processes

MDPI

Article

Detection Method of Key Ship Parts Based on YOLOv11

Yao Wang [1], Yi Jiang [1], Huiqi Xu [1,*], Chuanliang Xiao [2] and Ke Zhao [2]

[1] Naval Aviation University, Yantai 264001, China; wy@cqu-edu.cn (Y.W.); jiangyi1989913@163.com (Y.J.)
[2] Electrical and Electronic Engineering Academy, Shandong University of Technology, Zibo 255000, China; xiaocl@sdut.edu.cn (C.X.); 23504040599@stumail.sdut.edu.cn (K.Z.)
* Correspondence: jy@cqu-edu.cn; Tel.: +86-15564506237

Abstract: The current ship target detection algorithm is not adaptable to changes in target size and position, and its detection accuracy and speed are difficult to balance in the process of algorithm improvement. In particular, there is little research on the detection of key parts of ships. In this paper, an improved YOLOv11 ship key location detection algorithm is proposed. It is difficult to balance the relationship between detection efficiency and model complexity using current key parts detection algorithms. The processing performance of deforming key parts according to the algorithms is not strong. Moreover, the algorithms' detection performance of occluding overlapping targets and small-size targets is not high. In view of the practical problems existing in the algorithms and the realistic demand for real-time and accurate detection of key parts of targets on dynamic ships at sea, three aspects of improvement on YOLOv11 are proposed. The visible light ship key position dataset is then constructed for the experiment. The experimental results show that the improved YOLOv11 network in this paper can detect the key position information of ships more accurately and efficiently, effectively balance the relationship between model accuracy and speed, overcome the interference of dynamic factors, improve the fine interpretation of dynamic ship information at sea, and provide a feasible technical approach for accurately and precisely mastering ship information at sea and realizing real-time sea area detection.

Keywords: ship target; key part; YOLOv11; attention mechanism; feature fusion

check for updates

Academic Editor: Jun-Qiang Wang

Received: 13 December 2024
Revised: 29 December 2024
Accepted: 7 January 2025
Published: 13 January 2025

Citation: Wang, Y.; Jiang, Y.; Xu, H.; Xiao, C.; Zhao, K. Detection Method of Key Ship Parts Based on YOLOv11. *Processes* **2025**, *13*, 201. https://doi.org/10.3390/pr13010201

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

With the deepening of international exchanges and the high speed of global trade, maritime transportation plays an important role in the exchanges and communications between countries and regions. Ships are important carriers for realizing maritime transportation and supply, reconnaissance and patrol, and life-saving medical care. To perform various tasks, accurate and real-time mastery of ship information on the surface of the sea is of great significance to maintaining smooth maritime traffic and stability of the sea area, and to keep trade, economic and cultural exchanges unimpeded.

At present, many types of ships are densely distributed on the sea, and with the development and application of artificial intelligence, big data, material manufacturing technology, etc., the types and functions of ships are becoming more and more diversified and complex. Realizing the accurate detection and identification of the key parts of the ship is the key to differentiating between various types of ships, determining the direction of the ship and other important information, and to being able to pass the key part of the information to judge the ship's situation when the ship receives smokescreen and other blocking and overlapping interferences. This method provides accurate detection of the key parts of the ship in the complex marine environment, is able to effectively distinguish between

background and ship information, differentiates between multiple types of ship targets, and can effectively deal with the impact of the geometric changes generated by the ship's movement on the detection. Most traditional maritime ship detection methods require artificial setting of features, with high algorithmic complexity, low real-time performance, and low robustness.

With the progress and innovation of computer technology, artificial intelligence technology and other emerging technologies and their in-depth application in many fields, the target detection method based on deep learning gradually shows its unique advantages, has high detection accuracy and fast speed, and is widely used in the field of computer vision, covering a variety of image analysis and processing tasks, such as scene recognition, attitude estimation, emotion analysis, motion detection, image segmentation and so on. The target detection technology based on deep learning has been widely used in the detection and identification of ships for its outstanding advantages. Compared to traditional methods, deep convolutional neural networks effectively utilize a large amount of data, automatically extract target information through mature training, obtain target features with high generalization, and are able to better cope with the influence of shape change, background, illumination, occlusion, etc., with strong robustness. In complex scenes, the use of target detection technology based on deep learning can greatly improve accuracy and efficiency, so the target detection and identification methods for maritime vessels are rapidly developing from traditional methods to the field of deep learning. Reference [1] obtains high-resolution feature images by means of dense attentional feature aggregation and uses iterative fusion and dense connectivity to improve the generalizability of the network, which improves the accuracy of the network in detecting ship targets; Reference [2] uses the methods of low-hat transform (Bottom-HAT) and high-hat transform (YOP-HAT) to process the input image and uses the subsumption algorithm and the threshold segmentation method to extract the candidate region. Finally, through the classification idea of deep learning algorithm for ship target detection, compared with the traditional method, the detection accuracy and speed have been significantly improved; by optimizing the YOLO (you only look once) network and extracting the ship target's candidate characteristics across the region of interest, Reference [3] increases the precision of identifying and detecting the ship target on the high-resolution satellite map. In order to increase the detection accuracy of tiny vascular targets, the literature [4] presents the CA positional attention module in the YOLOX model and applies the BiFPN weighted feature fusion method in the network's neck structure, IM-YOLO-S.; Reference [5] constructs a cross-connected feature pyramid network (CCFPN) based on the multi-scale ship target detection algorithm in SAR images, fuses deep and shallow feature maps, uses channel splicing method to increase the detection accuracy of ship targets in high-resolution satellite images and increases the amount of information contained in the feature map by using the method of channel splicing, and improve the ability to extract shallow features of the target through multi-way cavity convolution, which has better performance for the detection of multi-scale ship targets in complex backgrounds; Reference [6] adds a feature-mapping module to the Yolov3-tiny network so that the prediction layer of the network is capable of extracting more important target features and enhances the detection accuracy of the densely distributed small ship targets.

Currently, a variety of target detection algorithms can realize the task of ship target detection in different scenarios, but for the future development of the increasingly complex marine environment, there are still the following deficiencies:

(1) The ship target is viewed as a whole for detection and identification, the key parts are not accurately detected, and the fine-grained features are not fully utilized, so it cannot provide accurate ship target guidance information;

(2) There are many types of ship targets at sea with different morphologies, the existing algorithms are weak in the correlation analysis between the multi-channel feature maps, the feature utilization of ship targets of different scales is not high, and it is easy to lose the key feature information and reduce detection accuracy;
(3) The current dataset for ship targets is less, especially the visible image dataset, and the classification data level for the key parts of the ship is lacking.

Target detection algorithms based on deep learning are divided into a two-stage object detection algorithm and a one-stage object detection algorithm. The two-stage object detection algorithm has high detection accuracy but slow speed, which is difficult to meet the requirements of real-time task scenarios. The one-stage object detection algorithm has a fast detection speed, and the detection progress has been rapidly improved with the continuous improvement of the network. In particular, the YOLOv11 network proposed in 2024, compared with the previous network, has been qualitatively improved in detection accuracy and speed, and the model is lighter, showing outstanding performance in processing real-time tasks and detecting multi-size targets and small targets. Compared to traditional detection methods, the YOLOv11 target detection network shows outstanding advantages in accuracy and speed. In addition, visible light images have exceptional features and diverse information; are intuitive and easy to read; are less affected by temperature, wind, and other natural environments and disturbances; contain rich spectral information; and are resistant to the influence of smokescreens and other masking interferences. These attributes allow for the comprehensive and complete expression of low-altitude and ultra-low-altitude image information, which can help to improve image performance. The study of visible ship image key part detection based on deep learning can better adapt to the complex environment and accurately detect and recognize multiple types of targets.

Aiming at the shortcomings of the current ship target detection method and the problem of drastic changes in the target characteristics during image acquisition, this paper proposes the following improvement strategies: to construct a visible light ship image key part detection method based on YOLOv11 (1) for the problem that the target size of the ship navigating at sea affects the accurate detection due to geometric changes caused by the movement using the deformable convolutional DCNv4 reconstruction of the new modules to replace the original bottleneck block of C3k2 in the YOLOv11 backbone network. DCNv4 combines the advantages of convolution networks and DCNv3 and uses dynamic adaptive aggregation window to calculate weights according to input features, which enhances the expression ability and dynamic characteristics. In DCNv4, multiple channels in the same group are processed by one thread, which effectively reduces the work of reading memory and calculating bilinear interpolation coefficients. In this way, the algorithm can more accurately estimate the spatial deformation of the moving ship target without adding too many training parameters of the model. (2) AKConv uses different sampling shapes to extract features and has a stronger adaptability to targets with different sizes and shapes. The VoVGSCSP structure can integrate multiple levels of feature information to obtain more complete target information. To address the problem that the key parts of the ship belong to the category of small target detection, which is difficult to detect, the AVB module is constructed by using the AKConv changeable kernel convolution to improve the structure of VoVGSCSP and then combining with the BiFPN combination to construct a new type of neck network (Bi-AVB-FPN) to transmit the information in both directions at once. In this way, we can enrich multi-scale feature fusion, enhance feature semantic information, and help the model more accurately locate the target area of the key ship parts. (3) To address the problem of the generally limited resources of edge devices and the low applicability of detection algorithms, a lightweight GSConv convolution module is introduced in the neck part. The GSConv lightweight convolution module adopts the optimization strategy of

grouping convolution and spatial convolution, which can dynamically allocate computing resources and effectively reduce computation and memory loss. The improvement not only achieves higher detection accuracy, especially in small-scale target detection, but reduces power consumption and energy consumption. (4) Aiming at the problem that SAR image and infrared image acquisition is easily affected by the complex electromagnetic environment and the interference of the natural environment, we construct the data set of the key parts of the visible ship.

Based on the purpose of improving the detection performance of ship key parts at sea, this paper analyzes the shortcomings of the current key parts detection algorithm, proposes three improvements for the YOLOv11 network, and self-builds the visible key ship parts data set. Experiments show that the algorithm proposed in this paper can realize accurate and real-time detection of ship key parts at sea, and the network has higher accuracy and real-time performance, strong adaptability to different sizes and types of targets, and little influence from complex background interference. The research provides favorable technical support and theoretical support for realizing complete, real-time, and comprehensive control of ship information at sea.

2. Preliinary Concepts

2.1. Algorithm Principle

Current commonly used key part detection methods can be categorized into top-down (Top-Down) and end-to-end methods.

The Top-Down method first detects the target information, and then searches for the key part information inside the target. The mainstream algorithms are CPN, Hourglass, CPM, HRNet, and so on.

The main processing steps of the algorithms are:

(1) Detecting the target category information by deep learning-based target detection algorithms such as Faster R-CNN, YOLO, etc., and cropping and scaling to the specified size according to the location of the target bounding box;

(2) Inputting the region obtained in Step (1) into a network specifically trained for key part detection, such as HRNet, Stacked Hourglass Network, Mask R-CNN key point branching, etc., to generate a heat map. Each heat map corresponds to a key part center point, and the location with the highest value in the heat map indicates the most likely location of the key part center point, and then decodes the heat map to obtain the specific coordinate value of the center point;

(3) Perform non-maximum suppression (NMS) on the acquired coordinates to reduce the redundancy and improve the detection accuracy, and may also take corrective operations such as backpropagation to adjust the location of the center point of the key part.

The top-down method uses the target detection algorithm to obtain the target location information, simplifies the follow-up process of key vital parts detection and can be processed separately for multi-target cases, which can more accurately obtain the location information of the center point of the key parts. However, this method has high computational complexity, long training time, slow detection speed, and is not suitable for scenes with high real-time requirements; the method is very dependent on the accuracy of the target detection results, which is prone to large errors. The algorithm flow is shown in Figure 1. Different color boxes in the resulting diagram represent different key parts of the vessel.



Figure 1. Top-down key part detection algorithm.

The end-to-end approach is shown in Figure 2 and utilizes a unified network framework to detect the target and key parts. Different color boxes in the resulting diagram represent different key parts of the vessel, similarly. The network structure of this type of approach is relatively simple, but it is easy to ignore the connection between the target and key parts during the solution process, and it performs poorly in coping with the situation of feature insignificance and the problem of detecting small targets.

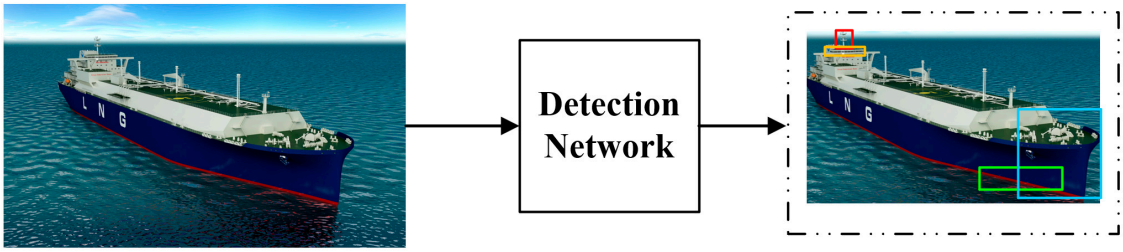


Figure 2. End-to-end key part detection algorithm.

2.2. Research Difficulties

The current key part detection algorithms generally have the following problems in realizing the accurate detection of the key parts of the ship:

(1) Insufficient deformable ability to deal with key parts. Ships at sea are densely distributed; there are many types of targets whose key parts are the location and number of differences, and in the process of ship target movement—subject to displacement, attitude transformation, and image acquisition equipment performance—motion and other factors interfere with the acquisition of image information in the key parts, and geometric changes are prone to occur. It is difficult to make flexible adjustments with the existing detection methods for coping with such differences;

(2) The balance between model complexity and efficiency is difficult to coordinate. While the network improves the target detection accuracy, it is often accompanied by an increase in model complexity and computation, which is difficult to apply to real-time monitoring environments;

(3) Detection of occluded and overlapped targets and small targets. The detection performance of tiny targets, overlapping targets, and occlusion is inadequate, and the network sensitivity is low. The important components of the ship are small target objects that only make up a small portion of the entire ship. In particular, when there is overlap and occlusion, the network can only gather a limited amount of feature information, and the detection results will be tangled amongst various important components. Furthermore, the distribution of background and target information is unbalanced in the diverse maritime environment, which makes it difficult to locate important ship components;

(4) Deep and shallow feature fusion problem. Through downsampling and pooling operations, convolutional neural networks extract both deep and shallow feature information. While shallow features lack high-level semantic information but contain

rich target location and detail information, deep features contain more abstract and overall information, which is crucial for identifying the features of the important parts. Existing methods have poor fusion of deep and shallow feature information and need to better fuse different layers of feature information to improve the network detection effect.

Aiming at the current network algorithms' applicable scenarios, existing problems and the need to improve the accurate detection of key parts of ships, the article proposes an improved method for realizing the detection of key parts of ships in complex marine environments based on the latest YOLOv11 target detection network to improve the network detection performance.

2.3. YOLOv11

By drastically cutting the number of parameters and cutting the reasoning time by 2% when compared to YOLOv10, the YOLOv11 target detection network significantly increases network efficiency and operating speed, making it the ideal option for real-time mission scenarios. In the COCO dataset, YOLOv11 has a higher mean average precision (mAP). Compared to YOLOv8, the number of parameters is reduced by 22%, the network is more lightweight and well balanced between speed and accuracy, the overall design of YOLOv11 gives it the advantage of capturing complex image information, and the ability to detect targets in difficult environments (poorly lit or cluttered scenes) is stronger. The YOLOv11 network is very adaptable and suitable for a variety of mission situations and working conditions as it can be effectively set up and run on edge devices like smartphones, cloud servers, or Internet of Things gadgets.

The network structure of YOLOv11 is shown in Figure 3.

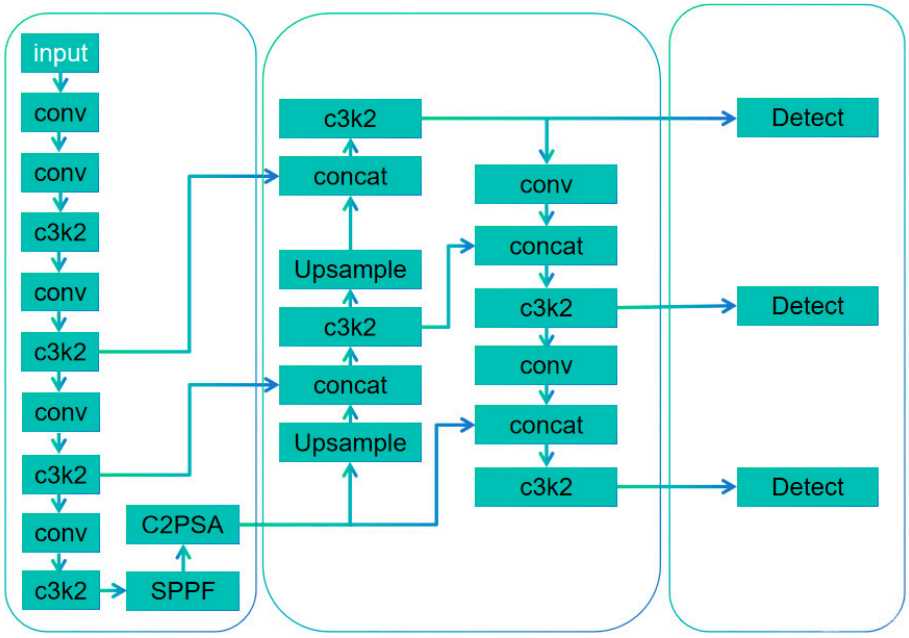


Figure 3. The network structure of YOLOv11.

YOLOv11 mainly includes the following parts: (1) Backbone. The pre-trained convolutional neural network CSPDarknet is used to extract image features. Pre-training model parameters can save computing resources and speed up the training process of new tasks. In addition, the network can learn a variety of common features with good mobility through pre-training weights. In new tasks, by fine-tuning pre-training weights, the model can be integrated into new problem scenarios more quickly and the performance

of handling problems can be improved. Moreover, it can avoid overfitting problems for small data. (2) Neck. This portion fuses features of different scales to enable the model to recognize multi-scale target information. YOLOv11 uses the improved PANet structure to enhance the level of feature fusion. PANet adopts a horizontal connection mode, and information can be conveyed in the feature maps of different layers so that the high-level features can make full use of the details of the lower layers. (3) Head. This portion is to predict the category probability and boundary box. The head part of YOLOv11 has multiple branches to perform prediction, which can deal with multi-scale target detection problems. YOLOv11 adds two depth-separable convolution DWConvs in the classification detection head of the decoupling head to reduce the number of parameters and the amount of computation.

2.4. Basic Model Structure Introduction

2.4.1. DCNv4

In 2024, the fourth generation of Deformable Convolutions DCNv4 is proposed as an efficient dynamic sparse operation operator that provides a wider range of feature-level sampling when extracting feature maps compared to the previous versions (e.g., DCNv3, DCNv2, DCN) [7]. DCNv4 integrates a modulation mechanism. This mechanism not only learns the offsets of the samples, but also modulates the amplitude of the learned features, which allows DCNv4 to control its sample space and relative impact [8].

DCNv4 can significantly improve network convergence and processing speed and has significant performance advantages in tasks such as image classification, semantic and instance segmentation, and image generation. The main advantages of DCNv4 are:

(1) Abandon softmax normalization in spatial aggregation. DCNv4 uses dynamic adaptive window to calculate weights based on input features, removing the limitation of weight normalization, and enhancing expressive ability and dynamic characteristics; (2) improve memory access, using shared aggregation weights and sampling offsets so that a single thread can synchronize multiple channels in the same group, avoiding non-essential memory reads and operations, and accelerating network convergence and processing speed. operations, which accelerates network convergence and processing speed; (3) DCNv4 introduces vectorized loading, which can read data from multiple channels at a time, reducing the number of instructions to access the memory and further improving the efficiency of accessing the memory.

As can be seen from Figure 4, the attention mechanism in DCNv3 aggregates spatial features through a limited number of dynamic weights (0, 1), the set of sampling points—i.e.—the window, is the same for the attention mechanism, whereas DCNv3 applies a specialized window for each position, the convolution's aggregation weights are unbounded and have a more flexible range, and a specialized sliding window is applied for each position, but the window shapes and aggregation weights are independent of inputs. DCNv4 combines the advantages of convolution with DCNv3 using adaptive aggregation windows and dynamic aggregation weights with unbounded ranges of values. Multiple channels within the same group are processed in DCNv4 by a single thread, which share the aggregation weights and sampling offsets in such a way as to reduce the work of reading memory and calculating bilinear interpolation coefficients, etc., and to be able to merge multiple memory access instructions.

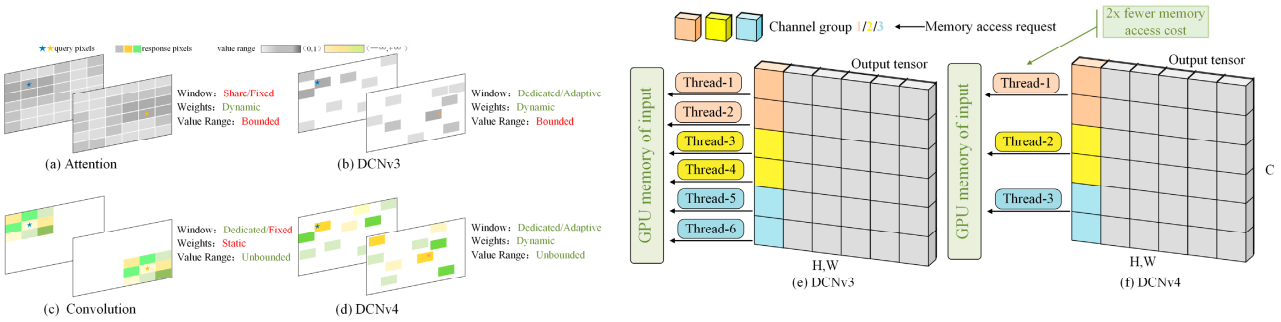


Figure 4. Network Performance Comparison Chart.

The detection rate of FlashInternImage Backbone equipped with DCNv4 reaches 48 FPS and the detection accuracy reaches 57.7%. Compared with 53.9% of InterImage-T network, the detection accuracy is improved by 3.8%. The number of network parameters is 48.5 M. Compared with the Interimage-t parameter 48.7 M, the number is reduced by 0.2 M, and the network performance is better than the original InterImage Backbone.

2.4.2. BiFPN

The BiFPN (bi-directional feature pyramid network) [9], is an extension of the feature pyramid network (FPN) by adding bi-directional interconnections within the pyramid layers so that information in the network can be transmitted from top to bottom and from bottom to top simultaneously. The network structure is shown in Figure 5. Different colors in the figure means different layers.

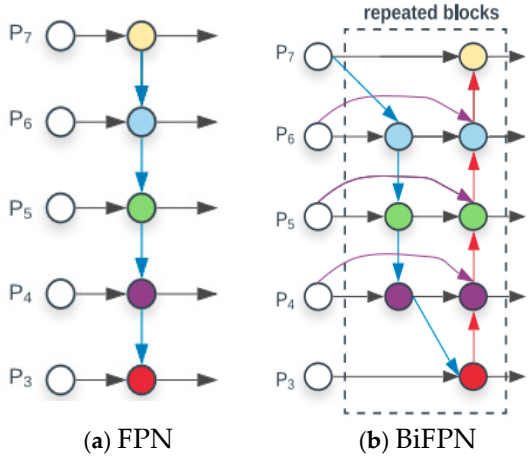


Figure 5. Comparison of network structure.

The BiFPN network uses convolutional neural networks such as ResNet as the backbone network and extracts the target features from multiple layers to form a feature pyramid and applies bi-directional connections between each layer of the pyramid to enhance the feature circulation between the lower and higher layers, and at the same time, the bi-directional connections can integrate the information of each layer of the pyramid from the upper and lower directions to effectively obtain the multi-scale feature information. The BiFPN also applies the feature weighted fusion method to fuse the features between different layers, and the weights can be obtained during the training to ensure the optimal integration of features. The BiFPN also applies a feature weighting fusion method to integrate features between different layers, and the weights can be obtained during training to ensure the optimal integration of features.

2.4.3. AKConv

AKConv [10] changeable kernel convolution performs feature extraction through irregular convolution kernel functions, which can provide convolution kernels with arbitrary sampling sizes and shapes for a wide range of varying targets, filling in the shortcomings of traditional convolution. In the target detection test and comparison test on COCO2017 and VisDrone-DET2021, AKConv shows better performance:

(1) Flexible sample shape: compared to the fixed sample shape of conventional convolution, AKConv allows convolution to have different sizes and shapes, which can flexibly adapt to deformed targets and greatly improve the detection accuracy;
(2) Number of parameters and operational efficiency: AKConv can reduce the number of parameters and the number of operations by changing the size of the convolution, thus improving the operational efficiency and performance of the model;
(3) Optimize network performance: as a lightweight model, AKConv can replace the conventional convolution to reduce the number of parameters and computational overhead of the model and can maintain model performance while improving the network generalization ability and efficiency;
(4) Scalability: AKConv can be scaled to larger kernel sizes, providing more options for improving network performance, adapting to a variety of scenarios, and having wide applicability.

Figure 6 shows the size of five convolutional initial samples; AKConv can design different initial sampling shapes to achieve any shape of sampling.

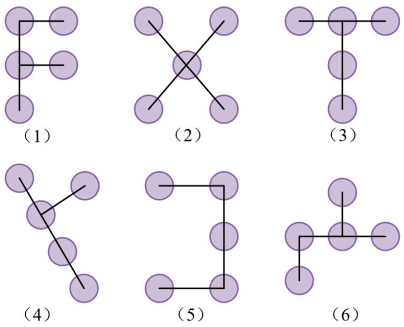


Figure 6. Sampling shape diagram.

The mAP value of the YOLOv7-tiny network using AKConv reaches 51.3% in COCO2017 data set, the detection accuracy of the basic YOLOv7-tiny network without AKConv is 50.2%, and the detection accuracy is increased by 1.1%. The GFLOPS value of the improved network is 13.7, which is 0.5 higher than that of the basic network (13.2), and the network inference speed is faster. The AKConv structure is shown in Figure 7.

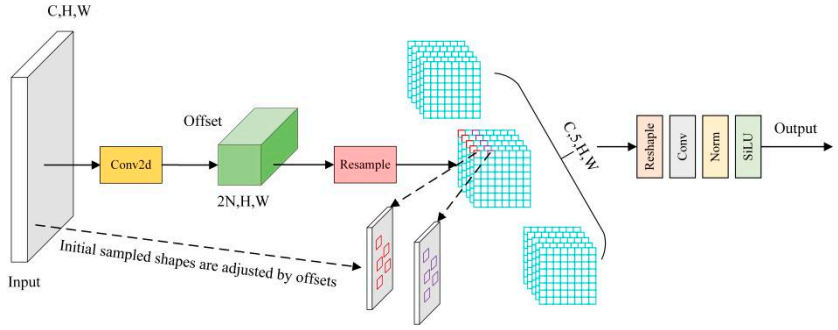


Figure 7. AKConv structure diagram.

2.4.4. VoVGSCSP

The VoVGSCSP module (volumetric grid spatial cross-stage partial) maps the input features into a voxelized network to obtain more local feature information in space and references the CSPNet (cross-stage partial network) network idea to utilize the method of fusion and separation of features to maintain the accuracy of multi-type dynamic ship targets at sea. VoVGSCSP can efficiently fuse multiple different levels of feature information, improve the feature expression capability, better balance between model network performance and computational efficiency, and is more suitable for real-time scenarios. The module structure is shown in Figure 8.

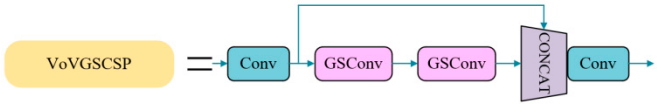


Figure 8. Structure of the VoVGSCSP model.

2.4.5. GSConv

The GSConv (grouped spatial convolution) convolution module [11] adopts the optimization method of combining grouped convolution and spatial convolution to reduce computation and memory loss.

The time complexity of the convolutional computation is defined by the floating-point operators FLOPs. The time complexity of SC, DSC, and GSConv without considering bias is:

$$Time_{SC} \sim O(W \cdot H \cdot K_1 \cdot K_2 \cdot C_1 \cdot C_2) \tag{1}$$

$$Time_{DSC} \sim O(W \cdot H \cdot K_1 \cdot K_2 \cdot 1 \cdot C_2) \tag{2}$$

$$Time_{GSConv} \sim O(W \cdot H \cdot K_1 \cdot K_2 \cdot \frac{C_2}{2} \cdot (C_1 + 1)) \tag{3}$$

where W, H denote the output feature map width and height, respectively; K_1, K_2 denote the convolution kernel size, and C_1, C_2 denote the number of channels of the input feature map and output feature map. The mAP value of GSConv in PASCAL VOC 2007 + 12 dataset reached 59.8%, compared with 58.9% of SC, 56.9% of DSC, and 59.3% of Ghostnet, which are increased by 0.9%, 2.9%, and 0.5%, respectively. The FLOPs of GSConv module is 3.7 G, and the total computing cost is lower than that of SC module and GhostNet module, and the model complexity is reduced.

In order to balance the computational cost and detection performance, GSConv may also dynamically assign computing resources based on hardware performance. This not only increases detection accuracy but also lowers power and energy usage. The accuracy for tiny targets is somewhat improved by GSConv convolution as it generates less computational load than traditional convolution, particularly when it comes to small-size feature maps.

The GSConv structure is shown in Figure 9.

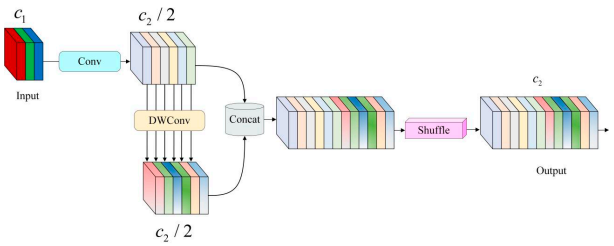


Figure 9. GSConv structure. Where c_2, $c_2/2$ denote the number of channels.

3. YOLOv11 Based Ship Key Part Detection Method

Aiming at the difficult problem of detecting key vital parts of dynamic ship targets in complex environments, YOLOv11 is improved as follows in order to further enhance the accuracy and efficiency of detection as well as the ability to adapt to complex environments.

3.1. C3k2_DCNv4 Structure

The majority of detected targets in ship detection of multiple target types in key and vital areas are irregular shapes that are imaged from various angles. Ordinary convolution's convolution kernel is typically a regular square structure, making it challenging to accurately cover the target area. In order to further improve the detection accuracy of key parts of multi-scale ships and improve the processing ability of target deformation based on the advantages that DCNv4 can learn the sample offset and modulate the learned feature amplitude so as to control its sample space and relative influence, the convolution of DCNv4 is quoted in YOLOv11. A new module reconstructed using DCNv4 convolution replaces the original bottleneck block of C3k2 in the YOLOv11 backbone network to more accurately estimate the spatial deformation of a dynamic ship target and to improve the network's adaptability to the geometrical changes of the ship's key parts due to motion.

By introducing DCNv4 to improve the backbone network of the YOLOv11 model, the shape of the convolution kernel can be adaptively changed to cover the key areas of the ship as much as possible, thus obtaining richer feature information and realizing more accurate feature extraction of the key parts of the ship.

3.2. Multi-Scale Feature Pyramid Network (Bi-AVB-FPN)

In order to solve the issue of the ship's key section belonging to a tiny target that is hard to identify, the model's neck network has to be improved. Affected by the ship target sailing and image acquisition equipment movement, it will cause geometric changes of the acquired target and affect the net detection effect. By introducing AKConv, it provides multiple sizes and shapes of convolution kernels for dynamic ship target detection, improves the ability to adapt to the dynamically changing scene, and enhances the detection accuracy.

The low-level information of the image obtained through the convolution operation contains more detailed features of the target, while the high-level information contains more semantic information and spatial features. Effectively fusing the information of different levels and realizing cross-layer connectivity and interoperability can further improve the detection accuracy of multiple types of dynamic ship targets at sea. This experiment adds AKConv to the VoVGSCSP module, using AKConv as a two-branch convolution after concat operation. A new convolution is obtained, replacing the orignal Conv in VoVGSCSP module, and creates a new module called the AVB (AKConv-VoVGSCSP Block, AVB) module that substitutes the C3k2 module for the first two layers of the neck network.

The BiFPN mode of action can better obtain multi-scale feature expression, effectively enhancing the network's ability to handle targets of different complexity and size. This is crucial for scenarios where there is a significant difference in the target's size.

In view of the wide distribution and diverse types of marine ship targets, the acquisition of images containing a variety of sizes and different forms of ship targets in the task scenario, the BiFPN network structure is used to construct the Bi-AVB-FPN network to enhance the information fusion between different layers, obtain more complete feature information, and enhance the detection performance of the network on multi-size-and-shape targets in complex scenes.

3.3. GSConv Architecture

In order to speed up the operation process of the convolutional neural network CNN, the images input to the CNN are basically subjected to such transformations in the backbone network: the spatial information is gradually passed to the channel. However, each time, the feature map for spatial (width, height) compression and channel expansion will produce part of the semantic information loss. The channel-dense convolution SC maximizes the preservation of the hidden connections of each channel, while the channel-sparse convolution DSC completely disconnects these connections. GSConv preserves these connections as much as possible with a much lower time complexity.

In order to ensure the model detection accuracy, reduce the model complexity as much as possible, improve the operation efficiency, and make the model more suitable for the use scenario with limited edge device resources, a lightweight GSConv module is introduced in the neck part.

3.4. Improved YOLOv11

Combined with the above improvements, the YOLOv11 network structure is improved and the overall network structure model is shown in Figure 10. The improved YOLOv11 network includes three parts: backbone, neck and detect. The backbone part extracts image features, the neck part recognizes multi-scale target information, and the head part predicts category probability and boundary box of targets.

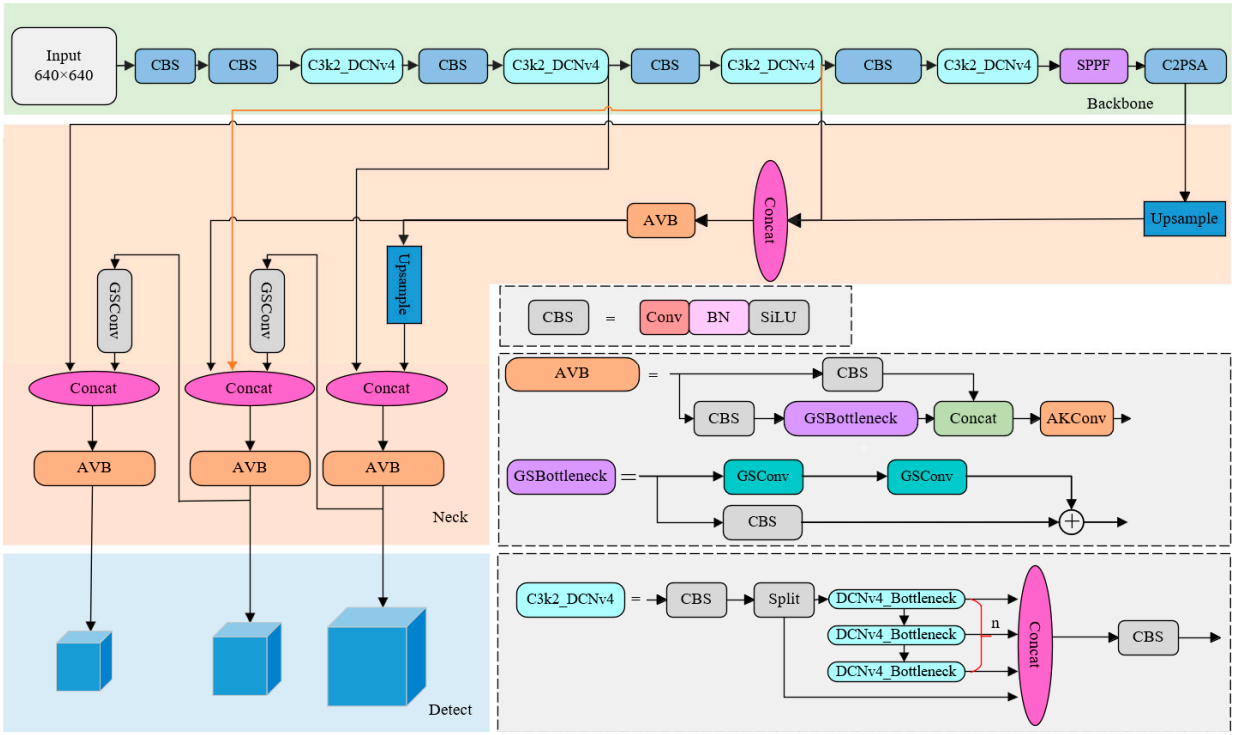


Figure 10. Improved YOLOv11 network architecture.

3.4.1. Loss Function of YOLOv11

The loss function design uses the Extended IoU (EIoU) loss function. Compared with the IoU (intersection over union) function, EIoU comprehensively considers the overlapping area between the real frame and the predicted frame and introduces the penalty term of the

aspect ratio and center point offset, which accelerates network convergence and improves the accuracy of prediction. The formula is as follows:

$$\begin{aligned} L_{EIoU} &= L_{IoU} + L_{dic} + L_{asp} \\ &= 1 - IoU + \frac{\rho^2\left(b,b^{gt}\right)}{(c_w)^2+(c_h)^2} + \frac{\rho^2\left(w,w^{gt}\right)}{(c_w)^2} + \frac{\rho^2\left(h,h^{gt}\right)}{(c_h)^2} \end{aligned} \tag{4}$$

where ρ represents the euclidean distance between the center point of the real frame and the prediction frame; c_w, c_h represent the height and width of the minimum external frame covering the two boxes respectivey; b, b^{gt} represent the center point of the prediction box and the real box respectively; w, w^{gt} represent the width of the predicted box and the real box respectively; h, h^{gt} represent the heights of the predicted box and the real box, respectively.

The loss function of YOLOv11 network includes three parts: classification loss (cls loss), positioning loss (box_loss), and distribution focal loss.

(1) cls loss optimizes the accuracy of the model to predict the target type and ensures the correctness of the classification.

$$cls_loss = \sum_{i=0}^{S^2} 1_i^{obj} \sum_{c \in classes} \left(p_i(c) - \hat{p}_i(c)\right)^2 \tag{5}$$

where S is the mesh size; 1_i^{obj} represents whether the ith grid cell has a target; $p_i(c)$ represents the probability that the target class in the ith grid cell predicted by the model is c; $\hat{p}_i(c)$ represents the true label, indicating whether the target category in the ith grid cell is c.

(2) Positioning loss (bounding box regression loss, box loss) optimizes the difference area of the real and predicted bounding box to ensure accurate target positioning.

$$\begin{aligned} box_loss = \quad & \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^{obj} \left((x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2\right) + \\ & \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^{obj} \left(\left(\sqrt{w_i} - \sqrt{\hat{w}_i}\right)^2 + \left(h_i - \sqrt{\hat{h}_i}\right)^2 \right) \end{aligned} \tag{6}$$

where S is the mesh size; B is the predicted number of bounding boxes for each grid cell; 1_{ij}^{obj} represents whether the jth bounding box in the ith grid cell predicts the target; x, y is the coordinates of the center point of the boundary box; w, h is the width and height of the unbounded box; λ_{coord} is the weight coefficient, balancing the loss of multiple parts.

(3) Distribution focal loss (dfl_loss) coordinates the imbalance in the detection of target categories, enhances the model's ability to judge whether the target exists, and improves the level of network detection of small targets and complex samples.

$$DFL = -\sum_{i=1}^{N} \sum_{c=1}^{C} y_{ic} \left(\alpha (1 - p_{ic})^{\gamma} \log(p_{ic}) + (1 - \alpha) p_{ic}^{\gamma} \log(1 - p_{ic})\right) \tag{7}$$

where N indicates the number of samples; C represents the number of classes; y_{ic} represents the true label of the ith sample; p_{ic} represents the probabilistic predicted value of the ith sample class is c; α is a balance factor, used to adjust the weight between positive and negative samples; γ is a focusing parameter, used to adjust the attention to complex samples.

3.4.2. Network Training

YOLOv11 uses mixed precision training technology, which can speed up the training process and reduce the footprint of video memory while ensuring the accuracy of the model. Model training is divided into the following steps:

(1) Load the pre-trained model: the pre-trained weights on ImageNet are usually used as the initial weights of the model;
(2) Multi-scale training: the YOLOv11 model can randomly adjust the input image size to improve the model's adaptability;
(3) Loss function optimization: optimize the weighted sum of classification, location losses, and IoU losses.

4. Data Set Construction

4.1. Data Set Composition

Currently, most of the datasets for ship targets are SAR images and infrared images, they are mainly for the main body of the ship, and there is no publicly available visible light dataset for the key parts of the ship. In order to fit the actual marine environment and image acquisition equipment to the low altitude, ultra-low altitude flight perspective to explore the use of the network proposed in this paper in order to achieve accurate detection and identification of the ship and its key parts through the network collection, existing dataset excerpts, real-life shooting, and other methods of selecting the visible ship images containing the front-view and side-view images were used. The dataset contains a variety of types of ships, including the existence of occlusion, a small target scene, a total of 5589 chapters of images. The data set contains many types of ships, as well as occlusion and small target scenes, totaling 5589 images. The key parts of the ship are divided into cockpit, antenna mast, bow and waterline.

The number of images containing key parts is shown in Table 1.

Table 1. Number of pictures containing vital parts.

Key Parts	Cockpits	Antenna Masts	Ship's Bow	Waterline
Quantity (sheets)	4559	4724	5193	5276

The dataset is in PASCAL VOC format and labeled using the labelImgv1.8.1 tool. When labeling important targets, because the detection box is rectangular, it is difficult to avoid selecting irrelevant information. Especially when labeling the detection box of the waterline, too-narrow and ant too-long structure will increase the difficulty of detection. Therefore, when selecting the waterline, special attention is paid to the position close to the horizontal, which is weakly interfered with by the environmental noise. This reduces the narrowness of the labeling box and avoids selecting the ship's thematic information on a large area; for the cockpit, the antenna mast, the bow, and other key parts of the ship, the target edge is used as the benchmark for the labeling. The image format is .jpg. In order to accelerate the network training process, the image size is unified as 640 × 640 and named as 000001 format, and the image is labeled; 80% of the images in the dataset are divided into the training set and 20% are divided into testing set.

When labeling the waterline, special attention is paid to the position close to the horizontal and weakly interfered by the environmental noise, which reduces the narrowness of the labeling box and avoids selecting the main information of the ship in a large area; for the cockpit, the antenna mast, the bow and other key parts of the target edge as the reference to labeling.

4.2. Experimental Environment

This experiment is completed in Ubuntu18.04 operating system, using Pytorch 1.10 deep learning framework, cuda11.1 + cudnn8.0.4 + opencv4.5.8 environment; hardware environment is: Intel (R) Xeon (R) Platinum i9-13900k, Nvidia GeForce RTX 4090; program-

ming operations were implemented using the Python3.7.0 language. The detail parameters are shown in Table 2.

Table 2. Experimental environment parameters.

Parameter	Parameter Value
Operating system	Ubuntu 18.04
Bits	32-bit (computing)
RAM	64 GB
Graphics card	GeForce RTX 4090
Video memory	24 GB
CPU	Intel (R) Xeon (R) Platinum i9-13900k
Deep learning framework	Pytorch 1.10
CUDA version	11.1
Cudnn version	8.0.4

4.3. Experimental Parameter Settings

The input image size of this experiment is 640 × 640, the initial learning rate (learning rate) is 0.01, and the stochastic gradient descent is used to update the learning rate. The momentum (momentum) is 0.937, and the weight delay (weight_decay) is 0.0005. Mosaic data augmentation is used during the training process, and each time, we read four images, which are spliced after flipping, scaling, and other operations, respectively, to make the detection background richer. Label smoothing is set to 0.01 to prevent model overfitting and increase model generalization. All models are trained for 500 epochs, batch size is set to 32, and the number of worker threads is 16.

4.4. Evaluation Metrics

TP (true positive): samples accurately determined as positive samples; TN (true negatives): samples accurately determined as negative samples; FP (false positive): negative samples determined as positive samples; FN (false negative): positive samples are determined as negative samples; Q is the total number of queries.

(1) Precision

Precision is the accuracy/checking rate, i.e., the ratio of all samples detected as positive that actually belong to the class.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{8}$$

(2) Recall

Recall is the recall/check all rate, also known as the detection rate, which is the ratio of the number of targets detected as positive classes to the total number of all classes detected as targets.

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{9}$$

(3) AP (Average precision)

AP is the graphical area between the precision–recall curve (PR curve) and the x-axis; the x-axis of the curve is the recall value, which indicates the ability of the classifier to cover positive samples; and the y-axis is the precision value, which indicates the prediction

accuracy of the positive instances. The PR curve indicates the variation between the two values.

$$AP = \begin{cases} \int_0^1 PRdr, \text{ Continuous PR curve} \\ \sum_{k=1}^{n} P(k)\Delta r(k), \text{ discrete PR curve} \end{cases} \quad (10)$$

(4) mAP (mean average precision)

mAP i.e., the average of the APs of all the query results sorted.

$$mAP = \frac{\sum_{q=1}^{Q} AP(q)}{Q} \quad (11)$$

That is, the proportion of correct predictions to the predicted results. mAP50 indicates the average value of AP50 when the IoU threshold is set to 50%

(5) Detection time

The time to detect a single image, unit ms. The smaller the value, the shorter the time to detect a single image and the faster the algorithm.

4.5. Analysis of Experimental Results

The network LOSS curve is shown in Figure 11.

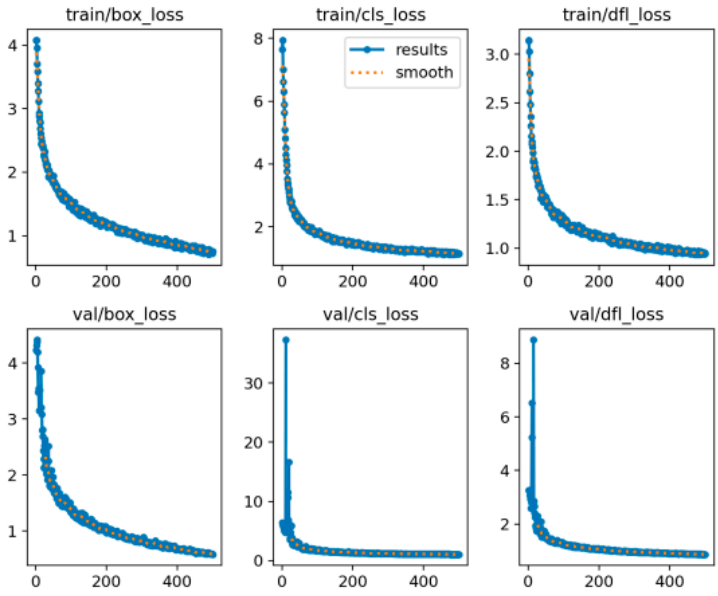


Figure 11. Loss function curve for 500 rounds of network training.

It can be concluded from the loss function curve graph that after sufficient iterative training, the network's loss value decreases gently and eventually tends to stabilizing to convergence. The final convergence of the loss function value is smaller, there are no overfitting or underfitting problems; in the process of the loss function decline, there are no large fluctuations, proving that the network model is reasonably designed and that the size and depth of the dataset is well constructed so that the network is adequately trained and converges smoothly.

In order to verify the adequacy of 500 rounds of training, the number of training rounds was set to 600 and the network loss function curve was obtained, as shown in Figure 12. It can be seen from the figure that the network has stabilized after 500 rounds, and the loss function is basically stable, which proves that it has been fully trained during 500 rounds.

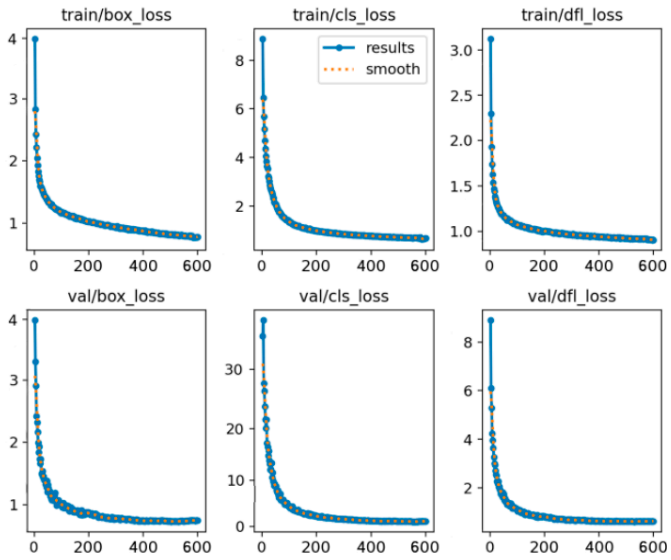


Figure 12. Loss function curve for 600 rounds of network training.

The network PR curve is shown in Figure 13. The detection accuracy of different types of targets is shown in Table 3.

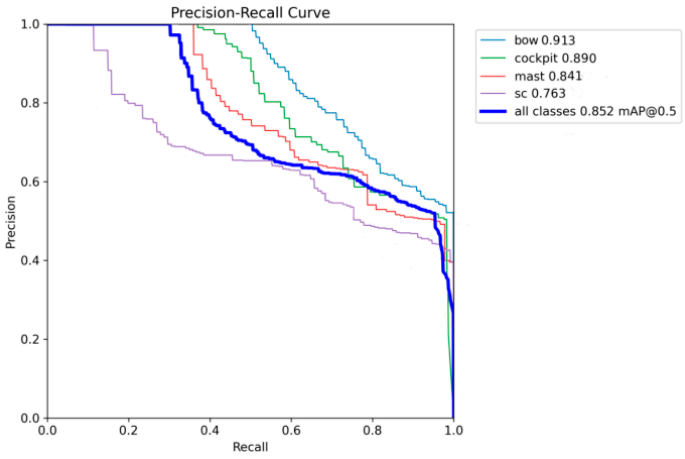


Figure 13. Network PR curve graph.

Table 3. Detection accuracy of vital parts.

Typology	Cockpits	Waterline	Mast	Bow	mAP50	Detection Time
AP	0.890	0.765	0.841	0.913	0.852	1.1 ms

The AP value in the table indicates the detection accuracy of each type of key part. mAP50 represents mean average precision 50, indicating the average detection accuracy of all types when the intersection over union (IoU) is 50%.

From the statistics of various types of key parts detection results, it is known that the improved ship key parts detection network based on YOLOv11 achieves 85.2% detection accuracy for key parts of multiple types of ships, with high detection accuracy, and the time for detecting a single picture is 1.1ms, which is able to meet the requirements of real-time detection of the marine environment. The bow detection accuracy reaches 0.913, the highest detection accuracy, mainly due to the more prominent features of the bow of the ship and other parts of the pixel ratio being large, easy-to-extract features; at a cockpit detection

accuracy of 0.890 and a mast detection accuracy of 0.841, the network is able to more accurately extract the characteristics of the two key parts of the two types of information. The detection accuracy of the waterline reaches 0.765, and when labeling the waterline, the size of the rectangular enclosing box is reduced. Only the waterline part is boxed as much as possible to reduce the selection of background factors, which improves the robustness of the network training, and the detection accuracy of the waterline is high. The detection effect of the network is shown in Figure 14.

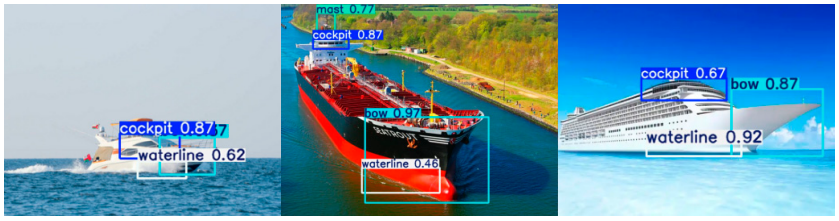


Figure 14. Network detection effect.

Figure 14 shows that the network can effectively detect the key parts of the ship target in the picture. For multi-angle ships, targets have shown better detection performance, high detection accuracy, and accurate labeling of the box location. The pixel occupies a relatively small part of the key parts of the target and can still be effectively detected and extracted information with basically no leakage detection.

Small targets have low proportions of pixels in the image, and their features are not significant in the important and difficult problem of target detection. In order to verify the adaptability of the improved YOLOv11 network to the size changes of targets, especially the detection performance of small targets, 1000 images of large target ships and 1000 images of small target ships were collected and the network performance was tested. The detection accuracy of various key parts was obtained as shown in Table 4.

Table 4. Detection accuracy of key parts of vessels of different sizes.

Targets' Size	Typology	Cockpits	Waterline	Mast	Bow	mAP50	Detection Time
Small targets	AP	0.847	0.751	0.719	0.913	0.808	1.1 ms
Large targets		0.923	0.840	0.762	0.911	0.859	1.1 ms

As can be seen from Table 4, the average detection accuracy of the network for large targets reaches 0.859 and that for small targets reaches 0.808, both of which achieve a high detection level. In addition, for several types of key parts, the detection accuracy of cockpit and bow parts with obvious features is higher, and the detection accuracy of the cockpit of large targets reaches 0.923. The detection accuracy of the small target cockpit reaches 0.847, while the detection accuracy of the bow reaches 0.913 and 0.911, respectively, indicating high detection accuracy. The detection results of waterline and mast are also good, which proves the adaptability of the network to different size targets and can effectively cope with the size changes of targets. The actual effect diagram of network detection is shown in Figure 15.



Figure 15. Detection effect.

In order to further verify the performance advantages of the constructed method, in order to cope with the requirements of key part detection accuracy and speed and the limitations of hardware equipment arithmetic power in the actual application environment, the training of SSD, CenterNet-Hourglass, YOLOv8n, YOLOv10n, and YOLOv11n networks is carried out in the same experimental environment and parameter settings, and the number of iterations is 500 epochs. The network performance was tested, and the basic performance parameters of the statistical networks are shown in Table 5.

Table 5. Comparative experiments of key parts detection.

Model	Precision (%)	Recall (%)	mAP50 (%)	Parameters	Times (ms)
SSD	60.2	78.9	74.3	16,4765,132	3.4
CenterNet-Hourglass	61.9	80.1	77.6	9,5432,146	3.3
YOLOv8n	62.7	86.3	79.7	3,006,623	1.0
YOLOv10n	64.2	80.5	78.1	2,696,366	1.5
YOLOv11n	65.3	82.1	80.8	2,583,127	1.4
Ours	70.6	87.7	85.7	2,437,559	1.1

Parameters are the total number of learnable parameters in the model. It is an important index for measuring the complexity of the model, which directly affects the storage requirements, computing costs, and training time of the model.

With a detection accuracy of 85.7%, the YOLOv11-based key component identification method suggested in this research outperforms the SSD, CenterNet-Hourglass, YOLOv8n, YOLOv10n, and YOLOv11n networks by 11.4%, 8.1%, 6.0%, 7.6%, and 4.9%, respectively. The algorithm's detection accuracy is greatly increased; the detection time of a single image is 1.1 ms, which is much faster than SSD and CenterNet-Hourglass networks. Additionally, when compared to the YOLO series algorithms in YOLOv10n and YOLOv11n, the network real-time is stronger and the improvements are 0.4 ms and 0.3 ms, respectively. Additionally, the network model suggested in this research has the fewest parameters when compared to other network types, making it more suitable for application situations with constrained hardware platform resources.

4.6. Ablation Experiment

This chapter is aimed at realizing the detection of key parts of moving ship targets in complex marine environments. The YOLOv11 network proposed a number of module improvements in order to verify the value of the role of each module, and ablation experiments were used to verify the impact of the various improvement programs on the detection results. The experimental environment and experimental parameter settings are the same as in the previous experiment, and the comparative network performance is

shown in Table 6. '×' in the table indicates that the module is not used in the network, and '√' indicates that the module is used.

Table 6. Comparison of ablation experiment results.

C3k2_DCNv4	Bi-AVB-FPN	GSConv	Precision (%)	Recall (%)	mAP50 (%)	Parameters	Times (ms)
×	×	×	65.3	82.1	80.8	2,583,127	1.4
√	×	×	68.6	87.4	83.9	2,571,439	1.0
√	√	×	70.3	86.7	85.2	2,527,319	0.9
√	√	√	70.6	87.7	85.7	2,437,559	1.1

As can be seen from Table 6, the detection accuracy of the unimproved YOLOv11 network is 80.8%, and the time for detecting a single image is 1.4 ms. After applying the C3k2_DCNv4 structure, the average detection accuracy of the network is improved by 3.1%, and the detection time is reduced by 0.4 ms. This is due to the fact that a new block based on the reconfiguration of the DCNv4 is utilized in the C3k2_DCNv4 structure to replace the original bottomleneck block in C3k2 in the YOLOv11 backbone network, which can enhance the network's ability to counteract the interference situation of geometric deformation generated by motion, thus enhancing the network's ability to accurately estimate the spatial deformation changes of the moving ship target. This can effectively improve the network's ability to adapt to the geometric changes caused by the motion of the marine ship target and can estimate the spatial deformation of the moving ship target more accurately. The use of the DCNv4 can streamline the training process and reduce the model training parameters, so the detection accuracy and detection efficiency of the network are significantly improved after applying the C3k2_DCNv4 structure. After applying the multi-scale feature pyramid network (Bi-AVB-FPN), the detection accuracy of the network is further improved by 1.3%, and the detection time is reduced by 0.1 ms. The Bi-AVB-FPN network is able to effectively enhance the neck network of the model and enrich the fusion of target feature information, which improves the network's ability to detect small targets and is more helpful in locating the target area in the key vital parts of the ship; in the Bi-AVB-FPN network, the idea of a CSPNet (cross-stage partial network) network is invoked, and the fusion and separation of features using the method of cross-stage local connection effectively reduces the computational redundancy, reduces the number of parameters of the model, and speeds up the detection rate while maintaining the richness of the feature expression. Finally, a lightweight GSConv convolution module is introduced in the neck part of the model to reduce the computation and memory loss. In order to balance the relationship between computational cost and detection performance, the number of parameters of the network model is further reduced, while the detection accuracy is improved by 0.5% and the detection time of the network is 1.1 ms—which is an increase of 0.2 ms—mainly due to the fact that after the improvement of the model has increased the complexity of the model to a certain degree, so the detection time is faster. The main reason is that while the detection time increases as a result of the model's increased complexity, the total number of network model parameters is effectively reduced, allowing the network to better adapt to scenarios with limited hardware conditions. Additionally, the network detection accuracy increases by 4.9% when compared to the time when it was not improved, and the performance of detecting small targets in the key parts is improved. Additionally, the detection time is reduced by 0.3 ms, further improving the network detection rate.

The detection effect of different modules of the network is shown in Figure 16.



Figure 16. Detection effect of different modules.

It can be seen that after continuous improvement of the network detection effect continues to improve, detection accuracy is higher, the leakage rate is lower, and the final optimized network can completely detect the key parts of the ship, and the detection accuracy and the size and location of the encircling frame are closer to the actual target features.

However, in the images of some test results, it is found that it is difficult to distinguish between the passenger cabin and the cockpit of a passenger ship with very similar characteristics, as shown in Figure 17.

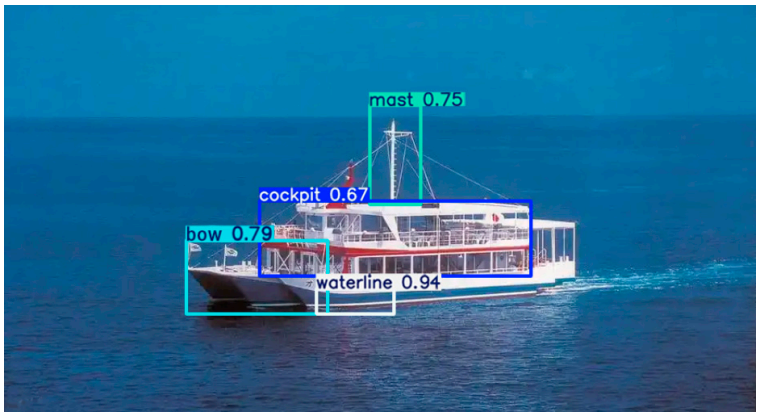


Figure 17. Bad detection result.

The network boxes of the entire cockpit and passenger compartment area as the cockpit can be seen in Figure 17.

5. Conclusions

The thesis aims to improve the accurate real-time detection of key parts of ships at sea. Because the existing algorithms of dynamic target scale change robustness is not strong, detection accuracy and speed are difficult to balance when constructing the YOLOv11-based detection algorithms of key parts of the ship. Self-constructed visible light datasets were used to train the network and design the comparison experiments and the ablation experiments to prove that the optimized network model has the ability to achieve accurate real-time detection of key parts of ship targets in the complicated scenario. The optimized network model is able to achieve accurate real-time detection of key parts of ship targets in complex marine environments. The detection accuracy of the algorithm is improved from 80.8% to 85.7% compared to the network before improvement; the detection accuracy is improved by 4.9%; the detection time of a single pair of images is reduced from 1.4 ms to 1.1 ms; and the number of network parameters is reduced, which effectively simplifies the network model and provides a lightweight and convenient condition for the use of intelligent terminals such as unmanned aerial vehicles.

The next step is to carry out research on the application of the network in real scenarios to improve adaptability to complex environments and to consider exploring the acquisition and interpretation of all-time maritime information in combination with infrared images.

Author Contributions: Y.W.: responsible for resources, formal analysis, program compilation and writing original draft. Y.J.: responsible for writing—review and editing. H.X.: responsible for methodology, project administration. Y.J.: responsible for investigation and resources. C.X.: responsible for resources, formal analysis. K.Z.: responsible for resources, formal analysis. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The original contributions presented in this study are included in the article.

Acknowledgments: Here I would like to express my gratitude to Zhu Pingyun, my advisor, for his control and guidance of the overall idea of my article during the research period. At the same time, I would like to thank Xiao Chuanliang for his careful teaching and guidance on the structure of the article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Gao, F.; He, Y.; Wang, J.; Hussain, A.; Zhou, H. Anchor-free convolutional network with dense attention feature aggregation for ship detection in SAR images. *Remote Sens.* **2020**, *12*, 2619. [CrossRef]
2. Bai, Y.; Chi, W.K.; Xie, B.R.; Zhang, L.; Zheng, L.Y.; Mu, W.T. Infrared ship detection by combining target extraction and deep learning. *Telecommun. Eng.* **2023**, *63*, 193–198.
3. Tang, G.; Liu, S.; Fujino, I.; Claramunt, C.; Wang, Y.; Men, S. H-YOLO: A single-shot ship detection approach based on region of interest preselected network. *Remote Sens.* **2020**, *12*, 4192. [CrossRef]
4. Liu, X.; Piao, Y.; Zheng, L.; Xu, W.; Ji, H. Ship detection for complex scene images of space optical remote sensing. *Opt. Precis. Eng.* **2023**, *31*, 892–904. [CrossRef]
5. Lin, X.W.; Xu, Z.J.; Huang, H. Multi-scale ship detection in SAR image with complex background. *Navig. China* **2023**, *46*, 17–24.
6. Wang, X.K.; Jiang, H.X.; Lin, K.Y. Remote sensing image ship detection based on modified YOLO algorithm. *J. Beijing Univ. Aeronaut. Astronaut.* **2020**, *46*, 1184–1191. (In Chinese)
7. Saehyun, A.; Jung-Woo, C.; Suk-Ju, K. An efficient accelerator design methodology for deformable convolutional networks. In Proceedings of the 2020 IEEE International Conference on Image Processing (ICIP), Online, 25–28 October 2020; pp. 3075–3079.
8. Xiong, Y.; Li, Z.; Chen, Y.; Wang, F.; Zhu, X.; Luo, J.; Wang, W.; Lu, T.; Li, H.; Qiao, Y.; et al. Efficient Deformable ConvNets: Rethinking Dynamic and Sparse Operator for Vision Applications. In Proceedings of the CVPR 2024, Seattle, WA, USA, 17–21 June 2024; pp. 5652–5659.
9. Tan, M.; Pang, R.; Le, Q.V. EfficientDet: Scalable and Efficient Object Detection. In Proceedings of the CVPR 2020, Seattle, WA, USA, 16–18 June 2020; pp. 10781–10788.

10. Zhang, X.; Song, Y.; Song, T.; Yang, D.; Ye, Y.; Zhou, J.; Zhang, L. AKConv: Convolutional Kernel with Arbitrary Sampled Shapes and Arbitrary Number of Parameters. *arXiv* **2023**, arXiv:2311_11587v2.
11. Li, H.; Li, J.; Wei, H.; Liu, Z.; Zhan, Z.; Ren, Q. Slim-neck by GSConv: A better design paradigm of detector architectures for autonomous vehicles. *arXiv* **2022**, arXiv:2206.02424v2.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.