ISSN 2616-5767 Vol.6, Issue 5: 17-22, DOI: 10.25236/AJETS.2023.060504

Performance Evaluation of Ship Target Detection Algorithms Based on Deep Learning

Peilin Li

College of Art and Science, The Ohio State University, Columbus, Ohio, 43201, United State

Abstract: *Ship target detection plays an important role in marine military and shipping. Deep learning allows us to extract deep features from large amount of data. In this paper, we select three different target detection algorithms based on deep learning, including Faster R-CNN, SSD, and YOLOv3, and apply the same dataset to these three algorithms. Then compare the results of the experiments and evaluate the performance of each algorithm. According to the result of the experiments, Faster R-CNN has a relatively better performance. The result of this paper would provide a reference for selecting a ship target detection algorithm.*

Keywords: *Deep Learning, Ship Target Detection, Convolutional Neural Network, Faster R-CNN, SSD, YOLOv3*

1. Introduction

As a part of marine transportation, shipping plays an important role in various marine affairs such as border control, environmental protection, traffic monitoring and rescue. Also, considering the ability to collect information over large spatial areas, remote sensing images are widely used in monitoring. Therefore, adopting remote sensing data for ship detection enable is to collect more valuable information. The technology of ship target detection can be used to position the ship and keep track of the ship's movement. That helps the marine military to protect marine safety and monitor maritime transportation [1-2].

Target detection has been a major topic of computer vision in the past decades. Most of the traditional target detection methods can be roughly divided into two steps: feature extraction and classification. People usually directly extract features from the image by using feature descriptors like HOG (Histogram of Oriented Gradients) and SIFT (Scale Invariant and Feature Transform), then classify the extracted features by directly passing the features to classifiers. For example, in 2017, Xu et al. proposed a maritime target detection method based on the HOG [3]. This method has relatively high precision and good robustness but still has difficulty in extracting accurate features. And two years later, they also proposed another ship detection method based on the Fourier HOG and SVM classifiers [4]. Although the improved method has greatly improved the precision and object segmentation, due to the limitation of feature extraction, it still had difficulty in identifying the ship and its wake [4].

Considering the benefits of deep learning, deep learning has been applied to various fields including image recognition [5]. The Regions with CNN features(R-CNN) method, which is the first CNN-based target detection model, was proposed by Girshick et al. in 2014[6]. After that, more and more CNN-based algorithms are released in the following decade. Most of the CNN-based detection methods can be roughly divided into two categories: one is the methods based on regional suggestion networks such as R-CNN, Fast R-CNN, and Faster R-CNN, and the other one is the methods based on regression methods such as YOLO, SSD [7]. One of the biggest differences between these two kinds of methods is that the regional-suggestion-based methods require regional proposal while extracting features from the images, but regression-based methods don't [8]. Both of these two kinds of detection models have already been applied to ship target detection. In 2019, Wang et al. proposed an improved YOLOv3 algorithm for ship target detection [2]. And Mou et al. proposed an improved Faster R-CNN algorithm for marine detection as well [9]. The difference between these two kinds of algorithms also leads to different performances on target detection. The regional-suggestion-based CNN algorithm usually has relatively high accuracy, but it has poor performance in real-time performance. In the contrast, the regression-based CNN algorithms have a relatively low accuracy but do well in real-time detection [6-8]. And due to the different performance, it may raise problems while selecting algorithms. Therefore, in this paper, we would select three algorithms from both categories and compare their performance on the same dataset, then provide

ISSN 2616-5767 Vol.6, Issue 5: 17-22, DOI: 10.25236/AJETS.2023.060504

a reference for algorithm selection on ship target detection.

In this paper, we would evaluate and compare the performance of the following target detection algorithms: Faster R-CNN, SSD, and YOLOv3. A dataset containing a number of ship photos would be used to train and test the three different models. And the resulting value of the mean average precision (mAP) for each algorithm would be used as the criteria to compare the performance of each algorithm.

The rest of this paper is organized as follows: Section 2 introduces the concept of three different target detection algorithms: F-RCNN, SSD, and YOLOv3. In section 3, we would first introduce the dataset and environment of the experiments, then we would display and analyze the results of each algorithm. Finally, we would present the conclusion in section 4.

2. Target Detection Algorithms

2.1 Faster R-CNN

As mentioned above, Faster R-CNN is one of the regional suggested networks-based CNN methods. Faster R-CNN was proposed by Ren et al. in 2015 [10]. The concept of Faster R-CNN was based on Fast R-CNN and compared to Fast R-CNN. It improved the mAP of the algorithm while effectively reducing the cost of proposal calculation.

The process of Faster R-CNN, shown as Fig. 1, can be roughly divided into 4 parts: convolutional (conv) layers, region proposal network (RPN), region of interest (RoI) pooling, and classification [10]. In the beginning, an image will be firstly scaled to a fixed size M × N and passed to the convolutional network. In this paper, we would use the VGG network as the backbone [11]. The original VGG network includes convolutional layers and fully connected (FC) layers [11]. But in Faster R-CNN, the convolutional layers are used to extract the feature map from the picture, only the convolutional layers would be used [10]. There are 13 shareable convolutional layers in the VGG model. The feature map generated by the convolutional layers will be directly passed to the RPN.

The regional proposal network (RPN) would take the feature map generated by the convolutional layers as input and output object/region proposals. To generate a region proposal, it will first operate a sliding window on the feature map. For each sliding window, it would be mapped to a lower-dimensional vector (512-d vector in this paper since it is generated by VGG), which will also be passed to two sibling fully connected layers: a box-regression layer (reg) and a box-classification layer (cls) [10]. And for each sliding widow, it will generate k predicted region proposals, called anchors [10]. Then the reg layer would have 4k outputs which are the corresponding coordinates of k boxes(including x,y,w,h, which (x,y) represent the center of the region proposal, w is the width of the box and h is the height of the region proposal, these four outputs represent the offset of each anchor) and the cls layer would have 2k output which is the probability of object or not-object for each proposal (the probability that each proposal is a ship or not a ship) [10]. Then proposal layer, which is also the last layer of the network, will output the region proposals based on the outputs of the reg layer and cls layer [10].

The concept of the region of interest (RoI) pooling was first introduced in the paper where Fast R-CNN was published [12]. It would take the region proposals and feature maps as input and output proposal feature maps. This process ensures that every outputted feature map has the same fixed size which would also accelerate the following process. In this process, it will first rescale each proposal to its corresponding feature map size and divide the corresponding feature map into an H × W grid. Then it would apply RoI max pooling to each grid cell and pass the final proposal feature maps to the classifier [12]. As the classifier takes the feature maps as input and passes them to the SoftMax classifier and bounding-box regressor so that we can get the final object position the object could be classified [12].

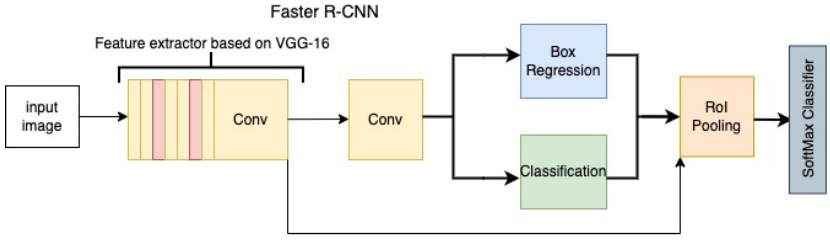

Figure 1: The Structure of Faster R-CNN

ISSN 2616-5767 Vol.6, Issue 5: 17-22, DOI: 10.25236/AJETS.2023.060504

2.2 SSD

Single Shot MultiBox Detector is one of the single-shot (one-stage) detection models. It was first published by Liu, Wei, et al in 2016 [13]. There are two kinds of SSD networks, one is SSD 300 and the other one is SSD 512. The number after SSD represents the size of the input image. As shown in Fig. 2, the SSD network also takes VGG-16 as the backbone of the base network. It only takes the first 5 convolutional layers from the original VGG-16 model and removes the last pooling layers [13]. Then replaces the last two fully connected layers with two convolutional layers and adds four extra convolutional layers at the end of the network [13]. This structure generates multi-scale feature maps for detection. To achieve higher accuracy, some layers with different scales would produce a fixed set of detection predictions by using a set of convolutional filters (3×3) [13]. It would either produce a score of a classification or a shape offset relative to the default box. For each grid cell in the feature maps, it would predict k default boxes and apply a convolutional filter to the predicted boxes, which is similar to the anchors in Faster R-CNN. Finally, it would collect the scores and predicted default boxes generated by the feature maps, then use non-maximum suppression (NMS) to produce the final detections [13].

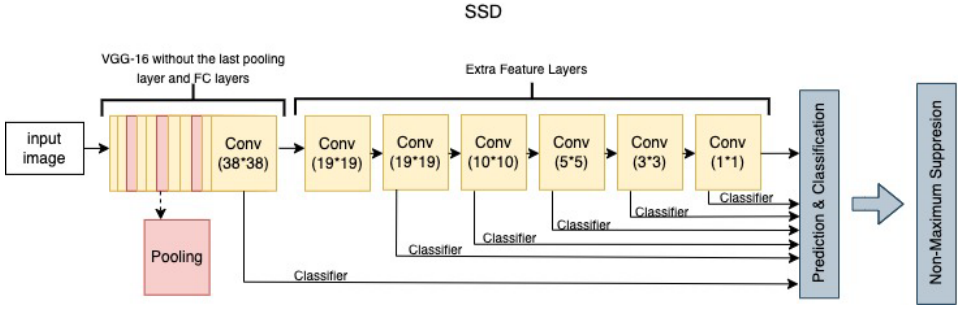

Figure 2: SSD Structure

2.3 YOLOv3

YOLO is the short-term of "you only look once" [14], which is also a one-stage target detection model. The first version of YOLO (YOLOv1) [14] was published in 2016 by Redmon, Joseph, et al. In this paper, we would use YOLOv3 as the third target detection method.

In YOLOv3, it would use its new feature extraction network: Darknet-53[15] as its base feature extractor, shown as Fig. 3. The Darknet neural network was first proposed in YOLOv2, which was called "Darknet-19" since it has 19 convolutional layers [16]. Similarly, Darknet-53 means there are 53 convolutional layers in this network, but the structures of Darknet-53 and Darknet-19 are quite different. Considering the benefit of the residual network [17], Darkent-53 uses multiple repeated residual blocks to create a deeper network. Each convolutional layer in Darknet-52 contains one 2-d convolutional layer, one batch normalization (BN) [18] layer, and a Leaky ReLu layer. The network would produce three different scale feature maps: 13 × 13, 26 × 26, and 52 × 52. Multi-scale feature maps enable the model to predict different size objects.

For each feature map, the system will predict 3 boxes at each grid cell. Therefore, in this case, the tensor of each feature map would be N × N × [3 * (4 + 1 + c)] which includes 4 bounding box offsets, 1 objectness prediction, and c is the number of classes prediction (where was 80 in [15] since there are 80 classes in COCO [19]). The system would take the first feature map (13 × 13) as input and pass it to a convolutional set and two more convolutional layers. Then the system would take the original feature map as input, upsample it by 2 times so that it has the same shape as the second feature map and concatenate it to the second feature map [15]. It would allow the feature map to contain more meaningful information. And repeat the same process as mentioned above for the third feature map [15]. At this point, the third feature map would gain the feature from the previous feature map, which is also "a 3-d tensor encoding bounding box, objectness, and class predictions." [15].

As the output contains the offset of the predicting bounding box, the system will first calculate the final bounding box and then classify the object class by using a logistic classifier [15]. Different from YOLOv2 [16], it uses a logistic classifier because there could be multiple objects in one bounding box, a logistic classifier allows multilabel while doing class prediction [15].

ISSN 2616-5767 Vol.6, Issue 5: 17-22, DOI: 10.25236/AJETS.2023.060504

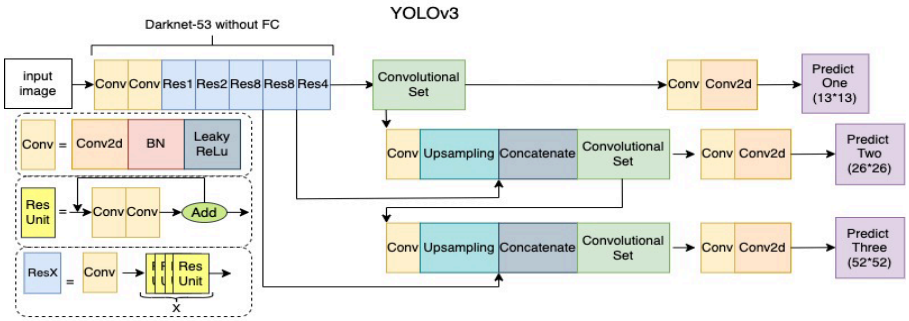

Figure 3: YOLOv3 Structure

3. Experiments

3.1 Data

In this paper, we would use a customized dataset to train and test three different models. The data set includes 3689 pictures of ships and corresponding XML files that describe the information of the picture such as the name (class) and size of the picture. In this paper, 90% of the data would be used to train the model (the ratio of the training set and validation set is 9:1), and the rest of the data would be considered as the testing set. The dataset would be processed as VOC (or PASCAL VOC, which is the acronym for pattern analysis, statical modeling, and computational learning visual object classes] [20] format before it is passed to the models.

3.2 Environment

The experiments are implemented under the PyTorch framework through the Python programming language on a 64-bit computer with Intel XeonI CPU E5-2699, 128 GB RAM, and Geforce RTX2080TI with GUDA10.2 and cuDNN7.2. The initial learning rate is set to 0.001. The batch size is set to 8, and the epoch is set to 100. The stochastic gradient descent (SGD) optimizer is employed. The weight decay and momentum coefficient are set to 0.0005 and 0.9, respectively. The threshold of the Intersection over union (IOU) is set to 0.5, and the threshold of confidence is set to 0.45.

3.3 Results & Analysis

In the experiment, the same dataset would be applied to three different models: Faster R-CNN, YOLOv3, and SSD. Then the performance of each model would be evaluated by comparing the resulting mAP (mean Average Precision). To explain the definition of mAP, it is necessary to mention the confusion matrix of the Precision-Recall curve. The confusion matrix includes 4 attributes:

True Positive (TP): the instance is positive and is predicted as positive.

False Negative (FN): the instance is positive and is predicted as negative.

True Negative (TN): the instance is negative and is predicted as negative.

False Positive (FP): the instance is negative and is predicted as positive.

And the definition of precision and recall are shown as below:

$$\text{Precision} = \frac{TP}{\text{TP+FP}} \tag{1}$$

$$\text{Recall} = \frac{TP}{\text{TP+FN}} \tag{2}$$

The Precision-Recall curve takes the precision scores as the y-axis and recalls scores as the x-axis. Average precision (AP) measures the area under the curve, therefore, the value of AP is always between 0 and 1. The mAP measures the average AP of each class. The formula of mAP is shown below, where n is the total number of classes. In this paper, there is only one class: ship.

$$\text{mAP} = \frac{\sum_{i=1}^{n} AP_i}{n} \tag{3}$$

ISSN 2616-5767 Vol.6, Issue 5: 17-22, DOI: 10.25236/AJETS.2023.060504

According to the data shown in Table 1, under the same threshold (IOU=0.5), the mAP of each model rank from high to low is Faster R-CNN > SSD > YOLOv3. The mAP of Faster R-CNN is 5.12% higher than the mAP of SSD and 8.3% higher than the mAP of YOLOv3. Based on the data shown in Table 1, if we take mAP as the only metric to evaluate the performance of the model, we can see that Faster R-CNN has relatively higher precision and therefore, has the best performance among the three models. Faster R-CNN, as a two-stage algorithm, would extract certain region proposal before it is passed to the classifier. It ensures the target has been distinguished from the background, therefore, the feature maps passed to the classifier contain more useful information and avoid being affected by the noises from the background. In this case, It indicates that the two-stage algorithm such as Faster R-CNN has advantages in terms of detection precision compared to the one-stage algorithms. However, though one-stage algorithms like SSD and YOLOv3 have lower precision, one-stage algorithms usually process faster in real-time. If real-time is considered as a metric, then the performance evaluation may be different.

Table 1: Mean average of three different networks

Network	Backbone	Batch Size	mAP (%)
Faster R-CNN	VGG	4	88.9
SSD	VGG	16	83.78
YOLOv3	Darknet-53	8	80.6

4. Conclusion

This paper shows the different applications and importance of ship target detections. Also, the experimental data shows that Faster R-CNN has the best performance among the three selected target detection networks by comparing the mAP of three different ship target detection networks. But there are more attributes that should be considered while comparing the target detection networks. For example, processing speed and efficiency are also one of the most necessary attributes to evaluate the performance of a target detection network. As more attributes are taken into consideration while evaluating the performance, a more overall reference could be given to people. Therefore, in the future, more attributes and experimental data should be included to improve the quality of the comparison between the target detection networks. Other than that, in this paper, we didn't evaluate the difference in precision between the small target and the big target. The precision of different size targets will usually be varied. The precision of different size targets and how to improve the precision is also another track worth investigating, and this will be left for future research.

References

[1] Behera B, Kumaravelan G, Prem K B. Performance Evaluation of Deep Learning Algorithms in Biomedical Document Classification[C]// 2019 11th International Conference on Advanced Computing (ICoAC). 2019.

[2] Kim Y, Kwak G H, Lee K D, et al. Performance Evaluation of Machine Learning and Deep Learning Algorithms in Crop Classification: Impact of Hyper-parameters and Training Sample Size[J]. The Korean Society of Remote Sensing, 2018(5).

[3] Fang C, Huang J, Cuan K, et al. Comparative study on poultry target tracking algorithms based on a deep regression network[J]. Biosystems Engineering, 2020, 190:176-183.

[4] Wang X, Ji S, Liang Y. Research on Fake Rating Detection Algorithm Based on Deep Learning[C]// CONF-CDS 2021: The 2nd International Conference on Computing and Data Science. 2021.

[5] Yang D W, Jia X B, Xiao Y J, et al. Noninvasive Evaluation of the Pathologic Grade of Hepatocellular Carcinoma Using MCF-3DCNN: A Pilot Study[J]. BioMed Research International, 2019, 2019(1):1-12.

[6] Yuan M X, Zhang L M, Zhu Y S, et al. Ship target detection based on deep learning method[J]. Ship Science and Technology, 2019.

[7] Lytvyn V, Pukach P, Vysotska V, et al. Identification and Correction of Grammatical Errors in Ukrainian Texts Based on Machine Learning Technology. 2023.

[8] Duan Y, Lv Y, Wang F Y. Performance evaluation of the deep learning approach for traffic flow prediction at different times[C]// IEEE International Conference on Service Operations & Logistics. IEEE, 2016.

[9] Zhou X, Yamada K, Takayama R, et al. Performance evaluation of 2D and 3D deep learning approaches for automatic segmentation of multiple organs on CT images[C]// Computer-Aided Diagnosis. 2018.

[10] Yang D, Yang L, Liu Y. Research and Implementation of Embedded Real-time Target Detection

ISSN 2616-5767 Vol.6, Issue 5: 17-22, DOI: 10.25236/AJETS.2023.060504

Algorithm Based on Deep Learning[J]. JPhCS, 2022.
[11] Jin H, Yan M, Lu J, et al. Performance Comparison of Moving Target Recognition between Faster R-CNN and SSD[C]// 2019 International Joint Conference on Information, Media and Engineering (IJCIME). 2019.
[12] Li Z, You Y, Liu F. Analysis on Saliency Estimation Methods in High-Resolution Optical Remote Sensing Imagery for Multi-Scale Ship Detection[J]. IEEE Access, 2020, 8:194485-194496.
[13] You G, Zhu Y. Target Detection Method of Remote Sensing Image Based on Deep Learning[C]// 2020 Cross Strait Radio Science & Wireless Technology Conference (CSRSWTC). 2020.
[14] Ren Q, Chang J, Han H. The Realization of Moving Target Tracking in Monitoring Video based on Deep Leaming[C]// 2019 IEEE 1st International Conference on Civil Aviation Safety and Information Technology (ICCASIT). IEEE, 2019.
[15] Zheng Z, Lei L, Sun H, et al. A Review of Remote Sensing Image Object Detection Algorithms Based on Deep Learning[C]// 2020 IEEE 5th International Conference on Image, Vision and Computing (ICIVC). IEEE, 2020.
[16] Sun S, Wang Y, Piao Y. A Real-time Multi-target tracking method based on Deep Learning[J]. Journal of Physics: Conference Series, 2021.
[17] Wu T, Liu H, Zhu J, et al. A Review of Camouflaged Target Detection Research[C]// 2021 IEEE 4th Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC). IEEE, 2021.
[18] Zhang Y, Chen G, Cai Z. Small Target Detection Based on Squared Cross Entropy and Dense Feature Pyramid Networks[J]. IEEE Access, 2021, PP(99):1-1.
[19] Dey A. Deep IDS: A deep learning approach for Intrusion detection based on IDS 2018[C]// 2020 2nd International Conference on Sustainable Technologies for Industry 4.0 (STI). 2020.
[20] Zheng Y, Chen Z, Lv D, et al. Air-to-Air Visual Detection of Micro-UAVs: An Experimental Evaluation of Deep Learning[J]. IEEE Robotics and Automation Letters, 2021, PP(99):1-1.