Object Detection in Remote Sensing Images with Mask R-CNN

Yuhang Gan, Shucheng You, Zhengyu Luo, Ke Liu*, Tao Zhang and Lei Du

Land Satellite Remote Sensing Application Center, MNR, Baishengcun 1#, Haidian District, Beijing, 100048
Email: 181191804@qq.com
*Corresponding Author Email: liuk@lasac.cn

Abstract. With the wide applications of remote sensing technology in engineering, the demands of efficient object detection algorithms for remote sensing images have also been significantly increased in recent years. Traditional detection methods have the shortcomings of low accuracy and poor robustness, which is difficult to be applied to remote sensing images with complex background and varieties of objects. Recently, the deep convolutional neural networks have already shown great advances in object detection and outperformed many traditional methods. In this work, we study the performance of a region proposal-based method, Mask R-CNN, for detecting airplane and ship in remote sensing images. Specifically, we add an FPN module to improve the accuracy of small objects detection, and a mask branch is used to describe the shape of objects. In addition, we used a series of data augmentation strategies during training for meeting the CNN's requirement of training samples diversity. The experimental results show that our model has superior performance in object detection of remote sensing images.

1. Introduction

With the development of remote sensing technology, remote sensing images analysis has attracted growing attention in computer vision and data mining community. Through the automatic analysis and understanding of satellite or aviation images, we can better comprehend the changes in ground objects. Among these, object detection of remote sensing images is an important and challenging task and has a wide range of application prospects in intelligent monitoring, urban planning, precision agriculture, etc.

In traditional practice, remote sensing images are usually analyzed and labelled manually, which is time-consuming and would cause subjective errors. Compared with that, automatic object detection methods can effectively and accurately detect specific objects in those images. Therefore, it is significant and urgent to develop some advanced remote sensing object detection methods.

In the past decades, object detection of remote sensing images has been widely studied, and some achievements have been achieved. Guang et al. [1] use low-level features such as color, texture, and brightness to build feature map to achieve rough location, then train an AdaBoost cascade classifier, and extract harr features to conduct remote sensing object detection. Li et al.[2] propose to use PNN to classify surface objects after fusing features of different levels. To some extent, these traditional algorithms have solved some problems of remote sensing object detection, but there are still many shortcomings, including complicated steps, features design, insufficient generalization, or low accuracy. In addition, compared with natural scene images, the background of remote sensing images is more complex and would result in strong interference to detection.
Recently, deep learning algorithms play an important role in computer vision tasks. In particular, the deep convolution neural network (DCNN) has achieved excellent performance in image classification, segmentation, and detection. Different from traditional works, object detection methods based on deep learning first extract high-level semantic features through the DCNN model and then use different network branches to predict the category and location of objects. Deep learning methods can learn more discriminative semantic features. Meanwhile, those frameworks always can be trained in an end-to-end manner and reduce accumulated errors. Therefore, those methods always achieve robust detection performance. Considering remote sensing images are also visual images, and the excellent generalization ability of the DCNN model, it is rational to apply deep learning methods to object detection in remote sensing images.

In this paper, we introduce a region proposal-based object detection framework, Mask R-CNN [3], to detect specific objects, i.e., airplane and ship, in given remote sensing images. It should be noted that objects in remote sensing images are usually relatively smaller. In order to improve the detection performance of small objects, we add feature pyramid networks (FPN) to ResNet101 to deal with different sizes of objects. Furthermore, considering the arbitrary shape of the objects, a mask branch is used to accurately describe the shape of objects. Experiments show that our model can detect the specific objects in the remote sensing images accurately and quickly.

2. Related Work
Object detection methods based on deep learning have developed and achieved great success in a short time. The existing methods are mainly divided into two categories: region proposal-based detectors and regression-based detectors.

2.1. Region Proposal-based Detectors
The region proposal-based detectors usually divide the framework into two stages: proposals generation and region regression. The first stage focuses on generating a series of region proposals that may contain objects, and selects proposals with high confidence as the input of the next stage. The second stage aims to classify the proposals and fine-tuning the coordinates of proposals. R-CNN[4] proposed by Girshick R et al. is the most representative region proposal method, which first proposes to apply the CNN model to object detection, and achieves breakthrough performance compared with previous works. The R-CNN pipeline consists of three steps. Firstly, it uses a selective search algorithm to scan the images to generate about 2000 region proposals. Secondly, the region proposal is reset to a fixed size, and the corresponding feature representations are extracted by a pre-trained CNN model. Finally, the detector inputs the features to SVM with a category label for each proposal, and adjusts location using a linear regressor. To improve inefficiency and visual distortion in R-CNN, He et al. [5] adds a spatial pyramid pooling network (SPP-Net) to unify the dimension of features output, which not only reduces the computation but also solves the limitation of the fixed input size of the CNN model. Inspired by SPP-Net, Girshick R et al. improved R-CNN and propose its extensions, i.e., Fast R-CNN[6] and Faster R-CNN[7]. Fast R-CNN replaces spatial pyramid pooling with a ROI Pooling layer, and uses multi-task loss to train the network. Further, Faster R-CNN proposes a regional proposal network (RPN) to generate proposals and trains the whole detector in an end-to-end manner.

2.2. Regression-based Detectors
Different from region proposal-based detectors, regression-based detectors simplify object detection as a regression task, and directly predict the location and categories without proposals generation and features resampling, which is conducive to keep efficient. YOLO[8] is the most representative work of this kind. Its working principle is as follows: for a given input image, the detector divides it into S*S grids, and each grid detects the objects in this region via predicting B bounding boxes and C category with their confidence scores. Through a simple and efficient pipeline, YOLO realizes object detection in real-time. However, it has the shortcoming of poor detection accuracy, especially for small size objects. As an extension of YOLO, YOLOv2[9] greatly improves the detection accuracy while maintaining efficiency. Many optimization strategies, including Batch Normalization, Anchor
Box, Fine-Grained Features, Mutil-scale Training, etc., are adopted in the framework of YOLOv2. Furthermore, YOLOv3[10] proposes a new backbone network named darktnet-53 to perform feature extraction, and outputs multi-scale feature map to detect different size objects. With continuous improvement, the accuracy and speed of the YOLO series have achieved great progress. Compared with YOLO, SSD[11] achieves better performance for detecting and locating small-sized objects via introducing default boxes mechanism and multi-scale feature maps. By using focal loss instead of the traditional cross-entropy, Retina-Net[12] increases the detection accuracy significantly.

3. Mask R-CNN Architecture

Figure 1 depicts the overall architecture of Mask R-CNN used in our work. Functionally, the framework consists of four components: a feature pyramid network (FPN) as backbone for extracting multi-scale feature maps, a region proposal network (RPN) for generating proposals, a detection branch for detecting specific objects, and a mask branch for conduct segmentation of specific objects. The detection pipeline consists of the following steps: Firstly, we produce the multi-scale pyramid feature maps of input image through FPN. Secondly, RPN uses the features to generate proposals based on anchor boxes for the subsequent detection and mask branches. Next, we map the high-quality proposals to the feature maps and resample the corresponding region features via ROIAlign for predicting class, coordinates, and object mask of proposals in detection and mask branches. Finally, a multi-task loss is used to train the whole network.

**FPN.** Generally speaking, objects in remote sensing images are relatively small, which causes relatively weak response in the feature maps. To tackle this problem, we add an FPN on the basics of ResNet backbone. By adding a top-down sampling network, FPN combines deep layer features with shallow layer features, which aims to strengthen shallow layer feature representations with rich semantic information from the deeper layers. Therefore, FPN can fuse the feature of different resolutions from a single-scale input, which improves the final detection accuracy with marginal cost.

**RPN.** RPN takes the pyramid feature maps and generates a set of object proposals on each pixel of the feature maps. The network uses an n*n sliding window to scan the feature map and generate a set of 256-d feature vectors for each pixel on the feature map. The feature vectors will be input into the object classification layer and the box regression layer. The object classification layer aims to judge whether the proposal is foreground or background, while the box regression layer predicts a set of position offsets relative to the ground truth for proposals.

**Detection and Mask branches.** The detection branch and mask branch are two parallel branches, which are used for refining the proposals and generating the objects mask respectively. For the proposals from RPN, the detector uses ROIAlign to resample the region features from the feature map of image. Note that we don’t share regional features in different branches. ROIAlign in the detection branch generates the features with the size of 7x7 while 14x14 in the mask branch. Each branch further extracts semantic information via a set of 3x3 convolution layers and 1x1 convolution layers. Finally, the detection branch conduct classification with softmax, and fine-tune the coordinates of the bounding boxes. The mask branch produces the objects’ mask with a polygon.

**Overall Objective.** Lastly, we formulate the overall objective of the Mask R-CNN in this work:

\[
L = L_{rpn} + L_{det} + L_{mask}
\]  

where \(L_{rpn}, L_{det}, L_{mask}\) are the loss function of RPN, detection branch and mask branch respectively.

Figure 1. The overall pipeline of the proposed framework.
4. Experiments

4.1. Datasets Description
In this work, we build the corresponding datasets for airplane and ship, so as to train each class independently. Each dataset contains three subsets, including a pre-training dataset, a fine-tuning dataset, and a testing dataset. The pre-training dataset and the fine-tuning dataset are used for training models, and the testing dataset is used for evaluation. We select the relevant images from several public remote sensing datasets, i.e., DOTA[13], OPT2017[14], RSOD[15], to form the pre-training dataset. Especially, considering the large size of remote sensing images, we conduct random clipping, sliding clipping, and random rotating on images to enhance the diversity of samples. In addition, we collect many remote sensing images of airplane and ship, and divide them into the fine-tuning dataset and the testing dataset. For analyzing the influence of resolution, we divide the ship’s testing images into low-resolution dataset L and high-resolution dataset H. The details of the dataset are shown in Table 1.

Table 1. Details of datasets. “L” refers to low-resolution testing dataset. “H” refers to high-resolution testing dataset.

| Type    | Pre-training | Fine-tuning | Testing  |
|---------|--------------|-------------|----------|
| Ship    | 30980        | 2210        | 6580(L), 6482(H) |
| Airplane| 25109        | -           | 48       |
| In total| 56089        | 2210        | 13110    |

4.2. Implementation Details
We implemented our framework based on the Maskrcnn-benchmark, and all experiments were conducted on high-performance computing server with Nvidia Tesla V100 (16G) GPUs. All models were trained with GPUs and inference with 1 GPU.

The whole training process was divided into two stages: pre-training and fine-tuning. We choose ResNet-101 as the backbone. Moreover, to enhance network robustness, data augmentation strategies such as multi-scale training, adding noise randomly, and random color adjusting are applied. Stochastic gradient descent (SGD) is adopted to optimize our framework. The weight decay is set to 0.0001, momentum is set to 0.9, and batch size is set to 8. In the pre-training stage, we train the model on the pre-training dataset for 20 epochs. The learning rate is set to 0.01 in the first 10 epochs, divided by 10 in the last 10 epochs. In the fine-tuning stage, we train on the fine-tuning dataset based on the pre-trained model. the training iterations on the fine-tuning dataset are set to 20K. The learning rate is set to 0.005 in the first 10K iterations, divided by 10 in the remains.

4.3. Experiment Results
**Ship.** We evaluate the performance of our model on two testing datasets with different resolutions but the same image size to analyze the influence of resolution. As Table 2 shows, our model achieves an F-measure of 63.4% on low-resolution dataset L. When inferring images with high resolution, a better evaluation result can be achieved, which outperforms the former by 29.3%, 12.5%, 21.7% on Recall, Precision and F-measure respectively. According to these preliminary experimental results, we analyze that the boundaries between ships and other objects are blurred on the images at a low-resolution. Therefore, the model is difficult to distinguish ships from the background, leading to detection failures such as false positive. The visual inference results of images at different resolutions can be seen in Figure 2.
Table 2. Evaluation of detecting ship at different resolution.

| Resolution | Recall | Precision | F-measure |
|------------|--------|-----------|-----------|
| Low        | 56.7   | 71.8      | 63.4      |
| High       | 86.0   | 84.3      | 85.1      |

Figure 2. Example results of detecting ship on different resolution. Images from (a) to (b) are selected from $L$, and (c) to (d) are selected from $H$.

Airplane. Different from ship, we directly use the pre-trained model to evaluate the performance of detecting airplane. For testing images, all airplane instances have been detected. Combined with Figure 3 and Table 3, our model achieves an outstanding F-measure of 100%, and the predicted results perfectly describe the shape of airplane objects, showing excellent performance.

Table 3. Evaluation of detecting airplane.

| Class  | Recall | Precision | F-measure |
|--------|--------|-----------|-----------|
| Plane  | 100.0  | 100.0     | 100.0     |

Figure 3. Example results of detecting airplane with our model.

5. Conclusion
In this work, we apply a region-based detector, i.e., Mask R-CNN, to detect airplane and ship in remote sensing images. Based on the properties of remote sensing images, we use a variety of data augmentation strategies to enhance the robustness of framework. Moreover, we add a pyramid feature network to solve the poor performance of small objects. Our experimental results show that the region proposal-based detector can accurately detect airplane and ship in remote sensing images.

6. Acknowledgements
Thanks to the support of National Key Research and Development Program of China (NO. 2016YFB0501403).
7. References

[1] Shu G, Dehghan A, Shah M. Improving an Object Detector and Extracting Regions using Superpixels. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pages: 3721-3727, 2013.

[2] Xin-De L, Wei-Dong Y, Jean D. An Airplane Image Target’s Mutil-feature Fusion Recognition Method. ACTA AUTOMATICA SINICA. 38(8):1298-1307, 2012.

[3] He K, Gkioxari G, Dollar P, et al. Mask r-cnn. In Proceedings of the IEEE International Conference on Computer Vision. pages: 2961-2969, 2017.

[4] Girshick R, Donahue J, Darrell T, et al. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pages: 580-587, 2014.

[5] He K, Zhang X, Ren S, et al. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence. 37(9): 1904-1916, 2015.

[6] Girshick R. Fast r-cnn. In Proceedings of the IEEE International Conference on Computer Vision. pages: 1440-1448, 2015.

[7] Ren S, He K, Girshick R, et al. Faster r-cnn: Towards Real-time Object Detection with Region Proposal Networks. In Advances in Neural Information Processing Systems. pages: 91-99, 2015.

[8] Redmon J, Divvala S, Girshick R, et al. You only look once: Unified, Real-time Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pages: 779-788, 2016.

[9] Redmon J, Farhadi A. YOLO9000: better, faster, stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pages: 7263-7271, 2017.

[10] Redmon J, Farhadi A. Yolov3: An incremental improvement. arXiv:1804.02767, 2018.

[11] Liu W, Anguelov D, Erhan D, et al. SSD: Single Shot Multibox Detector. In Proceedings of European Conference on Computer Vision. pages: 21-37, 2016.

[12] Lin T Y, Goyal P, Girshick R, et al. Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision. pages: 2980-2988, 2017.

[13] Xia G S, Bai X, Ding J, et al. DOTA: A large-scale dataset for object detection in aerial images. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pages: 3974-3983, 2018.

[14] Wu P C, Lee Y Y, Tseng H Y, et al. A Benchmark Dataset for 6DoF Object Pose Tracking. IEEE International Symposium on Mixed and Augmented Reality. pages: 186-191, 2017.

[15] Z Xiao, Q Liu, G Tang, X Zhai. Elliptic Fourier transformation-based histograms of oriented gradients for rotationally invariant object detection in remote-sensing images. International Journal of Remote Sensing, vol. 36, no. 2, 2015.

[16] F Massa, R Girshick. Maskrcnn-benchmark: Fast, modular reference implementation of instance segmentation and object detection algorithms in pytorch. 2018.