sensors

MDPI

Article

Multiscale Rotated Bounding Box-Based Deep Learning Method for Detecting Ship Targets in Remote Sensing Images

Shuxin Li, Zhilong Zhang *, Biao Li and Chuwei Li

ATR National Key Laboratory, National University of Defense Technology, Changsha 410073, China; lishuxin@nudt.edu.cn (S.L.); libiao_cn@163.com (B.L.); lichuwei17@nudt.edu.cn (C.L.)

* Correspondence: zhangzhilong@nudt.edu.cn

Received: 9 July 2018; Accepted: 14 August 2018; Published: 17 August 2018

Abstract: Since remote sensing images are captured from the top of the target, such as from a satellite or plane platform, ship targets can be presented at any orientation. When detecting ship targets using horizontal bounding boxes, there will be background clutter in the box. This clutter makes it harder to detect the ship and find its precise location, especially when the targets are in close proximity or staying close to the shore. To solve these problems, this paper proposes a deep learning algorithm using a multiscale rotated bounding box to detect the ship target in a complex background and obtain the location and orientation information of the ship. When labeling the oriented targets, we use the five-parameter method to ensure that the box shape is maintained rectangular. The algorithm uses a pretrained deep network to extract features and produces two divided flow paths to output the result. One flow path predicts the target class, while the other predicts the location and angle information. In the training stage, we match the prior multiscale rotated bounding boxes to the ground-truth bounding boxes to obtain the positive sample information and use it to train the deep learning model. When matching the rotated bounding boxes, we narrow down the selection scope to reduce the amount of calculation. In the testing stage, we use the trained model to predict and obtain the final result after comparing with the score threshold and nonmaximum suppression post-processing. Experiments conducted on a remote sensing dataset show that the algorithm is robust in detecting ship targets under complex conditions, such as wave clutter background, target in close proximity, ship close to the shore, and multiscale varieties. Compared to other algorithms, our algorithm not only exhibits better performance in ship detection but also obtains the precise location and orientation information of the ship.

Keywords: multiscale rotated bounding box; deep learning; ship detection; remote sensing image

1. Introduction

Ships are important maritime targets and monitoring the ship target is valuable in civil and military applications [1]. With the development of optical remote sensing, the resolution of remote sensing images has gradually increased, enabling the detection and recognition of ship targets. A conventional approach to interpret the optical remote sensing images relies on human expertise, which, however, faces difficulty in satisfying the need of processing mass image data. Therefore, it is necessary to study the problems involved in ship detection.

There are three difficulties in detecting ships from the optical remote sensing images. First, ships vary in appearance and size. The base shape of the ship target is long spindle, but ships of different categories have different scales, aspect ratios, and appearances. Second, different ship targets have different orientations. As remote sensing images are captured above the target from the sky platform, ship targets have many orientations, which make ship detection harder. Third, there is complex background clutter around a ship target, such as wave interference, ships in close proximity, and land background when the ships are close to the shore. There are two conventional approaches for detecting ships. The first is to detect the ship target through heuristic artificially designed multistep methods [1–4]. Relying on the difference between the ship and the background, these algorithms analyze the region geometric features, such as aspect ratio and scale, to decide in which region the ship target is after segmentation of ocean and land. However, artificially designed steps result in a lack of robustness. The other methods [5–8] rely on machine learning to train the classifier using features extracted from positive and negative samples. When testing a new image, sliding windows or regions of interest which is produced by a pre-detection method, such as Selective Search, are sent to the classifier to predict the class score, and then, a threshold is set to obtain the final detection result. Using a large scale of positive and negative samples to train the classifier, the performance of the detector is made more robust in complex backgrounds, which is usually used in target verification stage [9]. However, most methods use horizontal bounding boxes to detect the target, which is not suitable for more complex conditions such as targets in close proximity and when the ship is close to the shore.

Recently, deep learning-based methods have made significant progress in computer vision applications. Combining a large-scale dataset and high-performance computing hardware GPUs, deep learning methods have undergone a leap in development in areas such as target detection [10–14], target classification [15,16], and semantic segmentation [17,18]. Deep learning-based detection methods exhibit high performance in detecting common objects in daily images. For example, the Faster R-CNN algorithm [12] adopts a two-stage pipeline to detect objects. The first stage produces the regions of interest from the feature maps extracted through a pretrained network, and the second stage selects the features of regions of interest from the shared feature maps and predicts a more precise classification and localization. The SSD algorithm [13] uses end-to-end training networks to predict the class name of the target and regress the bounding box information from multiscale feature maps, which is produced by the hierarchical down sampling structure of deep network. Combined with the strategy of data augmentation, the performance of SSD is close to that of the Faster R-CNN algorithm and SSD exhibits faster detection than Faster R-CNN. Though exhibiting good performance, the above two algorithms use a horizontal bounding box to detect objects, which is not suitable for rotated objects in close proximity.

In the area of ship detection in remote sensing images, there are also deep learning-based methods [19–23], which use deep neural networks to extract robust features from the image. However, most of them are also based on a horizontal bounding box. Recently, deep learning-based methods using a rotated bounding box have been proposed to detect rotated remote sensing targets. Liu et al. proposed a DRBox algorithm [24] to combine the rotated bounding box with deep learning methods to realize rotation-invariant detection. This algorithm can effectively detect airplane, vehicle, and ship targets in remote sensing images and output the precise location and orientation information of the target. However, the approximate calculation in matching the rotated bounding boxes makes the performance unstable. Xia et al. proposed an improved Orientated Bounding Box based (OBB) Faster R-CNN algorithm [25] using eight parameters, which include the coordinates of the four corner points $(x_1, y_1, x_2, y_2, x_3, y_3, x_4, y_4)$, to record the rotated bounding box based on the Faster R-CNN algorithm. The algorithm can effectively detect the oriented targets, but the predicted bounding box sometimes becomes deformed as there is no constraint to maintain the rectangular shape of the box. Figure 1 shows a comparison of the target with horizontal and rotated bounding boxes. When using horizontal bounding boxes to label targets in close proximity, it is difficult to distinguish the rotated targets, while using rotated boxes, they can be clearly distinguished.

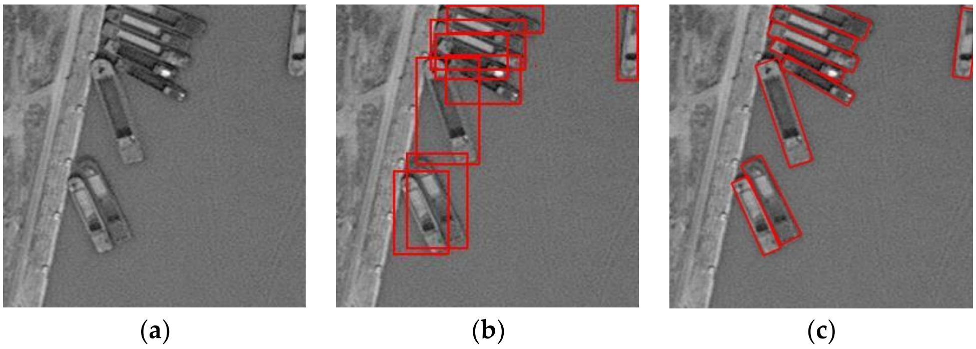


Figure 1. Comparison of target with horizontal and rotated bounding boxes. (**a**) Input image, (**b**) targets with horizontal bounding boxes, and (**c**) targets with rotated bounding boxes.

This paper proposes a rotated bounding box-based deep learning algorithm to detect ship targets in remote sensing images. We aim to detect rotated ships in a complex background more effectively. Based on the deep networks, we combine the rotated bounding box with the convolutional feature maps to simultaneously predict the location and orientation of the ship target. When labeling the ship targets, we add an angle parameter and use five parameters, including two center point coordinates, height, width, and angle of the box, to record a rotated box, which will limit the box to have a rectangular shape. In the training stage, we first set prior rotated boxes at each location of feature maps produced by networks and match them with the ground-truth boxes to obtain the positive sample information. Through the range control strategy, we narrow down the selection scope and the amount of calculation required for the matching is reduced. In the testing stage, the test image is sent to the network to predict the class, location, and orientation information of the target. Through comparing the scores with the threshold and post-processing of nonmaximum suppression, we can obtain the final detection and location result. This paper has the following contributions. (1) Combining the rotated bounding boxes with the deep learning methods to simultaneously detect and locate the rotated ship targets in a complex background, (2) adding an angle parameter to the original four parameters to record a rotated bounding box to maintain the rectangular shape of the box, and (3) using a range control strategy to calculate the matching between the prior rotated boxes and ground-truth boxes to reduce the amount of calculation.

2. Method to Label the Rotated Targets

To label the rotated targets, we need to add a parameter, besides the original four parameters, to record the angle information of the rotated target. The original labeling method records the coordinates of the top-left and bottom-right corners to uniquely determine a horizontal bounding box. In the deep learning object detection methods, the parameters are transformed as (x_c, y_c, w, h), where (x_c, y_c) is the coordinate of the center point, w is the width of the box, and h is the height of the box. The four parameters can better explain the feature of the box and ensure that the detection algorithm converges effectively.

Based on these four parameters, we add another angle parameter *theta* to uniquely determine a rotated bounding box $(x_c, y_c, w, h, theta)$, where (x_c, y_c) is still the coordinate of the center point; h represents the long side of the box; w represents the short side of the box, which is perpendicular to h; and *theta* represents the angle between h in the upward direction and the horizontal right direction, and is in the range of $[0, 180)$ degrees. The labeling parameters are shown in Figure 2.

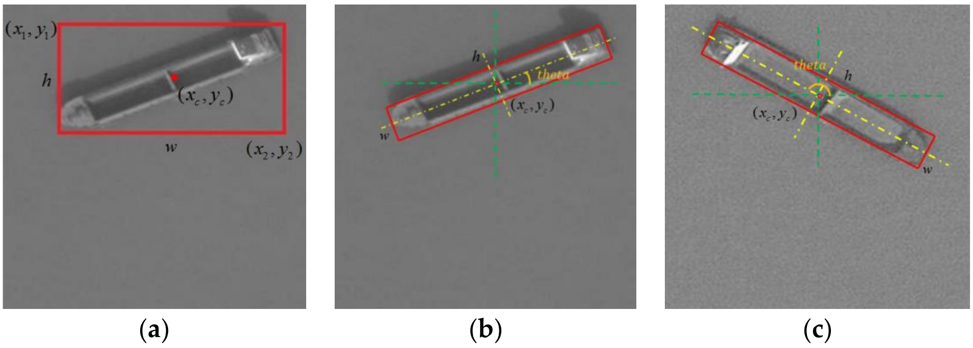


Figure 2. Comparison of parameters used in horizontal and rotated bounding boxes. (**a**) Parameters used in horizontal bounding boxes and (**b**,**c**) parameters used in rotated bounding boxes.

3. Multiscale Rotated Bounding Box-Based Deep Learning Detection Model

Our algorithm is based on deep learning methods and uses the multiscale rotated bounding boxes to detect and locate the ships precisely. This section gives a comprehensive explanation of the algorithm from the perspectives of network structure, design of the key parts, and implementation details.

3.1. Network Structure

The network structure is similar to that of the region proposal network of the Faster R-CNN method [12]. First, we use layers from cov1_1 to conv5_3 of VGG16 [16] to extract the features of the image. VGG16 model is pretrained on the ImageNet dataset [15], which has high performance in object classification. Many methods of target detection are fine-tuned from it, such as Faster R-CNN and SSD. So we choose this model and fine-tune it to get a better performance. Then, based on the feature maps, we design two flow paths with two convolutional layers to simultaneously predict the loss label and regress the location-angle offsets. Finally, we calculate the loss with the loss layer and backpropagation to train the model. The network structure is shown in Figure 3 and the diagram of the detection stage is shown in Figure 4.

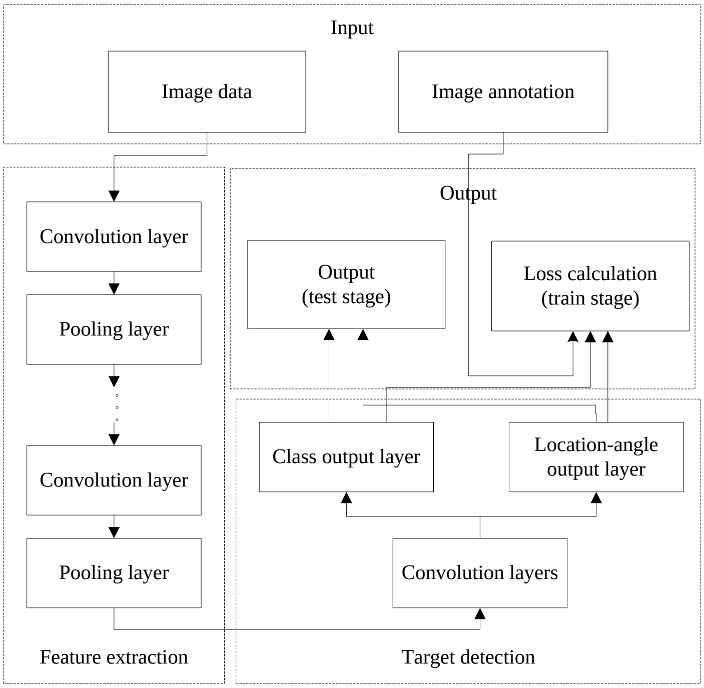


Figure 3. Diagram of network structure.

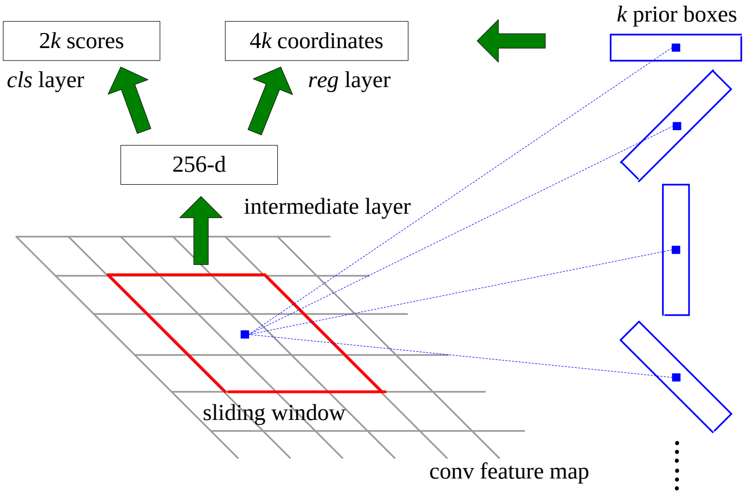


Figure 4. Diagram of target detection part.

3.2. Rotated Prior Box

After obtaining the convolutional feature maps, we set the rotated prior boxes on each position of the feature maps to locate the ship targets. The rotated prior boxes are shown in Figure 4, which is in blue color. Based on the rotated prior boxes, we can predict the class name and location information of the target in various orientations. When training the network, we first match the prior boxes with the ground-truth bounding boxes to obtain the positive sample information, which will be used to calculate the loss. The diagram of rotated prior boxes are shown in Figure 5, the one matched with the ground-truth bounding box is shown in red.

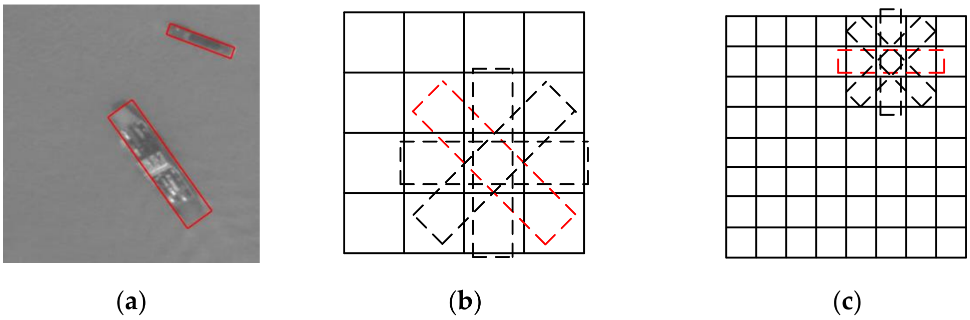


Figure 5. Diagrams of the ground-truth bounding box and the best matching prior box. (**a**) The image and the ground-truth box. (**b**) Prior boxes with dotted line and the best matching prior box with the bigger ship, which is in red. (**c**) Prior boxes with dotted line and the best matching prior box with the smaller ship, which is in red.

After splitting the large-scale image, we feed the network with images of size 600 × 600 pixels. The scale of the prior box is set as $\{32, 64, 128, 256\}$ and the aspect ratio means the ratio of h to w, which is set as 5:1, so the long side h of the box is $\{71, 142, 284, 568\}$; the step size of the box is 16. Using multi-scale prior boxes, we can detect the multi-scale targets better. The rotation angle is set as $\{0, 45, 90, 135\}$, which ignores the head or tail direction as it is difficult to identify when the ship scale is small. The rotated prior boxes of one scale are shown in Figure 6. These parameters are chosen based on the characteristics of the ship targets in the dataset and the network structure, which will be explained in details in Section 4.3.

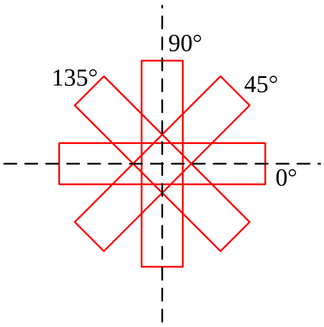


Figure 6. Diagram of the selected angles of the rotated bounding boxes.

3.3. Rotated Box Matching

Rotated box matching plays an important role in target detection, which is used in choosing positive samples in training, nonmaximum suppression in testing, and deciding whether it is the right detection in evaluation. Let us assume two rotated boxes, recorded as box_1 and box_2. We use a matrix with all zeros except the ones in the box region to record the box, so we can obtain the area of the box by computing the sum of the matrix elements. The two matrices are recorded as I_1 and I_2. The areas of box_1 and box_2 are recorded as $Area_1$ and $Area_2$. To obtain the overlap region, we calculate the product from I_1 and I_2. The result is recorded as matrix I_3, and the overlap area is the sum of the elements, recorded as $Area_\cap$. Thus, the matching degree of the two boxes is calculated as

$$matching\ degree = \frac{Area_\cap}{Area_\cup} = \frac{Area_\cap}{Area_1 + Area_2 - Area_\cap} \tag{1}$$

When matching the rotated boxes, the calculation between each prior box will lead to a large amount of calculation. Thus, we should first narrow down the selection scope. We set a selection range t_c and calculate the distance between the center points $d_c = \|(x_1^c, y_1^c) - (x_2^c, y_2^c)\|$, where (x_1^c, y_1^c) and (x_2^c, y_2^c) are the center point coordinates of the two boxes. If $d_c < t_c$, we select a prior box with the nearest angle to the ground-truth box angle to calculate the matching degree. By implementing these strategies the calculation amount is significantly reduced.

3.4. Loss Function

There are two flow paths to output the result, one predicts the class information of the target and the other predicts the location and angle information. The two layers are usually called the class label output layer and location angle regression layer. When calculating the loss, we combine the loss of class label output layer and the location angle regression layer to calculate the weighted sum. Using $x_{ij} = \{1, 0\}$ to record whether the ith prior box is matched with the jth ground-truth box, 1 means matched and 0 means mismatch. The loss is calculated as

$$L(x, c, l, g) = \frac{1}{N}(L_{cls}(x, c) + \alpha L_{loc}(x, l, g)), \tag{2}$$

where x is the matching status, c is class label output, l is the output of location and angle offsets, g is the ground-truth information, and N is the number of prior boxes matched with the ground-truth boxes. If $N = 0$, the loss is 0. L_{cls} is the loss of class label output layer, L_{loc} is the loss of the location and angle offset regression, and α is the weight, which is used to balance the two losses, here we set α to 1 experimentally.

The loss of class label output layer is calculated as

$$L_{cls}(x, c) = -\sum_{i \in Pos}^{N} x_{ij} \log(\hat{c}_i^1) - \sum_{i \in Neg} \log(\hat{c}_i^0), \tag{3}$$

$$\hat{c}_i^1 = \frac{\exp(c_i^1)}{\sum_p \exp(c_i^1)}, \tag{4}$$

where c_i is the output of class label layer, $\hat{c}_i$ is the class score, $\hat{c}_i^1$ is the probability of having a ship target in the box, and $\hat{c}_i^0$ is the probability of being the background. When calculating the weighted sum of the log function, we should see whether it is a positive or negative prior box after matching with the ground-truth boxes.

The loss of regression of location and angle offsets uses the SmoothL1 loss [11] for the calculation:

$$L_{loc}(x, l, g) = \sum_{i \in Pos}^{N} \sum_{m \in \{cx, cy, w, h\}} x_{ij}^k \text{smooth}_{L_1}(l_i^m - \hat{g}_j^m), \tag{5}$$

$$\text{smooth}_{L_1}(x) = \begin{cases} 0.5x^2 & \text{if } |x| < 1 \\ |x| - 0.5 & \text{otherwise,} \end{cases} \tag{6}$$

where l is the predicted offsets of the five parameters $(x_c, y_c, h, w, theta)$ and $\hat{g}_j$ is the offset between the ith prior box d_i and the jth ground-truth box g_j, as shown in Equations (7) and (8):

$$\hat{g}_j^{x_c} = (g_j^{x_c} - d_i^{x_c})/d_i^w, \hat{g}_j^{y_c} = (g_j^{y_c} - d_i^{y_c})/d_i^h \tag{7}$$

$$\hat{g}_j^w = \log\left(\frac{g_j^w}{d_i^w}\right), \hat{g}_j^h = \log\left(\frac{g_j^h}{d_i^h}\right), \hat{g}_j^{theta} = \tan(g_j^{theta} - d_i^{theta}) \tag{8}$$

When calculating the loss, the sum is calculated only if the prior box is matched with a ground-truth box.

3.5. Implementation Details

We use Caffe [26] to build the network structure and train our deep neural network. In the training stage, we process one image at one iteration. In the image, we select the prior boxes with a matching degree larger than 0.7 or those having the largest matching degree with a ground-truth box as the positive samples. The positive number N_p is not more than 64; the negative samples are randomly selected in the ones with matching degree below 0.3. The negative number N_f is $128 - N_p$, so the total number is 128. We use the stochastic gradient descent method to optimize the training. The initial learning rate is 0.001 and the number of iterations is 80,000. After 60,000 iterations, the learning rate is reduced 10-fold.

In the testing stage, we set the score threshold at 0.25 to decide whether there is a target in the box. Then, we perform the nonmaximum suppression to obtain the final result. We set the nonmaximum suppression threshold at 0.2 to remove the boxes whose matching degree is larger than 0.2 with another box of a larger score.

4. Experiments and Analysis

The algorithm was tested on images with ship targets in complex backgrounds to evaluate the detection and localization performance. We tested on a properly selected dataset and compared our algorithm with those using horizontal bounding boxes, such as Faster R-CNN and SSD, to show the advantage of detection using rotated boxes. We also carried out a comparison with existing algorithms using rotated bounding boxes, such as DRBox and OBB Faster R-CNN, to show the robustness of our algorithm.

4.1. Dataset

We collected 640 remote sensing images from Google Earth. The long side of the image was shorter than 1000 pixels. There were 4297 ship targets with complex backgrounds in the images. The complex clutter included wave clutter, ships in close proximity, and ships close to the shore.

4.2. Performance Evaluation Index

We used the average precision (AP) value to evaluate the performance of the algorithm. AP is the area under the precision-recall curve. Precision is the ratio of correct detection to the total number of targets detected. Recall is the ratio of correct detection to the total number of ground-truth targets. When we change the score threshold, we will get a different precision and recall. The precision–recall curve shows the precision variety following the recall variety, so it can reflect the overall performance of the algorithm. When considering whether it is a correct detection, we calculated the matching degree between the detected box and the ground-truth box, as in Equation (1). If the matching degree is higher than 0.5, we consider it as a correct detection. When a ground-truth bounding box has been matched with a detected box, then other detected boxes matched with it are considered as false alarms.

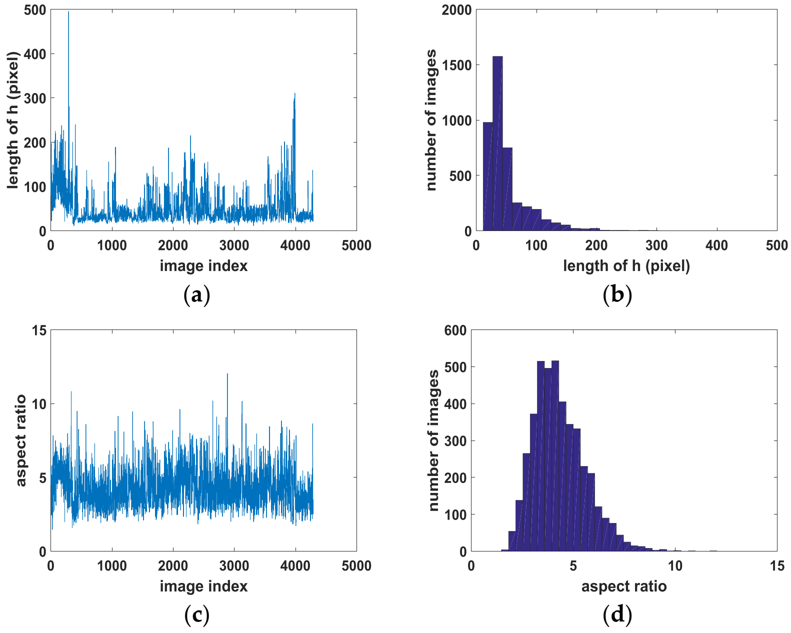
4.3. Performance of the Algorithm and Parameter Analysis

As the ships vary in scale and appearance, the algorithm should select suitable multiscale parameters and augment the data to learn more about the variety.

4.3.1. Dataset Characteristics Analysis and Model Parameter Selection

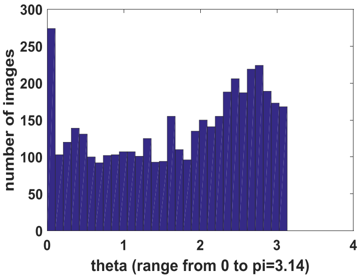
First, we analyzed the characteristics of the ship targets in the dataset to help select the model parameter. We recorded the long side of the rotated box as h, the short side as w, the aspect ratio as h/w, and the angle as *theta*.

Figure 7 shows the main characteristics distribution of the targets in the dataset. The long side h is distributed mainly from 20 to 150 pixels. The longest is about 500 pixels, while others are distributed between 200 and 300. The mean length is 50.6, and the aspect ratio h/w is distributed mostly between 3 and 7. The largest value is 12, while others are distributed between 8 and 10. The mean value is 4.3. The angle of the ships is mainly distributed around 0° and uniformly distributed at other angles.



(**a**) (**b**) (**c**) (**d**)

Figure 7. *Cont.*



(**e**)

Figure 7. Main characteristics distribution of the targets in the dataset. (**a**) The long side h of all 4279 targets, (**b**) distribution of long side h, (**c**) aspect ratio h/w of all 4279 targets, (**d**) distribution of aspect ratio h/w, and (**e**) distribution of angle *theata*.

After analyzing the characteristics of the targets in the dataset, we can use them to help select the parameters. We set the aspect ratio to 5 so that the rotated prior boxes can match the ground-truth boxes well. When choosing the scales of the prior box, we consider the long side h to be mainly distributed between 20 and 150 and the larger ones to be mainly between 200 and 300. Therefore, we select the scales at $\{32, 64, 128, 256\}$ to better match the characteristics of the real targets when the shorter side of the input image is bigger than 600 pixels and the longer side is smaller than 1000 pixels and is not preprocessed. When performing data augmentation, we rotate the images and label information to increase the variability of the targets. Since the size of the input image to the deep network is limited to 600–1000 pixels, the images with a larger or smaller scale will be preprocessed and the target scale is changed. Therefore, we should add zeros to the small images to expand them and split them with a large scale to retain the original size of the target.

4.3.2. Model Parameter Analysis

We trained the deep network with different parameters and compared the performances to analyze the influence of the parameters.

We trained on a dataset of 148 images with 80,000 iterations and an aspect ratio of 3. The performance AP was 0.091 when testing on all 640 images. When changing the aspect ratio to 5 and retaining all other parameters, the performance AP remained at 0.091. When the training dataset contained 200 images and the aspect ratio was 5, the performance AP increased to 0.212. These results show that training with less data will not help learn the ship's feature well, resulting in poor performance. When training with 200 images, the algorithm is more robust because a closer distribution of the training and testing datasets.

Then, we considered the influence of input image scale and preprocessed the images to keep the target within the original scale. After the preprocessing, we trained with a training dataset of 400 images with 80,000 iterations and an aspect ratio of 5. This caused the performance AP to increase to 0.362. The result shows that having more training samples and maintaining the original target scale will help the algorithm to learn more robustly and obtain better performance. Performances with different parameters are shown in Table 1.

Table 1. Performance comparison of our method with different parameters.

Number of Training Samples	Aspect Ratio	Keep Original Scale	AP
148	3:1	no	0.091
148	5:1	no	0.091
200	5:1	no	0.212
400	5:1	yes	0.362

4.4. Performance Comparison with Algorithms Using Horizontal Bounding Boxes

The commonly applied methods in the target detection area use the horizontal bounding boxes to search and detect targets, such as SSD and Faster R-CNN. Here, we compare the performances of the two algorithms with our method to show the advantage of using rotated bounding boxes.

We trained on a dataset with 400 images using an SSD algorithm and tested all 640 images. In this case, the performance AP was 0.206. While using the Faster R-CNN algorithm with the same parameters, the performance AP is 0.293. In our algorithm, the performance AP was 0.362. These results show that the method using rotated bounding boxes can learn the target characteristics better and detect the rotated ships in close proximity better. Figure 8 presents a comparison of the detection results between algorithms using horizontal bounding boxes and our method. These results show that both the method using horizontal boxes and that using rotated boxes can detect the ships off-shore well. When the ships are in close proximity or close to the shore, our method using rotated bounding boxes can detect the target better and locate it more precisely. A comparison of the performance AP values is shown in Table 2.

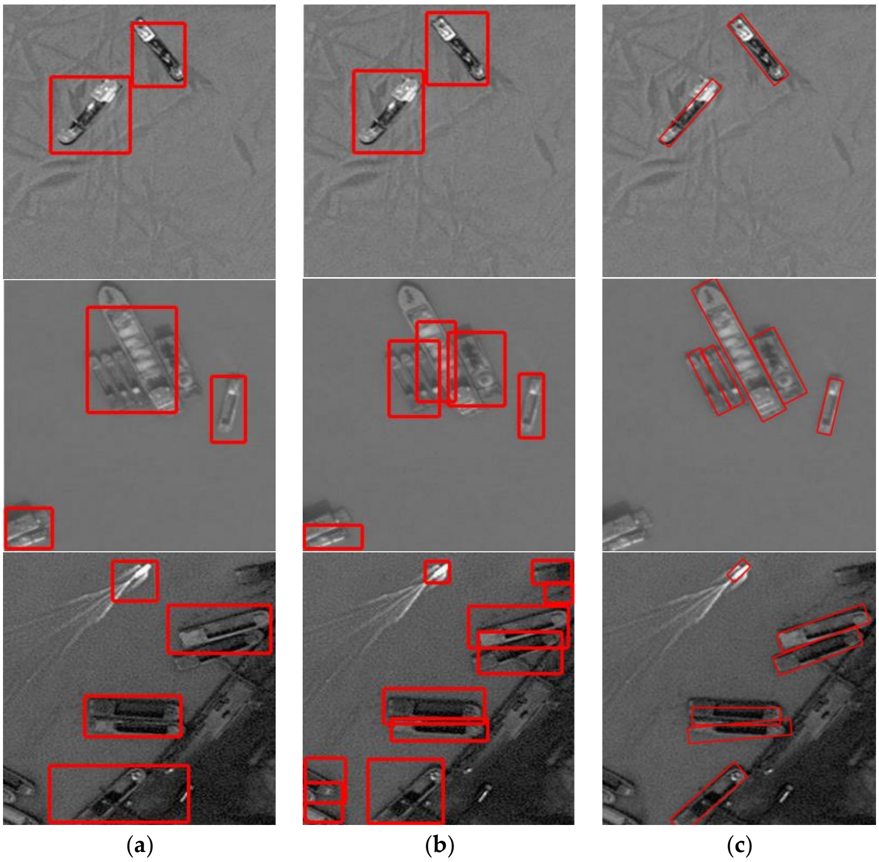


Figure 8. Comparison with the algorithm using horizontal bounding boxes. (**a**) Detection results of SSD, (**b**) detection results of Faster R-CNN, and (**c**) detection results of our method.

Table 2. Performance comparison of different algorithms.

Method	AP
SSD	0.206
Faster R-CNN	0.293
DRBox	0.305
Faster R-CNN (OBB)	0.278
Our method	0.362

4.5. Performance Comparison with Other Algorithms Using Rotated Bounding Boxes

We compared the performance of our algorithm with that of algorithms using rotated bounding boxes to show the advantage of our method. One of the methods is DRBox, which is based on SSD and combined with rotated bounding boxes to detect the oriented targets. We used the trained model provided by the author to test on all 640 images and found the performance AP to be 0.241. The detection results indicated that the model detects large-scale ships better while many small ships were missed. Hence, we added a larger resolution scale to detect the smaller ships and the performance AP was found to increase to 0.305. We continued by adding another larger resolution scale, and the performance AP dropped to 0.277. These results indicate that both the correct detection number and false alarm number are increasing. Overall, the performance of DRBox algorithm is lower than that of our method since the accuracy of detecting small targets is low and more false alarms are detected. Another method is the oriented bounding box (OBB) Faster R-CNN algorithm, which uses a rotated bounding box with eight parameters to detect the oriented targets. We tested the provided trained model on all 640 images and found the performance AP to be 0.278, which is also lower than that of our method. The main reason leading to the performance drop is the missing of several small ships.

Figure 9 shows that when detecting ships with a background of water, all three algorithms can detect well and locate the ships precisely; when the ships are in close proximity or close to the shore, our algorithm detects better, while other methods miss some ships; when detecting small ships, our algorithm performs better, while other methods miss some ships. The last column shows some false alarms on the land background in both DRBox method and our method, while the OBB Faster R-CNN method misses some ships. The performance AP values of the different methods mentioned above are shown in Table 2.

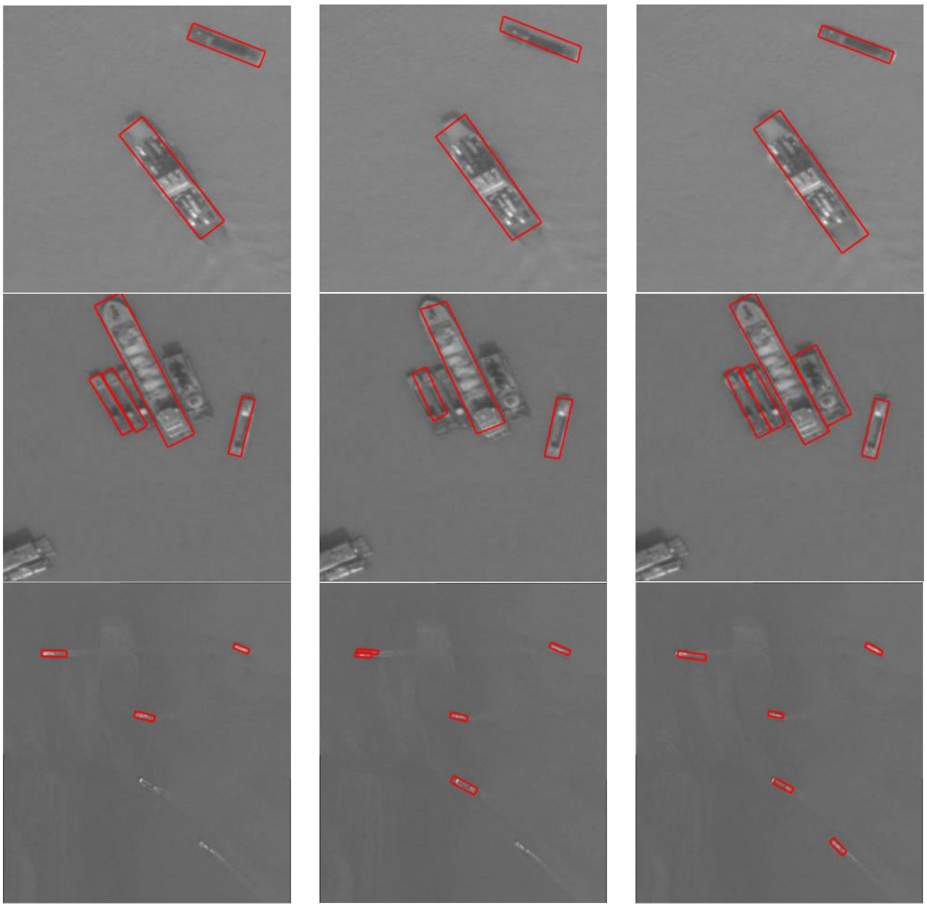


Figure 9. *Cont.*

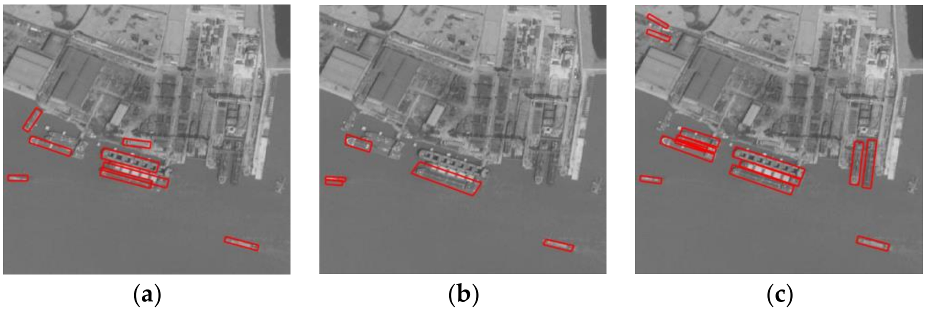


Figure 9. Comparison with other algorithms using rotated bounding boxes. Detection results of (**a**) DRBox, (**b**) OBB Faster R-CNN, and (**c**) our method.

4.6. Discussion

The above experiments and analysis show that our method using multiscale rotated bounding boxes can detect the ships and locate precisely when the ships are in complex backgrounds. Compared with the method using horizontal bounding boxes, our method detects better when ships are in close proximity or close to the shore; compared to other methods using rotated bounding boxes, our method detects ships with higher accuracy and lower missing detection rate, and especially detects small ships better. However, there are still some problems, such as false alarms; as a result, the performance AP still has room for improvement.

5. Conclusions

We proposed a multiscale rotated bounding box-based deep learning method to detect oriented ships in complex backgrounds in remote sensing images. The algorithm used five parameters to determine a rotated bounding box, which limits the box to a rectangular shape. The algorithm predicted different positions of the feature maps extracted by a pretrained deep network to output the class label and location-angle offset regression based on the multiscale rotated prior boxes. In the training stage, we searched the best-matched prior boxes with the ground-truth boxes to obtain the positive sample information, which was used to learn the ship features. When matching the rotated boxes, we first narrowed down the search scope to reduce the amount of calculation. In the testing stage, we input the testing image to the network and perform forward calculation to obtain the class labels and the location-angle offsets of each position. Then, we set a threshold of the class score to determine whether it is a ship and perform post-processing, such as nonmaximum suppression, to obtain the final detection result. The experimental result showed that our method can detect ships robustly under complex conditions, such as ships in wave clutter, in close proximity, close to the shore, and with different scales, and obtain an accurate location of the ships. Our method performs better than the other methods. However, the performance AP still has room for improvement, so we will study the target missing and false alarm problems in the future to improve the algorithm.

Author Contributions: S.L. and Z.Z. conceived and designed the experiments; S.L. performed the experiments; S.L. and C.L. analyzed the data; B.L. contributed material/analysis tools; S.L. and Z.Z. wrote the paper.

Funding: This project is supported by the National Natural Science Foundation of China (Grant No. 61101185) and Ministerial level Foundation (Grant No. 61425030301).

Acknowledgments: The authors express their gratitude to the reviewers and editors for their kind help.

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

References

1. Zhu, C.; Zhou, H.; Wang, R.; Guo, J. A novel hierarchical method of ship detection from spaceborne optical image based on shape and texture features. *IEEE Trans. Geosci. Remote Sens.* **2010**, *48*, 3446–3456. [CrossRef]

2. Shi, Z.; Yu, X.; Jiang, Z.; Li, B. Ship detection in high-resolution optical imagery based on anomaly detector and local shape feature. *IEEE Trans. Geosci. Remote Sens.* **2014**, *52*, 4511–4523.
3. Xu, F.; Liu, J.; Dong, C.; Wang, X. Ship detection in optical remote sensing images based on wavelet transform and multi-level false alarm identification. *Remote Sens.* **2017**, *9*, 985. [CrossRef]
4. Liu, G.; Zhang, Y.; Zheng, X.; Sun, X.; Fu, K.; Wang, H. A new method on inshore ship detection in high-resolution satellite images using shape and context information. *IEEE Geosci. Remote Sens. Lett.* **2014**, *11*, 617–621. [CrossRef]
5. Xu, F.; Liu, J.; Sun, M.; Zeng, D.; Wang, X. A hierarchical maritime target detection method for optical remote sensing imagery. *Remote Sens.* **2017**, *9*, 280. [CrossRef]
6. Nie, T.; He, B.; Bi, G.; Zhang, Y.; Wang, W. A method of ship detection under complex background. *ISPRS Int. J. Geo-Inf.* **2017**, *6*, 159. [CrossRef]
7. Sui, H.; Song, Z. A novel ship detection method for large-scale optical satellite images based on visual LBP feature and visual attention model. In Proceedings of the 23rd International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences Congress (ISPRS 2016), Prague, Czech Republic, 12–19 July 2016; pp. 917–921.
8. 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]
9. Zou, Z.; Shi, Z. Ship detection in spaceborne optical image with SVD networks. *IEEE Trans. Geosci. Remote Sens.* **2016**, *54*, 5832–5845. [CrossRef]
10. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the 27th IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2014), Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
11. Girshick, R. Fast R-CNN. In Proceedings of the 15th IEEE International Conference on Computer Vision, (ICCV 2015), Santiago, Chile, 11–18 December 2015; pp. 1440–1448.
12. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. In Proceedings of the 29th Annual Conference on Neural Information Processing Systems (NIPS 2015), Montreal, QC, Canada, 7–12 December 2015; pp. 91–99.
13. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. SSD: Single shot multibox detector. In Proceedings of the 14th European Conference on Computer Vision (ECCV 2016), Amsterdam, The Netherlands, 11–14 October 2016; pp. 21–37.
14. 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 2016), Las Vegas, NV, USA, 26 June–1 July 2016; pp. 779–788.
15. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. In Proceedings of the 26th Annual Conference on Neural Information Processing Systems (NIPS 2012), Lake Tahoe, NV, USA, 3–6 December 2012; pp. 1097–1105.
16. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556
17. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the 28th IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.
18. He, K.; Gkioxari, G.; Dollar, P.; Girshick, R. Mask R-CNN. In Proceedings of the 16th IEEE International Conference on Computer Vision (ICCV 2017), Venice, Italy, 22–29 October 2017; pp. 2980–2988.
19. Zhang, R.; Yao, J.; Zhang, K.; Feng, C.; Zhang, J. S-CNN-based ship detection from high-resolution remote sensing images. In Proceedings of the 23rd International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences Congress (ISPRS 2016), Prague, Czech Republic, 12–19 July 2016; pp. 423–430.
20. Tang, J.; Deng, C.; Huang, G.-B.; Zhao, B. Compressed-domain ship detection on spaceborne optical image using deep neural network and extreme learning machine. *IEEE Trans. Geosci. Remote Sens.* **2014**, *53*, 1174–1185. [CrossRef]

21. Liu, Z.; Hu, J.; Weng, L.; Yang, Y. Rotated region based CNN for ship detection. In Proceedings of the 24th IEEE International Conference on Image Processing (ICIP 2017), Beijing, China, 17–20 September 2017; pp. 900–904.
22. Liu, Z.; Liu, Y.; Weng, L.; Yang, Y. A high resolution optical satellite image dataset for ship recognition and some new baselines. In Proceedings of the International Conference on Pattern Recognition Applications and Methods, Porto, Portugal, 24–26 February 2017; pp. 324–331.
23. Shao, X.; Li, H.; Lin, H.; Kang, X.; Lu, T. Ship detection in optical satellite image based on RX method and PCAnet. *Sens. Imaging* **2017**, *18*, 21. [CrossRef]
24. Liu, L.; Pan, Z.; Lei, B. Learning a rotation invariant detector with rotatable bounding box. *arXiv* **2017**, arXiv:1711.10398.
25. 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. *arXiv* **2017**, arXiv:1711.10398.
26. Jia, Y.; Shelhamer, E.; Donahue, J.; Karayev, S.; Long, J.; Girshick, R.; Guadarrama, S.; Darrell, T. Caffe: Convolutional architecture for fast feature embedding. In Proceedings of the 2014 ACM Conference on Multimedia, Orlando, FL, USA, 3–7 November 2014.