Mechatronics and Intelligent Transportation Systems
https://www.acadlore.com/journals/MITS

Ship Detection Based on an Enhanced YOLOv5 Algorithm

Xin Liu[1,2*], Qingfa Zhang[1,2], Yubo Tu[1,2], Mingzhi Shao[1,2], Tengwen Zhang[1,2], Yuhan Sun[1,2], Haiwen Yuan[3]

[1] Shandong Jiaotong University, 264210 Weihai, China
[2] Weihai Institute of Marine Information Science and Technology, 264200 Weihai, China
[3] Wuhan Institute of Technology, 430205 Wuhan, China

[*] Correspondence: Xin Liu (axinzaixian@163.com)

Received: 09-10-2024 **Revised:** 10-20-2024 **Accepted:** 10-27-2024

Citation: X. Liu, Q. F. Zhang, Y. B. Tu, M. Z. Shao, T. W. Zhang, Y. H. Sun, and H. W. Yuan, "Ship detection based on an enhanced YOLOv5 algorithm," *Mechatron. Intell Transp. Syst.*, vol. 3, no. 4, pp. 223–234, 2024. https://doi.org/10.56578/mits030403.

Abstract: Advanced ship detection technologies play a critical role in improving maritime safety by enabling the rapid identification of vessels and other maritime targets, thereby mitigating the risk of collisions and optimizing traffic efficiency. Traditional detection methods often demonstrate high sensitivity to minor variations in target appearance but face significant limitations in generalization, making them ill-suited to the complex and dynamic nature of maritime environments. To address these challenges, an enhanced ship detection method, referred to as YOLOv5-SE, has been proposed, which builds upon the YOLOv5 framework. This approach incorporates attention mechanisms within the backbone network to improve the model's focus on key features of small targets, dynamically adjusting the importance of each channel to boost representational capacity and detection accuracy. In addition, a refined version of the Complete Intersection over Union (CIoU) loss function has been introduced to optimize the loss associated with target bounding box prediction, thereby improving localization accuracy and ensuring more precise alignment between predicted and ground-truth boxes. Furthermore, the conventional coupled detection head in YOLOv5 is replaced by a Decoupled Head, facilitating better adaptability to various target shapes and accelerating model convergence. Experimental results demonstrate that these modifications significantly enhance ship detection performance, with mean Average Precision (mAP) at IoU 0.5 reaching 94.9% and 95.1%, representing improvements of 3.1% and 1.2% over the baseline YOLOv5 model, respectively. These advancements underscore the efficacy of the proposed methodology in improving detection accuracy and robustness in challenging maritime settings.

Keywords: Ship detection; YOLO; Attention mechanism; Efficient decoupled head; Alpha-Complete Intersection over Union (CIoU)

1 Introduction

The land, which covers 29% of the Earth's total surface area, contrasts with the vast oceans, which occupy 71%, and is aptly described as "three parts land and seven parts ocean". As human dependence on marine resources intensifies, and maritime traffic becomes increasingly busy, and the significance of safe vessel operation and the prevention of marine accidents becomes increasingly evident. Over the years, deep learning technology has reached a mature stage and can be widely applied in various fields. Many distinguished scholars have proposed various research methods for ship target detection, for instance, Dehghani-Dehcheshmeh et al. [1], who introduced an improved Transformer module to mitigate the interference of image noise and light pollution on detection, therefore, to make improvements on the model's ability, and to get deep feature information from images. Kong et al. [2] proposed a lightweight ship detection network based on the YOLOx-Tiny model, which significantly improves the accuracy and speed of ship detection in SAR images through multi-scale feature extraction and an adaptive threshold strategy. Fan et al. [3] proposed the CSDP-YOLO algorithm to address the challenges of class imbalance and complex background interference in satellite remote sensing image data for small-target ship detection, employing an innovative CSDP module and MPDIoU loss function to significantly enhance detection accuracy and robustness. Li et al. [4] proposed a new SDVI (ship detection from visual image) algorithm, the enhanced YOLOv3-tiny network,

https://doi.org/10.56578/mits030403

which outperforms the original YOLOv3-tiny in detection accuracy and real-time performance, achieving precise classification and positioning of six types of ships in practical scenarios.

Target detection technology has experienced rapid development over the past few years. Initially, it primarily depended on hand-crafted features and traditional machine learning techniques, such as Haar features [5] and Histogram of Oriented Gradients (HOG) [6]. However, these methods were constrained by the limitations in feature representation and the management of intricate scenarios. As time went on, the introduction of two-stage detection ideas by Fast R-CNN [7] and Faster R-CNN [8] separated the object detection task into two stages: generating candidate regions and classifying these regions, which significantly improved detection accuracy. The emergence of single-stage detectors like SSD (Single Shot Multibox Detector) [9] and YOLO [10] changed the traditional two-stage detection paradigm. These methods complete both object detection and localization simultaneously in a single forward pass.

Traditional ship detection algorithms exhibit significant limitations in handling small targets, adapting to environmental changes, and resisting interference:

(1) Small target detection challenges: For smaller ships, traditional algorithms struggle to capture sufficient feature information, leading to higher false negative and false positive rates.

(2) Poor environmental adaptability: Under harsh lighting conditions (e.g., low light or strong reflection) and in changing weather environments, traditional algorithms' performance significantly deteriorates, affecting detection reliability.

(3) Weak interference resistance: Non-target objects (such as waves, birds, etc.) are easily misidentified as ships, increasing the false alarm rate and reducing the system's reliability.

To address these issues, this paper proposes an improved YOLOv5-based algorithm, YOLOv5-SE, aiming to enhance model performance and overcome the limitations of existing technologies:

(1) By introducing the Squeeze-and-Excitation (SE) module into the backbone network, YOLOv5-SE can adaptively adjust the weights of each channel, thereby enhancing the focus on critical features of small targets. This improvement significantly increases the model's capability to represent and detect targets of different sizes and shapes.

(2) By optimizing the CIoU loss function, YOLOv5-SE proposes a new boundary box regression loss calculation method. This method not only improves localization accuracy but also ensures a closer match between the detection boxes and the actual targets, effectively reducing false alarms.

(3) Unlike the native coupled detection head (Coupled Head) in YOLOv5, YOLOv5-SE adopts a Decoupled Head structure. This design allows the model to adapt more flexibly to variations in target shapes and sizes, accelerates convergence during training, and enhances the model's robustness and generalization ability.

2 YOLOv5

The YOLO (You Only Look Once) algorithm has undergone multiple iterations and upgrades, evolving from its initial version, YOLOv1, to the latest iteration, YOLOv8. Each version has achieved significant advancements in innovation and performance [10–15]. The selection of YOLOv5 as the base model is primarily driven by its ability to improve deployment efficiency on small- to medium-sized edge computing platforms. Although YOLOv8 demonstrates superior detection accuracy, it requires more substantial computational resources, which is not feasible for devices with limited processing power. In comparison, YOLOv5 achieves a more harmonious balance between performance and resource consumption, fully addressing the practicality and feasibility issues in computing-constrained environments. Therefore, from the standpoint of practical application and resource optimization, YOLOv5 is more suitable to meet the needs of the research. The algorithm is available in several variants: YOLOv5s (small), YOLOv5m (medium), YOLOv5l (large), and YOLOv5x (extra-large).

The network architecture of YOLOv5 is illustrated in Figure 1, including four principal components: the Input stage, the Backbone stage, the Neck stage, and the Detection stage. The Input stage is used for processing image data, utilizing Mosaic data augmentation (refer to Figure 2), adaptive anchor boxes, and adaptive image scaling techniques. The Backbone stage is responsible for feature extraction, where the C3 module combines three standard convolutional layers with multiple Bottleneck modules, reducing the parameter count and enhancing computational efficiency without compromising the original detection accuracy. The SPPF module mainly focuses on the integration of multi-scale features, enriching the semantic information of the feature map by synthesizing features across various scales. The Neck stage consists of two network structure+s, FPN (Feature Pyramid Network) [16] and PAN (Path Aggregation Network) [17], which together form an FPN+PAN structure, enabling a more comprehensive capture of multi-scale information in target detection tasks. The Detection stage is used for the final target detection, processing the feature information from the Backbone and Neck stages, and through feature fusion, making the adjustment on the position and size of each anchor box, and calculating the object's confidence and class probability. Finally, Non-Maximum Suppression (NMS) is used to remove the model of overlapping predictions and retain the most accurate detection outcomes.

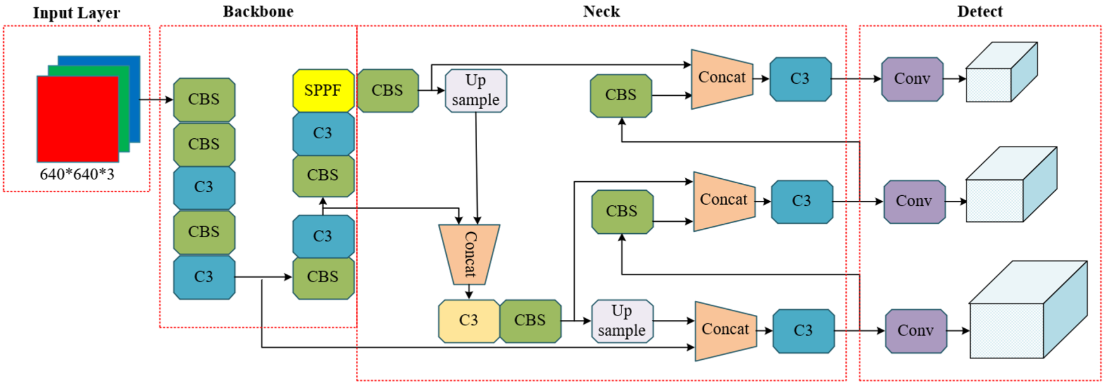

Figure 1. YOLOv5 network architecture

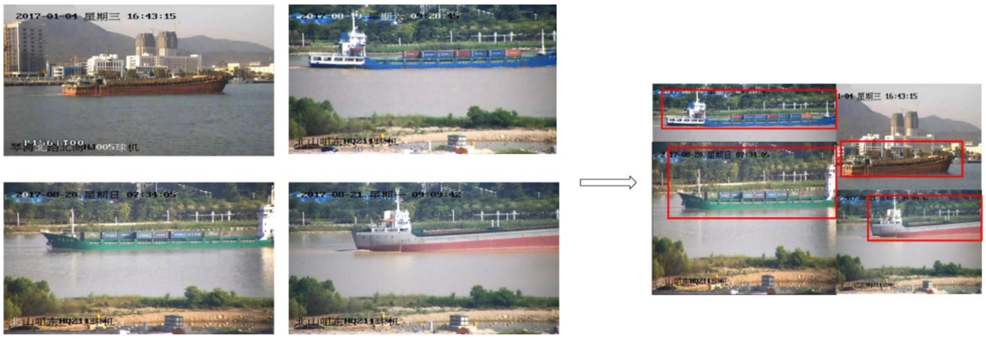

Figure 2. Mosaic data augmentation

3 Improvements in YOLOv5 Algorithm

3.1 SE

In traditional ship detection methods, especially when handling small targets, these methods are often very sensitive to changes in the target objects and have weak generalization capabilities. This limitation is particularly prominent when facing complex maritime environments. Therefore, adopting an effective feature recalibration method is crucial for improving detection performance. To address this issue, this paper introduces the channel attention mechanism —SE [18], which adaptively adjusts the importance of different feature channels in Convolutional Neural Networks (CNNs), thereby improving feature extraction efficiency and detection accuracy. The SE algorithm process is shown in Figure 3.

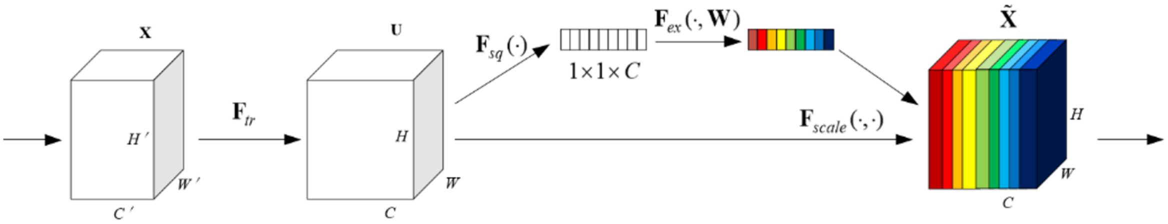

Figure 3. SE algorithm flowchart

The core idea of the SE mechanism is that not all feature channels in a CNN have the same level of importance for the current detection task. Some channels may contain more target-related information, while others may contain less useful information. Therefore, the SE mechanism allocates different weights to each feature channel, enhancing the expression capacity of important channels and suppressing the influence of irrelevant channels. Specifically, a feature map is composed of multiple channels, each responsible for extracting different features from the input data. The SE mechanism adjusts these channels through two steps: Squeeze and Excitation. The Squeeze operation aggregates the spatial information of each channel through Global Average Pooling (GAP) to obtain a global description of each channel; the Excitation operation generates a weight for each channel through a fully connected layer, dynamically recalibrating the original feature map.

The input feature map X from the previous layer is preprocessed before entering the SE layer, where a Transformation operation is performed on it, which simply means obtaining a feature map U through a convolution. The Squeeze operation (F_{Sq}) is the first step of SENet, and the dimensions of output feature map from the convolution layer after preprocessing can be defined as $[H, W, C]$, H and W are the height and width of the feature map separately, and C is the number of channels of the feature map. For each channel $(c = 1, 2, \dots, C)$ of the input feature map, the spatial information is aggregated into a single real value by the GAP. Thus, for C channels, a vector of length C is ultimately obtained $[z_1, z_2, \dots, z_c]$. This vector captures the global receptive field of each channel. The calculation formula is presented in Eq. (1), z_c is the representative value of the channel after GAP, and $X_{i,j,c}$ is the element located at the (i, j) position within the C-th channel.

$$z_c = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} X_{i,j,c} \tag{1}$$

The next operation is Excitation F_{ex}, which effectively utilizes the output from the Squeeze operation through a fully connected layer to generate channel-specific weights. To use these weights to adjust the channel responses of the original feature map, thereby highlighting important features. The input is the output of the Squeeze operation, a vector of length C can be $[Z_1, Z_2, \dots, Z_C]$, and the Excitation operation is mainly accomplished through two fully connected layers with an intermediate hyperparameter (dimensionality reduction ratio) between them. The 1st fully connected layer reduces the dimension of the input vector from $1 \times 1 \times C$ to $1 \times 1 \times C/r$ by using a weight matrix W_1, and then passes through a nonlinear activation function $ReLU$ to the next layer; The 2nd fully connected layer restores the dimension of the activated output from $1 \times 1 \times C/r$ to $1 \times 1 \times C$ through another weight matrix W_2; and finally through an activation function Sigmoid, produces a weight vector ranging from 0 to 1. Each element of this vector corresponds to the importance weight of a channel in the original feature map. The specific calculation formula for the Excitation operation is shown in Eq. (2).

$$s = \sigma \left(W_2 \cdot \operatorname{Re} LU \left(W_1 \cdot z + b_1 \right) + b_2 \right) \tag{2}$$

The weight matrices of W_1 and W_2 are of dimensions $C \times C/r$ and $C/r \times C$, respectively; The two fully connected layers of b_1 and b_2 are of dimensions C/r and C, respectively; After the 1st fully connected layer is activation function ReLU; and then after the 2nd fully connected layer is the activation function σ; S represents the channel weight vector.

The final step of the SENet module is to perform a multiplication operation F_{scale}, combining the generated feature vector s (with dimension $1 \times 1 \times C$) with the original feature map U (of dimension $H \times W \times C$) as shown in Eq. (3). Each of the $H \times W$ pixel values in every channel of the original feature map U is used to multiply by the corresponding channel's weight value in vector s, these will be resulting in the last output of the SE module.

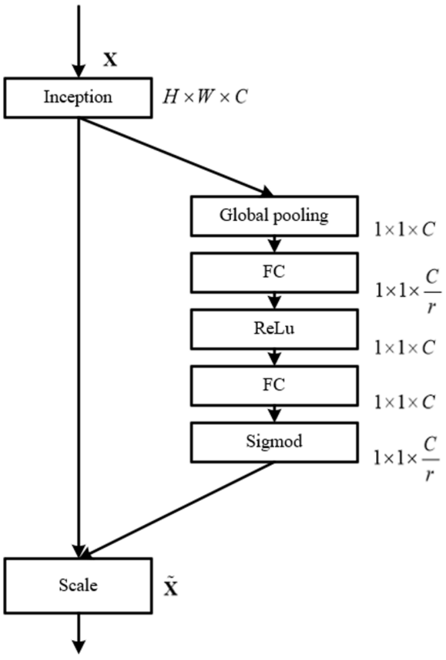

Figure 4. SE algorithm flowchart

$$\tilde{x}_c = F_{\text{scale}}(u_c, s_c) = s_c u_c \tag{3}$$

where, v_c represents the c-th channel of the original feature map U; s_c is the weights of the c-th channel; y_c is for the c-th channel of the feature map s_c after applying the weights is $\tilde{x}_c$. So for the complete network structure of SENet has been created and shown in Figure 4.

This paper integrates the SE channel attention mechanism into the backbone network of YOLOv5-SE. In this integration process, the SE module performs additional processing on the feature map before the final feature extraction and representation. Especially in the task of detecting small ship targets, the SE mechanism dynamically adjusts the weights of the channels, effectively enhancing the feature map's ability to represent key targets, thus significantly improving the network's detection accuracy. In addition, the SE mechanism accelerates the network's convergence speed and optimizes the training process by enhancing the effectiveness of the features. The modified YOLOv5 network structure is shown in Figure 5.

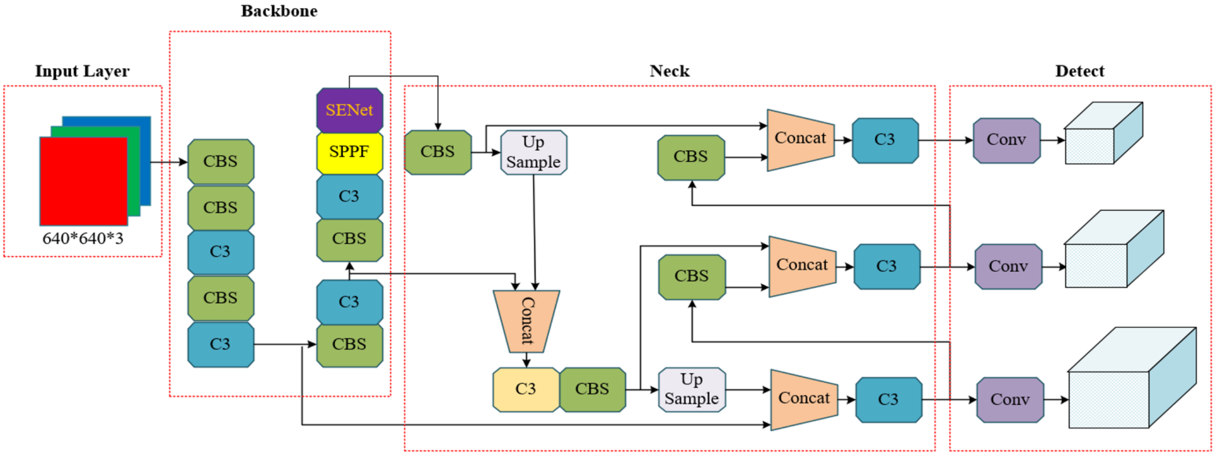

Figure 5. Adding the SE module to the YOLOv5 network architecture

3.2 Alpha-CIoU

The development of loss functions in the field of object detection has evolved from the beginning, gradually evolving into more refined variants such as GIoU [19], DIoU [20], and CIoU. The loss function used in YOLOv5 is CIoU, and its calculation formula is shown in Eq. (4) and Eq. (5).

$$v = \frac{4}{\pi^2}\left(\arctan\frac{w^{gt}}{h^{gt}} - \arctan\frac{w}{h}\right)^2 \tag{4}$$

$$L_{CIoU} = 1 - IoU + \frac{\rho^2\left(b, b^{gt}\right)}{c^2} + \beta v \tag{5}$$

The ratio of intersection IoU is to union between the predicted and true box, calculated as $\frac{|B \cap B^{gt}|}{|B \cup B^{gt}|}$. Among them, B is the predicted box; B^{gt} represents the true box; $\rho\left(b, b^{2t}\right)$ is the Euclidean distance between the centers of the predicted box $_b$ and the true box b^{gt}; The diagonal length of the smallest closed region c that covers both boxes; A weight parameter β is used to balance the aspect ratio term, The measurement of the consistency of the aspect ratio between the predicted and true box is v; The width and height of the predicted and true box are w, h and w^{gt}, h^{gt} respectively.

Although the CIoU loss function has demonstrated superior performance in various scenarios, it does not provide parameters to flexibly adjust the focus on these different factors in its design. This lack of flexibility may lead to suboptimal performance when the model is faced with data having specific attributes, such as high aspect ratio changes or extreme positional deviation. Therefore, this paper proposes an improved loss function CIoU, is α-CIoU [21]. α-CIoU is based on the traditional CIoU, but introduces an adjustable parameter α, which allows the model to perform bounding box regression more flexibly according to the characteristics of a specific dataset or the requirements of the detection task. By appropriately selecting the value of α, the loss function can be customized to prioritize aspect ratio accuracy or center point localization, which helps achieve better performance. Experiments have shown that in most cases, taking results α=3 in better experimental effects. The calculation formula for the loss function α-CIoU is shown in Eq. (6).

$$L_{\alpha-CIoU} = 1 - IoU^{\alpha} + \frac{\rho^{2\alpha}\left(b, b^{gt}\right)}{c^{2\alpha}} + (\beta v)^{\alpha} \tag{6}$$

From YOLOv3 to v5, these models have generally adopted the design of a Coupled Head in the detection head part. Subsequently, to meet the needs of industrial applications, YOLOX [22] was introduced, proposing an innovative Decoupled Head design. This design separates the tasks of class classification and bounding box regression into two independent branches to enhance detection efficiency and accuracy. This decoupling approach allows the model to exhibit higher flexibility and accuracy when dealing with complex industrial scenarios. YOLOv6, developed by the Meituan technical team, further optimizes this design. While reconstructing the Backbone and Neck of YOLOv5, it adopts and improves upon the Decoupled Head of YOLOX, including the removal of a 3×3 convolutional normalization activation function layer (hereinafter referred to as the convolutional layer), which maintains accuracy while reducing latency and alleviates the additional latency overhead caused by convolutions in the decoupling head. Figure 6 shows the comparison of the two Head structures.

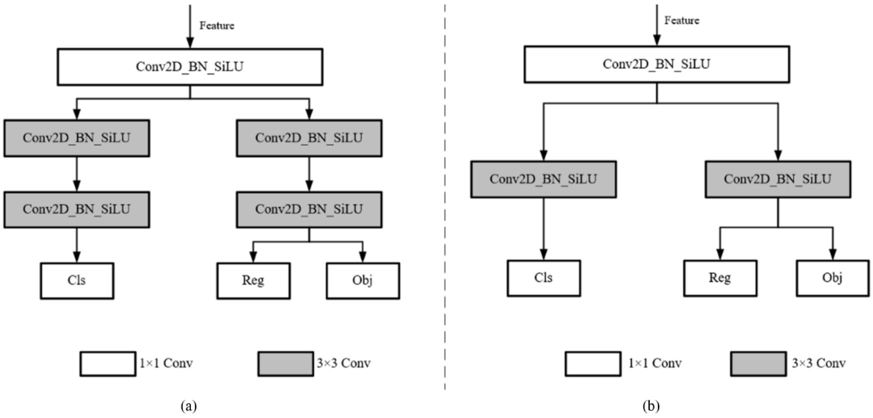

Figure 6. Head comparison of YOLOX and YOLOv6

This paper replaces the Coupled Head of the improved YOLOv6 with a Decoupled Head and integrates it into the Detect end of YOLOv5-SE. The feature maps are further enhanced by the Neck end, and through multi-scale feature fusion, three effective feature layers with enhanced features are obtained, which are represented in the code as: p4[-1,3,C3,[256,False]], p5[-1,3,C3,[512,False]], p6[1,3,C3,[1024,False]]. These three feature layers which are passed into the detection head in order to achieve the final prediction results. The specific operations are as follows: First, a 1×1 convolutional layer is used to integrate the channels of the feature layers that we input, as shown in subgraph (b) of Figure 6, where the left part is category classification, using a 3×3 convolutional layer for feature extraction, and then using a 1×1 convolutional layer to classify this feature point; and the right part is bounding box regression and determining the probability of the target's existence, the specific process is basically consistent with the left part. The height (H), width (W), and number of channels (C) obtained after the three feature layers pass through the Decoupled Head are shown in Table 1, where the value is taken as 6 in this paper. Finally, the results of category classification, bounding box regression, and target existence probability are stacked to produce the final prediction result.

Table 1. Key parameters of our model

		H	W	C
	Cls	80	80	num_classes
p4 layer	Reg	80	80	4
	Obj	80	80	1
	Cls	40	40	num_classes
p5 layer	Reg	40	40	4
	Obj	40	40	1
	Cls	20	20	num_classes
p6 layer	Reg	20	20	4
	Obj	20	20	1

4 Experimental Analysis

4.1 Evaluation Criteria

This paper primarily uses Precision, Recall, mAP@0.5, and mAP@0.5:0.95 as the main evaluation metrics to measure the performance of the improved YOLOv5 model. The precision focus model is used to identify the proportion of targets, while Recall is used to evaluate the model's ability to capture all relevant targets. mAP@0.5 measures the model's average precision at an IoU threshold of 0.5, and mAP@0.5:0.95 refers to the average performance of the model within the range of IoU from 0.5 to 0.95, at increments of 0.05 [23, 24]. Here are the calculation formulas:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{7}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{8}$$

$$mAP = \frac{1}{N}\sum_{i=1}^{N} AP_i \tag{9}$$

where, TP refers to the model correctly identifying positive class instances as positive; FP represents the model incorrectly identifying negative class examples as positive; FN indicates the model wrongly identifying positive class instances as negative; AP refers to the numerical representation of the area under the precisionrecall curve; N is the number of categories.

4.2 Experimental Environment and Dataset

This paper experimentally verified the algorithm, setting up a computing environment based on a Linux server, with technical specifications as follows: RTX2080 SUPER 8G graphics card, Python3.9 programming environment, Pytorch2.0 deep learning framework, and CUDA version 11.7.

This study used the open-source ship detection dataset Seaships, proposed by Shao et al. [25], for experimental validation. The dataset contains 7,000 images, covering six categories of ships: bulk carriers, general cargo ships, freighters, container ships, fishing vessels, and passenger ships. To ensure the model's generalization ability, we split the dataset into training (5,600 images), validation (700 images), and test (700 images) sets in an 8:1:1 ratio. Additionally, by analyzing the position and size distribution of different class labels in the images, we confirmed the uniformity of the annotations and the diversity of target sizes in the dataset. Figure 7 shows the position distribution and size information of objects in the Seaships dataset. In subgraph (a) of Figure 7, most targets are concentrated in the central region of the image; The subgraph (b) of Figure 7 indicates that there are more small and medium-sized targets in the dataset, while large targets are relatively sparse.

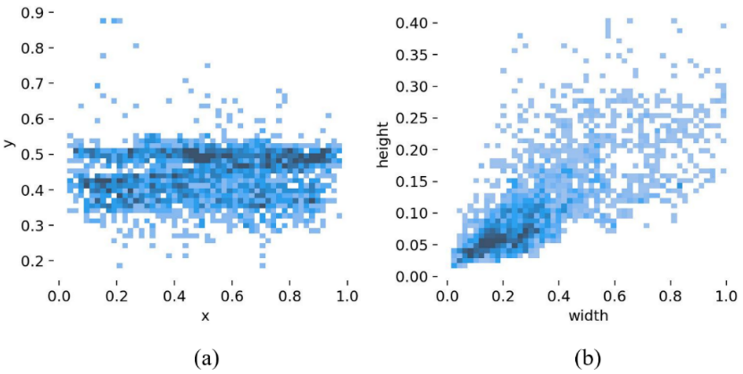

Figure 7. Distribution of the dataset under different labels

All input images were resized to a resolution of 640×640 to ensure consistency and reduce computational complexity; The batch size was set to 16, balancing memory usage and gradient estimation stability; The initial learning rate was set to 0.01. The optimizer used was stochastic gradient descent (SGD) with momentum (momentum = 0.9) and weight decay (weight decay = 0.0005); The total number of epochs was set to 100. During the training process, we closely monitored the trends of training loss, validation loss, and various evaluation metrics to ensure that the model did not overfit or underfit. At the same time, an early stopping mechanism was employed to prevent overtraining; if the performance on the validation set did not improve after several consecutive epochs, the training process was automatically terminated.

4.3 Ablation Experiment

Introduce an important research method in the field of computer vision is the ablation experiment, which is used to understand and evaluate the contribution of each component of a model or algorithm to the final performance [26]. In simple terms, by sequentially removing (or ablating) a particular component or feature from the model or algorithm and then observing the impact of this removal on the model's performance, one can understand the importance and role of that component or feature [11].

To verify whether the proposed three improvements are effective for the original model while keeping the dataset, training process and parameters, and hardware environment consistent, this paper conducts ablation experiments using the lightweight model YOLOv5s as the baseline. The experimental results are indicated in Table 2.

Table 2. Key parameters of our model

Model	SENet	α-CIoU	Decoupled Head	Precision / %	Recall / %	mAP@0.5 / %	mAP@0.5:0.95 / %
YOLOv5s	×	×	×	91.8	91.0	93.9	66.3
Improved model 1	√	×	×	92.1	90.9	94.3	65.9
Improved model 2	×	√	×	95.3	88.6	94.4	68.4
Improved model 1	×	×	√	93.1	93.1	95.0	69.2
YOLOv5-SE	√	√	√	94.9	92.7	95.1	68.4

Use the improved model 1 to incorporate the SENet channel attention mechanism, this would allow the model to focus more on channels with rich information in the image, thereby enhancing the ability to capture key features. The addition of this attention mechanism has increased the precision by 0.3%, indicating better recognition of real targets, especially small objects, with a 0.4 points increase in mAP@0.5, indicating an improvement in the precision of target localization. This proves the efficiency of the attention mechanism in enhancing the accuracy of object detection; the improved model 2 uses a loss function α-CIoU, by adjusting the weight parameters, the model has made more detailed optimizations in the shape and size of bounding boxes, with accuracy improved by 3.5%, reducing the occurrence of misjudgments., and mAP@0.5 and mAP@0.5:0.95 increasing by 0.5% and 2.1% separately, the figures show the detection performance of the model has been comprehensively improved under different IoU thresholds; the improved model 3, after replacing with a decoupled detection head, that has an accuracy increase of 1.3%, with the model's performance under different IoU thresholds improved, mAP@0.5 and mAP@0.5:0.95 increasing by 1.1% and 2.9% respectively. It can be seen that the decoupled structure more effectively balances classification and localization tasks, enhancing the model's comprehensive recognition ability for targets. Integrating the SENet attention mechanism, the loss function, and the Decoupled detection head into the YOLOv5 model has obtained significant performance improvements: the Precision increased by 3.1%, the Recall increased by 1.7%, and mAP@0.5 and mAP@0.5:0.95 increased by 1.2% and 2.1% respectively. To conclude, these improvements significantly enhance the model's detection efficiency and accuracy especially in complex scenarios.

Although the improvements proposed in this experiment significantly enhanced the model's performance across most metrics, some errors and limitations still remain. First, due to the bias in the target sizes and distributions in the Seaships dataset, especially the larger number of small-sized targets, the model may perform relatively poorly when handling large-sized targets. Secondly, although we achieved good mAP scores across multiple IoU thresholds, the model's performance may fluctuate under certain boundary conditions (e.g., when targets are very close or heavily overlapped), necessitating further optimization of the model architecture and training strategy to improve its robustness.

4.4 Comparative Experiment

To further ascertain the strength of the improved model compare to other models, this paper conducted comparative experiments, comparing the improved YOLOv5 with several other popular object detection models, including Faster R-CNN, SSD, YOLOv5s, YOLOv8s and the earlier version of YOLO, YOLOv3. The detection results on different algorithms are illustrated in Table 3 (units: %).

Refer to the data in Table 3, it shows that the YOLOv5-SE model is not much different in the precision compared to these five algorithms, but it has improved Recall rates and mAP@0.5 by 8%, 5.9%, 9.1%, 1.7% and 1.7%, 2.8%, 1.9%, 1.2% respectively. As can be seen from the comparison charts of different algorithms in Figures 8-12, the revised YOLOv5 model performs very well in detecting small and overlapping objects, significantly outperforming Faster R-CNN, SSD, YOLOv3, YOLOv8s and YOLOv5s.

Table 3. Key parameters of our model

Model	Precision / %	Recall / %	mAP@0.5 / %
Faster R-CNN	94.2	84.7	93.4
SSD	95.3	86.8	92.3
YOLOv3	93.2	83.6	93.2
YOLOv5s	91.8	91.0	93.9
YOLOv8s	94.8	92.4	94.9
YOLOv5-SE	94.9	92.7	95.1

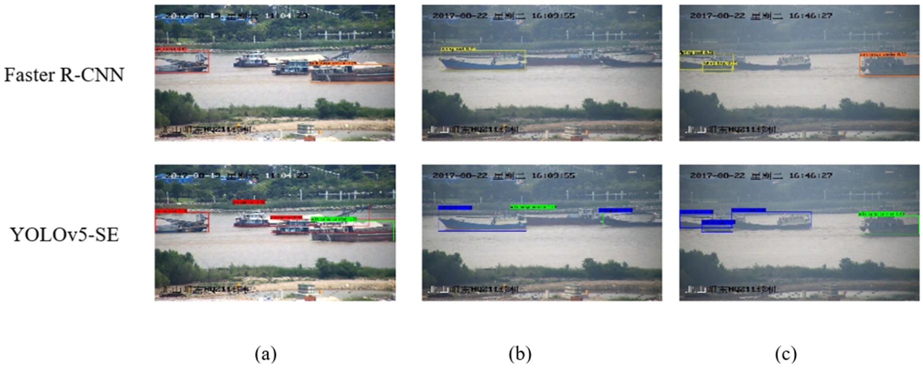

Figure 8. Comparison between YOLOv5-SE and faster R-CNN

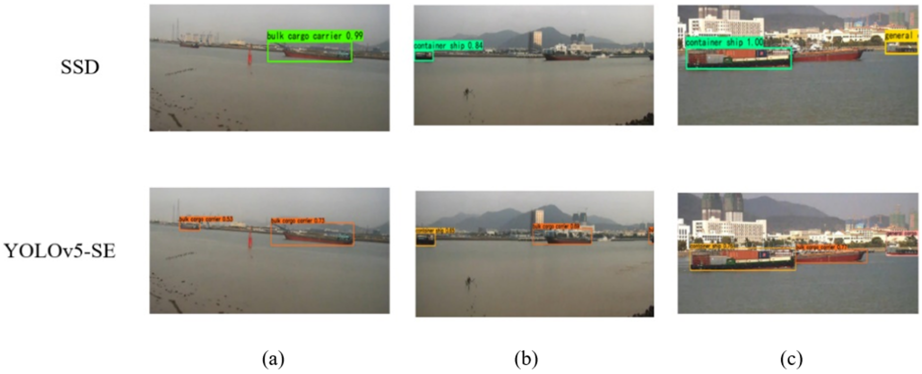

Figure 9. Comparison between YOLOv5-SE and SSD

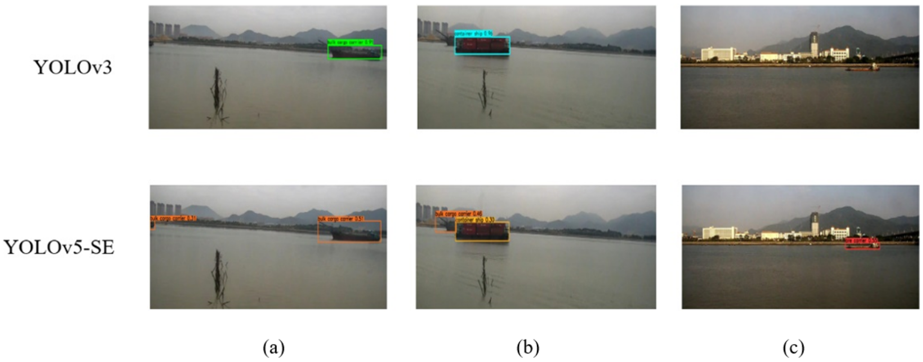

Figure 10. Comparison between YOLOv5-SE and YOLOv3

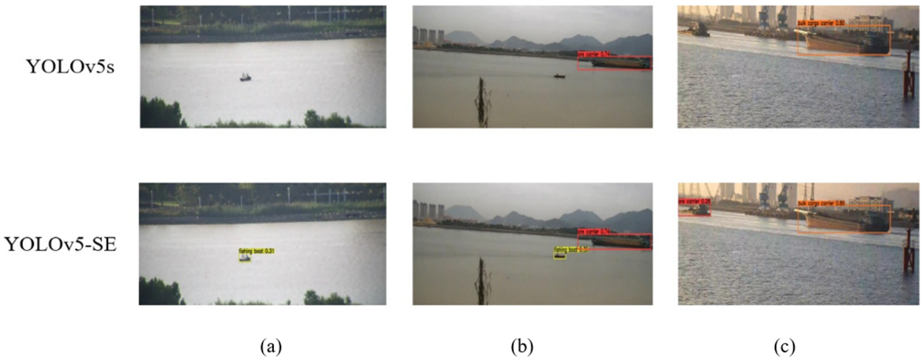

Figure 11. Comparison between YOLOv5-SE and YOLOv5s

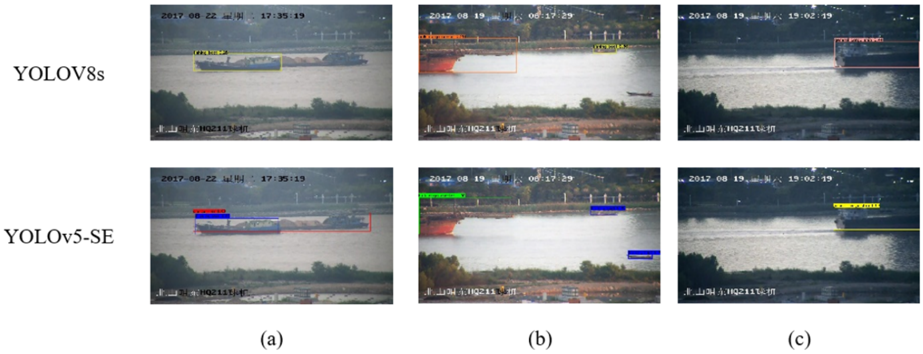

Figure 12. Comparison between YOLOv5-SE and YOLOv8s

5 Conclusion

This paper significantly improves the performance and efficiency of ship detection by making a series of modifications to the YOLOv5 model, particularly in complex marine environments. The variability of the marine environment, such as turbulent seas, unpredictable lighting, harsh weather conditions, and overlapping ships, often presents challenges for object detection tasks. In such environments, traditional detection algorithms often struggle to maintain high accuracy and robustness. By integrating the SENet attention mechanism, improving the loss function, and incorporating a decoupled detection head, the modified model demonstrates stronger feature extraction capabilities and target localization accuracy, thereby improving its performance in complex marine environments.

To address the challenges in complex marine environments, the proposed improved algorithm enhances the model's robustness and adaptability. The SENet attention mechanism allows the model to adaptively focus on important feature channels in dynamic backgrounds, improving sensitivity to ship targets. The improved loss function refines the optimization of bounding boxes, enhancing the model's target localization accuracy under harsh conditions, especially when targets are partially occluded or in poorly lit environments. The introduction of the decoupled detection head effectively balances the classification and localization tasks, enhancing the model's ability to handle small and overlapping targets. These innovations enable the model to demonstrate higher accuracy and stability in complex scenarios such as maritime surveillance, channel management, and port operations.

However, despite the improved model's strong adaptability in many complex scenarios, certain limitations still exist. First, dynamic changes in the marine environment, such as wave reflections and background differences across various sea regions, may affect detection accuracy. Particularly under extreme weather conditions (e.g., heavy fog, rain) and in high-density ship clustering areas, the model may encounter errors and difficulties in recognition. To further enhance robustness, future work could involve increasing the diversity of the dataset by adding more varied marine environmental data and introducing scene-based adaptive training methods to improve the model's adaptability to various complex scenarios.

Overall, the proposed improved algorithm not only enhances the performance of ship target detection but also strengthens the model's robustness and adaptability in complex marine environments. Nonetheless, further optimization of the model is required in extreme and special environments to ensure stable performance across different marine scenarios.

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.

References

[1] S. Dehghani-Dehcheshmeh, M. Akhoondzadeh, and S. Homayouni, "Oil spills detection from SAR earth observations based on a hybrid CNN transformer networks," *Mar. Pollut. Bull.*, vol. 190, p. 114834, 2023. https://doi.org/10.1016/j.marpolbul.2023.114834

[2] W. Kong, S. Liu, M. Xu, M. Yasir, D. Wang, and W. Liu, "Lightweight algorithm for multi-scale ship detection based on high-resolution SAR images," *Int. J. Remote Sens.*, vol. 44, no. 4, pp. 1390–1415, 2023. https://doi.org/10.1080/01431161.2023.2182652

[3] X. Fan, Z. Hu, Y. Zhao, J. Chen, T. Wei, and Z. Huang, "A small ship object detection method for satellite remote sensing data," *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, vol. 17, pp. 11 886–11 898, 2024. https://doi.org/10.1109/JSTARS.2024.3419786

[4] H. Li, L. Deng, C. Yang, J. Liu, and Z. Gu, "Enhanced YOLO v3 tiny network for real-time ship detection from visual image," *IEEE Access*, vol. 9, pp. 16 692–16 706, 2021. https://doi.org/10.1109/ACCESS.2021.3053956

[5] E. Quiles-Cucarella, J. Cano-Bernet, L. Santos-Fernández, C. Roldán-Blay, and C. Roldán-Porta, "Multi-index driver drowsiness detection method based on driver's facial recognition using haar features and histograms of oriented gradients," *Sensors*, vol. 24, no. 17, p. 5683, 2024. https://doi.org/10.3390/s24175683

[6] B. Bhattarai, R. Subedi, R. Gaire, E. Vazquez, and D. Stoyanov, "Histogram of oriented gradients meet deep learning: A novel multi-task deep network for 2D surgical image semantic segmentation," *Med. Image Anal.*, vol. 85, p. 102747, 2023. https://doi.org/10.1016/j.media.2023.102747

[7] N. Rane, "YOLO and faster R-CNN object detection for smart Industry 4.0 and Industry 5.0: Applications, challenges, and opportunities," *SSRN*, no. 4624206, 2023. https://doi.org/10.2139/ssrn.4624206

[8] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, 2016. https://doi.org/10.1109/TPAMI.2016.2577031

[9] W. Zhu, H. Zhang, J. Eastwood, X. Qi, J. Jia, and Y. Cao, "Concrete crack detection using lightweight attention feature fusion single shot multibox detector," *Knowledge-Based Syst.*, vol. 261, p. 110216, 2023. https://doi.org/10.1016/j.knosys.2022.110216

[10] T. Diwan, G. Anirudh, and J. V. Tembhurne, "Object detection using YOLO: Challenges, architectural successors, datasets and applications," *Multimed. Tools Appl.*, vol. 82, no. 6, pp. 9243–9275, 2023. https://doi.org/10.1007/s11042-022-13644-y

[11] M. Hussain, "YOLO-v1 to YOLO-v8, the rise of YOLO and its complementary nature toward digital manufacturing and industrial defect detection," *Machines*, vol. 11, no. 7, p. 677, 2023. https://doi.org/10.3390/machines11070677

[12] J. Redmon, "YOLOv3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018. https://doi.org/10.48550/arXiv.1804.02767

[13] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal speed and accuracy of object detection," *arXiv preprint arXiv:2004.10934*, 2020. https://doi.org/10.48550/arXiv.2004.10934

[14] M. G. Ragab, S. J. Abdulkader, A. Muneer, A. Alqushaibi *et al.*, "A comprehensive systematic review of YOLO for medical object detection (2018 to 2023)," *IEEE Access*, vol. 12, pp. 57 815–57 836, 2024. https://doi.org/10.1109/ACCESS.2024.3386826

[15] M. Flores-Calero, C. A. Astudillo, D. Guevara, and J. Maza, "Traffic sign detection and recognition using YOLO object detection algorithm: A systematic review," *Mathematics*, vol. 12, no. 2, p. 297, 2024. https://doi.org/10.3390/math12020297

[16] H. J. Park, J. W. Kang, and B. G. Kim, "ssFPN: Scale sequence (S2) feature-based feature pyramid network for object detection," *Sensors*, vol. 23, no. 9, p. 4432, 2023. https://doi.org/10.3390/s23094432

[17] H. Yu, X. Li, Y. Feng, and S. Han, "Multiple attentional path aggregation network for marine object detection," *Appl. Intell.*, vol. 53, no. 2, pp. 2434–2451, 2023. https://doi.org/10.1007/s10489-022-03622-0

[18] A. G. Roy, N. Navab, and C. Wachinger, "Recalibrating fully convolutional networks with spatial and channel "squeeze and excitation" blocks," *IEEE Trans. Med. Imaging*, vol. 38, no. 2, pp. 540–549, 2018. https://doi.org/10.1109/TMI.2018.2867261

[19] S. Saxena, S. Dey, M. Shah, and S. Gupta, "Traffic sign detection in unconstrained environment using improved YOLOv4," *Expert Syst. Appl.*, vol. 238, p. 121836, 2024. https://doi.org/10.1016/j.eswa.2023.121836

[20] L. Zhuo, B. Liu, H. Zhang, S. Zhang, and J. Li, "MultiRPN-DIDnet: Multiple RPNs and distance-IoU discriminative network for real-time UAV target tracking," *Remote Sens.*, vol. 13, no. 14, p. 2772, 2021. https://doi.org/10.3390/rs13142772

[21] J. He, S. Erfani, X. Ma, J. Bailey, Y. Chi, and X. S. Hua, "Alpha-IoU: A family of power intersection over union losses for bounding box regression," in *Proceedings of the 35th International Conference on Neural Information Processing Systems*, 2021, pp. 20 230–20 242.

[22] Z. Ge, S. Liu, F. Wang, Z. Li, and J. Sun, "YOLOX: Exceeding YOLO series in 2021," *arXiv preprint arXiv:2107.08430*, 2021. https://doi.org/10.48550/arXiv.2107.08430

[23] O. E. Olorunshola, M. E. Irhebhude, and A. E. Evwiekpaefe, "A comparative study of YOLOv5 and YOLOv7 object detection algorithms," *J. Comput. Soc. Inform.*, vol. 2, no. 1, pp. 1–12, 2023. https://doi.org/10.33736/jcsi.5070.2023

[24] P. Hönig and W. Wöber, "Explainable object detection in the field of search and rescue robotics," in *International Conference on Robotics in Alpe-Adria Danube Region*, 2023, pp. 37–44. https://doi.org/10.1007/978-3-031-32606-6_5

[25] Z. Shao, W. Wu, Z. Wang, W. Du, and C. Li, "Seaships: A large-scale precisely annotated dataset for ship detection," *IEEE Trans. Multimed.*, vol. 20, no. 10, pp. 2593–2604, 2018. https://doi.org/10.1109/TMM.2018.2865686

[26] V. C. Mahaadevan, R. Narayanamoorthi, R. Gono, and P. Moldrik, "Automatic identifier of socket for electrical vehicles using SWIN-transformer and SimAM attention mechanism-based EVS YOLO," *IEEE Access*, vol. 11, pp. 111 238–111 254, 2023. https://doi.org/10.1109/ACCESS.2023.3321290