Ship Target Detection Based on Improved YOLO Network

Hong Huang,1 Dechao Sun,1 Renfang Wang,1 Chun Zhu,1 and Bangquan Liu2

1College of Big Data and Software Engineering, Zhejiang Wanli University, Ningbo, China
2College of Digital Technology and Engineering, Ningbo University of Finance and Economics, Ningbo, China

Correspondence should be addressed to Dechao Sun; sundechao123@163.com and Chun Zhu; pigspring@zwu.edu.cn

Received 27 May 2020; Revised 13 July 2020; Accepted 20 July 2020; Published 17 August 2020

Abstract
Ship target detection is an important guarantee for the safe passage of ships on the river. However, the ship image in the river is difficult to recognize due to factors such as clouds, buildings on the bank, and small volume. In order to improve the accuracy of ship target detection and the robustness of the system, we improve YOLOv3 network and present a new method, called Ship-YOLOv3. Firstly, we preprocess the input image through guided filtering and gray enhancement. Secondly, we use k-means++ clustering on the dimensions of bounding boxes to get good priors for our model. Then, we change the YOLOv3 network structure by reducing part of convolution operation and adding the jump join mechanism to decrease feature redundancy. Finally, we load the weight of PASCAL VOC dataset into the model and train it on the ship dataset. The experiment shows that the proposed method can accelerate the convergence speed of the network, compared with the existing YOLO algorithm. On the premise of ensuring real-time performance, the precision of ship identification is improved by 12.5%, and the recall rate is increased by 11.5%.

1. Introduction

With the vigorous development of shipping industry, the water traffic is more and more busy. Due to the frequent occurrence of collisions and other accidents between ships, it is necessary to detect the types of ships effectively to ensure the safety of water traffic. The ship detection technology based on computer vision is of great significance to improve port management and maritime inspection.

The traditional methods of ship detection are based on the automatic identification system and ship features [1, 2]. Li et al. proposed an improved dimensional space clustering algorithm to identify abnormal behavior of ships [3]. Zhang et al. used AIS data to identify ships with attempted collision [4, 5]. Zhou et al. proposed a detection method for classification and identification of bow and hull [6]. Zang et al. carried out ship target detection from the nonfixed platform [7, 8]. Although these studies have achieved good results, there are generally problems such as low recognition accuracy and human intervention. As a result, the traditional ship detection method is difficult to achieve the ideal detection effect.

In recent years, with the rapid development of artificial intelligence technology, the method based on deep learning has become the mainstream detection method. At present, there are two methods to solve the problem of target detection through deep learning: two-stage detection method and one-stage detection method. The two-stage algorithm uses region suggestion detection, mainly including AlexNet [9], VGG [10], ResNet [11], Fast-RCNN [12], and Faster-RCNN [13]. Although the detection accuracy is better than the traditional method, the detection speed is slightly insufficient. The feature extraction process takes a long time and is difficult to achieve the effect of real-time detection. In order to ensure the accuracy and improve the detection speed, one-stage algorithm is proposed. The one-stage detection does not use the idea of combining fine detection with rough detection, but directly detects the results in a single stage. The whole process does not need region suggestion detection and directly realizes end-to-end detection for inputting images. Therefore, the detection speed is greatly improved. The one-stage algorithm mainly includes SSD [14], YOLO [15], YOLOv2 [16], and YOLOv3 [17].
The YOLOv3 algorithm keeps the accuracy and improves the speed, which is favored by many researchers. Zhang et al. proposed multiscale ship target detection based on deep learning [18, 19]. Feng et al. improved the effect of ship classification and recognition by using spatial transform segmentation [20]. Li et al. used one-dimensional target to detect SAR image and used classifier to determine ship type [21, 22]. Chen et al. proposed automatic ship identification and behavior analysis detection based on video, which can accurately detect the ship and successfully identify the historical behavior of the ship [23]. Huang et al. proposed an improved YOLOv3 network for intelligent detection and classification of ship images/videos [24, 25]

Although the YOLOv3 algorithm has achieved good detection results, it has low recognition accuracy in complex scenes such as fog or night, and has phenomenon of missing detection for small target ships. Therefore, this paper proposes a Ship-YOLOv3 algorithm to solve these problems. We optimize the anchor boxes of YOLOv3 algorithm and improve the network structure of the algorithm. The experimental shows that this method can accelerate the convergence speed of the network under the premise of ensuring the real-time performance. Compared with the YOLOv3 algorithm, the precision of ship identification is improved by 12.5%, and the recall rate is increased by 11.5%.

Compared to other works on ship detection, our algorithm possesses the following advantages:

(1) Target box dimension clustering: the K-means++ algorithm is used to cluster the target box of self-made ship dataset. The optimal width and height value is calculated and the predefined anchor in YOLOv3 is modified accordingly.

(2) Improvement of network structure: aiming at the problem of false recognition rate of ship small target detection, the basic network Darknet53 in YOLOv3 algorithm is improved. Some redundant network layers of Darknet53 are reduced, and the jump join mechanism of residual network is added to enhance the detection of small ship targets.

2. Preliminaries

In this section, some essential concepts relating to YOLOv3 are briefly reviewed; these will be used in the rest of this work.

The target detection algorithm of YOLOv3 is proposed on the basis of YOLOv1 and YOLOv2. In order to achieve better classification effect, the residual network is used to realize jump connection. At the same time, the convolution with step size of 2 is used for down sampling to reduce the negative gradient effect caused by pooling. In addition, a batch normalization and activation function of each convolution layer are added to avoid overfitting in training. Based on the up sampling and fusion mechanism of FPN, three scales of output are designed to improve the accuracy of small target detection. The network structure of YOLOv3 is shown in Figure 1.

Firstly, the input image is divided into $S \times S$ grids. When the center of the object falls into the grid in the image, the grid is responsible for predicting the bounding box of the object. The corresponding score of this grid is 1, and the other grids are 0. Each grid predicts 3 bounding boxes, and each box represents $(5+N)$ values. The value 5 contains the location information of the bounding box: the center coordinates $(x,y)$, width and height $(w,h)$, and confidence of the bounding box. The value $N$ represents the number of categories in the dataset.

The NMS algorithm filters the confidence level of the target in the grid and obtains the bounding box with the highest score as the object detection frame. Since a target may belong to multiple categories, the Softmax layer is replaced by a $1 \times 1$ convolution layer and a logistic regression activation function structure. The score of each category is predicted by logistic regression, and the target is predicted by a threshold. The category higher than the threshold is the real category of the bounding box.

The error between the predicted value and real value is usually calculated with the following cross-entropy loss function:

$$
\text{loss} = \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{ij}^{\text{obj}} \left[ (\sigma(t_j)_1 - \sigma(\hat{t}_j)_1)^2 + (\sigma(t_j)_2 - \sigma(\hat{t}_j)_2)^2 \right] + \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{ij}^{\text{obj}} \left[ (t_{w_j} - \hat{t}_{w_j})^2 + (t_{h_j} - \hat{t}_{h_j})^2 \right] + \sum_{i=0}^{S^2} \sum_{j=0}^{B} G_{ij} (c_j^i - \hat{c}_j^i)^2 + \sum_{i=0}^{S^2} \sum_{j=0}^{B} \sum_{c \in \text{classes}} \hat{I}_{ij}^{\text{obj}} (p_j^i(c) - \hat{p}_j^i(c))^2,
$$

where $I_{ij}^{\text{obj}}$ means that when the $j$th bounding box of the $i$th grid is responsible for predicting the target. The $I_{ij}^{\text{obj}}$ value is 1, otherwise it is 0. The $S^2$ is the number of grids and $B$ is the number of bounding boxes in each grid. The $\sigma(t_j)$ and $\sigma(\hat{t}_j)$ are the offsets of the center position. The $t_{w_j}$ and $t_{h_j}$ are offsets for width and height. The parameter $C$ is the confidence score. The $p$ is the probability of ship class.
clustering, YOLOv3 network structure optimization, and data processing. Next, the implementation process of each module is introduced in detail.

3.1. Target Box Dimension Clustering. In the target detection task, selecting an appropriate anchor can significantly improve the speed and accuracy of target detection. Although the YOLOv3 algorithm is trained on MSCOCO and PASCAL VOC dataset, nine groups of anchors are obtained. The anchor values are (10, 13), (16, 30), (33, 23), (30, 61), (62, 45), (59, 119), (116, 90), (156, 198), and (373, 326). However, the anchor values used in COCO and VOC dataset are not suitable for the ship dataset used in this paper. Therefore, this paper uses the K-means++ algorithm to cluster the ship dataset and selects nine different anchors. The anchor values are (66, 13), (136, 22), (162, 30), (218, 30), (217, 38), (220, 47), (212, 64), (347, 40), and (444, 62). Compared with manual
selection of the prior frame, clustering can accelerate the convergence of the network and effectively improve the gradient descent in the training process. The cluster evaluation criteria are as follows:

\[ d(\text{bbox}, \text{center}) = 1 - \text{IOU}(\text{bbox}, \text{center}), \] (2)

where \( d(\text{bbox}, \text{center}) \) is the distance between the bounding box and the center box and \( \text{IOU}(\text{bbox}, \text{center}) \) is the intersection over union between the two boxes. This clustering method can produce a larger intersection ratio and a smaller distance between the bounding boxes in the same cluster. Each grid in the ship feature map of each scale in Figure 3 predicts three prior frames. It contains \((4 + 1 + N)\)-dimensional vector, which represents the central coordinates \((x, y)\), width and height \((w, h)\), and confidence of the bounding box. The \( N \) represents the number of categories in the dataset:

\[
\begin{align*}
\mathbf{b}_x & = \sigma(t_x) + c_x, \\
\mathbf{b}_y & = \sigma(t_y) + c_y, \\
\mathbf{b}_w & = \mathbf{p}_w e^{\mathbf{e}_w}, \\
\mathbf{b}_h & = \mathbf{p}_h e^{\mathbf{e}_h}, \\
p_r(\text{object}) \cdot \text{IOU}(\text{b, object}) & = \sigma(t_0), \\
\text{IOU}(\text{b, object}) & = \frac{\text{area}(\text{b}) \cap \text{area}(\text{g})}{\text{area}(\text{b}) \cup \text{area}(\text{g})},
\end{align*}
\]

where \( \mathbf{b}_x \) and \( \mathbf{b}_y \) are the ensemble framework detection results and \( t_x \) and \( t_y \) are the predicted center coordinates of each ship’s bounding box prior. After the activation function in the network, the offsets \( \sigma(t_x) \) and \( \sigma(t_y) \) corresponding to the center coordinates of the ship’s bounding box will be obtained. The \( \mathbf{b}_w \) and \( \mathbf{b}_h \) are the width and height of the detected box, respectively. The \( \mathbf{p}_w \) and \( \mathbf{p}_h \) are the weight matrices on the width and height, respectively. The \( e^{\mathbf{e}_w} \) and \( e^{\mathbf{e}_h} \) are the rate of width and height of the bounding box. The \( \sigma(t_a) \) is the offset of ship prediction score, and it is obtained by the product of ship prediction probability and IOU. The area \((\text{b})\) is the area of the ship’s bounding box, and area \((\text{g})\) is the area of ground truth. The ratio of the intersection area between area \((\text{b})\) and area \((\text{g})\) represents IOU, as shown in Figure 4.

### 3.2. YOLOv3 Network Structure Optimization

For ships in different scenes, the feature extraction is very important for ship classification. The convolution layer of ship feature extraction network can effectively analyze the features of ships, as shown in Figures 5 and 6. The residual network is added to YOLOv3 network, and the hop connection of residual is used many times. The residual network can solve the problem that the gradient is difficult to descend in the process of feature extraction, accelerate the convergence speed, and reduce the error. The size of the image is increased to \(448 \times 448\). The convolution with the step size of 2 in the network is used to down sample the image by 32, 16, and 8 times. The output of the last layer is lightweight improved. The convolution of the original output prediction layer \(3 \times 3\) and \(1 \times 1\) is pruned, and only the convolution of \(3 \times 3\) is reserved for ship position and category prediction. This can reduce the network operation, avoid overfitting, and make the model have better generalization ability. Finally, the feature maps of three scales are obtained, and the feature information of the ship is shown in the matrix as follows:

\[
f_b = f \left( \sum_{t} S_{b-1}^{t} \ast P_{ta}^{b} + W_{a}^{b} \right),
\]

where \( f_b \) is the correspondence between the \( b \)th network and the \( a \)th output, \( f \) is the activation function of the layer \( b \) network, \( S_{b-1}^{t} \) is the mapping of the \( t \) ship by the \( b-1 \) convolution network layer, \( P_{ta}^{b} \) is the weight matrix between the \( t \)th and the \( a \)th ship feature layer, and parameter \( W_{a}^{b} \) is the bias of the \( a \)th output ship feature at the \( b \)th convolutional network layer.

### 3.3. Data Processing

#### 3.3.1. Guided Filtering

In this paper, a guided filter is constructed to filter the images in the night or fog scene. It can be seen from Figure 7 that the guidance filter can protect the contour features of the ship while maintaining the smooth filtering of the ship. It can effectively solve the problem of blur and uneven gray level produced by night or foggy images and has good performance in denoising.

The ship image is filtered to get the output image through the guide image, and the weighted average value is filtered as follows:

\[
q_i = \sum_j W_{ij}(I)p_j, \\
q_i = a_k I_i + b_k, \quad \forall i \in w_k, \\
\arg\min_{i \in w_k} \sum (q_i - p_i)^2, \\
\arg\min_{i \in w_k} \sum (a_k I_i + b_k - p_i)^2,
\]

**Figure 4: Schematic diagram of ship IOU.**
where \( q_i \) is the output image filtered by pixels at position \( i \), \( W_{ij} \) is the filter core of guide image \( I \), \( p_j \) is the input image of the ship with pixels at position \( j \). And \( w_k \) is the window, \( a_k \) and \( b_k \) are the constant coefficients. When the guide image \( I \) has an edge, the output ship image will maintain the edge unchanged. The unsmoothed area of the input image is regarded as noise, and the noise is reduced to the minimum.

The loss function generated by filtering is shown as the following formula:

\[
E(a_k, b_k) = \sum_{i \in w_k} \left( (a_k I_i + b_k - p_i)^2 + \epsilon a_k^2 \right),
\]

\[
a_k = \frac{(1/|w|) \sum_{i \in w_k} I_i p_i - \mu_k \bar{p}_k}{\sigma_k^2 + \epsilon},
\]

\[
b_k = \bar{p}_k - a_k \mu_k,
\]

\[
q_i = \frac{1}{|w|} \sum_{k: i \in w_k} (a_k I_i + b_k) = \bar{a}_k I_i + \bar{b}_k,
\]

\[
\bar{p}_k = \frac{1}{|w|} \sum_{i \in w_k} p_i.
\]

The \(|w|\) is the number of pixels, \( \mu_k \) represents the average value of the guide image in the window, \( \sigma_k^2 \) represents the variance, and \( \bar{p}_k \) is the average value of the image in the window. For different windows, average all \( q_i \) and finally establish the mapping of pixels from \( I \) to \( q \).

3.3.2. Gray Scale Enhancement. The feature of the ship is weakened by the influence of illumination and environment. In this paper, the gray enhancement is used to improve the
situation. The histogram is the number of pixels in each gray level of the image, which reflects the contrast of the image. Contrast Limited Adaptive Histogram Equalization (CLAHE) is used to cut histogram by setting threshold. The cut part is divided into other histograms, and the contrast of each region is limited. At the same time, interpolation is used to improve the operation speed. The gray histogram is shown in Figure 8. After the processing of gray histogram, the image contrast is enhanced while the noise is suppressed. The gray range of the image becomes more uniform, which is conducive to the extraction of ship features.
Figure 9: The loss curve (a) for our Ship-YOLOv3 network and (b) for YOLOv3 network.

Figure 10: Real-time detection results of (a) YOLOv3 algorithm and (b) Ship-YOLOv3 algorithm.

Figure 11: Real-time detection results of (a) the YOLOv3 algorithm and (b) Ship-YOLOv3 algorithm under normal conditions.
4. Experiment and Discussion

4.1. Experimental Environment and Dataset. Our framework was developed on the Win10 OS with 128G RAM and 3 GHz CPU. The GPU version is NVIDIA GeForce GTX 1080 Ti, which contains 11GB RAM. The framework of deep learning is Tensorflow-GPU (1.9 Version). The ship dataset is obtained by framing the surveillance video on the banks of Datong bridge and Hengmen waterway. The data scene is complex and changeable, and the time is also different. We use labeling software to label images. There are three types of ships: heavy bulk ships, empty bulk ships, passenger ship and management ship, respectively labeled boat, boat2, and boat3. The parameters are shown in Table 1. There are 4915 ship images, 4421 training images, 441 verification images, and 53 test images.

4.2. Training Parameter Selection. During the training process, the weights pretrained on the PASCAL VOC dataset are loaded for migration training. In the first stage, the weight of darknet53 was restored. When the learning rate is set to 0.0001, only the last regression layer is trained until the loss is reduced to a lower level. In the second stage, the weights are restored from the first stage, and then the learning rate is set to 0.00001 to train all network layers. The attenuation factor of learning rate was 0.96, which was decayed every 5 epochs. The total training epoch is 100, the batch is 6, and the optimizer is momentum. After 73700 iterations, the loss of Ship-YOLOv3 is reduced to about 0.09794. The loss of YOLOv3 algorithm is about 0.1120, and the loss curve is shown in Figure 9. The Ship-YOLOv3 algorithm has better convergence effect, improves the convergence speed, and reduces the loss value.

4.3. Experiment and Analysis

4.3.1. Algorithm Evaluation. In order to evaluate the effect of ship detection, this paper uses the precision rate and recall to test the Ship-YOLOv3 algorithm:

\[
P = \frac{N_{tp}}{N_{tp} + N_{fp}}
\]

\[
R = \frac{N_{tp}}{N_{tp} + N_{fn}}
\]

(7)

where P is the precision rate of the ship, the \(N_{tp}\) is the number of ship samples that are detected correctly, \(N_{fp}\) is the number of samples with detection errors, \(R\) is the recall rate of the ship, and the parameter \(N_{fn}\) is the number of samples of the ship that missed detection. In addition, the NMS threshold is adjusted to 0.45. When the intersection ratio between the predicted bounding box and the actual position of the ship is greater than 0.45, the ship is detected correctly.

4.3.2. Contrast Experiment on Foggy Day or Night. The YOLOv3 algorithm and our Ship-YOLOv3 algorithm are used to detect the ship video in real time, and the key frames in the video are extracted respectively, as shown in Figure 10. At frame 106, frame 272, and frame 433, the YOLOv3 algorithm and the Ship-YOLOv3 algorithm have detected the target. For large target ships, the detection effect of Ship-YOLOv3 algorithm is better. The positioning is more accurate and the border is narrow. For small target ships, the Ship-YOLOv3 algorithm has better recognition rate and robustness. In this scene, the ship’s feature extraction is greatly disturbed. In the training process, guided filtering and gray enhancement processing are carried out to improve the anti-interference ability of the improved algorithm. It can effectively detect the ship and overcome the adverse factors of light and fog.

4.3.3. Contrast Experiment under Normal Conditions. It can be seen from Figure 11 that the YOLOv3 algorithm only recognizes large-scale ships in frame 117. However, the inspection of small-scale ships is missing. The Ship-YOLOv3 algorithm can also accurately locate and identify small-scale ships, with remarkable effect. In 167 frames and 232 frames, the detection effect of Ship-YOLOv3 algorithm is better and the prediction frame deviates from the target ship in YOLOv3 algorithm.

Compared with YOLOv3 algorithm, the detection time of Ship-YOLOv3 algorithm in nighttime or foggy scene is reduced by 8.6ms, as shown in Table 2. In the conventional scene, the detection time is reduced by 3.53 ms and 6.06 ms on average. In this paper, the Ship-YOLOv3 algorithm maintains a high recognition rate and improves the detection speed.

4.3.4. Classification and Recognition Experiment. It can be seen from Figure 12 that the Ship-YOLOv3 can extract different types of ship features. It can detect and locate three kinds of ships, and the recognition effect is good. And the recognition effect of small target ship is significant, which effectively reduces the probability of missing detection.

Through comparative analysis of Table 3, the Recall, Precision, and MAP of Ship-YOLOv3 are all higher than the YOLOv3 algorithm, but the total loss is lower, so the performance of our algorithm is more superior.

| Scenario            | YOLOv3  | Ship-YOLOv3 |
|---------------------|---------|-------------|
| Night or fog        | 31.25ms | 22.65ms     |
| Conventional        | 26.12ms | 22.59ms     |
| Average             | 28.68ms | 22.62ms     |

Table 2: Comparison of detection time.
In order to solve the problem of ship detection in complex scenes such as fog or night, a Ship-YOLOv3 algorithm is proposed to detect ship targets. It is mainly improved from two aspects: target box dimension clustering and YOLOv3 network structure optimization.

Firstly, the ship data is processed: the collected ship video is divided into frames to establish the ship dataset and guided filtering and gray enhancement processing are carried out on the night or foggy images to maintain the feature of the ship. Then, the self-made ship dataset is clustered into target box dimension: the size of prior box is improved to predict the position of ships. Lastly, the structure of YOLOv3 network is optimized: increase the input size of the image and reduce the convolution layers in the down sampling process. The multiscale output is used to adapt to different ship types and improve the detection accuracy.

The effectiveness of the improved algorithm in real-time detection and classification of ship dataset is verified by the comparative experiments in different scenarios. After 73700 step iteration, the loss of Ship-YOLOv3 is 0.01406 lower than that of YOLOv3, and the convergence effect is better. Compared with YOLOv3 algorithm, the detection time is reduced by 6.06 ms on average, which improves the detection speed while maintaining high recognition. The algorithm accelerates the convergence speed of the network and improves the precision of ship detection by 12.5% and recall rate by 11.5% on the premise of ensuring real-time performance.

Although the algorithm in this paper has achieved satisfactory results in real-time ship detection and classification, some remaining research work can be carried out to further improve our method performance. Firstly, the self-made ship dataset has less categories, so it is the focus of future research to increase the category of ships for detection. Secondly, we will continue to optimize the network structure of Ship-YOLOv3. The parameters of the network are simplified to adapt to ship classification and detection in different scenarios. Finally, the tracking identification of ships will be increased in the future, and the behavior of ships will be analyzed to ensure the safety of waterway transportation and improve the efficiency of port management and maritime inspection.
Data Availability

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

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (no. 61906170), Project of the Science and Plan for Zhejiang Province (no. LGG18F020001 and LGF19F020008), Natural Science Foundation of Zhejiang Province (no. LY19F020001), NingBo Science and Technology Project (no. 2019C50008), Natural Science Foundation of Zhejiang Province (no. LGG18F020001), NingBo Science and Technology Project (no. Y201534540 and Y201432028).

References

[1] J. Wang, C. Zhu, Y. Zhou, and W. Zhang, “Vessel spatio-temporal knowledge discovery with AIS trajectories using co-clustering,” Journal of Navigation, vol. 70, no. 6, pp. 1383–1400, 2017.
[2] R. J. Bye and A. L. Aalberg, “Maritime navigation accidents and risk indicators: an exploratory statistical analysis using AIS data and accident reports,” Reliability Engineering & System Safety, vol. 176, pp. 174–186, 2018.
[3] H. Li, J. Liu, K. Wu, Z. Yang, R. W. Liu, and N. Xiong, “Spatio-Temporal vessel trajectory clustering based on data mapping and density,” IEEE Access, vol. 6, pp. 58939–58954, 2018.
[4] W. Zhang, F. Goerlandt, J. Montewka, and P. Kujala, “A method for detecting possible near miss ship collisions from AIS data,” Ocean Engineering, vol. 107, pp. 60–69, 2015.
[5] D. Luo, S. Zeng, and J. Chen, “A probabilistic linguistic multiple attribute decision making based on a new correlation coefficient method and its application in hospital assessment,” Mathematics, vol. 8, no. 3, 340 pages, 2020.
[6] S. Li, Z. Zhou, B. Wang, and F. Wu, “A novel inshore ship detection via ship head classification and body boundary determination,” IEEE Geoscience and Remote Sensing Letters, vol. 13, no. 12, pp. 1920–1924, 2016.
[7] Y. Zhang, Q.-Z. Li, and F.-N. Zang, “Ship detection for visual maritime surveillance from non-stationary platforms,” Ocean Engineering, vol. 141, pp. 53–63, 2017.
[8] S. Zeng, D. Luo, C. Zhang, and X. Li, “A correlation-based TOPSIS method for multiple attribute decision making with single-valued neutrosophic information,” International Journal of Information Technology & Decision Making, vol. 19, no. 01, pp. 343–358, 2020.
[9] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” in Proceedings of the International Conference on Neural Information Processing Systems, Curran Associates Inc., Lake Tahoe, NV, USA, pp. 1097–1105, December 2012.
[10] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” 2015, http://arxiv.org/abs/1409.1556.
[11] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778, IEEE Computer Society, Las Vegas, NV, USA, June 2016.
[12] R. Girshick, “Fast R-CNN,” in Proceedings of the IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, December 2015.
[13] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: towards real-time object detection with region proposal networks,” IEEE Transactions on Pattern Analysis & Machine Intelligence, vol. 39, no. 6, pp. 1137–1149, 2015.
[14] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, and S. Reed, “SSD: single shot multibox detector,” in Proceedings of the European Conference on Computer Vision (ECCV), pp. 21–37, October 2016, http://arxiv.org/abs/1512.02325.
[15] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: unified, real-time object detection,” in Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 779–788, IEEE, Las Vegas, NV, USA, June 2016.
[16] J. Redmon and A. Farhadi, “YOLO9000: better, faster, stronger,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 6517–6525, IEEE, Honolulu, HI, USA, July 2017.
[17] J. Redmon and A. Farhadi, “YOLOv3: an incremental improvement,” 2018, http://arxiv.org/abs/1804.02767.
[18] Y. Zhang, J. Shu, L. Hu et al., “A ship target tracking algorithm based on deep learning and multiple features,” in Proceedings of the Twelfth International Conference on Machine Vision, Rome, Italy, November 2020.
[19] S. Zeng, S.-M. Chen, and L.-W. Kuo, “Multiattribute decision making based on novel score function of intuitionistic fuzzy values and modified VIKOR method,” Information Sciences, vol. 488, pp. 76–92, 2019.
[20] Y. Feng, W. Diao, X. Sun, M. Yan, and X. Gao, “Towards automated ship detection and category recognition from high-resolution aerial images,” Remote Sensing, vol. 11, no. 16, p. 1901, 2019.
[21] Y. Li, M. Cheng, X. Peng et al., “Ship detection and recognition combing one-dimensional range profile with SAR image,” The Journal of Engineering, vol. 2019, no. 19, 2019.
[22] C. Zhang, C. Chen, D. Streimikiene, and T. Balezentis, “Intuitionistic fuzzy Multimoora approach for multi-criteria assessment of the energy storage technologies,” Applied Soft Computing, vol. 79, pp. 410–423, 2019.
[23] X. Chen, L. Qi, Y. Yang et al., “Video-based detection infrastructure enhancement for automated ship recognition and behavior analysis,” Journal of Advanced Transportation, vol. 2020, Article ID 7194342, 12 pages, 2020.
[24] Z. S. HuangB. Wen et al., “An intelligent ship image/video detection and classification method with improved regressive deep convolutional neural network,” Complexity, vol. 2020, Article ID 1520872, 11 pages, 2020.
[25] O. Bachem, M. Lucic, H. Hassani et al., “Fast and provably good seedings for k-Means,” in Proceedings of the 30th Conference on Advances in Neural Information Processing Systems, pp. 55–63, Barcelona, Spain, 2016.