Journal of *Marine Science and Engineering*

MDPI

Article

A YOLOv7-Based Method for Ship Detection in Videos of Drones

Quanzheng Wang [1], Jingheng Wang [2], Xiaoyuan Wang [1,3,*], Luyao Wu [1], Kai Feng [1] and Gang Wang [1]

[1] College of Electromechanical Engineering, Qingdao University of Science and Technology, Qingdao 266000, China; 0020030005@mails.qust.edu.cn (Q.W.); 2022030030@mails.qust.edu.cn (L.W.); isfengkai@163.com (K.F.); joekwang@163.com (G.W.)
[2] Department of Mathematics, Ohio State University, Columbus, OH 43220, USA; wang.14053@osu.edu
[3] Intelligent Shipping Technology Innovation and Comprehensive Experimental Base, Qingdao 266000, China
* Correspondence: wangxiaoyuan@qust.edu.cn

Abstract: With the rapid development of the shipping industry, the number of ships is continuously increasing, and maritime accidents happen frequently. In recent years, computer vision and drone flight control technology have continuously developed, making drones widely used in related fields such as maritime target detection. Compared to the cameras fixed on ships, a greater flexibility and a wider field of view is provided by cameras equipped on drones. However, there are still some challenges in high-altitude detection with drones. Firstly, from a top-down view, the shapes of ships are very different from ordinary views. Secondly, it is difficult to achieve faster detection speeds because of limited computing resources. To solve these problems, we propose YOLOv7-DyGSConv, a deep learning-based model for detecting ships in real-time videos captured by drones. The model is built on YOLOv7 with an attention mechanism, which enhances the ability to capture targets. Furthermore, the Conv in the Neck of the YOLOv7 model is replaced with the GSConv, which reduces the complexity of the model and improves the detection speed and detection accuracy. In addition, to compensate for the scarcity of ship datasets in top-down views, a ship detection dataset containing 2842 images taken by drones or with a top-down view is constructed in the research. We conducted experiments on our dataset, and the results showed that the proposed model reduced the parameters by 16.2%, the detection accuracy increased by 3.4%, and the detection speed increased by 13.3% compared with YOLOv7.

Keywords: ship detection; drones; YOLOv7; object detection; deep learning

Citation: Wang, Q.; Wang, J.; Wang, X.; Wu, L.; Feng, K.; Wang, G. A YOLOv7-Based Method for Ship Detection in Videos of Drones. *J. Mar. Sci. Eng.* **2024**, *12*, 1180. https://doi.org/10.3390/jmse12071180

Academic Editor: Sergei Chernyi

Received: 12 May 2024
Revised: 11 July 2024
Accepted: 12 July 2024
Published: 14 July 2024

1. Introduction

With the rapid development of the shipping industry, an increasing number of ships are being produced and put into operation, leading to the frequent occurrence of maritime accidents. Ship accidents not only threaten the safety of people but also bring huge economic losses. It is understood that the majority of ship accidents are ship collisions [1], so real-time detecting of other vessels around the ship is very important. However, there are many problems in using cameras fixed on ships for ship detection. Ship detection becomes difficult because of the shading between ships, and the entire navigational environment of a ship cannot be accurately obtained by the cameras fixed on the ship. Therefore, a broader view is urgently needed for ship detection.

In recent years, computer vision and drone flight control technology have continuously developed, making drones widely used in related fields such as maritime target detection. This method can not only detect other vessels around the ship in real time but also patrol the navigation route and assist in berthing and unberthing. Compared to cameras fixed on ships, a greater flexibility and a wider field of view are provided by drones with cameras. In the process of unmanned aerial vehicles accompanying intelligent ship navigation, the reduction in computational burden will directly significantly reduce

the energy consumption of the calculation process, which is crucial for current application scenarios. Therefore, how to improve detection accuracy while considering computational burden and through comprehensive consideration of the relationship between the two, while ensuring a certain level of accuracy to improve computational efficiency and reduce device power consumption is the key problem to be solved in this article.

There is a difference in angle between the drone camera and the camera on the ship. The shooting angle of the drone is a top-down view. From this view, the shape of the ship is very different from the ordinary perspective, and other floating objects at sea can be easily detected as ships. Moreover, the ships in the drone's videos will take on different sizes and shapes as the drone flies at different altitudes and the ship moves. In addition to the above situations, the speed of real-time detection is also affected by the complexity of the model. The computing power for real-time detecting is limited, and if the model is complex, the detection speed will be greatly reduced. Therefore, a lightweight ship detection model that is fast and accurate is needed.

With the rapid development of deep learning technology, YOLO series are widely applied to target detection. In this research, an improved YOLOv7 [2] model is proposed, which is called YOLOv7-DyGSConv. The proposed model provides higher detection accuracy, faster speed, and is more lightweight than the traditional YOLOv7, which makes it an ideal model for ship detection in real-time drone-captured videos. The structure of this model is shown in Figure 1. The DyHead attention mechanism [3] is added to the original model to enhance the network's ability to capture targets. Then, GSConv [4] is used to replace the Conv in the Neck part, which can reduce the complexity of the model and maintain the detection speed and accuracy of the mode. In addition, a ship detection dataset containing 2842 images captured by drones or similar drone views is constructed. Experiments are conducted on our dataset, and the results show that compared with YOLOv7, the proposed model reduces the parameters of the proposed model by 16.2%, the detection accuracy is increased by 3.4%, and the detection speed is increased by 13.3%. Self built data were hosted in Gitee, and the warehouse link is as follows: https://gitee.com/piky-00/yolov7-dygsconv.git (accessed on 12 May 2024).

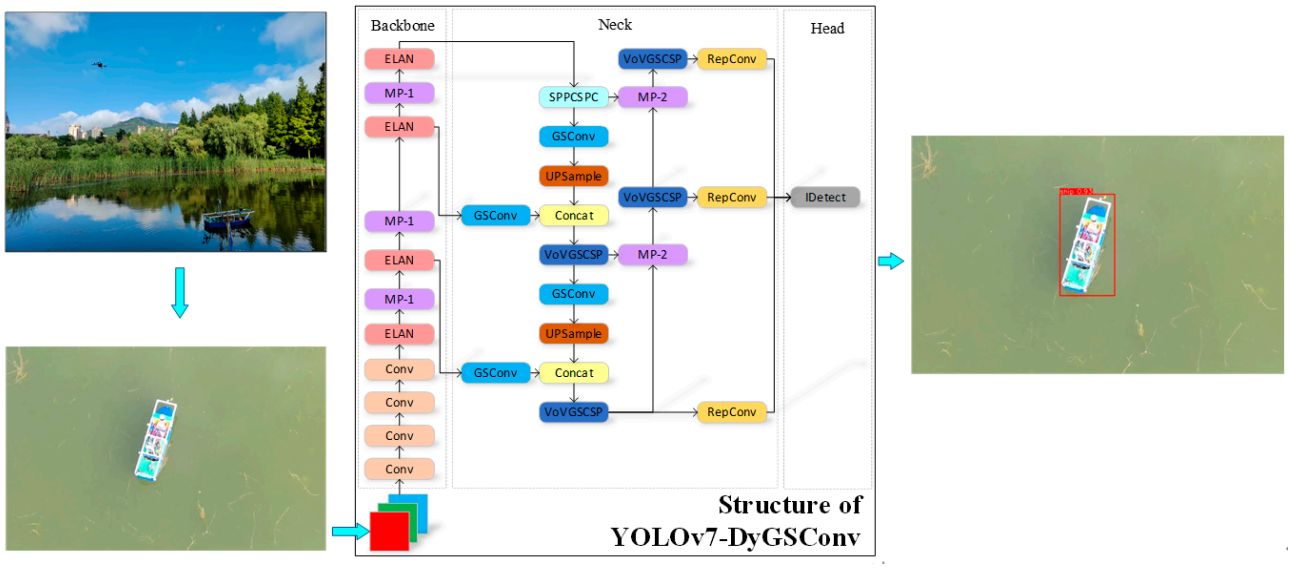


Figure 1. The framework of YOLOv7-DyGSConv.

Before deep learning was widely applied, feature detection, a more traditional machine vision method, is mainly utilized in target detection, such as the Sobel operator, the Canny operator, and other edge detection algorithms. Considering the high complexity of the above algorithms and the poor robustness of the designed feature extractor, they have good performance across specific scenarios. But the target detection tasks and requirements in

complex scenarios cannot be met with these approaches [5]. Arshad et al. [6] performed ship detection based on the Sobel edge detection operator and morphological operations, but this algorithm cannot be used in scenes where the background changes in real time. Schwegmann et al. [7] used Haar-Like feature extraction to extract ship features and output ship detection through adaptive cascade AdaBoost classifier training. Guanyu Chen [8] for obstacle detection in complex backgrounds, through the improved GrabCu algorithm that enhances the details of the segmented image to segment the target and the background, then uses HOG features for feature extraction of obstacles, and finally use SVMs to identify and classify the obstacles. The detection rate of this method is improved over the previous algorithm, but it still fails to detect objects with occlusions. Kun Yang et al. [9] used the Gaussian modeling algorithm to obtain the video background in a static environment and employed the differential mean method to realize motion target detection and obstacle tracking. This method can eliminate the dynamic noise formed by camera shake and leaves moving with the wind in static environments, but it is not suitable for complex dynamic environments.

Since the information in images cannot be completely extracted by the feature extraction methods in traditional image processing, and the robustness of the traditional methods is not strong, coupled with the development of deep learning, deep learning models are gradually applied to target detection. Compared with traditional feature extraction, the features of the target image extracted by the deep learning model constructed by neural networks are characterized by deep feature abstraction and strong feature capability expression [10]. Convolutional neural network models are more mainstream in deep learning models due to their good feature extraction and generalization capabilities [11]. Convolutional neural networks [12] were originally proposed in 1962 by Hubel et al. in a study on the cat visual cortex, which is a type of feed-forward neural network. In 2014, Ross Girshick et al. [13] applied a convolutional neural network to target detection and proposed the famous region-based target detection algorithm R-CNN. The feature extraction layer and feature mapping layer are generally included in convolutional neural networks, and the secondary feature extraction structure of local feature extraction is used behind the convolutional layer. When an image is input into a convolutional neural network, the network can compute directly on the image, which greatly improves the algorithm's performance [14]. From an overall point of view, compared with traditional algorithms, both detection accuracy and detection speed are surpassed by object detection algorithms based on convolutional neural networks.

In recent years, widely used target detections algorithms based on convolutional neural networks have been mainly categorized as One-stage and Two-stage [15]. Two-stage algorithms are based on the basic idea of region detection, first extracting the feature information of the candidate box and then classifying and performing positional regression on candidate frames, such as R-CNN [13] and Faster R-CNN [16]. The One-stage algorithm is based on the idea of regression to retrieve the confidence level and position coordinates of the category of the detection target, such as SSD [17], YOLOv1 [18], YOLOv2 [19], and YOLOv3 [20]. Therefore, the One-stage algorithm has a more concise structure and faster detection speed compared with the Two-stage algorithm [21].

Many scholars have widely applied convolutional neural networks to water surface target detection. In order to overcome the problem that existing ship target detection models often missed detection in complex marine environments, Jiang et al. [22] inserted the improved CA-M in the backbone of the YOLOv7-Tiny model, embedded ODconv into the ELAN, and introduced content-aware feature reorganization (CARAFE) and SIoU into the model. Chen et al. [23] proposed the CSD-YOLO model to perform multi-scale ship object detection operations in SAR images of complex scenes. In addition, El-Gayar et al. [24] addressed the limitations of existing comparative tools and delivered a generalized criterion to determine beforehand the level of efficiency expected from a matching algorithm given the type of images evaluated, which provide inspiration for our research. Zhang et al. [25] proposed an R-CNN method for detecting ships from high-resolution remotely sensed

images in response to the poor performance of traditional SAR image-based ship detection methods in detecting small and aggregated ships. Aiming at the problem that water surface images have large highlight areas caused by reflections, which leads to failure of obstacle detection, Haodong Xu [26] proposed a simple method for segmenting water surface areas and non-water surface areas and derived a formula for calculating the specular reflection component in the highlight pixel points, which removes specular reflections from the water surface images and then detects water surface obstacles using a semantic segmentation-based method. Huang et al. [27] proposed a new Ship-YOLOv3 method by changing the network structure of YOLOv3 and reducing some convolution operations. Through different comparative experiments, the detection time of this algorithm is reduced by 6.06 ms, the ship detection precision is improved by 12.5%, and the recall rate is improved by 11.5%. Tianwei Feng [28] presented a binocular vision-based method for detecting and localizing water shorelines and water surface obstacles by surface unmanned boats. This method first extracts the water shoreline through a morphology-based water shoreline detection method and removes land areas beyond the water shoreline to reduce interference and narrow the search range. Then, the obstacles are roughly estimated by FT saliency detection, and the feasibility of the obstacles is improved by an ORB-based feature matching strategy, and the obstacles with high confidence are labeled. Finally, binocular stereovision is used to localize the shoreline and water surface obstacles. Based on deep learning, Jun Li [29] tested multiple combinations of optimization algorithms for situations such as leakage and misdetection, which often occur in ship detection, and finally concluded that the combination algorithm based on YOLOv3, which introduces the Mixup data enhancement method, Attention Module, Residual Connection, CIOU Loss Function, and Bottom-Up Path Enhancement Algorithm, has the best performance effect. For target ship tracking, the accuracy of ship tracking is improved by improving the DeepSort method. Mengyao Gao [30] proposed the YOLOv3 algorithm that improves the DarkNet-53 network structure by borrowing the idea of dense networks and the YOLOv4 algorithm that improves the DSPDarNet53 network based on deep learning for small and dense targets in ships. The experimental results show that the improved YOLOv3 algorithm has a slower detection speed, which affects the detection performance of the model, while the improved YOLOv4 algorithm has a better detection performance. JingYa Duan [31] researched the algorithm to improve the precision of ship target detection and fine classification of ships based on deep learning, and improved DCGAN to make the ship images generated by the algorithm clearer and more realistic, which can improve the accuracy of ship recognition. The dark channel a priori algorithm is also improved to de-fog ship images to improve the recall rate of ship detection. The improved YOLOv2 algorithm for target detection is more robust to complex background interference, ship target multi-scale, and other factors. Finally, the fine classification of ships is realized based on the attention mechanism and cascade classification network. Aiming at the problems of poor robustness, low detection accuracy, and time-consuming traditional ship target detection algorithms on complex inland waterways, ShiRu Sun [32] designed the AE-YOLOv3 ship target detector based on feature attention and feature fusion enhancement technology and used the improved DeepSort tracking algorithm as the ship tracking module. Guowen He built a ship panoramic image system based on machine vision and combined YOLOv4, Deepsort, and other algorithms to realize ship detection, ranging, and tracking to assist ship navigation [33].

Almost all of the above research is based on the camera on the ship, combined with the improved algorithm for ship detection. The research in this paper improves not only the algorithm but also the experimental tool, which uses the high-altitude perspective for ship detection with drones.

2. Proposed Network

There is a difference in angle between the drone camera and the camera on the ship. The shooting angle of the drone is a top-down view. From this view, the shape of the ship is

very different from the ordinary perspective, and other floating objects at sea can be easily detected as ships, which leads to the accuracy of ship detection not being high. Therefore, how to improve the performance of YOLOv7 target detection has become a key issue in ship detection in images taken by drones. It is worth noticing that the size of different types of ships varies greatly, and the ships in the drone's videos will take on different sizes and shapes as the drone flies at different altitudes and the ship moves. In addition, computing power affects the real-time detection effect, which can be solved by making the model lightweight. The above problems undoubtedly make ship detection in drone videos more difficult, so a fast and accurate lightweight detection algorithm is needed.

In this study, we proposed our YOLOv7-DyGSConv model to improve the accuracy and efficiency of YOLOv7 in ship detection and making the model lightweight for real-time detection. DyHead and GSConv are added to YOLOv7, whose network structure consists of four parts: the Input, the Backbone, the Neck, and the Head.

2.1. DyHead Block

The complexity of combining localization and classification in target detection has led to a boom in methods, with various works attempting to improve the performance of the target detection head, at which point DyHead was proposed. Differences in the scale size of the ship and changes in the shape and position of the ship as it moves in the drone perspective can be compensated by DyHead's scale awareness and spatial awareness. In this work, DyHead was added to the head of the YOLOv7 model, enhancing the model's ability to capture targets. The schematic diagram of DyHead is shown in Figure 2. Three different attention mechanisms were included in the above model, with rule awareness, spatial awareness, and task awareness all added simultaneously to various attention mechanisms, and the focus of each attention mechanism is different.

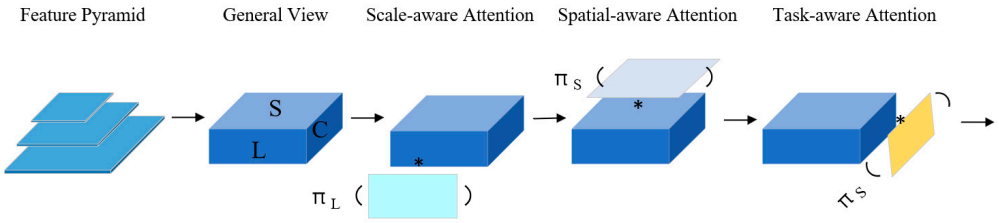


Figure 2. A schematic diagram of DyHead. (The '*' indicates the addition of attention mechanisms corresponding to the image in different dimensions of the feature).

The feature pyramid was rescaled, and its main process can be represented as a three-dimensional vector, $F \in R^{L \times H \times W \times C}$. The number of pyramid layers in the above vector can be represented as "L", "H", "W", and "C", which represent height, width, and the number of channels of feature, respectively. The reshaping tensor, $S = H \times W$, has been redefined by us as a three-dimensional tensor, $F \in R^{L \times S \times C}$. For the feature tensor, the general formula for self-attention is as follows:

$$W(F) = \pi(F) \cdot F \tag{1}$$

where $\pi(.)$ is an attention function. This function is implemented by a fully connected layer, and coupled with the high dimensionality of the tensor, it is computationally prohibitive to learn the attention function in all dimensions. Unlike the previous situation, attention mechanisms are transformed into three different attention mechanisms using different methods. These three attention mechanisms are continuous, and only one angle is focused on in each attention mechanism:

$$W(F) = \pi_C(\pi_S(\pi_L(F) \times F) \times F) \times F \tag{2}$$

where π_C, π_S, and π_L are three different attention functions applying to dimensions "L", "S", and "C", respectively. The formulas for scale-aware attention $\pi_C()$, spatial-aware attention $\pi_S()$, and task-aware attention $\pi_L()$ are shown below:

$$\pi_L(F) \times F = \sigma(f(\frac{1}{SC}\sum_{S,C} F)) \times F \tag{3}$$

where $f(.)$ is a linear function of an approximate 1×1 convolutional layer, and $\sigma(x)$ is a sigmoid function with the expression $\sigma(x) = \max(0, \min(1, \frac{x+1}{2}))$.

$$\pi_S(F) \times F = \frac{1}{L}\sum_{l=1}^{L}\sum_{k=1}^{K} \omega_{l,k} \times F(l; p_k + \Delta p_k; c) \times \Delta m_k \tag{4}$$

In the above equation, K is set as the number of sparse samples; Δp_k is the self-learned spatial offset; $p_k + \Delta p_k$ is the offset position of Δp_k from a discriminant region and Δm_k is a self-learned importance scalar.

$$\pi_C(F) \times F = max(\alpha^1(F) \times F_C + \beta^1(F), \alpha^2(F) \times F_C + \beta^2(F)) \tag{5}$$

where F_C is the feature slice at the c-th channel; $[\alpha^1, \alpha^2, \beta^1, \beta^2]^T = \theta(.)$ is a hyperfunction that learns to control the activation thresholds, which first conducts a global average pooling on $L \times S$ dimensions to reduce the dimensionality, then applies two fully connected layers and a normalization layer, and finally uses a sigmoid function to normalize the output.

Since the above three attention mechanisms are sequential, they can be nested in multiple layers, allowing multiple $\pi_C()$, $\pi_S()$, and $\pi_L()$ to be effectively stacked on top of each other. The structure of the DyHead module is detailed in Figure 3.

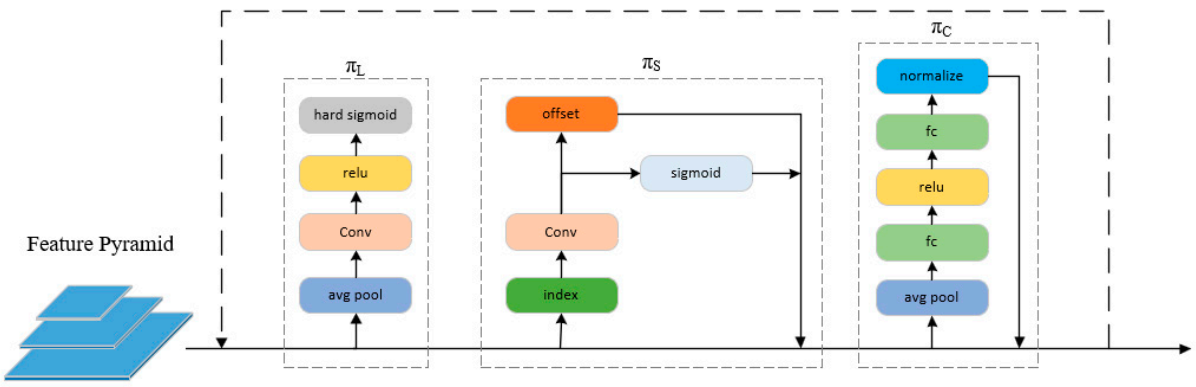


Figure 3. A detailed design of DyHead.

2.2. GSConv Block

YOLOv7 uses a faster convolutional speed and a smaller model compared to YOLOv5, which can achieve higher detection speeds with the same computational resources, so we conducted a series of studies based on YOLOv7. However, it was found in the experiments that the limited computing power of the experimental equipment, coupled with the huge amount of data in the real-time video of the drone, means that YOLOv7 is still not able to meet the detection requirements. Therefore, in this work, YOLOv7 needs to be improved to be lighter to reduce computational costs.

GSConv is a lightweight convolution that reduces the complexity of the model and maintains accuracy. Some lightweight models have used DWConv (Depth Separable Convolution), which can be lightweight by reducing the parameters and floating point operations. In the calculation process, the channel information of the input image is often used separately. The difference between DWConv and SD (Standard Convolution) is shown

in Figure 4. GSConv is a hybrid convolution that combines DWConv with SD. GSConv first performs a down-sampling of an ordinary convolution, then uses DWConv to infiltrate the information generated by the ordinary convolution into each part of the information generated by DWConv, and finally performs a shuffle operation. This operation greatly reduces the negative impacts of DWConv's defects on the model and effectively utilizes the advantages of DWConv to achieve lightweight models. The structure of GSConv is shown in Figure 5.

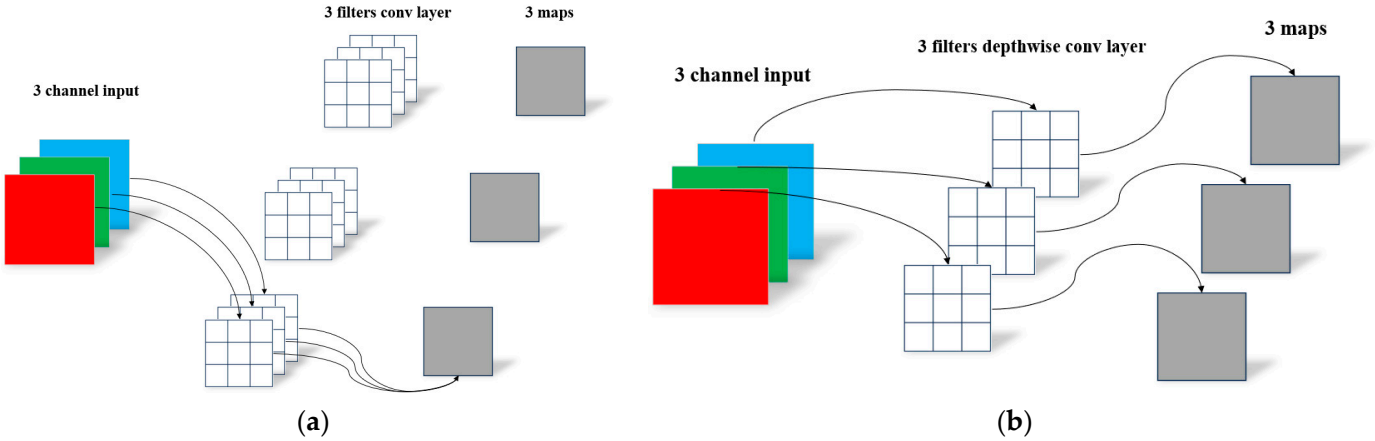


Figure 4. The difference between SD (**a**) and DWConv (**b**).

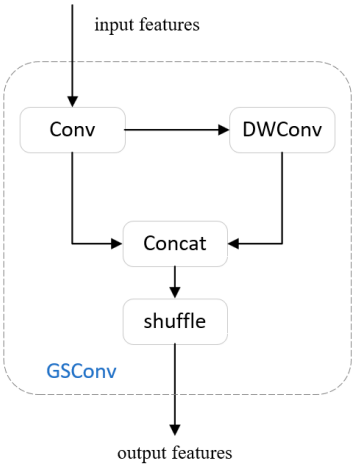


Figure 5. The structure of GSConv.

The cross-stage partial network module VoV-GSCSP is a network module based on the GSConv, which combines enhanced CNN learning capability generalized methods such as DenNet, VoVNet, and CSPNet. The structure of VoV-GSCSP is shown in Figure 6. Flexibly applying this module to the YOLOv7 neck allows us to piece the thin neck structure like building blocks, which can make the model more lightweight and improve the detection accuracy and detection speed.

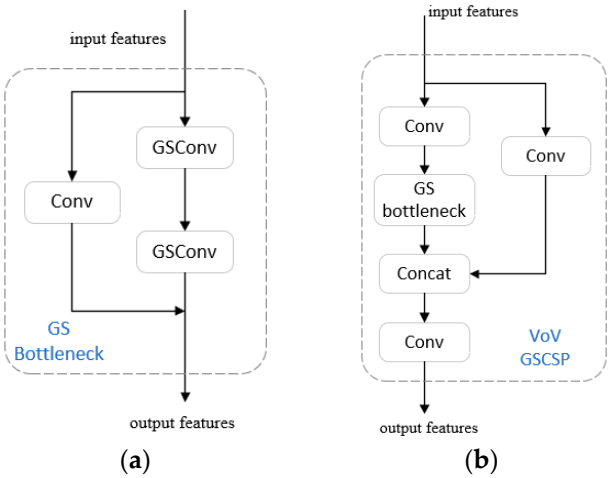


Figure 6. The structures of the GS-Bottleneck (**a**) and VoV-GSCSP (**b**).

3. Experiment

3.1. Experimental Environment and Dataset

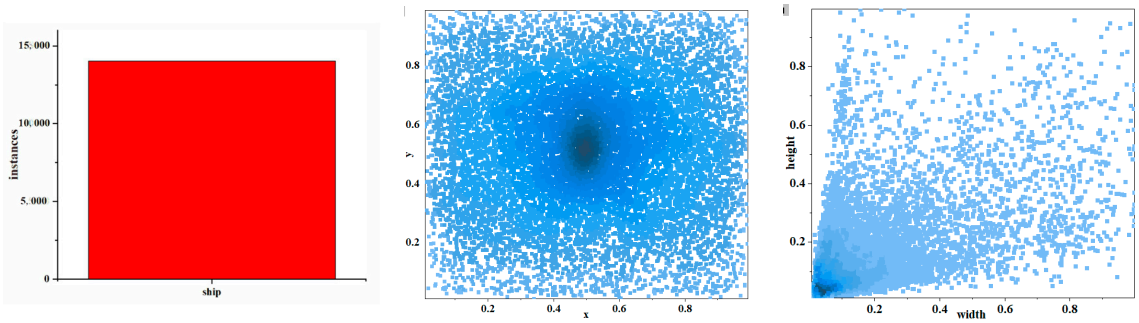
The drone that took pictures at high altitude in our study is the DJI M30. As shown in Figure 7, this drone has a solid structure, a strong power system, and is equipped with some cameras, such as the flight camera and gimbal, to provide stable experimental conditions. The experiment was set up in the sea area near Yinhai Port in Nanqu District, Qingdao, Shandong, China. The drone's altitude was basically maintained between 60 m and 120 m, and the specific altitude was determined by the drone operator according to the actual situation. At the same time, the drone's automatic following function was used to follow the ship's movement, and it sailed at an economic speed of 8 knots. All computers used in the experiment were installed with the Windows 10 system, and a deep learning framework was constructed using the NVIDIA GeForce RTX 3050 GPU. The experiment was conducted using PyTorch tool software. The Python version was 3.7, and the Torch versions were 3.7 and 1.11.0. Only one GPU was used for training and inference. The specific configuration is shown in Table 1. Due to the special angle of the drones overlooking the ship, it is almost impossible to find an existing publicly available ship detection dataset that satisfies the training requirements; therefore, in this work, we collected 2842 images captured by drones or similar drone views and used these images to construct a dataset for detection. Since it is very difficult to categorize the ship meticulously in the drone view, the dataset has only one category, "ship". The images in our dataset have four sources, including the MS-COCO dataset, the VOC dataset, some images from the web, and images captured by drones. Labelimg was used to label the ship targets in the images one by one, and the dataset was divided into training set, validation set, and test set according to the radio of 7:1:2.

Figure 7. A drone used to take pictures at high altitudes.

Table 1. The configuration of the experimental environment.

Parameter	Configuration
Computer operating system	Windows 10
CPU	Intel(R) i5-12400F
GPU	NVIDIA RTX 3050
CUDA	V 1.8.2
Python	V 3.7
Pytorch	V 1.11.0

Our ship dataset visualization of statistical information is shown in Figure 8. As shown in Figure 8a, there are over 10,000 occurrences of ships in our dataset. Figure 8b shows the distribution of these instances, and we can see that most instances are located at the center of the image. Figure 8c evaluates the number of instances of different sizes, and it is evident that small objects occupy the vast majority of our dataset, which is consistent with the characteristics of images captured by drones.



(**a**) (**b**) (**c**)

Figure 8. Visualization of statistical results of our ship datasets. The number of ships is shown in (**a**); the position of the ship's center in the picture is shown in (**b**); and the size of the ship in the picture is shown in (**c**). The darker the color at the midpoint (**b**,**c**), the greater the quantity.

3.2. Metrics

Three important indicators were adopted in the experiment, including model complexity, detection speed, and detection accuracy. GFLOPs (number of floating-point operations per second) and parameters can be used to measure model complexity, and FPS (number of transmitted frames per second) can be used to measure model detection speed. For accuracy, we chose AP (Average Precision) as the metric, and AP is calculated as follows:

In order for objects of the same category to be detected, if the intersection of the predicted bounding box and the exact value exceeds the threshold set in our experiment, the results of the relevant experiment can be considered correct detection. Correctly detected instances are called TPs (True Positives); incorrectly detected instances are called FPs (False Positives); and undetected instances are called FNs (False Negatives). The above parameters give us two metrics named precision and recall. Precision is defined as follows:

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{6}$$

Recall is defined as follows:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{7}$$

Precision represents the percentage of detected instances that are correctly detected, and recall represents the percentage of instances that are correctly detected. With these two

metrics, the P–R curve can be plotted, and the area between the P–R curve and the axes is the AP. Hence, the AP is denoted as follows:

$$\mathrm{AP} = \int_0^1 \mathrm{P(R)dR} \tag{8}$$

In the experiments, AP50:95 was chosen as a measure of detection accuracy, which is the average of AP values at IoU thresholds ranging from 0.5 to 0.95 in steps of 0.05.

3.3. Fusing DyHead Block with YOLOv7

Based on the feature pyramid extracted from the Neck structure of YOLOv7, DyHead further resizes it to the same dimensions to form a 3-dimensional tensor, $F \in R^{L\times S\times C}$, that is then used as an input to DyHead. Next, the DyHead blocks of size-aware, spatial-aware, and task-aware are stacked sequentially.

Due to the sequential application of the three attention mechanisms in DyHead, DyHead can be nested multiple times. In this experiment, the application of different numbers of DyHead blocks in YOLOv7 is experimented with and analyzed. The efficiency of the model by controlling the depth (number of blocks) is evaluated, and their performance and computational costs (GFLOPs) are compered to the baseline (YOLOv7). The input size of the dataset in the model is set to 640 × 640; the number of iterations for model training is 200; and the model's effect is tested on the test set. The number of DyHead blocks is set to 1, 2, 3, and 4, and the performance is shown in Table 2. When the number of DyHead blocks is 1, 2, and 3, the models all have lower computational costs than baseline, but the AP of the models increases only when they contain two DyHead blocks.

Table 2. Model effects of stacking different numbers of DyHead blocks in YOLOv7.

Model	Size (Pixels)	GFLOPs	AP50:95
YOLOv7 (Baseline)	640	103.2	61.8
YOLOv7 + DyHead × 1	640	99.7	61.3
YOLOv7 + DyHead × 2	640	100.9	62.1
YOLOv7 + DyHead × 3	640	102.2	61.0
YOLOv7 + DyHead × 4	640	103.4	61.2

The visualization of the heat map after applying different numbers of DyHead blocks is shown in Figure 9. As shown in Figure 9, before applying the attention mechanism, the model cannot focus on the ship accurately. As more attention DyHead blocks are superimposed in the model, it is obvious that the model can focus on the ship more accurately, which is a good proof of the effectiveness of the attention mechanism. Combining the results from Table 2 and Figure 9, the model containing two DyHead blocks is selected for the subsequent experiments.

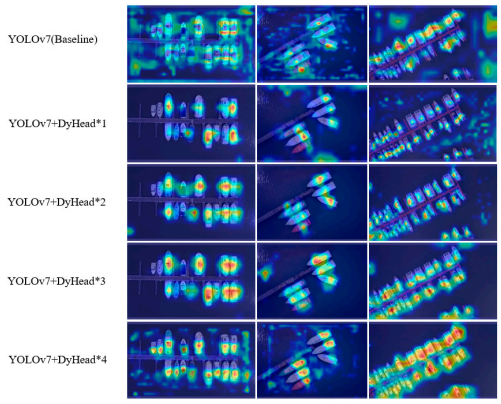


Figure 9. Heat maps of stacking different numbers of DyHead blocks in YOLOv7.

From Table 2, it can be seen that the GFLOP parameter gradually increases with the improvement of the model, and there is no obvious convergence trend. Combined with the parameter AP50:95, it reaches its maximum value at "YOLOv7 + DyHead × 2".

3.4. Fusing GSConv with YOLOv7

In this experiment, we fused our YOLOv7 model with GSConv to reduce the complexity of the model and improve the efficiency of ship detection. In order to obtain the best effect of the GSConv module in the model, the application effect of GSConv in different layers is analyzed. There are four Conv modules in the Neck of YOLOv7, and the original Conv module is replaced by the GSConv module, which is shown in Figure 10. By gradually increasing the number of replacement positions, the replacement position that best meets expectations is determined. The experimental input size is set to 640 × 640. All models were trained 200 times on the training set and received detection results on the test set, as shown in Table 3.

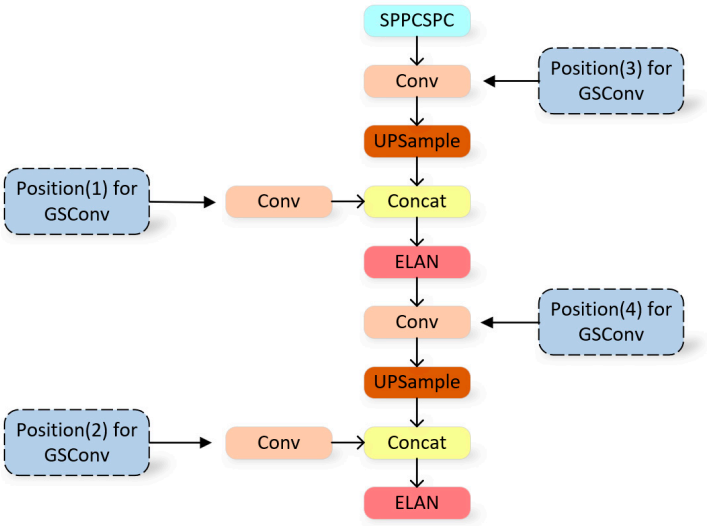


Figure 10. Different positions chosen to implement GSConv in the Neck of YOLOv7.

Table 3. Detection results of replacing Conv with GSConv in different locations.

Model	Size (Pixels)	Parameters	GFLOPs	AP50:95	FPS
Position (1)	640	37 M	103.2	61.8	68.96
Position (1) + (2)	640	35 M	100.1	62.3	71.34
Position (1) + (2) + (3)	640	33 M	96.4	62.9	75.86
Position (1) + (2) + (3) + (4)	640	32 M	93.2	63.2	78.74

Based on continuous attempts, the Conv modules at four positions of the Neck of YOLOv7 are replaced with GSConv modules. The comparison before and after model replacement is shown in Table 4. From Table 4, the parameters of GSConv are half of Conv. In addition, in order to further reduce the inference time and ensure the model's accuracy, the ELAN module in the neck of the YOLOv7 model is replaced by the VoV-GSCSP module. A comparison of the performance of this model with the baseline (YOLOv7) is shown in Table 5.

Table 4. Comparison of the structure and parameters of the model before and after replacing Conv with GSConv in the Neck of YOLOv7.

	Before		After	
Layer	**Module**	**Params**	**Module**	**Params**
...	...	...	...	...
12	Conv	131,584	GSConv	69,248
13	Upsample	0	Upsample	0
14	Conv	262,656	GSConv	134,784
15	Concat	0	Concat	0
16	ELAN	1,264,128	ELAN	1,264,128
17	Conv	33,024	GSConv	18,240
18	Upsample	0	Upsample	0
19	Conv	65,792	GSConv	34,624
...	...	...	...	...

Table 5. Detection results of applying GSConv in the Neck of YOLOv7.

Model	Size (Pixels)	Parameters	GFLOPs	AP50:95	FPS
YOLOv7 (Baseline)	640	37 M	103.2	61.8	68.96
YOLOv7 + GSConv	640	32 M	93.2	63.2	78.74

3.5. Ablation Study

In order to obtain the most qualified model, ablation experiments are conducted based on our established ship dataset. This experiment evaluated the performance of four models: YOLOv7 (Baseline), YOLOv7 + DyHead, YOLOv7 + GSConv, and YOLOv7 + DyHead + GSConv.

Table 6 shows the comparison results of the above four models, and we can see that the three improved models are significantly optimized in terms of their model complexity, detection accuracy, and detection speed compared to baseline. It is worth noticing that YOLOv7 + GSConv has the most obvious improvement in detection speed. Although the model with the added attention mechanism (YOLOv7 + DyHead) is slightly lower than baseline in FPS, we still retain the attention module in light of the findings above. Added with DyHead and GSConv, although YOLOv7-DyGSConv is slightly inferior to the YOLOv7 + GSConv model in detection speed, the overall performance when run on our own dataset is still the best among the models in this study. Compared with YOLOv7, YOLOv7-DyGSConv has a 16.2% reduction in parameters, a 3.4% improvement in detection accuracy, and a 13.3% improvement in detection speed. The comparison between the network structure of YOLOV7-DYGSCONv and that of YOLOv7 is shown in Figure 11.

Table 6. The results of different models in ablation experiments.

Model	Size (Pixels)	Parameters	GFLOPs	AP50:95	FPS
YOLOv7 (Baseline)	640	37 M	103.2	61.8	68.96
YOLOv7 + DyHead	640	35 M	100.1	62.3	71.34
YOLOv7 + GSConv	640	33 M	96.4	62.9	75.86
YOLOv7 + DyHead + GSConv	640	32 M	93.2	63.2	78.74

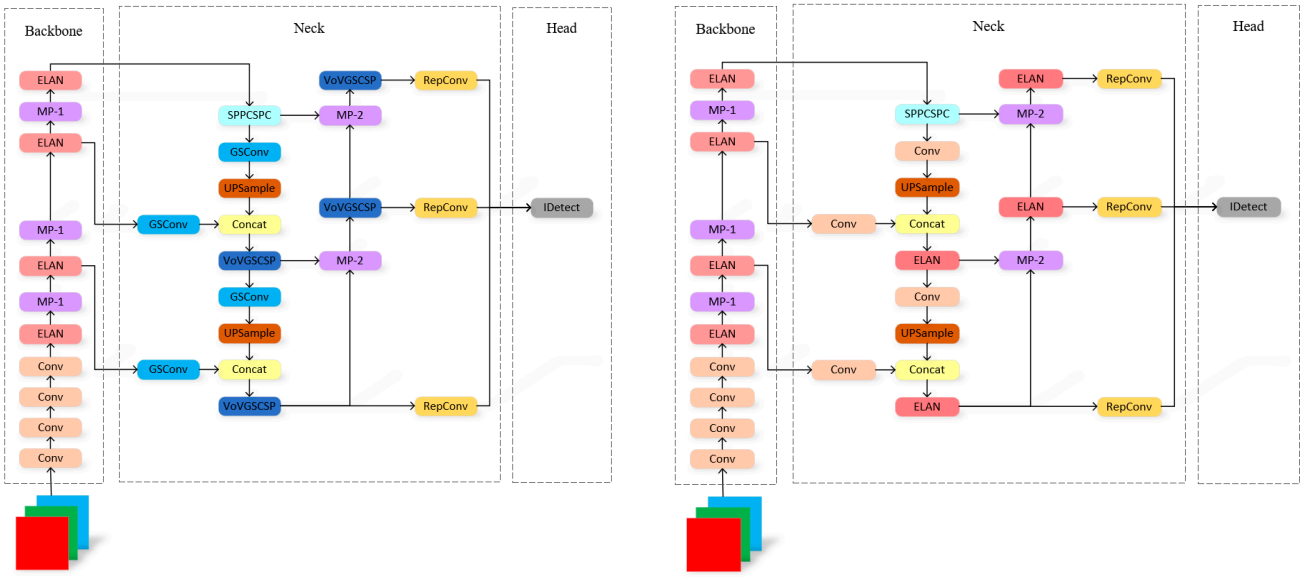


Figure 11. Different positions chosen to implement GSConv in the Neck of YOLOv7. (**a**) YOLOv7-DyGSConv structure diagram; (**b**) YOLOv7 structure diagram.

3.6. Comparisons with the State-of-the-Art

In this experiment, we compared our model with six other state-of-the-art detectors on a self-constructed dataset. The results are presented in Table 7, where the detectors included are YOLOv8l, YOLOv5s-ODConvNeXt [34], YOLOv6-tiny [35], Faster-RCNN [16], EfficientDet-d0 [36], and Scaled-YOLOv4 [37]. It is clear that the proposed model is better than YOLOv8l, YOLOv5s-ODConvNeXt, YOLOv6-tiny, Faster-RCNN, EfficientDet-d0, and Scaled-YOLOv4 in detection accuracy on the self-constructed dataset. Since the proposed model is improved based on YOLOv7, except for YOLOv8l, the parameters of the other five detectors are smaller than the proposed model, but the detection accuracy or detection speed is slightly lower than our model. YOLOv8l is close to our model in detection accuracy, but the number of parameters of YOLOv8l is large, and the detection speed is significantly lower than that of our model. Among these six models, YOLOv5s-ODConvNeXt has the fastest detection speed and the parameters are only 6.99M, but the detection accuracy is 10.5% lower than YOLOv7-DyGSConv. For model size, EfficientDet-d0 has the least number of parameters, but is 46.6% and 27.4% lower than the model in this paper in detection accuracy and detection speed, respectively. Therefore, our model achieves a better balance between detection accuracy, detection speed, and model complexity, has better comprehensive performance, and is more suitable for ship detection based on real-time drone videos. The visualization of ship detection from YOLOv7-DyGSConv is shown in Figure 12.

Table 7. The detection results of the YOLOv7-DyGSCon detector and other detectors on a self-constructed dataset.

Model	Size (Pixels)	Parameters	AP50:95	FPS
YOLOv7 (Baseline)	640	37 M	61.8	68.96
YOLOv7-DyGSConv	640	31 M	63.9	78.13
YOLOv8l	640	43.6 M	62.2	65.36
YOLOv5s-ODConvNeXt	640	6.99 M	57.7	107.53
YOLOv6-tiny	640	14.95 M	56.5	86.96
Faster-RCNN	640	28.68 M	57.2	41.15
EfficientDet-d0	512	3.82 M	43.6	61.35
Scaled-YOLOv4	640	9.11 M	58.3	81.97

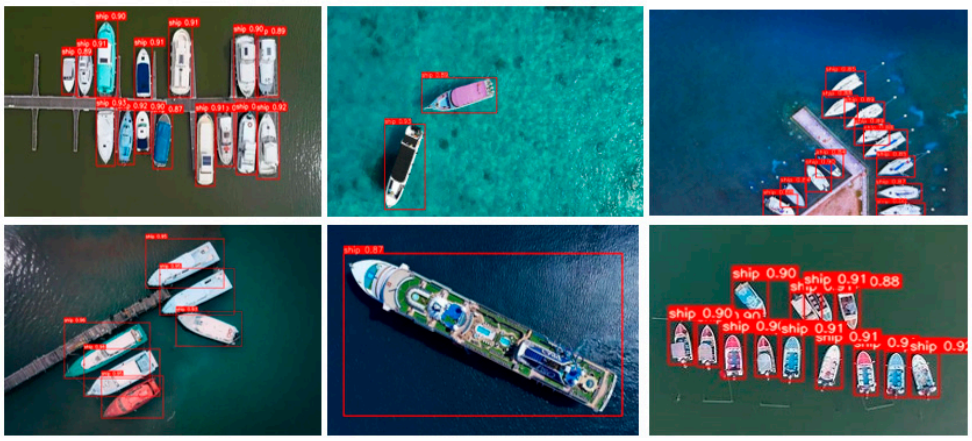


Figure 12. The visualization of ship detection from YOLOv7-DyGSConv.

4. Conclusions

This article mainly focuses on the problem of obstacle ship perception in the autonomous navigation process of intelligent ships and proposes a ship detection model based on improved YOLOv7. The main innovation is the addition of the attention mechanism at the head of the model, which enhances the ability to capture targets that are difficult to detect by radar. At the same time, GSConv is used to reduce the complexity of the model and improve computational efficiency. The main innovation of this article lies in the improved application of the YOLOv7 model to the autonomous navigation of ships, as well as the collection of corresponding datasets and model training.

In this research, an attention mechanism is added to the YOLOv7 model to enhance the network's ability to capture targets. Furthermore, the Conv in the Neck of the YOLOv7 model is replaced with GSConv, which reduces the complexity of the model and ensures detection speed and accuracy. In addition, a self-constructed ship detection dataset containing 2842 images taken by drones or with a drone perspective is constructed in this research. Ablation experiments are conducted on our dataset, and the results show that compared with YOLOv7, the proposed model reduces the parameters of the proposed model is reduced by 16.2%, the detection accuracy is increased by 3.4%, and the detection speed is increased by 13.3%. In the process of an unmanned aerial vehicle (UAV) accompanying intelligent ship navigation, the reduction in computational burden will directly significantly reduce the energy consumption of the computational process, which is crucial for the current application scenario. Therefore, the above results consider the computational load while improving detection accuracy and make a compromise by comprehensively considering the relationship between the two. A better trade-off between detection accuracy, detection speed, and model complexity is achieved by YOLOv7-DyGSConv, which has a better overall performance than the other six state-of-the-art detectors. It is worth noting that currently, in the process of using remote sensing images for ship target detection, a large number of studies have utilized RBox (Rotated Bounding Box) theory for ship target detection [38–41]. In the latest release of YOLOv7, there are optimizations for RBox performance. Therefore, from existing related research, the detection effect of using RBox is roughly the same as the detection effect of the improved YOLOv7 algorithm proposed in this paper.

However, the work proposed in this paper still has room for development. Only the detection of ships is carried out in this article, but in actual navigation, especially in narrow and complex waters, there are not only ships at sea but also obstacles such as reefs and bridge piers. Therefore, the detection target of the model can be enlarged in subsequent work to break through the limitations and cover a wider range of use scenarios. In addition, in the follow-up work, the model can be made to combine with other algorithms to make the maritime work intelligent, such as ship collision avoidance, path planning, maritime patrol, and so on.

Author Contributions: Q.W.: Conceptualization, Methodology, Writing—original draft. J.W.: Formal analysis, Writing—review and editing. X.W.: Conceptualization, Methodology, Supervision, Validation, Funding acquisition. L.W.: Visualization, Software, Data curation. K.F.: Investigation, Data curation. G.W.: Software, Writing—review and editing. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the New Generation Information Technology Innovation Project of the China Ministry of Education's University-Industry Cooperation Fund, grant number 2022IT191; the Qingdao Top Talent Program of Innovation and Entrepreneurship, grant no. 19-3-2-8-zhc; the project "Research and Development of Key Technologies and Systems for Unmanned Navigation of Coastal Ships" of the National Key Research and Development Program, grant no. 2018YFB1601500; the General Project of the Natural Science Foundation of Shandong Province of China, grant no. ZR2020MF082; the Shandong Intelligent Green Manufacturing Technology and Equipment Collaborative Innovation Center; grant No. IGSD-2020-012; and the Graduate Independent Research Innovation Project of Qingdao University of Science and Technology, grant no. B2022KY005.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data will be made available upon reasonable request.

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

References

1. Huang, Y.; Chen, L.; Chen, P.; Negenborn, R.R.; Van Gelder, P.H. Ship collision avoidance methods: State-of-the-art. *Saf. Sci.* **2020**, *121*, 451–473. [CrossRef]
2. Wang, C.Y.; Bochkovskiy, A.; Liao, H.Y.M. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. *arXiv* **2022**, arXiv:2207.02696. [CrossRef]
3. Dai, X.; Chen, Y.; Xiao, B.; Chen, D.; Liu, M.; Yuan, L.; Zhang, L. Dynamic Head: Unifying Object Detection Heads with Attentions. *arXiv* **2021**, arXiv:2106.08322. [CrossRef]
4. 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:2206.02424.
5. Wang, C.; Bu, L. Survey of Object Detection Algorithms Based on Convolutional Neural Networks. *Ship Electron. Eng.* **2021**, *41*, 161–169.
6. Arshad, N.; Moon, K.S.; Kim, J.N. Multiple Ship Detection and Tracking Using Background Registration and Morphological Operations. *Signal Process. Multimed.* **2010**, *2010*, 121–126.
7. Schwegmann, C.P.; Kleynhans, W.; Salmon, B.P. Synthetic Aperture Radar Ship Detection Using Haar-Like Features. *IEEE Geoscience And Remote Sensing Letters* **2017**, *14*, 154–158. [CrossRef]
8. Chen, G. *Research of Obstacle Recognition Algorithm Based on Machine Vision*; Guizhou University: Guiyang, China, 2017.
9. Yang, K.; Huang, L. Dynamic obstacle identification for the moving USV. *Intell. Comput. Appl.* **2019**, *9*, 193–196.
10. Xiong, C. *Research on Ship Target Detection and Tracking Algorithm Based on Deep Learning*; Jimei University: Xiamen, China, 2022.
11. Lin, J.D.; Wu, X.Y.; Chai, Y.; Lin, H.P. Structure Optimization of Convolutional Neural Networks: A Survey. *Acta Autom. Sin.* **2020**, *46*, 24–37.
12. Hubel, D.H.; Wiesel, T.N. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *J. Physiol.* **1962**, *160*, 106–154. [CrossRef]
13. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J.; Berkeley, U.C. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. *IEEE Conf. Comput. Vis. Pattern Recognit.* **2014**, *2014*, 580–587. [CrossRef]
14. Wang, X.L.; Jiang, F.C.; Ning, F.C.; Ma, Q.D.; Zhang, F.; Zou, H.B. Ship Detection with Improved Convolutional Neural Network. *Navig. China* **2018**, *41*, 41–45+51.
15. Li, B.Z.; Wen, Z.J.; Gu, J.J.; Liu, K. Review of Target Detection Algorithms Based on Deep Learning. *Comput. Digit. Eng.* **2022**, *50*, 1010–1017.
16. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [CrossRef] [PubMed]
17. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.; Berg, A.C. SSD: Single Shot MultiBox Detector. In *Computer Vision–ECCV 2016, Proceedings of the 14th European Conference, Amsterdam, The Netherlands, 11–14 October 2016*; Springer International Publishing: Cham, Switzerland, 2016. [CrossRef]
18. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. *arXiv* **2016**, arXiv:1506.02640. [CrossRef]

19. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. YOLO9000: Better, Faster, Stronger. *IEEE Conference on Computer Vision and Pattern Recognition* **2017**, 6517–6525. [CrossRef]
20. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *arXiv* **2018**, arXiv:1804.02767.
21. Yang, Z.; Bu, Z.Y. Improved design on computer vision for small object detection. *Exp. Technol. Manag.* **2022**, *39*, 64–70. [CrossRef]
22. Jiang, Z.; Su, L.; Sun, Y. YOLOv7-Ship: A Lightweight Algorithm for Ship Object Detection in Complex Marine Environments. *J. Mar. Sci. Eng.* **2024**, *12*, 190. [CrossRef]
23. Chen, Z.; Liu, C.; Filaretov, V.F.; Yukhimets, D.A. Multi-Scale Ship Detection Algorithm Based on YOLOv7 for Complex Scene SAR Images. *Remote Sens.* **2023**, *15*, 2071. [CrossRef]
24. El-gayar, M.; Soliman, H.; Meky, N. A comparative study of image low level feature extraction algorithms. *Egypt. Inform. J.* **2013**, *14*, 175–181. [CrossRef]
25. Zhang, S.; Wu, R.; Xu, K.; Wang, J.M.; Sun, W.W. R-CNN-Based Ship Detection from High Resolution Remote Sensing Imagery. *Remote Sens.* **2019**, *11*, 631. [CrossRef]
26. Xv, H.D. *Research and Implementation of Obstacle Detection System Based on Semantic Segmentation for Unmanned Surface Vehicles*; Nanjing Normal University: Nanjing, China, 2020. [CrossRef]
27. Huang, H.; Sun, D.C.; Wang, R.F.; Zhu, C.; Liu, B.Q. Ship Target Detection Based on Improved YOLO Network. *Math. Probl. Eng.* **2020**, *2020*, 6402149. [CrossRef]
28. Feng, T.W. *Obstacle Detection and Positioning for Unmanned Surface Vehicle Based on Binocular Stereo Vision*; Fujian Normal University: Fuzhou, China, 2020. [CrossRef]
29. Li, J. *Research on Surface Ship Detection and Tracking Method Based on Deep Learning*; Harbin Engineering University: Harbin, China, 2021. [CrossRef]
30. Gao, M.Y. *Research on Ship Target Detection in Remote Sensing Image Based on Deep Learning*; North University of China: Taiyuan, China, 2022. [CrossRef]
31. Duan, J.Y. *Research on Ship Recognition Algorithm Based on Deep Learning*; South China University of Technology: Guangzhou, China, 2020. [CrossRef]
32. Sun, S.R. *Research on Ship Target Detection and Tracking Algorithms Based on AE-YOLOv3*; Jiangxi University of Science and Technology: Ganzhou, China, 2022. [CrossRef]
33. He, G.W. *Navigation Active Safety Assistance Technology and System Construction Method for Ships Based on Machine Vision*; Qingdao University of Science and Technology: Qingdao, China, 2022. [CrossRef]
34. Cheng, S.; Zhu, Y.; Wu, S. Deep learning based efficient ship detection from drone-captured images for maritime surveillance. *Ocean Eng.* **2023**, *285*, 115440. [CrossRef]
35. Li, C.; Li, L.; Jiang, H.; Weng, K.; Geng, Y.; Li, L.; Ke, Z.; Li, Q.; Cheng, M.; Nie, W.; et al. YOLOv6: A Single-Stage Object Detection Framework for Industrial Applications. *arXiv* **2022**, arXiv:2209.02976. [CrossRef]
36. Tan, M.; Pang, R.; Le, Q.V. EfficientDet: Scalable and Efficient Object Detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 10778–10787. [CrossRef]
37. Wang, C.Y.; Bochkovskiy, A.; Liao, H.Y.M. Scaled-YOLOv4: Scaling Cross Stage Partial Network. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; pp. 13024–13033. [CrossRef]
38. Li, S.; Zhang, Z.; Li, B.; Li, C. Multiscale Rotated Bounding Box-Based Deep Learning Method for Detecting Ship Targets in Remote Sensing Images. *Sensors* **2018**, *18*, 2702. [CrossRef] [PubMed]
39. Liu, Z.; Wang, H.; Weng, L.; Yang, Y. Ship Rotated Bounding Box Space for Ship Extraction from High-Resolution Optical Satellite Images with Complex Backgrounds. *IEEE Geosci. Remote Sens. Lett.* **2016**, *13*, 1074–1078. [CrossRef]
40. Wang, Y.; Wang, L.; Lu, H.; He, Y. Segmentation based rotated bounding boxes prediction and image synthesizing for object detection of high resolution aerial images. *Neurocomputing* **2020**, *388*, 202–211. [CrossRef]
41. Sun, F.; Li, H.; Liu, Z.; Li, X.; Wu, Z. Arbitrary-angle bounding box based location for object detection in remote sensing image. *Eur. J. Remote Sens.* **2021**, *54*, 102–116. [CrossRef]