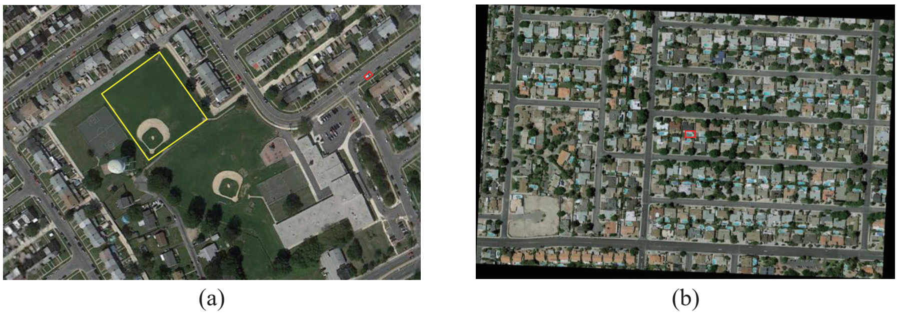


Figure 1. Two remote sensing images in DOTA dataset. (**a**) The yellow rectangle is the bounding box of the baseball diamond and the red rectangle is the bounding box of the small vehicle. There is a big difference in the scale between the two. (**b**) The red rectangle is the ding box of the swimming pool, from which it can be seen that the sparse swimming pools is surrounded by complex background.

Recently, attentional mechanism has been widely used in neural network models to improve the efficiency of the network. The essence of attention mechanism is human brain visual attention mechanism. According to cognitive neuroscience, attention can be divided into focus attention, which is active attention, refers to the purposeful and conscious focus on an object, and marked attention, which is passive attention, refers to the attention driven by external stimuli without active intervention. In artificial neural network, attention mechanism generally refers to focus attention. Ref. [24] used attention mechanism on Recurrent neural network (RNN) model to improve the performance of image classification. In [25], Bahdanau et al. used an attention-like mechanism to simultaneously translate and align on a machine translation task, allowing attention mechanism to be applied in Natural language processing (NLP) field. Intra attention proposed by [26] focuses on all positions in a sequence to get the response of a certain position in the sequence. Ref. [27] further demonstrated that machine translation model by self-attention can achieve excellent performance. Ref. [28] designed a non-local neural network (NLNet) to model pixel-level pairwise relationships with attention mechanism. Based on NLNet, Ref. [29] proposed Self-Attention Generative Adversarial Network (SAGAN), which allows attention-driven, long-range dependency modeling for image generation tasks. Ref. [30] achieved the feature weight of each channel in the feature map through global average pooling, and make the model pay different attention to each channel in the feature map. Squeeze-and-Excitation Networks (SENet) is often added to other networks as channel-attention to improve the efficiency.

Inspired by the related works mentioned above, we proposed a novel object detector of remote sensing images for the difficulties in remote sensing filed. First, we used the shape mask prediction of Mask R-CNN to locate the target area. Thus, our method can flexibly detect arbitrary-oriented objects, without any predefined rotation anchor. Second, in order to retain more positional information for small objects, we designed a refine feature pyramid network, which merges the high-layer semantic features with the low-layer positional features and obtains the multi-scale feature maps, solving the problem that the scales of objects in the same image vary greatly. Finally, inspired by the attention mechanism of the human brain, we designed a multi-layer attention network that enables the network to accurately detect small objects of interest from complex backgrounds and to focus on learning the features of small objects, just like focused attention in cognitive neuroscience.

Combined with the above techniques, our method can significantly improve detection performance. Furthermore, the proposed method can obtain the performance of mAP 69.09% on DOTA (a large remote sensing dataset), which is better than the previous leading algorithms. The contributions of this paper are as follows:

(1) For more robust handling of arbitrary-oriented objects, we use the instance segmentation branch of Mask R-CNN to generate shape masks of the objects, and then use them to determine the accurate object's rotation bounding box. Compared with the existing rotation detector, this is a simple and efficient method for obtaining a rotating bounding box, because it is not necessary to design a rotation anchor or a rotation branch in advance.

(2) Considering that the scales of objects vary greatly, a refined feature pyramid network is developed to merge the high-layer sematic features with the low-layer positional features and to get multi-scale feature maps. Compared with the existing multi-scale feature map methods, our refine feature pyramid network can effectively reduce the checkerboard effect or aliasing effect in feature fusing and improve the effectiveness of feature fusion.

(3) For complex background, a multi-layer attention network is designed to reduce the impact of background noise and to highlight target features. Compared with existing attention networks, the proposed multi-layer attention network simultaneously focus on the spatial position and features of objects, which is extremely helpful for the detection of small objects overwhelmed by complex backgrounds.

2. Proposed Methods

In this section, we will describe the various parts of our pipeline in detail. Figure 2 shows the overall framework of our method. Our pipeline consists of two key components: a Refine Feature Pyramid Network (RFPN) and a Multi-layer Attention Network(MANet), and is based on Mask R-CNN. Specifically, RFPN can generate a set of multi-scale feature maps by fusing features for each input image, and then MANet further suppresses background noise and highlight target features through attention mechanism. Then, obtaining the high-quality regional proposals from RPN for subsequent Fast R-CNN and mask branch. In the second stage, horizontal bounding box regression, class prediction, and shape mask prediction are obtained. Finally, the predicted shape masks are applied to calculate the object's rotation bounding box.

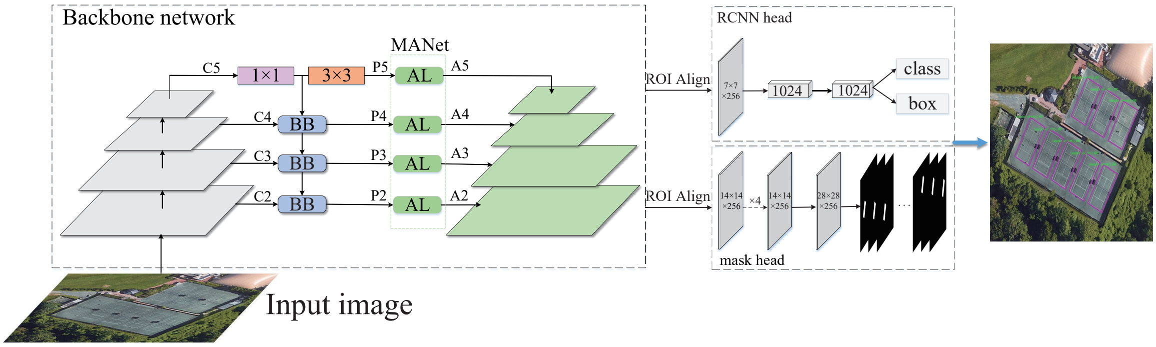


Figure 2. The overall framework of RADet. BB denotes the building block of Refine Feature Pyramid Network. AL denotes the attention layer of MANet (Multi-layer Attention Network).

2.1. Rotation Bounding Box Prediction Based on Mask

Mask R-CNN is an extension of Faster R-CNN, which can simultaneously achieve object detection and instance segmentation. This multi-task learning method can effectively improve the performance of object detection. In this paper, we use the object mask predicted by the mask branch of Mask R-CNN to obtain the rotation bounding box of the object, to achieve arbitrary-oriented object detection in remote sensing images.

2.1.1. Instance Label Generation

The instance label of remote sensing image is shown in Figure 3. Unlike natural image datasets such as COCO and Pascal VOC, which provide pixel level labels, remote sensing image object detection datasets only provide coordinate labels of object. Therefore, generating instance label is a precondition for using Mask R-CNN. In this paper, we get the polygon connected by the object's coordinates, and regarded the pixels in the polygon as the object, and the pixels outside the polygon as the non-object, then we get an instance label of the object. Although this approach will bring some noise, the implementation process is quite simple. This kind of instance labels has little effect on the final instance segmentation results as demonstrated by experiments.

2.1.2. Rotation Bounding Box Prediction

As we all know, the mask branch of Mask R-CNN will predict a shape mask for each object in the image. For the predicted shape mask, we calculate its minimum area rectangle and use it as the object's rotated bounding box. In this process, we can easily obtain the rotation bounding box of arbitrary-oriented object without using any rotation anchor. The prediction of rotation bounding box is:

$$x, y, h, w, \theta \quad = \quad \mathrm{minAreaRect}(mask) \tag{1}$$

$$\begin{aligned} x_0 = x - \tfrac{\sin\theta}{2} \cdot h - \tfrac{\cos\theta}{2} \cdot w, y_0 = y + \tfrac{\cos\theta}{2} \cdot h - \tfrac{\sin\theta}{2} \cdot w \\ x_1 = x + \tfrac{\sin\theta}{2} \cdot h - \tfrac{\cos\theta}{2} \cdot w, y_1 = x - \tfrac{\cos\theta}{2} \cdot h - \tfrac{\sin\theta}{2} \cdot w \end{aligned} \tag{2}$$

$$\begin{aligned} x_2 = 2x - x_0, y_2 = 2y - y_0 \\ x_3 = 2x - x_1, y_3 = 2y - y_1 \end{aligned} \tag{3}$$

where x, y, h, w, θ denote the rotation bounding box's center coordinates and its width, height and angle. minAreaRect$(\cdot)$ represents a function that computes the minimum area rectangle of the shape mask. (x_i, y_i) denote the ith coordinates of the rotation bounding box.

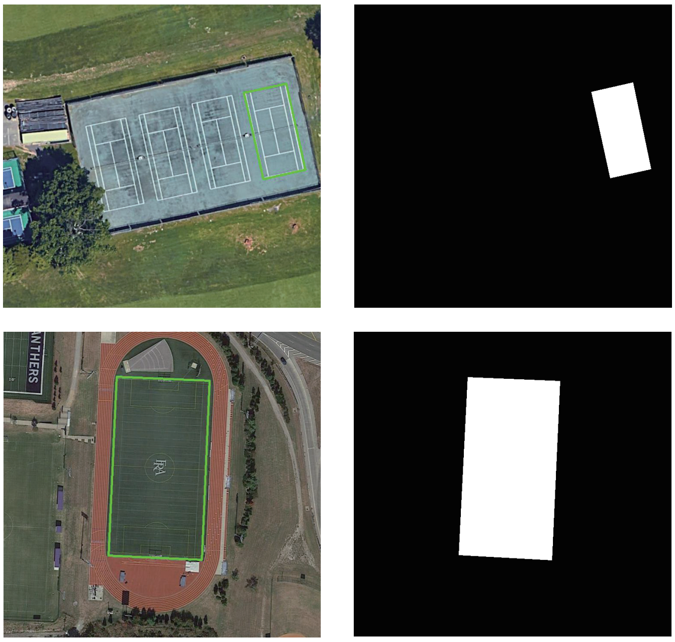


Figure 3. Ground Truth. (**Left**) Image samples with green polygon. (**Right**) Binary instance labels.

2.2. Refine Feature Pyramid Network

Now, there are many deep convolutional neural networks with strong ability to extract image features, such as ResNet [31]. However, due to pooling layers used in deep layers, small object will lose most of its positional features in deep layers, while the large object still retains good positional and semantic features in deep layers. Therefore, if the multi-scale feature maps with context information can be obtained by merging the high-level features with the low-level features, the problem of that scales of objects vary greatly in the same image can be solved.

Inspired by RefineDet, we designed a Refine Feature Pyramid Network, which can fuse the higher-layer feature maps with the lower-layer feature maps, and then obtain the multi-scale feature maps with rich context information. Figure 4 shows the improved building block that constructs our top-down feature maps. Moreover, similar to the idea of default box settings in SSD, we use single-scale different-ratio anchors at each level and use different-scale anchors on different levels. In other words, the large-scale anchors used in high-layer feature maps (small scale) is mainly responsible for large object detection, and the small-scale anchors used in low-layer feature maps (large scale) is mainly responsible for small object detection, so as to overcome the problem of that scale of objects varied greatly in remote sensing images.

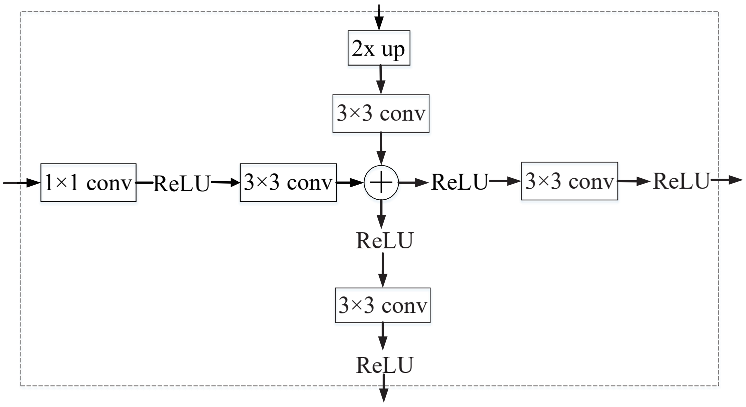


Figure 4. The building block of refine feature pyramid network. The ⊕ denotes element-wise addition.

Specifically, for Resnet, our Refine Feature Pyramid Network only acts on the feature activation output by the last residual block output at each stage of Resnet, which are denoted as $C2$, $C3$, $C4$ and $C5$. We all know that feature maps of the same size can be fused, so the high-level feature maps need to be up-sampled before being fused with the low-level feature maps. Interpolation and deconvolution are commonly used up-sampling methods. Ref. [23] used deconvolution to go from a low-resolution feature maps to a higher-resolution feature maps for fusion. However, unfortunately, deconvolution will produce uneven overlap (checkerboard artifacts) when up-sampling a feature map, as shown in Figure 5. The uneven overlap means that the convolution kernel operates more in some places than others. FPN uses nearest neighbor interpolation to obtain larger-sized feature maps, but it produces aliasing effects. In fact, convolution can filter out aliased high-frequency signals. Therefore, we use a combination of nearest neighbor interpolation and convolution instead of deconvolution or simple interpolation, which will effectively reduce the checkerboard effect or aliasing effect. Moreover, we use a 1×1 convolution (which can reduce channel dimensions) and a 3×3 convolution to further extract the low-layer detailed location information, and the ReLU layer is applied between the two convolution layers to obtain the non-linear representation. Then, we fuse the high-level semantic features with the low-level location features through element-wise addition, and obtain the merged map through a 3×3 convolution and two ReLU layers. This process is repeated until the finest resolution feature map is generated. Finally, we obtain a set of multi-scale feature maps corresponding to the merged maps of each layer, which is defined as $\{P2, P3, P4, P5\}$. It is worth noting that $P5$ is obtained by $C5$ through a 1×1 convolution and a 3×3 convolution, which is the same way as $P5$ in FPN.

Since all levels of multi-scale feature map is shared by RPN and detection head, we fixed the feature dimension (denoted as d) of each level feature map and all extra convolutional layers. In this study, we set $d = 256$, which can meet the requirement of a fixed number of feature map channels in each layer, and also reduce memory consumption and maintain good performance.

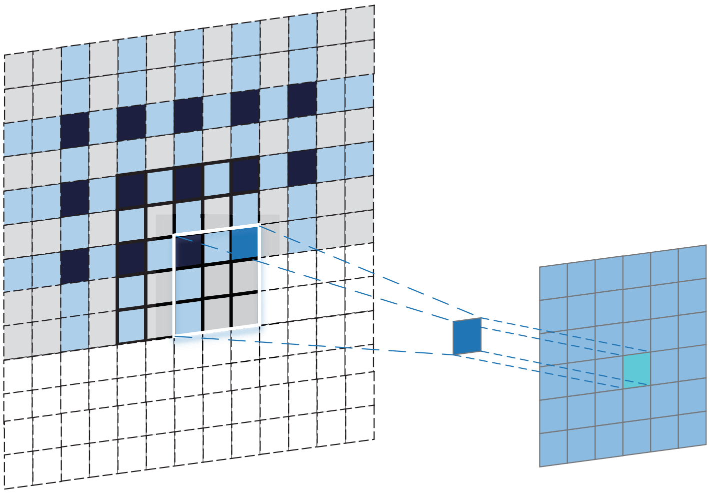


Figure 5. An example of a checkerboard artifacts. Checkerboard artifacts occurs when a 3×3 convolution deconvolution on a 5×5 feature map with stride = 2.

2.3. Multi-Layer Attention Network

Referring to the human brain's focus attention mechanism, we design a multi-layer attention network, which enables the network to focus on processing some key information or information of interest when faced with a large amount of input information, so as to improve the performance of network. The proposed multi-layer attention network consists of four identical attention layer, which are connected after $\{P2, P3, P4, P5\}$, and the output is $\{A2, A3, A4, A5\}$. As illustrated in Figure 6, each attention layer contains two parts: position attention block and channel attention block.

The position attention block is adopted to model the pairwise long-range dependencies, guiding the network to pay special attention to the location of the target. Then, the channel attention block aims to model the channel-wise relations, guiding the network to pay more attention to the features of targets, which are the key to determining which category the target belongs to.

As we all know, object detection is a visual task sensitive to position, i.e., once the position of the object in the image changes, the network needs to give a meaningful respond accordingly. However, convolutional neural network favors translation invariance—shift of an object inside an image should be indiscriminative, which is obviously contrary to the principle of object detection. The proposed position attention block can effectively model the relationships among widely separated spatial positions, making the network more sensitive to the position of targets, thus enhancing the network locating performance. On the other hand, the network distinguishes between objects and non-objects based on the learned features and classifies the objects correctly. Therefore, our goal is to enable the network to learn the importance of different features, and strengthen the learning of important features. To achieve this, we propose a channel attention block inspired by SENet [30]. In deep convolutional neural networks, each channel dimension of the feature map is learned by a convolution kernel due to the weight-sharing characteristics of the convolution. That is, different channels of the feature map represent different features learned from images. In fact, different features contribute differently to the network. To enhance features with high contributions (target features) and weaken features with low contributions (non-target features), our channel attention block, which follows the position attention block, will first quantifies the contribution of each feature in the feature map through global averaging pooling, and then aggregates it to the original input by broadcast element-wise addition, thus enabling the features with greater contributions to receive more attention.

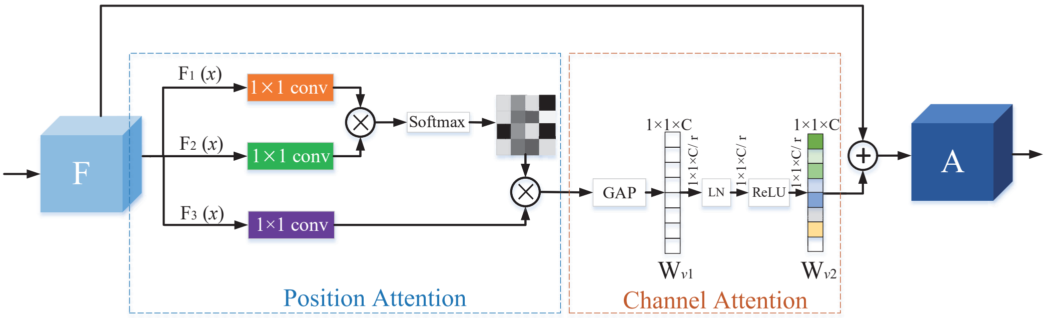


Figure 6. The overview of attention layer (part of the multi-layer attention network). The ⊕ denotes the broadcast element-wise addition.

In the position attention block, the input is the image feature of the previous hidden layer $x \in R^{C\times H\times W}$, x are then transformed into three feature spaces F_1, F_2 and F_3. First, the attention map that models long-range dependence between pixels is obtained through $F_1(x)$ and $F_2(x)$, where $F_1(x) = W_1x$ and $F_2(x) = W_2x$.

$$A_{i,j} = soft\max(S_{ij}) = \frac{\exp(S_{ij})}{\sum\limits_{j=1}^{N} \exp(S_{ij})} \tag{4}$$

where $S_{ij} = F_1(x_i)^T \otimes F_2(x_j)$, and $A_{i,j}$ indicates the pairwise relationship between position i and position j. Here, C, H and W are respectively the number of channels, height and width of the feature map of the previous hidden layer. Then, the obtained attention map is applied to the feature space F_3 to obtain the response $z \in R^{C\times H\times W}$ of each query position at all positions on the feature map, where,

$$z_i = \sum_{j=1}^{N} A_{i,j} \otimes F_3(x_j),\ F_3(x) = W_3x \tag{5}$$

In the above formula, $W_1 \in R^{C' \times C}, W_2 \in R^{C' \times C}, W_3 \in R^{C \times C}$ is the weight matrix learned by 1×1 convolution. i is the index of query position. $N = H \times W$ and N denotes number of feature locations. $\otimes$ denotes matrix multiplication. Because the feature spaces interact with each other through matrix multiplication, there will be a large memory footprint, especially for feature maps with large sizes. Therefore, we can improve memory efficiency by reducing the number of feature map channels. However, in order to ensure the same number of input and output channels in each attention layer, we can reduce C' to $C' = C/k, k = 1, 2, 4, 8, 16$. We found that when $k = 8$, the memory consumption is minimal and the performance loss is minimal. Therefore, in order to balance the efficiency and performance of the model, we set $C' = C/8$

The input of the channel attention block is the output of the position attention block. To make better use of the spatial location information learned by the location attention block, we first generate channel-wise statistics via a global average pooling, which squeezes the global spatial information into a channel descriptor. This process can be expressed by the Formula (6). Then, to fully capture channel-wise dependencies, we designed a transform architecture (Eqation (7)) that can meet two criteria: (1) it can learn a non-linear interaction between channels. (2) it can learn a non-mutual-exclusive relationship between channels.

$$s_c = \frac{1}{H \times W} \sum_{m=1}^{H} \sum_{n=1}^{W} z_c(m, n) \tag{6}$$

where z_c, H and W denotes channel c, height and width of the feature map z.

$$y = W_{v2}(\mathrm{Re}LU(LN(W_{v1}z))) \tag{7}$$

where LN denotes layer normalization that can ease normalization of the two-layer architecture for the transform block, and $W_{v1} \in R^{\frac{C}{r} \times C}$, $W_{v2} \in R^{C \times \frac{C}{r}}$. To make the channel attention block lighter, we reduced the dimension of the first fully connected (FC) layer by ratio r. In fact, there will be certain redundancy features in the FC layer. Therefore, setting r too small will affect the network performance and have a high memory consumption. When r is set too large, some important features may be lost, but the memory efficiency is high. In this paper we set $r = 4$, which can achieve the balance between efficiency and performance of the model.

In addition, to further enhance the features of each position, we use residual connections between inputs and outputs of each attention layer. Therefore, the final output of the channel attention block is $o_i = y_i \oplus x_i$, where $\oplus$ denotes the broadcast element-wise addition.

2.4. Loss Function

When training RPN, we assign a binary category label to each anchor. We assign a positive label to the anchors that meet two conditions: 1) the anchor has a highest Intersection-over-Union (IOU) overlap with a ground-truth box; Or 2) the IOU overlap between an anchor and the ground-truth box is greater than 0.7. When the IOU overlap with any ground-truth box is less than 0.3, the anchor is considered to be the background (non-object), and we assign a negative label to it. It is worth noting that anchors that are neither positive samples nor negative samples don not contribute to objective training. We minimize the objective function following the multi-task loss function of mask R-CNN, which is defined as follows:

$$L = L_{rpn} + \lambda_1 \cdot L_{cls} + \lambda_2 \cdot L_{reg} + \lambda_3 \cdot L_{mask} \tag{8}$$

where L_{rpn}, L_{cls}, L_{reg} and L_{mask} are the same as defined in [11], λ_1, λ_2 and λ_3 are the balance parameters between each task loss. In this paper, we set $\lambda_1 = \lambda_2 = \lambda_3 = 1.0$. Here, the mask branch of Mask R-CNN has a Km^2 dimensional output for each ROI, encoding K binary mask of resolution $m \times m$, one of each of K classes. To achieve this, we apply per-pixel sigmoid on each output of mask branch, and the L_{mask} is defined as the average binary cross-entropy loss. When an ROI is associated with ground-truth class k, the output mask only belongs to k class will contributes to the loss.

In addition, L_{reg} is defined as:

$$L_{reg} = L_{reg}(t, t^*) = \sum_{i \in \{x,y,w,h\}} smooth_{L_1}(t - t^*) \tag{9}$$

In which $smooth_{L_1}(x) = \begin{cases} 0.5x^2, & if|x|<1 \\ |x|-0.5, & otherwise \end{cases}$

For the bounding box regression, we adopt the parameterization of four coordinates, defined as follows:

$$\begin{aligned} t_x &= (x - x_a)/w_a, & t_y &= (y - y_a)/h_a, \\ t_w &= \log(w/w_a), & t_h &= \log(h/h_a), \\ t^*_x &= (x^* - x_a)/w_a, & t^*_y &= (y^* - y_a)/h_a, \\ t^*_w &= \log(w^*/w_a), & t^*_h &= \log(h^*/h_a) \end{aligned} \tag{10}$$

where x, y, w and h denotes box's center coordinates and its width and height. Variables x, x_a and x^* are for the predicted box, anchor box, and ground-truth box respectively (likewise for y, w, h).

3. Experiments and Results

In this section, we will introduce the dataset and implementation details used in our experiments. All experiments in this paper were implemented by Pytorch on a server with Nvidia Geforce GTX 2080Ti GPU and 11 G memory.

3.1. Datasets

We evaluate our approach on two public remote sensing datasets: DOTA and NWPUVHR-10. The datasets used for the experiments in this paper are briefly introduced as follows:

DOTA [15] is a large dataset for object detection in aerial images. It can be used for developing and evaluating object detectors in remote sensing field and contains 2806 aerial images from different sensors and platforms. Each image ranges in size from 800 × 800 to 4000 × 4000 pixels and contains a wide variety of scales, directions, and shapes. These DOTA images are then annotated by aviation image interpreters using 15 common object categories. The fully annotated DOTA benchmark contains 188,282 instances, each of which is labeled with an arbitrary quadrilateral. DOTA has two detection tasks: horizontal boundary box (HBB) and directional boundary box (OBB). To ensure that the training data and test data distributions approximately match, half of the original image were selected as the training set, 1/6 as the verification set, and 1/3 as the test set. We divided the DOTA images into sub-images of size 1024 × 1024, with an overlap of 200 pixels, and scaled it to 1333 × 800. We then removed the blank sample that did not contain any object. With all these processes, we obtain 10,276 patches for training, 3626 patches for validating and 10,833 patches for testing.

NWPUVHR-10 [32,33] dataset is a public detection dataset with 10 class geospatial objects for detection, which is only used for research purposes. The dataset contains a total of 800 very-high-resolution (VHR) remote sensing images culled from the Google earth and Vaihingen datasets, which are then manually annotated by experts using 10 common object categories. The 10 categories are airplane, ship, storage-tank, baseball-diamond, tennis-court, basketball-court, ground-track-field, harbor, bridge and vehicle. The NWPUVHR-10 dataset contains two sub-datasets: a positive dataset of 650 images and a negative dataset of 150 images. For the positive dataset, each image contains at least one object to be detection. Hence, we only use the positive dataset of NWPUVHR-10 dataset. Each image in the positive dataset is about 1000 pixels. In this paper, the split ratios of the training dataset, validation dataset and test dataset are 60%, 20% and 20%, respectively.

3.2. Implementation Detatils

3.2.1. Rpn

RPN is used to generate object proposals for subsequent fast R-CNN and mask branches. We adapt the RPN by replacing the singe-scale feature map with our multi-scale feature maps, and assign anchors of different sizes at different stages. Specifically, on five stages $\{A2, A3, A4, A5, A6\}$, the area of the anchors is set to $\{32^2, 64^2, 128^2, 256^2, 512^2\}$pixels, respectively. It is worth noting that A6 is obtained by A5 through max pooling. Meanwhile, different aspect ratios $\{1:2, 1:1, 2:1\}$ of anchors are adopted at each stage.

3.2.2. Training

Since Mask R-CNN is our baseline network, we set hyper-parameters mainly following Mask R-CNN. Our base network is ResNet 101 and is initialized with its pre-trained weights on ImageNet. All new layers are initialized with $kaimingnormal$. In training stage of all the experiments, we used SGD as the optimizer, with a batch size of 2 (the number of GPUs is 1 and each GPU calculates 2 images), momentum of 0.9 and weight decay of 0.0001. We train our model for 12 epochs with a learning rate of 0.0025, and use a linear warmup learning strategy to accelerate the network convergence. The warmup step is 500, and the learning rate will decrease to 0.00025 and 0.00003 at the 8th and 11th epoch. The mini-batch size of RPN and Fast R-CNN are set to 256 and 512 per image with 1:3 sample ratio of positives and negatives.

3.2.3. Inference

In the inference stage, first, RPN generates many object proposals. After NMS with a threshold of 0.7, 1000 object proposals are fed into Fast R-CNN. Then, Fast R-CNN further fine-tunes the target position according to the object proposals generated in the first stage, obtains object's category and horizontal candidate boxes by regression, and removes the redundant candidate boxes through the NMS with a threshold of 0.5. The kept candidate boxes are input to the mask branch to generate the shape mask maps of objects. Finally, the objects' rotation bounding box is generated based on the predicted shape mask.

3.2.4. Evaluation Indicators

To quantitatively evaluate the performance of the proposed method, we use the Average Precision (AP), precision-recall curve (PRC), and mean Average Precision (mAP) as the evaluation indicators for the experiments in this paper. AP, PRC and mAP are three well-known and widely applied indicators to evaluate the performance of object detection methods [34]. PRC can be obtained through four evaluation components: true positive (TP), false positive (FP), false negative (FN) and true negative (TN) [35]. TP and FP indicate the number of targets detected correctly and the number of targets detected incorrectly, respectively. FN represents the number of targets not detected. Based on these four evaluation components, the formulas for recall and accuracy are defined below:

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

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

AP is the average precision of the target in the range of recall = 0 to recall = 1, and is generally the area under PRC. mAP is the average value of AP values for all classes, and the larger the mAP value, the better the object detection performance.

3.3. Peer Methods Comparison

The proposed RADet with Refine Feature Pyramid Network and Multi-layer Attention Network is compared with other object detectors on two datasets: DOTA and NWPUVHR-10. The results show that Our model achieves competitive performance and outperforms other models.

3.3.1. Results on Dota

In addition to the official baseline given by DOTA, we also compared our results with R-DFPN [16], R2CNN [13], RRPN [14] and the method proposed by Yang et al. in [36]. The performance of these methods is shown in Table 2. As can be seen from Table 2, compared to other methods, due to the addition of the proposed Multi-layer Attention Network, RADet has a significant effect on improving the detection performance of small objects surrounded by complex backgrounds such as bridge, ship, swimming pool, small vehicle, and large vehicle. Moreover, with the proposed Refine Feature Pyramid Network, the detection performance of objects that may exist on the same image and have large scale differences, such as baseball diamond and small vehicle, and harbor and ship, can also be improved simultaneously. In conclusion, our method is better than the existing published results, reaching 69.09% mAP.

Some detection examples of RADet on DOTA dataset are shown in Figure 7. In Figure 7, it can be seen that the false alarm rate of the proposed RADet are very low, while recall rate is high. Figure 7 also shows that the proposed RADet can well deal with complex background noise, and can also detect targets with large scale changes.

Table 2. Overall performance evaluation of different methods on DOTA datasets. The short names for categories are defined as: PL-Plane, BD-Baseball diamond, BR-Bridge, GTF-Ground track field, 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. Ours * indicates the RADet with ResNext101. Bold numbers in Tables means the best results.

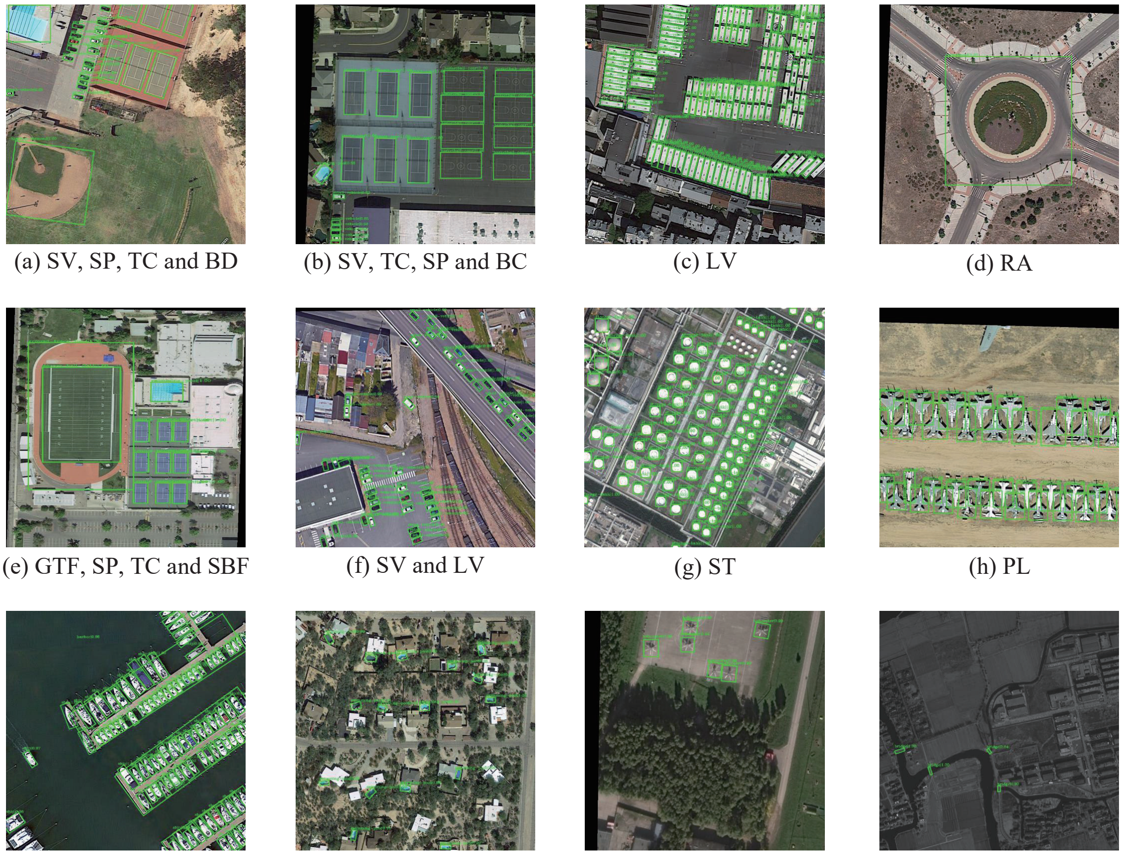
Method	PL	BD	BR	GTF	SV	LV	SH	TC	BC	ST	SBF	RA	HA	SP	HC	mAP
FR-O [15]	79.09	69.12	17.17	63.49	34.20	37.16	36.20	89.19	69.60	58.96	49.40	52.52	46.69	44.80	46.30	52.93
R-DFPN [16]	80.92	65.82	33.77	58.94	55.77	50.94	54.78	90.33	66.34	68.66	48.73	51.76	55.10	51.32	35.88	57.94
R2CNN [13]	80.94	65.67	35.34	67.44	59.52	50.91	55.81	90.67	66.92	72.39	55.06	52.23	55.14	53.35	48.22	60.67
RRPN [14]	**88.52**	71.20	31.66	59.30	51.85	56.19	57.25	**90.81**	72.84	67.38	**56.69**	52.84	53.08	51.94	53.58	61.01
Yang et al. [36]	81.25	71.41	36.53	67.44	61.16	50.91	56.60	90.67	68.09	72.39	55.06	55.60	62.44	53.35	51.47	62.29
Ours	79.66	**77.36**	47.64	**67.61**	65.06	74.35	68.82	90.05	74.72	**75.67**	45.60	61.84	64.88	68.00	53.67	67.66
Ours *	79.45	76.99	**48.05**	65.83	**65.46**	**74.40**	**68.86**	89.70	**78.14**	74.97	49.92	**64.63**	**66.14**	**71.58**	**62.16**	**69.09**

3.3.2. Results on Nwpuvhr-10

On the NWPUVHR-10 dataset, we evaluate our method using AP, mAP and PRC as evaluation indicators. Table 3 shows the overall performance comparison results of our method and other classical object detection algorithms on the NWPUVHR-10 dataset. There is no doubt that our method also achieves the first place on the NWPUVHR-10 dataset, with 90.24% mAP. In Figure 8, it can be seen that our method achieved the best detection results in more than half of the categories, such as airplane, vehicle, basketball court, ground track field, baseball diamond and tennis court. In short, by comprehensively analyzing the AP values, mAP values and PRCs, we can see that our RADet has achieved the best detection performance.

Table 3. Overall performance evaluation of different methods on NWPUVHR-10 datasets. Bold numbers in Tables means the best results.

Method	SSD512 [12]	Faster R-CNN [6]	FPN [7]	Ours
Tennis-court	33.85	79.77	86.69	**90.74**
Vehicle	45.45	81.02	89.39	**89.98**
Harbor	32.95	**79.37**	69.52	78.01
Basketball-court	61.85	79.96	90.60	**97.46**
Ground-track-field	99.31	90.67	90.42	**99.53**
Bridge	45.45	59.93	**79.49**	77.05
Ship	53.90	**81.82**	81.40	81.39
Airplane	90.91	90.91	90.91	**100.00**
Storage-tank	93.06	97.89	**98.95**	97.90
Baseball-diamond	90.35	90.24	90.12	**90.36**
mAP	64.71	83.25	86.75	**90.24**



(a) SV, SP, TC and BD

(b) SV, TC, SP and BC

(c) LV

(d) RA

(e) GTF, SP, TC and SBF

(f) SV and LV

(g) ST

(h) PL

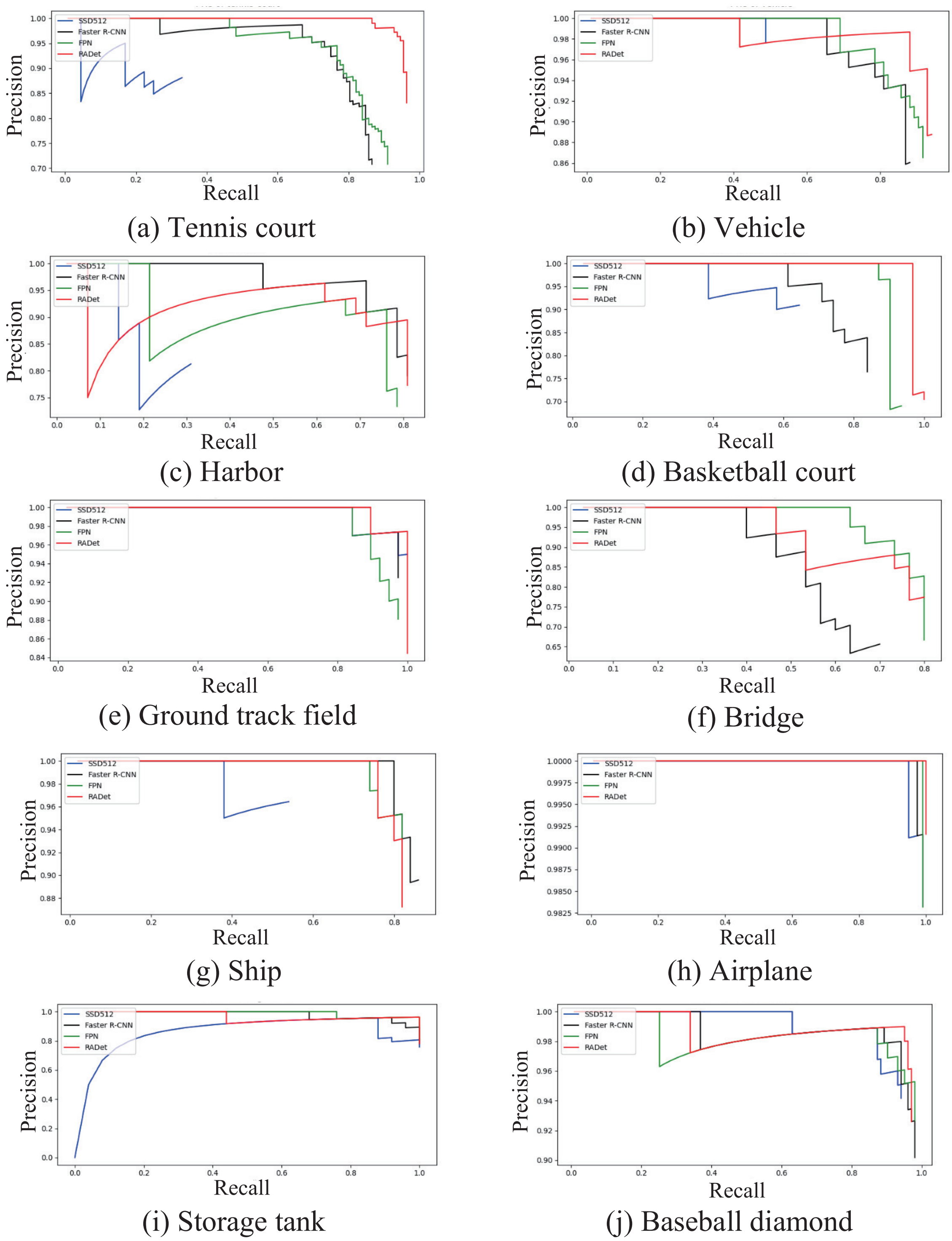
(i) SP and HA

(j) SP

(k) HC

(l) BR

Figure 7. Examples of RADet's detection results on DOTA dataset. (**a**) Detection results of Small vehicle (SV), Swimming pool (SP), Tennis court (TC) and Baseball diamond (BD). (**b**) Detecton results of Small vehicle (SV), Tennis court (TC), Swimming pool (SP) and Basketball court (BC). (**c**) Detection results of Large vehicle (LV). (**d**) Detection results of Roundabout (RA). (**e**) Detection results of Ground track field (GTF), Swimming pool (SP), Tennis court (TC) and Soccer-ball field (SBF). (**f**) Detection results of Small vehicle (SV) and Large vehicle (LV). (**g**) Detection results of Storage tank (ST). (**h**) Detection results of Plane (PL). (**i**) Detection results of Swimming pool (SP) and Harbor (HA). (**j**) Detection results of Swimming pool (SP). (**k**) Detection results of Helicopter (HC). (**l**) Detection results of Bridge (BR).



(a) Tennis court

(b) Vehicle

(c) Harbor

(d) Basketball court

(e) Ground track field

(f) Bridge

(g) Ship

(h) Airplane

(i) Storage tank

(j) Baseball diamond

Figure 8. The P-R curves of RADet and other classical object detection algorithms on NWPUVHR 10 dataset. (**a**) P-R curves on Tennis court category. (**b**) P-R curves on Vehicle category. (**c**) P-R curves on Harbor category. (**d**) P-R curves on Basketball court category. (**e**) P-R curves on Ground track field category. (**f**) P-R curves on Bridge category. (**g**) P-R curves on Ship category. (**h**) P-R curves on Airplane category. (**i**) P-R curves on Storage tank category. (**j**) P-R curves on Baseball diamond category.

3.4. Ablation Study

3.4.1. Quantitative Analysis

To verify the effectiveness of the proposed approach, we do two sets of ablation experiments on the test set of the DOTA dataset. All results were obtained by submitting the prediction results to the official DOTA evaluation server. In both sets of ablation experiments, we used AP and mAP as evaluation indicators. Table 4 shows the results of our model on the DOTA dataset in two different up-sampling methods of the Refine Feature Pyramid Network. Table 5 summarizes the results of our model with different settings on our DOTA dataset.

Baseline setting. We chose mask R-CNN with FPN as our baseline. For fairness, all our experiments use ResNet101 as the base model, and all the experimental data and parameter settings are strictly consistent.

Effect of Refine Feature Pyramid Network. We replace the FPN in the baseline with the proposed Refine Feature Pyramid Network, which can increase the total mAP by 0.54%. As discussed in Section 2.2, our resize-convolution can effectively reduce the checkerboard effect generated during the up-sampling process, which can also be proved by the results in Table 4. Compared with Refine Feature Pyramid Network using deconvolution, the Refine Feature Pyramid Network using resize-convolution can increase mAP by 0.78% , which shows that resize-convolution can effectively reduce the checkerboard effect analyzed in Section 2.2.

Table 4. Results of ablative study of different component for up-sampling in our proposed Refine Feature Pyramid Network on the DOTA dataset.

Method	Deconvolution	Resize-convolution	mAP
RFPN with different component	✓		64.86
		✓	65.64

Table 5. Results of ablative study of different component in RADet. The +RFPN means that we replace the FPN in baseline with our proposed Refine Feature Pyramid Network, and +MANet means that we add our proposed Multi-layer Attention Network to baseline. The short names here are the same as that in Table 2. Bold numbers in Tables means the best results.

Method	PL	BD	BR	GTF	SV	LV	SH	TC	BC	ST	SBF	RA	HA	SP	HC	mAP
Baseline	79.78	74.88	44.13	63.22	63.60	67.25	68.56	90.01	69.44	67.76	44.65	**64.09**	63.84	64.89	50.34	65.10
+ RFPN	79.84	75.91	43.08	65.22	**65.11**	72.93	**69.09**	**90.69**	68.97	68.86	43.58	63.10	64.62	67.59	46.02	65.64 (↑ 0.54)
+ MANet	**80.03**	75.32	43.58	62.47	64.13	72.77	68.74	90.19	70.29	73.51	**51.26**	61.24	64.44	**68.04**	43.75	65.98 (↑ 0.88)
+ RFPN + MANet	79.66	**77.36**	**47.64**	**67.61**	65.06	**74.35**	68.82	90.05	**74.72**	**75.67**	45.60	61.84	**64.88**	68.00	**53.67**	**67.66 (**↑ 2.56**)**

Effect of Multi-layer Attention Network. To further effectively suppress the influence of background noise and highlight the object feature, we propose Multi-layer Attention Network. The results in Table 5 show that our Multi-layer Attention Network can significantly improve the detection results of small objects such as swimming pool and storage tank that may be surrounded by complex background. Adding our proposed Multi-layer Attention Network to the baseline can increase the total mAP of the model by 0.88% to 65.98%, and increase the AP of the storage tank category by 5.75%, and the AP of the swimming pool category by 3.15%. In addition, adding Multi-layer Attention Network to our model using Refine Feature Pyramid Network can also improve the performance of the model, which further demonstrate the effectiveness and portability of Multi-layer Attention Network.

3.4.2. Qualitative Analysis

Qualitative analysis of Resize-convolution. It can be seen from Figure 9b that using deconvolution for up-sampling will produce serious uneven overlap (checkerboard artifacts), which will affect the detection performance of the final network, as shown in Table 4. Similarly, as shown in Figure 9c using only nearest neighbor interpolation for up-sampling also produces some aliasing effects. Our resize-convolution (combination of nearest neighbor interpolation and convolution) works best, as shown in Figure 9d, because 3 × 3 convolution can filter out some aliased high-frequency signals.

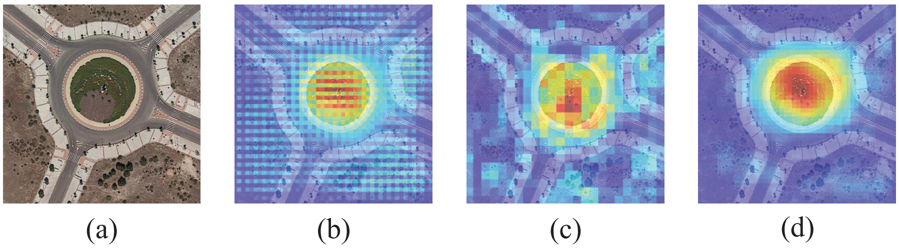


Figure 9. The heatmaps of different component for up-sampling. (**a**) Input image. (**b**) The heatmap after up-sampling with deconvolution. (**c**) is the heatmap after up-sampling with nearest neighbor interpolation. (**d**) is the heatmap after up-sampling with nearest neighbor interpolation and convolution (resize-convolution).

Qualitative analysis of Multi-layer Attention Network. Due to the complexity of real-world data such as remote sensing images, there will be a lot of noise information near the objects. Extensive noise will overwhelm the object information and the boundary between objects will be blurred, as shown in Figure 10b, leading to missed detection and increasing false alarms. It can be seen from Figure 10c that our proposed Multi-layer Attention Network can effectively suppress background noise and highlight object information, which is helpful to improve the final detection performance of the model. In addition, it can be seen from Figure 10d that due to the position attention block, the proposed RADet can pay attention to some information around the swimming pool, such as small vehicles and houses, which greatly helps the precise positioning of the swimming pool.

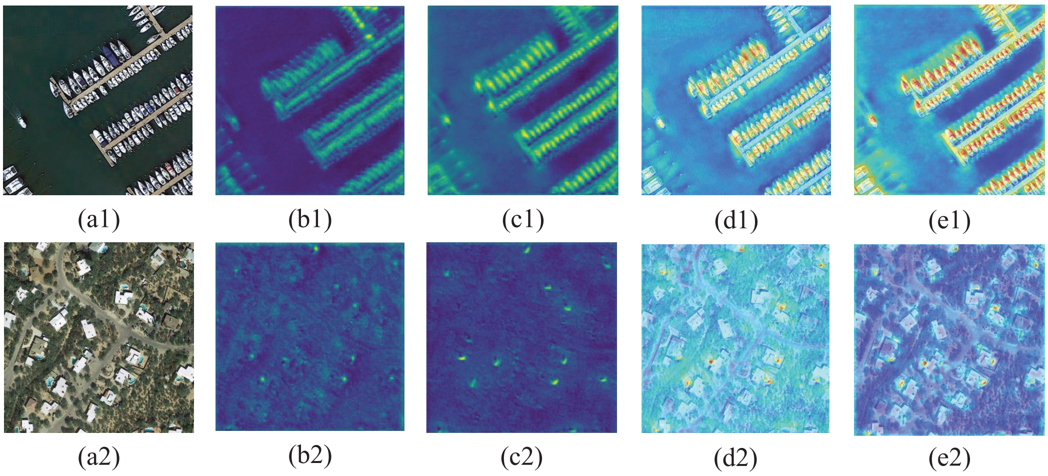


Figure 10. Visualizations of Multi-layer Attention Network. (**a1**,**a2**) The input images. (**b1**,**b2**) The input feature maps of Multi-layer Attention Network. (**c1**,**c2**) The output feature maps of Multi-layer Attention Network. (**d1**,**d2**) The heatmaps of the input feature maps of Multi-layer Attention Network. (**e1**,**e2**) The heatmaps of the output feature maps of Multi-layer Attention Network.

4. Discussion

4.1. Effectiveness of Refine Feature Pyramid Network and Multi-Layer Attention Network on Faster R-Cnn

To further verify the effectiveness of the proposed Refine Feature Pyramid Network (RFPN) and Multi-layer Attention Network (MANet), we added it to Faster R-CNN and performed experiments on the DOTA test set. The experimental results are shown in Table 6. The larger the mAP, the better the detection performance of the model. As can be seen from Table 6, the mAP of Faster R-CNN increased to 61.79% (increased by 1.33%) after adding the Refine Feature Pyramid Network, and increased to 61.82% (increased by 1.36%) after adding Multi-layer Attention Network, which further proves the effectiveness of the proposed Refine Feature Pyramid Network and Multi-layer Attention Network. In fact, although RFPN and MANet can improve the overall detection performance of the model, from the perspective of the AP of each class, RFPN and MANet have weakened the performance of Faster R-CNN in detecting large targets to some extent.

Table 6. The efficientiveness of RFPN and MANet on Faster R-CNN. The short names here are the same as that in Table 2.

Method	PL	BD	BR	GTF	SV	LV	SH	TC	BC	ST	SBF	RA	HA	SP	HC	mAP
Faster R-CNN [6]	80.32	77.55	32.86	68.13	53.66	52.49	50.04	90.41	75.05	59.59	57.00	49.81	61.69	56.46	41.85	60.46
Faster R-CNN + RFPN	79.29	75.95	47.97	58.54	54.88	50.10	52.14	79.93	59.96	68.77	42.63	64.04	66.04	69.14	57.44	61.79
Faster R-CNN + MANet	79.65	74.29	49.63	55.59	55.02	50.16	52.29	79.60	66.57	68.59	39.50	63.15	65.50	71.87	55.84	61.82

4.2. Sensitivity Analysis of Nms Threshold for Radet

Non-maximum suppression (NMS) is the most commonly used post-processing method in object detection field. NMS can eliminate redundant boxes, leaving the best object detection position. Therefore, there is a necessary correlation between the non-maximum suppression threshold and the final detection performance of the object detection algorithm. Figure 11 analyzes the effect of NMS threshold used in post-processing of the proposed RADet on AP of each category and mAP of the algorithm. In Figure 11, it can be seen that when the NMS threshold is too high or too low, both AP and mAP show a downward trend, i.e., the detection effect of the algorithm shows a tendency to deteriorate. This is because that when the NMS threshold is too high, fewer redundant boxes are removed and the possibility of false detection is high; when the NMS threshold is too low, many redundant boxes are removed, the recall rate is low, and the possibility of missed detection is high.

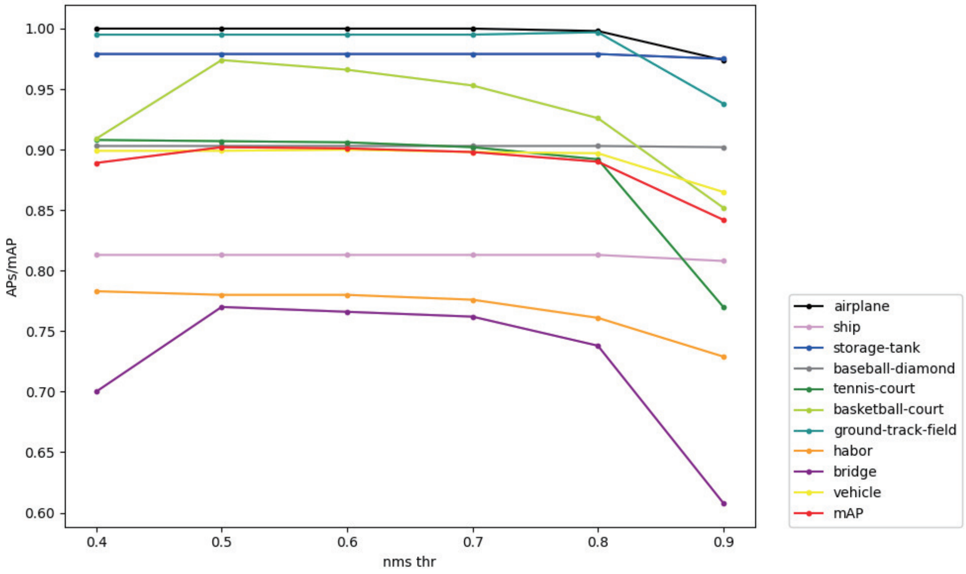


Figure 11. The impact of nms threshold on mAP value of RADet and AP values of different categories.

5. Conclusions

In this paper, we propose an end-to-end multi-category detector designed for arbitrary-oriented objects in remote sensing images. Our method is improved based on Mask R-CNN and obtain the rotation bounding box of the objects through the shape mask predicted by the network. In addition, considering that the scales of objects in remote sensing images vary greatly, we adopt the backbone of the pyramid structure to obtain multi-scale feature maps and further improve the up-sampling method to reduce the checkerboard effect produced by deconvolution. Based on this, we propose a Refine Feature Pyramid Network, which can overcome the difficulty of large differences in object's scale and effectively reduce the checkerboard effect. Moreover, the proposed RADet weakens the influence of noise from complex background and highlights the object features through the proposed Multi-layer Attention Network, which can further improve the detection performance of small objects surrounded by complex backgrounds. Our method achieved the best detection performance on two public remote sensing image datasets: DOTA and NWPUVHR-10.

There is no doubt that there is still room for improvement in our approach. Since our method obtains the object's rotation bounding box based on the predicted shape mask, once the shape mask of the object is not well predicted, it will affect the quality of the rotation bounding box. In addition, like most two-stage target detectors, our method does not implement real-time detection. Therefore, in the future, we are interested in the following directions: 1) Further improve the mask branch to obtain a better shape mask. 2) Implement RADet in anchor free mode, to make RADet lighter and more flexible.

Author Contributions: Methodology, Q.H.; Data precessing and Experimental results analysis, Q.H., X.P. and Y.L.; Oversaw and Suggestions, Y.L. and L.J.; Writing–review and editing, Y.L. and Q.H.; investigation, Q.H., X.P., Y.L. and R.S.; funding acquisition, Y.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China under Grant 61772399, Grant U1701267, Grant 61773304, Grant 61672405 and Grant 61772400, the Key Research and Development Program in Shaanxi Province of China under Grant 2019ZDLGY09-05, the Program for Cheung Kong Scholars and Innovative Research Team in University Grant IRT_15R53, and the Technology Foundation for Selected Overseas Chinese Scholar in Shaanxi under Grant 2017021 and Grant 2018021.

Acknowledgments: The authors would also like to express their sincere thanks to the editors and the anonymous reviewers for their valuable comments and contributions.

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

References

1. Braga, A.M.; Marques, R.C.; Rodrigues, F.A.; Medeiros, F.N. A median regularized level set for hierarchical segmentation of SAR images. *IEEE Geosci. Remote Sens. Lett.* **2017**, *14*, 1171–1175. [CrossRef]
2. Jin, R.; Yin, J.; Zhou, W.; Yang, J. Level set segmentation algorithm for high-resolution polarimetric SAR images based on a heterogeneous clutter model. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2017**, *10*, 4565–4579. [CrossRef]
3. Lang, F.; Yang, J.; Yan, S.; Qin, F. Superpixel Segmentation of Polarimetric Synthetic Aperture Radar (SAR) Images Based on Generalized Mean Shift. *Remote Sens.* **2018**, *10*, 1592. [CrossRef]
4. Ciecholewski, M. River channel segmentation in polarimetric SAR images: Watershed transform combined with average contrast maximisation. *Expert Syst. Appl.* **2017**, *82*, 196–215. [CrossRef]
5. 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 (CVPR 2014), Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
6. 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]
7. Dai, J.; Li, Y.; He, K.; Sun, J. R-fcn: Object detection via region-based fully convolutional networks. In Proceedings of the Advances in Neural Information Processing Systems (NIPS), Barcelona, Spain, 5–10 December 2016; pp. 379–387.
8. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask r-cnn. In Proceedings of the IEEE International Conference on Computer Vision (ICCV 2017), Venice, Italy, 22–29 October 2017; pp. 2961–2969.
9. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788. [CrossRef]
10. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv Prepr.* **2018**, arXiv:1804.02767.

11. Redmon, J.; Farhadi, A. YOLO9000: Better, faster, stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2017), Honolulu, HI, USA, 21–26 July 2017; pp. 7263–7271.
12. 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, Amsterdam, The Netherlands, 8–16 October 2016; pp. 21–37.
13. 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 Prepr.* **2017**, arXiv:1706.09579.
14. 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]
15. 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 (CVPR 2018), Salt Lake City, UT, USA, 18–23 June 2018; pp. 3974–3983.
16. Yang, X.; Sun, H.; Fu, K.; Yang, J.; Sun, X.; Yan, M.; Guo, Z. Automatic ship detection in remote sensing images from google earth of complex scenes based on multiscale rotation dense feature pyramid networks. *Remote Sens.* **2018**, *10*, 132. [CrossRef]
17. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 640–651.
18. Kong, T.; Yao, A.; Chen, Y.; Sun, F. Hypernet: Towards accurate region proposal generation and joint object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2016), Las Vegas, NV, USA, 27–30 June 2016; pp. 845–853.
19. Liu, W.; Rabinovich, A.; Berg, A.C. Parsenet: Looking wider to see better. *arXiv Prepr.* **2015**, arXiv:1506.04579.
20. Bell, S.; Lawrence Zitnick, C.; Bala, K.; Girshick, R. Inside-outside net: Detecting objects in context with skip pooling and recurrent neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2016), Las Vegas, NV, USA, 27–30 June 2016; pp. 2874–2883.
21. Cai, Z.; Fan, Q.; Feris, R.S.; Vasconcelos, N. A unified multi-scale deep convolutional neural network for fast object detection. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; pp. 354–370.
22. 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 (CVPR 2017), Honolulu, HI, USA, 21–26 July 2017; pp. 2117–2125.
23. Zhang, S.; Wen, L.; Bian, X.; Lei, Z.; Li, S.Z. Single-shot refinement neural network for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2018), Salt Lake City, UT, USA, 18–23 June 2018; pp. 4203–4212.
24. Mnih, V.; Heess, N.; Graves, A.; Kavukcuoglu, K. Recurrent models of visual attention. In Proceedings of the Advances in Neural Information Processing Systems (NIPS), Montréal, QC, Canada, 8–13 December 2014; pp. 2204–2212.
25. Bahdanau, D.; Cho, K.; Bengio, Y. Neural machine translation by jointly learning to align and translate. *arXiv Prepr.* **2014**, arXiv:1409.0473.
26. Cheng, J.; Dong, L.; Lapata, M. Long short-term memory-networks for machine reading. *arXiv Prepr.* **2016**, arXiv:1601.06733.
27. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In Proceedings of the Advances in Neural Information Processing Systems (NIPS), Long Beach, CA, USA, 4–10 December 2017; pp. 5998–6008.
28. Wang, X.; Girshick, R.; Gupta, A.; He, K. Non-local neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2018), Salt Lake City, UT, USA, 18–23 June 2018; pp. 7794–7803.
29. Zhang, H.; Goodfellow, I.; Metaxas, D.; Odena, A. Self-attention generative adversarial networks. *arXiv Prepr.* **2018**, arXiv:1805.08318.
30. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2018), Salt Lake City, UT, USA, 18–23 June 2018; pp. 7132–7141.

31. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2016), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
32. Cheng, G.; Han, J.; Zhou, P.; Guo, L. Multi-class geospatial object detection and geographic image classification based on collection of part detectors. *Isprs J. Photogramm. Remote Sens.* **2014**, *98*, 119–132. [CrossRef]
33. Cheng, G.; Zhou, P.; Han, J. Learning rotation-invariant convolutional neural networks for object detection in VHR optical remote sensing images. *IEEE Trans. Geosci. Remote Sens.* **2016**, *54*, 7405–7415. [CrossRef]
34. Han, J.; Zhou, P.; Zhang, D.; Cheng, G.; Guo, L.; Liu, Z.; Bu, S.; Wu, J. Efficient, simultaneous detection of multi-class geospatial targets based on visual saliency modeling and discriminative learning of sparse coding. *Isprs J. Photogramm. Remote. Sens.* **2014**, *89*, 37–48. [CrossRef]
35. Kang, M.; Ji, K.; Leng, X.; Lin, Z. Contextual region-based convolutional neural network with multilayer fusion for SAR ship detection. *Remote Sens.* **2017**, *9*, 860. [CrossRef]
36. Yang, X.; Sun, H.; Sun, X.; Yan, M.; Guo, Z.; Fu, K. Position detection and direction prediction for arbitrary-oriented ships via multitask rotation region convolutional neural network. *IEEE Access* **2018**, *6*, 50839–50849. [CrossRef]