DD-Net: A Dual Detector Network for Multilevel Object Detection in Remote-Sensing Images

Dongdong Zhang, Chunping Wang, and Qiang Fu

Department of Electronic and Optical Engineering, People Liberation Army Engineering University, Shijiazhuang Hebei 050003, China

Correspondence should be addressed to Qiang Fu; 1418748495@qq.com

Received 22 April 2022; Accepted 6 July 2022; Published 20 July 2022

Copyright © 2022 Dongdong Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

With the recent development of deep convolutional neural network (CNN), remote sensing for ship detection methods has achieved enormous progress. However, current methods focus on the whole ships and fail on the component’s detection of a ship. To detect ships from remote-sensing images in a more refined way, we employ the inherent relationship between ships and their critical parts to establish a multilevel structure and propose a novel framework to improve the performance in identifying the multilevel objects. Our framework, named the dual detector network (DD-Net), consists of two carefully designed detectors, one for ships (the ship detector) and the other for their critical parts (the critical part detector), for detecting the critical parts in a coarse-to-fine manner. The ship detector offers detection results of the ship, based on which the critical part detector detects small critical parts inside each ship region. The framework is trained in an end-to-end way by optimizing the multitask loss. Due to the lack of publicly available datasets for critical part detection, we build a new dataset named RS-Ship with 1015 remote-sensing images and 2856 annotations. Experiments on the HRSC2016 dataset and the RS-Ship dataset show that our method performs well in the detection of ships and critical parts.

1. Introduction

As a fundamental task in computer vision, remote-sensing ship detection has been widely used in both military and civilian fields [1, 2]. With the development of CNN (convolutional neural network), the effectiveness of ship detection has been improved dramatically. However, it is difficult to meet the needs of some special tasks such as detecting components of a ship. So a more refined detection of ship and its components is needed.

Ships can be considered multilevel objects, where cockpit, powerhouse, radar antenna, etc. are subobjects of the ship. There also many other multilevel objects in our life such person and face, car and wheel, or streetlight and light, which are shown in Figure 1. After a comprehensive analysis of the prevalence and importance of each part of ship, we regard the cockpit as a critical part and conduct related research on detecting ship and its cockpit.

Traditional ship target detection algorithms [3–5] rely on manual setting of extracted features, which leads to time-consuming computation, poor robustness, and a high probability of missed or false detections. In recent years, with the rapid development of the deep learning technology and its wide application in computer vision, deep learning-based ship target detection algorithms have become mainstream in the ship detection field. In [6], the ship detection process was divided into two steps. First, sea-land segmentation was performed to reduce the influence of artificial objects on the coastline on ship detection; then, the ship was detected from the sea part pixels. In the ship detection network proposed in [7], angle information was added to the border regression to make the anchor frame fit more
closely to the arbitrarily oriented ship targets. This in turn enhanced the feature extraction capability of the network and thereby improved its performance in the detection of small-scale targets. In [8], an improved Mask R-CNN [9] framework was proposed to segment and locate targets using two key points of a ship: the bow and the stern; additionally, the key point bow was used in combination with the minimum bounding box of Mask to determine the direction of the target. Although the above algorithms have exhibited good detection performance in a variety of applications, they all detect the ship as a whole target but not a multilevel target and cannot detect ships in a more refined way. In optical remote-sensing ship images, the critical parts of the ship occupy only a few pixels, which makes it challenging to extract features of the critical parts from these images. Therefore, it is difficult for existing algorithms to accurately detect the critical parts of the ship from such images directly. If the critical parts are considered the detection target without factoring in their relationship with the ship, the interference of the artificial objects on the coastline will increase, which will in turn lead to a higher false detection rate.

The top-down pose estimation algorithm can locate person from images and detect the pose of the person, which is similar to the detection of a ship and its critical parts. To solve the above mentioned problems, inspired by the top-down pose estimation method [10–12], we propose a new network structure, named dual detector network (DD-Net), to find ships from images first and then find tiny critical parts from these ships. Our network contains two detectors which are the ship detector and the critical part detector. The ship detector adopts a single-stage network to predict a set of ship bounding boxes to find out ships from images. Then the feature maps of the detected ships are wrapped and sent to the ship region-based critical part detector to find the boxes of the critical parts. In the critical part detector, most useless information is removed, and only the pixels inside the boxed are remained. Thus, there is less interference inside ship proposals, which facilitates the detection of small parts. The whole network can be trained in an end-to-end way. To verify the proposed methods, we create a new remote-sensing ship dataset: RS-Ship. It contains 1015 well-labeled images for ships and their critical parts. To the best of our knowledge, this is the first dataset containing both ships and their critical parts, which paves the way for future researches on the detection of critical parts of ships. At last, we performed experiments on the HRSC2016 dataset [13] and RS-Ship dataset with the method. The result shows that our method achieves state-of-the-art performance both dataset for ship and critical part detection in complex scenes.

2. Summary Review of Previous Studies

2.1. Object Detection Methods. CNN-based object detection algorithms fall into two categories: two-stage networks [9, 14–17] and single-stage networks [18–21]. Two-stage networks rely on region proposals, which are generated by Region Proposal Network and then extract features from each region proposal for classification and bounding box regression. Single-stage networks directly estimate object candidates without relying on any region proposal, so this design brings fast computational efficiency. However, single-stage networks cannot achieve comparable detection accuracy to their two-stage counterparts. To allow the proposed network to maintain a balance between the accuracy and speed, we improve the detection accuracy by restricting the target region and reducing the influence of background based on a single-stage detection framework.

2.2. Ship Detection Methods. With the rapid development of deep learning in recent years, deep learning-based algorithms have emerged as a much more accurate and faster alternative to traditional algorithms in the detection of ship targets from optical remote-sensing images. In [22], the Inception structure [23] was used to improve the YOLOv3 [21] network. This improvement enhanced the feature extraction capability of the backbone network without losing any feature of small-scale ships during propagation over the network, thereby making the network better capable of detecting small-scale targets. In [24], Mask R-CNN was used to separate ships and the backgrounds, and soft-NMS was used in the screening process to further improve the robustness of ship detection. In [25], angle information was used together with some orientation parameters to make anchor better fit ship targets and thereby enable significant improvement in detection accuracy. Subsequently, various improved algorithms were proposed to address the problems including insufficient positive samples, feature misalignment, and inconsistency between classification and regression due to the introduction of rotating frames [26–29]. With the continuous improvements, today’s deep learning-based ship detection algorithms can meet the required levels of accuracy and efficiency for civilian applications. However, they all detect the ship as a whole target but not a multilevel target. To address this problem, the proposed network is specially designed to detect ship and its critical parts.

2.3. Top-Down Pose Estimation Algorithms. The top-down pose estimation algorithm consists of two parts: a region detection network and a pose estimation network. Persons
3. Proposed Method

The architecture of the proposed DD-Net network is illustrated in Figure 2. It consists of three parts: (1) the backbone CSPDarknet53 network [34], which is used to extract target features; (2) the ship detector, which is designed to detect the ship as a whole; and (3) the critical part detector, which is designed to detect the critical parts inside the selected ship bounding boxes predicted by the ship detector.

3.1. Backbone Network. The backbone network is CSPDarknet53, which is pretrained on ImageNet [35]. The input images for the backbone network are in a size of 640 × 640, and the output is four convolutional feature maps (C2, C3, C4, and C5) with different downsampling steps (4, 8, 16, and 32), respectively. Taking C2 as an example, the representation of feature map C2 will be available only when the scale of the target is larger than 4 × 4. The same principle is applied to the other output layers for different size of 8 × 8, 16 × 16, and 32 × 32. We use the network input scale of 640 × 640 as the benchmark and count the pixels occupied by ships and critical parts in the RS-Ship dataset, as summarized in Figure 3. The results clearly show that the ship targets are in a scale larger than 8 × 8 and the critical parts are larger than 4 × 4. To avoid missing detections due to the loss of target features from an excessively large downsampling step, we use C3, C4, and C5 as the input for the ship detector and C2 as the input for the critical part detector.

3.2. Ship Detector. To effectively utilize the semantic information of high-level feature maps and the detail information of low-level feature maps, we use C3, C4, and C5 to construct a feature pyramid network (FPN) [36] from top to down. Then, the FPN is reconstructed in a bottom-top format to reduce the span between high-level feature maps and low-level ones, which can enrich detail information of high-level feature maps and avoid loss of semantic information from channel reduction. In addition, a channel attention module (CAM) [37] is introduced to reconstruct the FPN to connect adjacent feature layers while generating salient features. The CAM uses pooling operations (MaxPool and AvgPool) to generate channel context descriptors and then outputs channel attention feature maps through a shared network, which consists of a multilayer perceptron (MLP). The computation process of the CAM can be expressed as follows:

\[
M_c(F) = s(MLP(AvgPool(F)) + MLP(MaxPool(F)))
\]

where \(F_{avg}\) and \(F_{max}\) denote the global average pooling feature and global max pooling feature, respectively. \(W_0\) and \(W_1\) represent two convolutional layers followed by a Relu activation function, and \(s\) is a Sigmoid function.
After the above processes, feature maps P3, P4, and P5 are obtained and propagated to the prediction layer for target prediction. Figure 4 shows the process of ship detection on feature map P5 with a resolution of $19 \times 19$. The detection process on P3 and P4 is same as that on P5. The input image is divided into $19 \times 19$ cells. Each cell corresponds to a pixel of P5 which is the center to generate three anchor boxes of different scales. By computing features within the anchor boxes, each anchor box generates six different prediction parameters $t_x, t_y, t_w, t_h, P_{\text{obj}}, P_k$. Specifically, $(t_x, t_y, t_w, t_h)$ represents the border offset, $P_{\text{obj}}$ is the target confidence, and $P_k$ is the classification probability. Each anchor box is regression predicted for one target bounding box, and the bounding box coordinates are calculated by the following equation.

$$
\begin{align*}
  b_x &= \sigma(t_x) + c_x \\
  b_y &= \sigma(t_y) + c_y \\
  b_w &= p_w e^t_w \\
  b_h &= p_h e^t_h,
\end{align*}
$$

where $(c_x, c_y)$ is the coordinates of the top left corner of the cell. $(b_x, b_y)$ is the coordinates of the center of the bounding box. $b_w, b_h$ indicates the width and height of the bounding box, respectively.

3.3. Critical Part Detector. From the statistical results of the size of critical parts in the RS-Ship dataset (as shown in Figure 3), it is clear that critical parts are small-scale targets, and their features can be lost easily during the feature extraction process. C2 is a low-level feature map, which contains rich detail information, but lacks semantic information. With few convolutional layers passed, the feature map C2 cannot fully represent the target features. To address the above problems, we come up with a special design for the critical part detector. First, masking is employed to restrict the target region. The predicted ship bounding boxes with high classification scores from the ship detector will be selected. Nonmaximum suppression (NMS) is performed on the selected proposals to find high-quality prediction boxes. By locating ships using horizontal boxes, the region of proposals can be expanded, and the loss of critical parts from inaccurate localization of ships can be avoided. After
box regression, the coordinates of the prediction boxes are mapped in F2, and a limited binarization operation is performed on F2 (the pixel values of regions within the prediction boxes are set to 1 and the rest to 0) to generate a masked map. The masking process is shown in Figure 5. Then F2 will be filtered by the masked map and only the ship regions covered by the ship-boxes are preserved. Second, a feature extraction network is constructed for extracting deep features to enhance the representation of feature maps. The network is composed of four residual modules and four deconvolution modules. To prevent loss of spatial information, the four residual modules and four deconvolution modules are connected via skip connections. The settings of parameters such as number of channels, step size, and convolution kernel size for each layer and their specific signs are shown in Figure 2. Finally, F2 is fed to the prediction layer for ship detection. Feature map C2 is transferred to P2 after mask filtering and feature extraction, and P2 contains rich details and semantic information, which can be used for the detection of critical parts.

4. Experiments and Analysis

In this paper, the experimental environment is Ubuntu 18.04, E5-2630v4 CPU, 64 GB RMB, and the network is built based on PyTorch1.6 deep learning framework and accelerated with an Nvidia Geforce-GTX 1080Ti (11 G memory) graphics card. To verify the effectiveness of our proposed method, several sets of experiments are conducted on the HRSC2016 dataset [13] and the self-built RS-Ship dataset. The detection speed of the network is evaluated by frames per seconds (FPS) and its detection performance by average precision (AP). The FPS and AP values can be calculated by the following equations:

\[
FPS = \frac{N_{Figure}}{Time},
\]

\[
AP = \int_{0}^{1} P(R) dR,
\]

where \(N_{Figure}\) is the number of images tested, \(Time\) is the time taken by the test, \(P\) is the accuracy rate, and \(R\) is the recall rate.

4.1. Datasets. The HRSC2016 dataset is currently the only publicly available dataset that contains only naval targets. Its data are collected from six well-known ports. The image resolution ranges from 0.4 m to 2 m, and the image size ranges from \(300 \times 300\) to \(1500 \times 900\). The dataset contains 1061 remote-sensing images with 2976 ship targets that vary significantly in scale. Some example images from the HRSC2016 dataset are shown in Figure 6. The main scenes covered by the dataset are sea and near-shore areas, with complex backgrounds and diverse ship types. This dataset has been frequently used by researchers to test the performance of algorithms for ship target detection.

We created a new dataset named RS-Ship to verify our method for more samples. The RS-Ship dataset is mainly collected from some famous military harbors on Google Maps. The dataset has been expanded with images of ships from the Internet by crawlers, and all the images have been formatted a uniform size of \(800 \times 600\). The dataset contains 1015 ship images and 2856 ship targets, and each ship target has a certain critical part. Ships and critical parts are labeled in a PASCAL VOC format. The ships in the dataset are in widely varying scales. The covered scenes are mainly near-shore areas with complex backgrounds, and the artificial objects on the coastline will have a certain interference on the detection of ships and critical parts. Some example images from the RS-Ship dataset are shown in Figure 6.

In the experiments, the HRSC2016 dataset is applied to evaluate the performance of the proposed method in the detection of ship targets, and the RS-Ship dataset is applied to evaluate its performance in the detection of both ships and critical parts. Both datasets are divided into training
and testing sets with a 4:1 ratio, and the details of each set are shown in Table 1.

4.2. Experiment Implementation. Transfer learning [38] includes various migration methods such as instance-based transfer and parameter-transfer transfer learning methods. In this paper, the idea of parameter-transfer is introduced to the training process: the CSPDarknet53 model trained with the ImageNet dataset is used as initial parameters of the network. The whole training process can be divided into two steps: first, the backbone network CSPDarknet53 is frozen and the other network parameters (other layer parameters except the backbone network CSPDarknet53) are trained with 50 training epochs. Then all convolutional layers are opened with 100 training epochs, and the whole training process is optimized with the Adam optimizer.

To make the most of the prior knowledge of ship target shapes, the $k$-means clustering algorithm is implemented to generate nine anchor boxes on the HRSC2016 and RS-Ship training sets, respectively. The clustering results are shown in Figure 7, and the sizes of the anchor boxes are shown in Table 2. From Figure 7, it is clear that the normalized widths and heights of most ship targets in the RS-Ship and HRSC2016 training sets are concentrated within 0.4, indicating that both datasets contain a large number of small-scale targets. Compared with those in HRSC2016, the ship targets in RS-Ship exhibit wider distributions of length and width and have larger aspect ratios, which indicates that the RS-Ship dataset built here in this paper can be used to verify the performance of algorithms in ship detection.

4.3. Ablation Studies. To evaluate the performance of the two detectors (ship detector and critical part detector) and the overall rationality of the network, we set up six sets of experiments on the HRSC2016 dataset and the RS-Ship dataset and evaluate the experimental results mainly using AP values.

4.3.1. Experiment 1. The model used for the first set of experiments is Model1, which consists of CSPDarknet53 and the ship detector, without attention module present in the Ship detector.

4.3.2. Experiment 2. The model used for the second set of experiments is Model2, which consists of CSPDarknet53 and the ship detector.

4.3.3. Experiment 3. The model used for the third set of experiments is Model3, which consists of CSPDarknet53 and the critical part detector, with no feature extraction network present in the critical part detector.

4.3.4. Experiment 4. The model used for the fourth set of experiments is Model4, which consists of CSPDarknet53 and the critical part detector.

4.3.5. Experiment 5. The model used for the fifth set of experiments is Model5, a simplified version of our proposed model (DD-Net) in which the association between the ship detector and the critical part detector is blocked.

4.3.6. Experiment 6. The model used for the sixth set of experiments is our proposed model (DD-Net).

The results of each set of experiments are shown in Table 3.
Compared with Model1, Model2 has an additional attention module. This helps to enhance the salient features of the ship targets and reduce the number of false detections but increases the number of missed detections. All in all, the second set of experiments achieved APs of 82.98% and 91.23% on the HRSC2016 dataset and the RS-Ship dataset, respectively, 3.57% and 2.54% higher than those obtained from the first set of experiments.

The C2 layer feature maps of CSPDarknet53 contain rich detail information but little semantic information, which makes it difficult for the detector to distinguish between targets and interfering signals. Unlike Model3, Model4 can deeply extract the features in the C2 layer to construct P2 feature maps which contain rich detail information and semantic information, resulting in a significantly reduced number of missed detections and a 25.69% higher AP value.

Model5 integrates the ship detector and critical part detector in one framework, which allows the network to concentrate on the ship region during feature extraction, which reduces the influence of background on detection and facilitates the detection of targets. Compared with Model2, Model5 delivers 0.17% and 0.5% higher AP values.

\[ \text{M-IoU} = 0.6558 \]
\[ \text{c0: 380} \quad \text{c5: 185} \]
\[ \text{c1: 64} \quad \text{c5: 375} \]
\[ \text{c2: 264} \quad \text{c7: 170} \]
\[ \text{c3: 181} \quad \text{c8: 273} \]
\[ \text{c4: 252} \]

\[ \text{M-IoU} = 0.6540 \]
\[ \text{c0: 106} \quad \text{c5: 396} \]
\[ \text{c1: 128} \quad \text{c5: 134} \]
\[ \text{c2: 286} \quad \text{c7: 307} \]
\[ \text{c3: 324} \quad \text{c8: 253} \]

**Figure 7:** The clustering results, which generated by k-means and the clustering points is set to 9.

**Table 2:** Width and height settings of anchor boxes in HRSC2016 and RS-Ship datasets.

| Dataset   | 1   | 2   | 3   | 4   | 5   | 6   | 7   | 8   | 9   |
|-----------|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| HRSC2016  | 22,35 | 47,97 | 50,244 | 75,42 | 124,163 | 126,305 | 176,75 | 251,164 | 257,341 |
| RS-Ship   | 33,127 | 48,41 | 57,246 | 73,120 | 110,418 | 139,196 | 150,63 | 287,356 | 299,129 |

**Table 3:** Experimental results of different models.

| Set       | HRSC2016 |          |          |          |          |          | RS-Ship |          |          |
|-----------|----------|----------|----------|----------|----------|----------|---------|----------|----------|
|           | TP       | FP       | P (%)    | R (%)    | AP (%)   |          | TP       | FP       | P (%)    | R (%)    | AP (%)   |
| Model1    | 698      | 98       | 87.69    | 83.59    | 79.41    | 505      | 36       | 93.35    | 90.01    | 88.69    |          |
| Model2    | 707      | 86       | 89.16    | 84.67    | 82.98    | 515      | 23       | 95.72    | 91.80    | 91.23    |          |
| Model3    | —        | —        | —        | —        | —        | —        | —        | —        | —        | —        | —        |
| Model4    | —        | —        | —        | —        | —        | —        | —        | —        | —        | —        | —        |
| Model5    | 711      | 84       | 89.43    | 85.15    | 83.13    | 517      | 21       | 96.10    | 92.16    | 91.73    | 430      | 68       | 86.35    | 76.65    | 72.14    |
| Model6    | 714      | 82       | 89.70    | 85.51    | 83.13    | 518      | 20       | 96.28    | 92.34    | 91.70    | 454      | 47       | 90.62    | 80.93    | 79.65    |

Compared with Model1, Model2 has an additional attention module. This helps to enhance the salient features of the ship targets and reduce the number of false detections but increases the number of missed detections. All in all, the second set of experiments achieved APs of 82.98% and 91.23% on the HRSC2016 dataset and the RS-Ship dataset, respectively, 3.57% and 2.54% higher than those obtained from the first set of experiments.

The C2 layer feature maps of CSPDarknet53 contain rich detail information but little semantic information, which makes it difficult for the detector to distinguish between targets and interfering signals. Unlike Model3, Model4 can deeply extract the features in the C2 layer to construct P2 feature maps which contain rich detail information and semantic information, resulting in a significantly reduced number of missed detections and a 25.69% higher AP value.

Model5 integrates the ship detector and critical part detector in one framework, which allows the network to concentrate on the ship region during feature extraction, which reduces the influence of background on detection and facilitates the detection of targets. Compared with Model2, Model5 delivers 0.17% and 0.5% higher AP values.
for ship detection on the two datasets, respectively. And Model5 has a 0.69% higher AP value for critical part detection then Model4.

Our proposed model enhances the correlation between the two detectors based on Model5. The experimental results show that our proposed model delivers an AP value of 79.65%, suggesting that the enhanced correlation allows for effective detection of critical parts while having no effect on ship detection.

The experimental results show that the combination of the two detectors in this paper can improve the feature extraction ability of the backbone network. By mapping the prediction results of the ship detector to the critical part detector, the relationship between ships and their critical parts is fully utilized, and the region for target detection is filtered for reducing the interference of background on critical part detection. The effect of adding a mask on feature extraction is visualized in Figure 8. Specifically, columns 1-4 show input images, F2 layer feature maps, mask filtered maps, and P2 layer feature maps in the sixth set of experiments, respectively, and column 5 shows the P2 layer feature maps in the fifth set of experiments without filtered by mask, noted as P2’. By comparing P2 and P2’, it is clear that the mask can minimize the interference of coastline artifacts on the detection of critical parts. By restricting the target region, the feature extraction network can better characterize target features and make the salient features of targets more representative.

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

In this section, the effectiveness of our proposed method is verified through comparisons with Faster R-CNN [17] (with an additional FPN module), SSD [18], RetinaNet [39], YOLOv3 [21], YOLOv4 [34], YOLOF [40], and TOOD [41]. The quantitative results of ship and critical parts detection by each network model on the HRSC2016 and RS-Ship datasets are shown in Table 4. From Table 4 and the PR curves (shown in Figure 9), it can be seen that the detection for ship by each method is significantly better than that of critical part, which indicates that the detection of critical part is more difficult. The main reason is that the scale of critical part is small and the features are similar to some man-

| Methods  | Backbone   | Input resolution | HRSC2016 P (%) | Ship R (%) | AP (%) | FPS | RS-Ship P (%) | Ship R (%) | AP (%) | FPS |
|----------|------------|------------------|---------------|------------|--------|-----|---------------|------------|--------|-----|
| Faster R-CNN | ResNet-50 | 600 × 600        | 82.85         | 85.10      | 83.22  | 14.28 | 89.91         | 92.16      | 91.59  | 77.85 | 77.72 | 72.73 | 16.21 |
| SSD      | VGG-16     | 300 × 300        | 85.23         | 61.60      | 59.61  | 31.36 | 91.06         | 76.30      | 75.05  | 61.21 | 12.66 | 11.20 | 33.83 |
| RetinaNet | ResNet-50 | 512-768          | 84.81         | 76.79      | 73.81  | 15.84 | 88.58         | 89.84      | 88.13  | 77.80 | 66.84 | 60.78 | 18.33 |
| YOLOv3   | Darknet-53 | 416 × 416        | 82.40         | 68.41      | 65.50  | 28.24 | 94.58         | 90.20      | 89.25  | 72.19 | 65.75 | 30.35 |
| YOLOv4   | CSPDarknet-53 | 608 × 608      | 87.72         | 84.46      | 82.65  | 23.51 | 94.68         | 91.98      | 90.43  | 78.79 | 76.58 | 27.37 |
| YOLOF    | ResNet-50 | 640 × 640        | 57.15         | 78.15      | 73.09  | 35.8  | 72.79         | 89.66      | 86.97  | 40.68 | 59.89 | 38.63 | 34.1 |
| TOOD     | ResNet-50 | 640 × 640        | 88.47         | 84.67      | 83.30  | 19.4  | 95.55         | 95.72      | 95.04  | 76.11 | 70.41 | 64.34 | 19.3 |
| CPD-Net  | CSPDarknet-53 | 640 × 640      | 89.70         | 85.51      | 83.13  | 22.67 | 96.28         | 92.34      | 91.70  | 80.93 | 79.65 | 26.60 |

Figure 8: Visualization of F2, P2, and P2’ feature maps in the experimental model.
made objects on the coastline, which causes it to be more easily influenced by the background. By comparing the detection results with other methods, our proposed method has a superior detection performance. With VGG-16 as the backbone network, SSD performs poorly in feature extraction and thus suffers from a high rate of missed detection for small-scale targets, delivering AP values of 59.61% and 75.05% for ship detection and only 11.20% for critical part detection. RetinaNet uses the ResNet-50 network for feature extraction and integrates an FPN module to enrich features. This makes RetinaNet perform better than SSD. However, the FPN module does not completely avoid the drawbacks of poor performance on the detection of small-scale targets. RetinaNet delivers AP values of 73.81% and 88.13% for ship detection on the two datasets, respectively, and an AP value of 60.78% for critical part detection. Faster R-CNN is a classical algorithm for two-stage object detection, has strong detection capability, and performs well on both datasets. But there is still room for improvement in the detection of small-scale targets. YOLOV3 has a stronger feature extraction capability with an FPN structure added on Darknet-53, but its generalization capability is insufficient. In addition, YOLOV3’s ship detection performance varies greatly between the two datasets, and its capability for critical part detection is average. YOLOV4 outperforms YOLOV3 by combining the advantages of various detection algorithms. It delivers the equal level of ship detection performance on both datasets as Faster R-CNN, and its critical part detection performance is better than those of the first four algorithms. YOLOF substantially improves the speed of detection by simplifying the FPN, but it is not effective in detecting multiscale targets, resulting in its low detection accuracy on both datasets. TOOD enhances the interaction between classification and localization to improve the consistency between the two tasks, and this strategy achieves good results in ship detection but performs generally for the detection of critical parts, with an AP value of only 64.34%.

From the FPS statistics of each network model (shown in Table 4), the following observations can be drawn. SSD is significantly faster than other algorithms, since it is a single-stage network with a simple network structure. Since Faster R-CNN is a two-stage model, its complex network structure and processing procedures can significantly drag down its detection speed. Our proposed method uses the prediction results of the ship detector to restrict the target region, which improves the detection performance of the critical part detector. Although this design compromises a little on the network’s detection speed, it allows the FPS to be greater than 20 on both datasets, which meets the real-time detection requirement. To sum up, our proposed model can nicely balance the detection speed and accuracy and deliver optimal AP values on the detection of critical parts, proving that it is suitable for the detection of ships and critical parts in complicated scenes.

To compare the detection performance of each algorithm more intuitively, the detection results of different methods are visualized in Figure 10. The first two columns show the results on the HRSC2016 dataset, and the last two show the results on the RS-Ship dataset. From Figure 10, it can be observed that SSD, RetinaNet, and YOLOV3 perform poorly in detecting small-scale targets, and they miss quite a number of the targets. Faster R-CNN and YOLOV4 are significantly better than the first three algorithms for ship detection on both datasets, but they both have an increased number of false detections for critical parts because of the interference of a large number of artifacts on the coastline. YOLOF has more false detections of targets on both datasets, especially when detecting aligned targets. On both datasets, TOOD has a good detection performance for ships but is still affected by the background and has some false detection targets. The DD-Net has the lowest number of missed and false detections on both datasets and performs the best in the detection of ships and critical parts.

5. Conclusion

In this paper, we propose a dual detector network called DD-Net for detection of ships which is considered a multilevel object. We take a ship as main object and a critical part as subobject of the multilevels object, and we make special design for DD-Net in order to achieve accurate detection of both. The DD-Net consists of two specially designed detectors for recognize ships and critical parts of ships, respectively. For the ship detector, we use two different
directional pyramid structures to enrich the ship features and introduce attention modules to enhance the target saliency. For the critical part detector, we design an additional feature extraction module for increasing semantic information contained in low-level feature maps. To make full use of the relationship between a ship and its critical parts, we introduce an additional association between the two detectors to allow the critical parts to be detected inside each ship region with minimal influence of the background, thus improving the accuracy of critical part detection. The experimental results show that the proposed algorithm can accurately detect the ship’s critical parts while accomplishing the ship target detection process.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Figure 10: Visualization of detection results of different methods.
Authors’ Contributions

The manuscript was approved by all authors for publication.

References

[1] B. Cheng, Z. Li, B. Xu, X. Yao, Z. Ding, and T. Qin, “Structured object-level relational reasoning CNN-based target detection algorithm in a remote sensing image,” Remote Sensing, vol. 13, no. 2, p. 281, 2021.

[2] W. Zhang, C. Zhao, and Y. Li, “A novel counterfeit feature extraction technique for exposing face-swap images based on deep learning and error level analysis,” Entropy, vol. 22, no. 2, p. 249, 2020.

[3] Z. Huang, “A sea-land segmentation algorithm based on graph theory,” 2nd ISPRS International Conference on Computer Vision in Remote Sensing (CVRS 2015), vol. 9901, pp. 240–249, 2016.

[4] J. Xu, K. Fu, and X. Sun, “An invariant generalized Hough transform based method of inshore ships detection,” in 2011 International Symposium on Image and Data Fusion, pp. 1–4, Tengchong, China, 2011.

[5] G. Cheng, P. Zhou, X. Yao, C. Yao, Y. Zhang, and J. Han, “Object detection in VHR optical remote sensing images via learning rotation-invariant HOG feature,” in 2016 4th International Workshop on Earth Observation and Remote Sensing Applications (EORSA), pp. 433–436, Guangzhou, China, 2016.

[6] L. Jiao, F. Zhang, F. Liu et al., “A survey of deep learning-based object detection,” IEEE Access, vol. 7, pp. 128837–128868, 2019.

[7] W. Liu, L. Ma, and H. Chen, “Arbitrary-oriented ship detection framework in optical remote-sensing images,” IEEE Geoscience and Remote Sensing Letters, vol. 15, no. 6, pp. 937–941, 2018.

[8] M. Nie, J. Zhang, and X. Zhang, Ship Segmentation and Orientation Estimation Using Keypoints Detection and Voting Mechanism in Remote Sensing Images, Springer, 2019.

[9] K. He, G. Gkioxari, P. Dollár, and R. Girshick, “Mask r-Cnn,” in Proceedings of the IEEE international conference on computer vision, pp. 2961–2969, Venice, Italy, 2017.

[10] Z. Cao, G. Hidalgo, T. Simon, S. E. Wei, and Y. Sheikh, “OpenPose: realtime multi-person 2D pose estimation using part affinity fields,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 43, no. 1, pp. 172–186, 2021.

[11] X. Nie, J. Feng, X. Xing, and S. Yan, “Pose Partition Networks for Multi-Person Pose Estimation,” in Proceedings of the european conference on computer vision (eccv), pp. 684–699, Munich, Germany, 2018.

[12] M. Kocabas, S. Karagoz, and E. Akbas, “MultiPoseNet: fast multi-person pose estimation using pose residual network,” in Proceedings of the European conference on computer vision (ECCV), pp. 437–453, Munich, Germany, 2018.

[13] Z. Liu, L. Yuan, L. Weng, and Y. Yang, “A high resolution optical satellite image dataset for ship recognition and some new baselines,” International Conference on Pattern Recognition Applications and Methods, vol. 2, pp. 324–331, 2017.

[14] S. Kreiss, L. Bertoni, and A. Alahi, “PipPaF: composite fields for human pose estimation,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 11969–11978, Long Beach, CA, USA, 2019.

[15] R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 580–587, Columbus, OH, USA, 2014.

[16] R. Girshick, “Fast R-CNN,” in Proceedings of the IEEE international conference on computer vision, pp. 1440–1448, Santiago, Chile, 2015.

[17] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: towards real-time object detection with region proposal networks,” Advances in Neural Information Processing Systems, vol. 28, pp. 91–99, 2015.

[18] C. Y. Fu, W. Liu, A. Ranga, A. Tyagi, and A. C. Berg, “DSSD: deconvolutional single shot detector,” 2017, https://arxiv.org/abs/1701.06659.

[19] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: unified, real-time object detection,” in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 779–788, Las Vegas, NV, USA, 2016.

[20] J. Redmon and A. Farhadi, “YOLO9000: better, faster, stronger,” in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 7263–7271, Honolulu, HI, 2017.

[21] J. Redmon and A. Farhadi, “Yolov3: an incremental improvement,” 2018, https://arxiv.org/abs/1804.02767.

[22] M. Gong, Y. Y. Liu, and G. N. Li, “A ship detection method for remote-sensing images based on improved YOLO-v3,” Electronics Optics & Control, vol. 27, no. 5, pp. 102–107, 2020.

[23] C. Szegedy, W. Liu, Y. Jia et al., “Going deeper with convolutions,” in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1–9, Boston, MA, USA, 2015.

[24] S. Nie, Z. Jiang, H. Zhang, B. Cai, and Y. Yao, “Inshore ship detection based on Mask R-CNN,” in IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium, pp. 693–696, Valencia, Spain, 2018.

[25] Z. Liu, J. Hu, L. Weng, and Y. Yang, “Rotated region based CNN for ship detection,” in 2017 IEEE International Conference on Image Processing (ICIP), pp. 900–904, Athens, Greece, 2017.

[26] Q. Ming, L. Miao, Z. Zhou, and Y. Dong, “Cfc-net: A critical feature capturing network for arbitrary-oriented object detection in remote sensing images,” IEEE Transactions on Geoscience and Remote Sensing, vol. 60, 2021.

[27] Y. Zhang, W. Sheng, J. Jiang, N. Jing, Q. Wang, and Z. Mao, “Priority branches for ship detection in optical remote sensing images,” Remote Sensing, vol. 12, no. 7, p. 1196, 2020.

[28] Y. Yu, H. Guan, D. Li, T. Gu, E. Tang, and A. Li, “Orientation guided anchoring for geospatial object detection from remote sensing imagery,” ISPRS Journal of Photogrammetry and Remote Sensing, vol. 160, pp. 67–82, 2020.

[29] K. Song, P. Huang, Z. Lin, and T. Lv, “An oriented anchor-free object detector including feature fusion and foreground enhancement for remote sensing images,” Remote Sensing Letters, vol. 12, no. 4, pp. 397–407, 2021.

[30] T. Y. Lin, M. Maire, S. Belongie et al., “Microsoft coco: common objects in context,” in European conference on computer vision, pp. 740–755, Zurich, Switzerland, 2014.

[31] G. Moon, J. Y. Chang, and K. M. Lee, “Posefix: model-agnostic general human pose refinement network,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7773–7781, Long Beach, CA, USA, 2019.
[32] J. Li, C. Wang, H. Zhu, Y. Mao, H. S. Fang, and C. Lu, "Crowd-pose: efficient crowded scenes pose estimation and a new benchmark," in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 10863–10872, Long Beach, CA, 2019.

[33] Z. Liu, H. Chen, R. Feng et al., "Deep dual consecutive network for human pose estimation," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 525–534, Nashville, TN, USA, 2021.

[34] A. Bochkovskiy, C. Wang, and H. M. Liao, "Yolov4: optimal speed and accuracy of object detection," 2020, https://arxiv.org/abs/2004.10934.

[35] J. Deng, W. Dong, R. Socher, L. J. Li, K. Li, and L. Fei-Fei, "ImageNet: a large-scale hierarchical image database," in 2009 IEEE conference on computer vision and pattern recognition, pp. 248–255, Miami, FL, USA, 2009.

[36] T. Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 936–944, Los Alamitos, CA, USA, 2017.

[37] Y. Li, H. Li, W. Hu, and H. L. Yu, "DSPCANet: dual-channel scale-aware segmentation network with position and channel attentions for high-resolution aerial images," IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 14, pp. 8552–8565, 2021.

[38] J. Chen, J. Sun, Y. Li, and C. Hou, "Object detection in remote sensing images based on deep transfer learning," Multimedia Tools and Applications, vol. 1-17, 2021.

[39] T. Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in Proceedings of the IEEE international conference on computer vision, pp. 2999–3007, Venice, Italy, 2017.

[40] Q. Chen, Y. Wang, T. Yang, X. Zhang, J. Cheng, and J. Sun, "You only look one-level feature," in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 13034–13043, Nashville, TN, USA, 2021.

[41] C. Feng, Y. Zhong, Y. Gao, M. R. Scott, and W. Huang, "TOOD: task-aligned one-stage object detection," in 2021 IEEE/CVF International Conference on Computer Vision (ICCV), pp. 3490–3499, Montreal, QC, Canada, 2021.