Article

A Real-Time Lightweight Detection Algorithm for Deck Crew and the Use of Fishing Nets based on Improved YOLOv5s Network

Wang Jiaming [1,2], Yin Xiangbo [2] and Li Guodong [2,*]

[1] College of Information Technology, Shanghai Ocean University, Shanghai 201306, China
[2] Fishery Machinery and Instrument Research Institute, Chinese Academy of Fishery Sciences, Shanghai 200092, China
* Correspondence: liguodong@fmiri.ac.cn

Abstract: A real-time monitoring system for the operational status of fishing vessels is an essential element for the modernization of the fishing industry. The operational status of fishing vessels can be identified by using onboard cameras to detect the deck crew and the use of fishing ne ts. Due to the typically limited processing capacity of shipboard equipment and the significant memory consumption of detection models, however, general target detection models are unable to perform real-time image detection to identify the operational status of fishing v essels. In this p aper, we propose a lightweight real-time deck crew and the use of fishing nets detection method, YOLOv5s-SGC. It is based on the YOLOv5s model, which uses surveillance cameras to obtain video of fishing vessels operating at sea and enhances the dataset. YOLOv5s-SGC replaces YOLOv5s's backbone and the ordinary convolutional blocks in the feature fusion network with ShufflNetV2 and Ghost module.

Keywords: YOLOv5s; deck crew detection; fishing net detection; deep learning model lightweight

1. Introduction

China is a vast maritime nation with an abundance of marine resources, and the total value of fishery production plays a crucial role in the agricultural sector[1]. The significance of the total value of fisheries production to the agricultural sector cannot be overstated. At present, the marine economy has become one of the favorable engines of the national economy, but the rapid growth of the market demand for aquatic products has caused many fishing vessels to operate illegally during the fishing moratorium and in foreign waters, which has damaged the ecological balance of the sea and seriously undermined the sustainable and healthy development of the ecological environment of fishing waters[2]. Some fishermen in the coastal areas have been working illegally during the closed season. In addition, they are influenced by the traditional idea of " Living by the Sea" and exploit the loopholes of supervision, with "vessels with different certificates", illegal new "three-noes" vessels, and illegal fishing at sea. This is especially true during the fishing moratorium, which can severely disrupt the order of fisheries production and harm the ecological equilibrium of the ocean[3]. In recent years, as the number of fishing vessels and the degree of mechanization and automation of fishing vessels have increased, so has the intensity of fishing, and irresponsible fishing practices have caused a severe decline in marine fishery resources and a crisis in marine fishery production[4,5]. To advance the modernization of the fishing industry and improve the intelligence and information technology on fishing vessels, it is necessary to identify the operational status of fishing vessels at sea. China's maritime rights and interests can be effectively safeguarded by advancing the level of fisheries informatization management construction and eliminating the retrograde development situation[6]. Consequently, it is imperative to investigate onboard camera technology to autonomously identify fishing vessels' operational status. The onboard camera allows for the rapid and accurate identification of deck crew and fishing nets on fishing vessels, which is essential for the automatic identification of the

 doi:10.20944/preprints202305.1115.v1

operating status of fishing vessels. The operational status of fishing vessels is identified based on the identification results of the onboard camera.

Manual detection, satellite monitoring, and shipboard video surveillance are the primary methods for identifying the operational status of a fishing vessel. The manual inspection method is primarily applied by law enforcement officers who board fishing vessels to identify their operational status. This manual judgment method is highly accurate and can promptly find out if a fishing vessel has engaged in illegal fishing in a restricted area. But it also requires a significant amount of human labor, and when there are too many fishing boats, there may not be enough staff to monitor the operational status of all fishing boats. The primary method of satellite detection involves deploying a monitoring system for fishing vessels on the fishing vessel. This system transmits the fishing vessel's current status data to the satellite, which then conveys it to the ground-based base station. However, the data capacity on land is limited and costly, and the real-time efficacy is poor.And for example, the price and communication fee of Inmarsat equipment is high, and the terminal equipment is large, some areas such as China's existence of a large number of small vessels, operating sea area is basically not extensive, poor economic capacity and other characteristics, to promote the application of such fishing vessels in a large area there is a very big difficulty[7]. The application of intelligent fishing vessels is very difficult. Because of the abundant space for the implementation of intelligent transformation of fishing vessels, the artificial intelligence of fishing vessel operation mode identification technology based on shipboard video is considered, combined with multi-source data to achieve the monitoring of fishing gear and fishing methods, and to improve the level of intelligent control of fishing vessel compliance operations. Zhang Jiaze[8] achieved 95.35% accuracy in the behavioral recognition method of the fishing vessel by installing high-definition camera equipment at four fixed locations on a mackerel fishing vessel and building a 3-2D fusion convolutional neural network to extract and classify the behavioral features of the fishing vessel. Shuxian Wang[9] attached cameras, built a convolutional neural network and added pooling layers, LSTM long short-term memory modules, and attention modules on Japanese mackerel fishing boats. In the behavior recognition test set of Japanese mackerel fishing vessels, they received an F1 score of 97.12%.

Deep learning algorithms have shown outstanding performance and promising application prospects in the fields of object detection and recognition[10–15]. In boat-based video identification systems, deep learning algorithms have produced greater results in terms of accuracy and real-time performance. But the following issues continue to exist: the computing platform of the fishing vessel's on-board equipment has limited computing power resources and the operating environment of the fishing vessel is complex, with issues such as light changes and field of view occlusion affecting the final detection results, yet the detection speed of complex models cannot meet the real-time requirements of the task, and the network models are too large to be deployed. Most deep learning algorithms with high accuracy require more model parameters and high computational complexity, requiring high computational power of hardware devices and slow detection speed; while deep learning models with fast detection speed are lacking in accuracy. Not much research has been done on the lightweight detection model for deck crew and the use of fishing nets that balances detection accuracy and detection speed well. To solve this problem, many excellent lightweight and efficient network structures have been proposed such as MobileNet[16–18], EfficientNet[19], PP-LCNet[20], etc. This study proposes a real-time detection algorithm YOLOv5s-SGC based on the YOLOv5s model, using the lightweight network ShuffleNetV2[21] 0.5× replaces the YOLOv5s backbone network CSP-Darknet53[22] to reduce the number of parameters and increase the speed of operation while maintaining accuracy, the general convolution and C3 modules in the feature fusion network were replaced with GSConv and CSP_GSC modules to further reduce the complexity of the model and the number of parameters, and finally the CBAM attention module was introduced in front of the detection layer to strengthen the feature representation capability of the network. The problem of detection accuracy degradation due to the reduced number

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 16 May 2023 doi:10.20944/preprints202305.1115.v1

of parameters is increased at the cost of a small amount of computation. To provide a real-time and effective detection method for deck crew and the use of fishing net detection of fishing vessels, experiments will be conducted on the data set of fishing vessels out at sea operations suggested in this paper, and the detection performance will be compared with other lightweight improvement methods.

2. Dataset

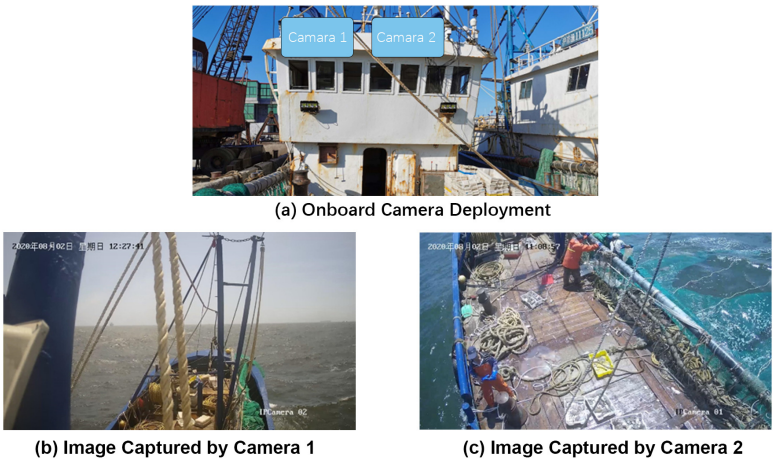
2.1. Fishing Vessel Operations Image Dataset

The dataset used for the experiment is self-built, and the source of the data is the video recordings of fishing vessels during their operations at sea. The video of the fishing vessel's operation for 7 days was taken by mounting a camera on the fishing vessel to capture the situation on the fishing vessel's deck. From it, 773 photos with a resolution size of 1920×1080 were extracted. The table below displays the used video-capturing hardware.

Table 1. Parameters of video capture devices.

Type of Equipment	Equipment Parameters
Gun-type cameras	1080P@22Hz wide-angle recording
Storage devices	2TB internal storage, cycle storage support, timed shooting

This project placed onboard cameras on two trawlers to expand the diversity of the image dataset of fishing vessel operations. Directly on the deck, two gun-mounted cameras were placed, with camera 1 inclined upward to record the activities at the bow and camera 2 angled downward to record the activities at the deck. The collection of images of fishing vessel operations is shown in Figure 1.



(a) Onboard Camera Deployment

(b) Image Captured by Camera 1

(c) Image Captured by Camera 2

Figure 1. Image of fishing vessel operations.

2.2. Data Labeling

The dataset was annotated using the Roboflow annotation tool, dividing the training set, validation set and test set in the ratio of 8:0.5:1.5. The specific information of the dataset is shown in Table 2.

Table 2. Information of the fishing boat dataset.

Category	Tag	Number
Deck Crew	Person	1531
Fishing Nets	Fishing_Net	543

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 16 May 2023 doi:10.20944/preprints202305.1115.v1

2.3. Data Enhancement

This study specifically incorporates some complex environmental factors that may be encountered in the process of simulating fishing boats at sea, such as changes in light and obstruction of vision, in the construction of the dataset to identify the operational status of fishing boats more accurately. Enhancing the dataset can significantly increase the model's resilience and accuracy because these complicated aspects frequently create increased interference with the identification of fishing vessel operating status in actual applications. A total of 2319 data photos were obtained after the dataset was expanded using a combination of Gaussian noise, random rectangular occlusion, random pixel zeroing, pretzel noise, Gaussian blur, motion blur, changing color temperature, and random contrast. Of these, 1857 images were used for the training set, 345 images were used for the test set, and 117 images were used for the validation set. The data enhancement samples are shown in Figure 2.

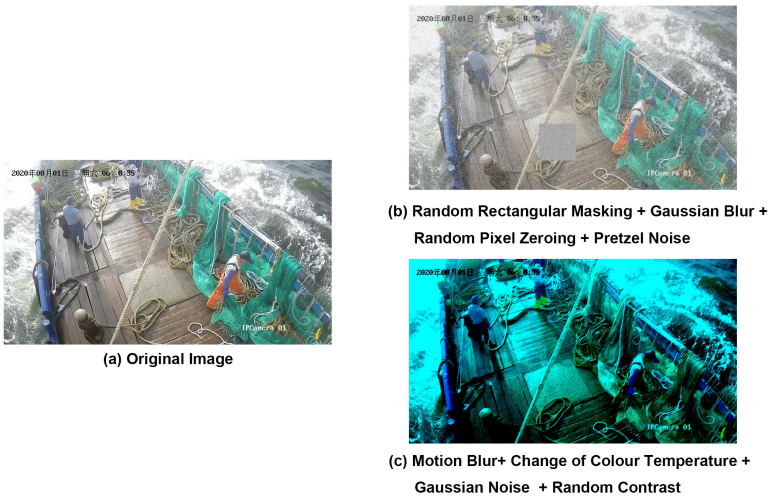


Figure 2. Data enhancement samples.

3. Proposed Methods

3.1. YOLOv5s Model

The YOLOv5 model, the current dominant deep learning framework, can be used for state detection to provide real-time deck crew and the use of fishing net detection on fishing vessels. YOLOv5s, YOLOv5m, YOLOv5l, and YOLOv5x are four variations of a one-stage structural target detection network that are all essentially the same structurally and have networks that go wider and deeper in that order. The YOLOv5s, the lightest model size, and the fastest detection speed is used in this study as the fundamental model due to the low computational power of hardware devices on fishing vessels. The Input, Backbone, Neck, and Head components make up the bulk of the YOLOv5 algorithm.

The input side is enhanced by Mosaic[23] data enhancement, adaptive anchor frame calculation, and adaptive image scaling to increase the richness of the data and improve the generalization ability of the network. Four photos are randomly chosen for operations like rotation and scaling, as well as random cropping, which is then conducted and stitched to create new training data. By doing this, the robustness of the network, the training speed of the model, and the capacity to recognize small targets are all improved.

The components of the backbone network include Focus, CSP-DarkNet53, and SPPF. Focus directly performs a slicing operation on the input image, allowing for more comprehensive feature extraction during the downsampling process[24]. The Cross-Stage Partial network layer structure is used to address the issue of excessive computational complexity during inference; Spatial Pyramid Pooling[25] structure converts feature maps of arbitrary size into fixed-size feature vectors.

 doi:10.20944/preprints202305.1115.v1

The neck network is mostly used to generate feature pyramids to improve the network model's capacity to recognize objects of various scales, making it possible to distinguish between the same object in various sizes. The neck network adopts the FPN+PAN[25] structure, where the FPN fuses deep semantic information and underlying target information through upsampling to enhance the network's learning ability of target features; the PAN conveys wall localization features from the bottom up to improve the network's learning performance of semantic information and localization information. Finally, a multi-scale fusion of features of images is achieved.

YOLOv5 uses GIOU_Loss[26] as the loss function. Compared with IOU, GIOU solves the problem that the loss function is not derivable when the prediction frame and the target frame do not intersect on the one hand and solves the problem that the two prediction frames are of the same size and the IOU is the same resulting in the IOU loss function not being able to distinguish the difference between the intersection of the two prediction frames on the other hand.

The detection side outputs multiple prediction frames and confidence levels for each grid of the fused feature maps at each scale individually. The output side is responsible for the final prediction output, and ultimately three scales of feature maps are obtained from the neck network for monitoring. The deeper feature maps are primarily responsible for the detection of large targets, while the shallower feature maps are primarily responsible for the detection of small targets. A non-maximum suppression[27] technique filters the final prediction frames.

Although YOLOv5 performs admirably on publicly available datasets and is excellent at feature extraction and target detection, there is still room to improve the performance of its model files on devices with limited processing power. Additionally, it frequently confuses the target with the background of the fishing boat when images taken by the onboard camera are dimly lit or have uneven illumination, leading to false alarms. The network can maintain high-precision target detection while reducing the model's weight size and improving its processing speed by making lightweight modifications and incorporating attention modules, As a result, it can be deployed and perform real-time detection of deck crew and fishing nets operations even on low-intelligence fishing vessels.

3.2. The Backbone Based on ShufflenetV2

The sizes of fishing nets and deck crew in the collected fishing vessel operation images in this study belong to large and medium targets in the image. A lightweight backbone network may be utilized to construct the feature extraction network to decrease computation and the number of parameters of the network while maintaining high accuracy, taking into account the low computational power of the onboard hardware devices. To meet the requirements of high accuracy, real-time operation on low computing power devices, and fewer model parameters, the YOLOv5s backbone feature extraction network is replaced by the ShuffleNetV2 network. The authors of ShuffleNetV2 have made four findings that affect the efficiency of the network, namely "Equal channel width minimizes memory access cost(MAC)", "Expensive group convolution increases MAC", "Network fragmentation reduces the degree of parallelism", and "Element-wise operations are non-negligible ". ShuffleNetV2 was proposed based on these four findings. ShuffleNetV2 introduces the concept of "channel split" based on ShuffleNetV1[28], and redesigns the basic structure module into two types: the basic unit and the downsampling unit, as shown in Figure 3.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 16 May 2023 doi:10.20944/preprints202305.1115.v1

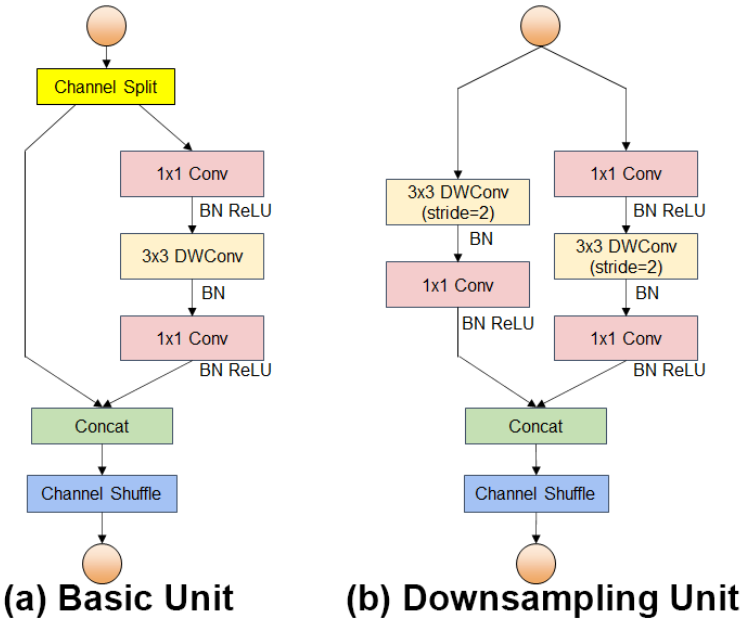


Figure 3. ShuffleNetV2 module composition.

As shown in Figure 3, in the basic unit, the channel split operation is used, so that each basic unit splits the input of the module into two branches, with one branch doing constant mapping and the other branch passing through two ordinary convolutional layers (Conv) with 1×1 convolutional kernels and one deep separable convolutional layer (DWConv) with 3×3 convolutional kernels. This ensures that the number of input channels is the same as the number of output channels while reducing the number of network model parameters and speeding up the network inference. In the downsampling unit, the spatial downsampling unit step is 2, which removes the channel splitting operation, thus making the number of output channels twice the number of input channels. The primary design idea of ShuffleNet is to execute a channel shuffle operation on several channels to address the issue of non-communication between the layers of network feature information produced by group convolution, which leads to a reduction in the capacity to extract network features. The channel shuffle operation is shown in Figure 4.

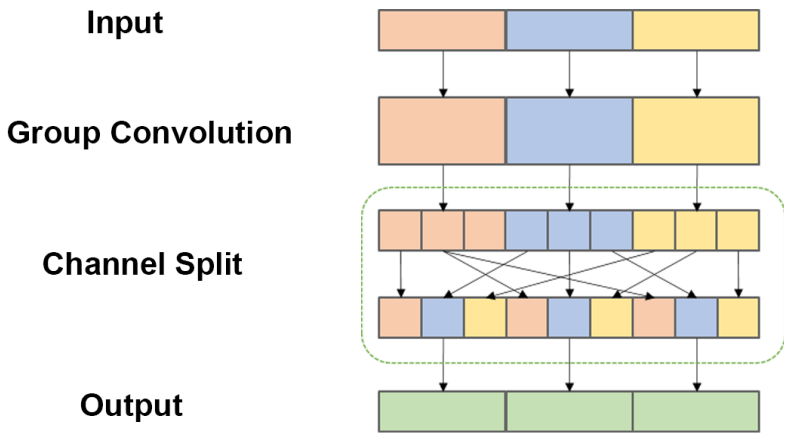


Figure 4. Channel shuffle.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 16 May 2023 doi:10.20944/preprints202305.1115.v1

As shown in Figure 4, the channel shuffle operation ensures that the feature information of each group can communicate with each other by regrouping the feature information of different groups in the output layer, to promote the full flow of information between channels without affecting the accuracy of the network and improve the learning ability of the feature information between groups, thereby reducing the computational effort of the network.

Table 3. Improved backbone network structure.

Layer	Output Size	Kernel Size	Stride	Repeat	Output Channels
Input	640 x 640				
Conv1	320 x 320	3 x 3	2	1	24
Maxpooling	160 x 160	3 x 3	2	1	24
Shuffle_Block	80 x 80		2	1	48
Shuffle_Block	80 x 80		1	3	48
Shuffle_Block	40 x 80		2	1	96
Shuffle_Block	40 x 40		1	7	96
Shuffle_Block	20 x 20		2	1	192
Shuffle_Block	20 x 20		1	3	192

Table 3 shows the improved backbone network, replacing the YOLOv5s backbone network with the ShuffleNetV2 0.5× feature extraction structure, thereby reducing the computational effort and number of parameters, and allowing real-time target detection on fishing vessel hardware with low computing power.

3.3. The Ghost Module

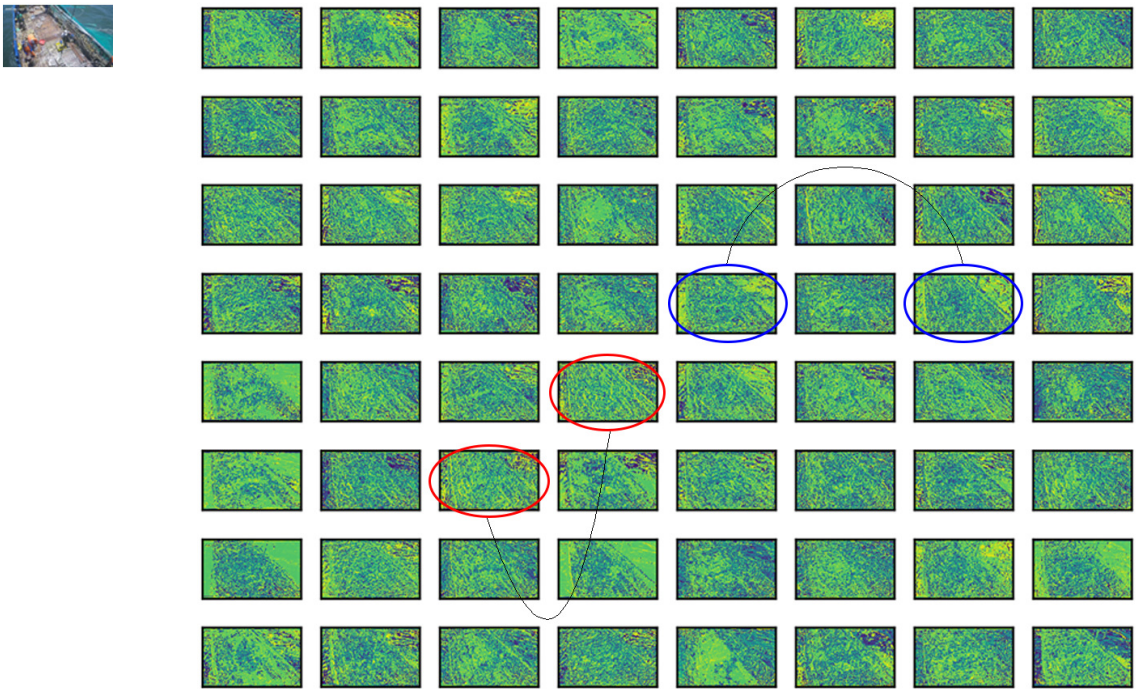


Figure 5. Similar feature maps of YOLOv5s

After the feature extraction network, the feature maps for detecting the deck crew and fishing net need to undergo further processing in the feature fusion network. This is necessary to combine and integrate the image features of the crew and fishing nets extracted

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 16 May 2023 doi:10.20944/preprints202305.1115.v1

by the backbone network and to pass these features to the recognition task to output the identification results of the crew and fishing net. As shown in Figure 5, there will be some photos with high similarity among these feature maps, which is feature map redundancy in the neural network, leading to wasteful computation and parameter increase, from a single image following feature extraction by the neural network. In response to this situation, we introduced GhostNet[29] in the feature fusion section module in GhostNet. The Ghost module is shown in Figure 6.

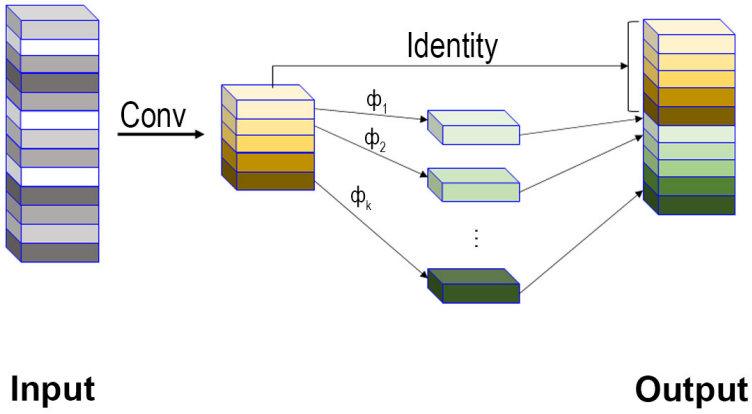


Figure 6. Ghost module.

In the Ghost module, the first step is to generate a part of the feature map by a small amount of traditional convolution, after which a linear operation is performed on these feature maps, and the newly generated feature map becomes the Ghost feature map. Finally, the two parts of the feature map before and after the linear operation are stitched together to produce the final feature map, and this structure is called Ghost convolution GSConv.

If the input size is $h \times w \times c$ of the feature map, the size of the kernel is $k \times k$, the number of channels of the cheap operation transforms is m and the number of transformations is s and the output is n. The width and height of the feature vector channels are w' and h' and $m \ll n$, then it follows that

$$n = m \times s \tag{1}$$

Since there is one constant transformation throughout, the actual number of effective transformations is$(s-1)$, equation (1) is transformed as

$$m \times (s-1) = \frac{n}{s} \times (s-1) \tag{2}$$

Assuming that the linear operation kernel has a mean value of $d \times d$ then the theoretical speedup ratio of Ghost convolution compared to conventional convolution can be calculated.

$$\begin{aligned} r_s &= \frac{nh'w'ckk}{\frac{n}{s}h'w'ckk + (s-1)\frac{n}{s}h'w'dd} \\ &= \frac{ckk}{\frac{1}{s}ckk + \frac{s-1}{s}dd} \approx \frac{sc}{s+c-1} \approx s \end{aligned} \tag{3}$$

It can be seen from the calculation procedure above that linear operations require less computing than conventional convolutional operations. By changing the original operation of generating feature maps using convolutional kernels to retaining only a small number of convolutional kernels and replacing the rest of the convolutional operations with linear operations, the amount of computation and time required to generate feature maps can be significantly reduced[30]. The result is a significant reduction in the amount of computation and time required to generate feature maps.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 16 May 2023 doi:10.20944/preprints202305.1115.v1

The study replaces the normal convolution and C3 modules in the YOLOv5s feature fusion network with the GSConv and CSP_GSC modules to reduce the size of the model. The CSP_GSC module is a replacement of the Conv module of BottleNeck in the C3 module with the GSConv module, so the parameters and computational effort in the CSP_GSC module are reduced. The C3 module and the CSP_GSC module are shown in Figure 7.

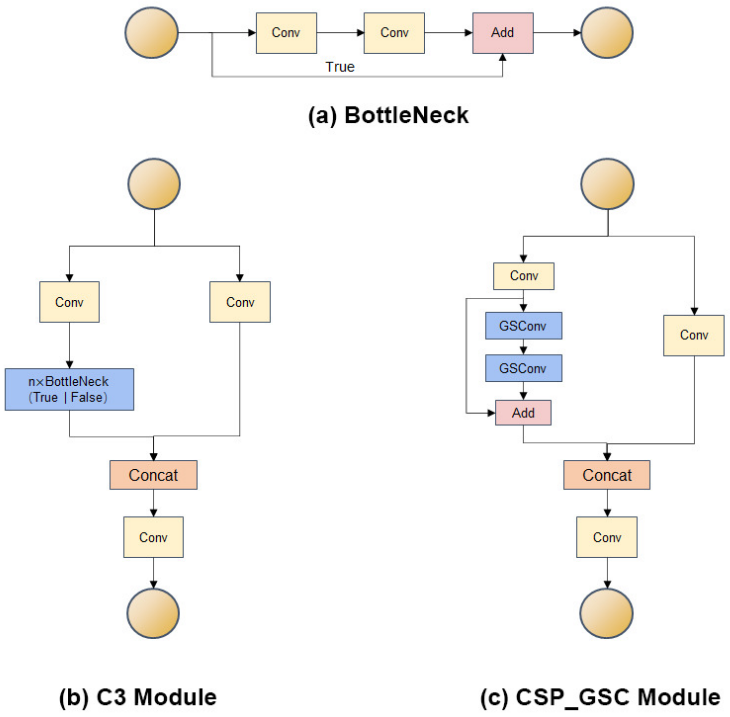


Figure 7. C3 module and CSP_GSC module.

3.4. The CBAM Module

Significantly reducing the number of parameters and computational effort of the backbone feature extraction network and the feature fusion network also reduces the detection accuracy of the model. As shown in Figure 8, the heat map of the dataset reveals that the accuracy of the model's target detection can be enhanced by incorporating an attention mechanism into the network, as the fishing nets are primarily located at the periphery or outside of the deck and the crew is primarily located on the deck.

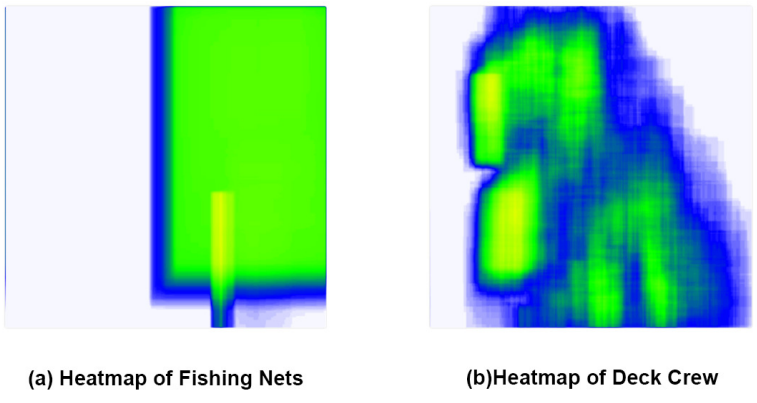


Figure 8. Dataset heatmap.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 16 May 2023 doi:10.20944/preprints202305.1115.v1

This study introduces the CBAM attention module[31] into the feature network to accurately obtain the relative position information of the target in the fishing vessel operation image, and to focus on the feature part of the fishing net and crew, thereby reducing the influence of information unrelated to the target, while also increasing the accuracy of target detection[31]. The CBAM module is an efficient and straightforward attention module for feed-forward convolutional neural networks. Given an intermediate feature map, the CBAM module sequentially infers the attention map along two separate dimensions, channel, and space, and finally multiplies the attention map with the input feature map to perform adaptive feature optimization, thereby enhancing the extraction of important features and suppressing target-irrelevant information. The structure of the CBAM module is shown in Figure 9.

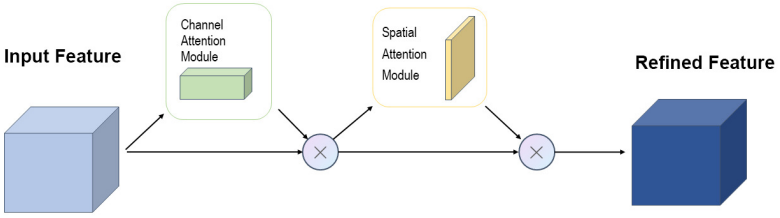


Figure 9. CBAM model structure.

The CBAM module introduces a channel attention module which can improve the model's ability to perceive location information to enhance its recognition effectiveness for precisely locating regions of interest in images of fishing vessel operations. Through average pooling and maximum pooling, the channel attention module derives a one-dimensional vector from the spatially compressed feature map. The focus of this channel is on what is significant on this graph. The channel attention mechanism can be expressed as follows:

$$M_C(F) = \sigma(MLP(AvgPool(F) + MLP(MaxPool(F))) \tag{4}$$

As shown in equation (4), where F is the input feature map; AvgPool denotes the average pooling operation; MaxPool denotes the maximum pooling operation; and MLP denotes the multilayer perceptron; σ is the Sigmoid activation function.

The task-relevant portions of the image contribute more to the outcome than other regions of the image to varied degrees. To increase accuracy, a spatial attention module can be included for the various deck crew and fishing nets positions. The spatial attention module compresses the channels, and in the channel dimension, average pooling and maximum pooling are conducted to compress the channels, respectively. The spatial attention mechanism can be expressed as equation (5), where $f^{7\times7}$ denotes the convolution of a convolution kernel of size 7×7:

$$M_s(F) = \sigma\left(f^{7\times7}(AvgPool(F); MaxPool(F))\right) \tag{5}$$

The final output of the CBAM attention mechanism is expressed as a feature map in equation (6):

$$M(F) = (F \times M_c(F)) \times M_s(F \times M_c(F)) \tag{6}$$

With the backbone feature extraction network replaced with ShuffleNetV2 and the feature fusion network incorporating the GSConv and CSP_GSC modules as well as the CBAM attention module, the structure of the YOLOv5s-SGC model, a lightweight real-time detection algorithm for fishing vessel personnel and retrieval nets based on the improved YOLOv5s network, is shown in Figure 10.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 16 May 2023 doi:10.20944/preprints202305.1115.v1

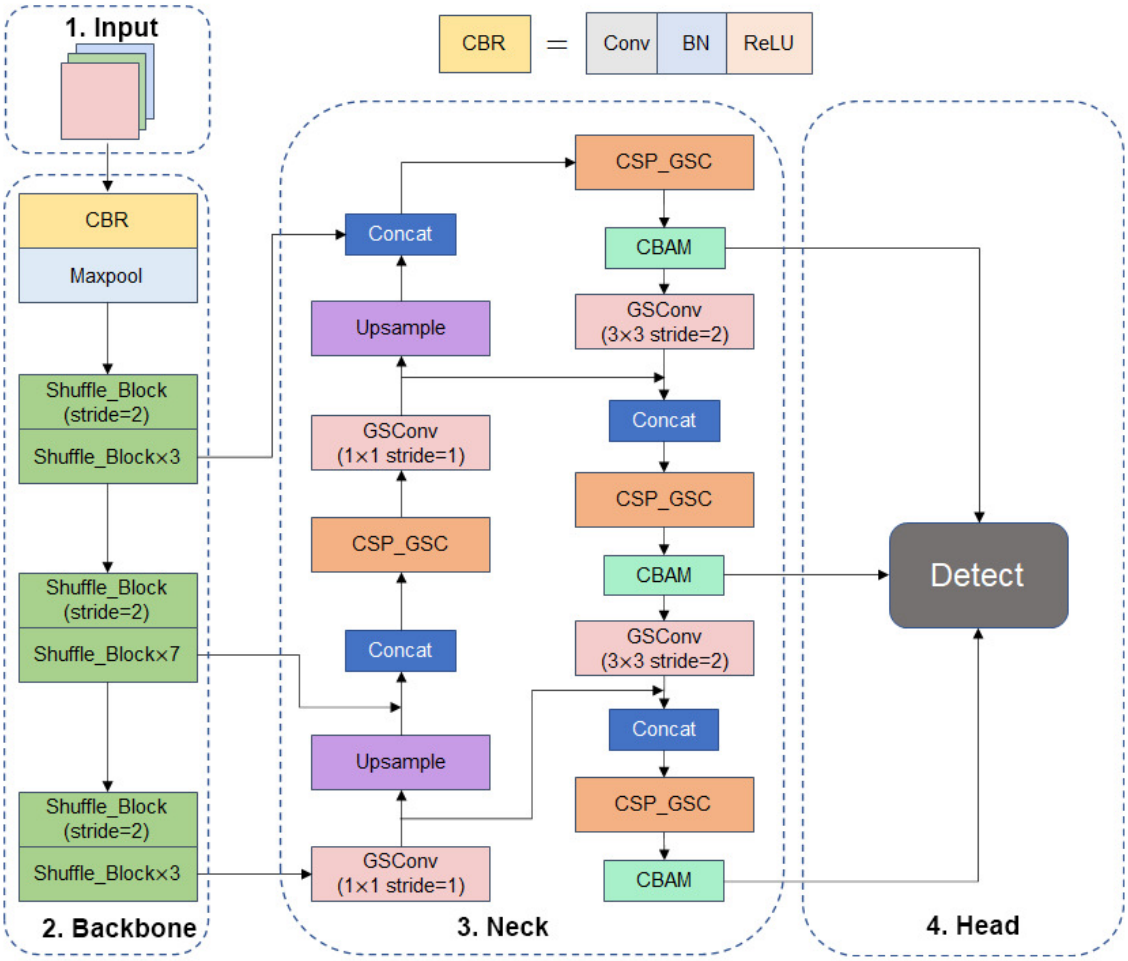


Figure 10. YOLOv5s-SGC model structure

4. Experiments

4.1. Evaluation Metrics

In this paper, we use common evaluation metrics to judge target detection models: Precision, Recall, mAP (mean Average Precision), weight size, FLOPs, and detection speed.

The accuracy rate equals the number of samples correctly identified as positive cases by the network; the recall rate equals the ratio of the number of samples correctly identified as positive cases by the network to the total number of positive samples. mAP equals the area under the PR (Precision-Recall) curve, where P is Precision and R is Recall, as calculated as follows:

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

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

TP represents "the number of positive samples identified by the network as positive," FP represents "the number of negative samples identified by the network as positive," and FN represents "the number of positive samples identified by the network as negative." FN represents "the number of positive samples identified as negative by the network." The AP of each category is obtained by integrating its PR curve. The AP is obtained by integrating the PR curves of each category, and the mAP is obtained by averaging the AP of each category:

$$mAP = \frac{1}{M} \sum_{m=1}^{M} AP^{(m)} \tag{9}$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 16 May 2023 doi:10.20944/preprints202305.1115.v1

The detection speed is defined as the number of frames per second (FPS) that an image is processed.

4.2. Environment

The operating system used for model training is Windows 10, the CPU model is AMD_Ryzen_7_5800H, the GPU model is NVIDIA GeForce RTX 3070 Laptop, the video memory size is 8G, the memory size is 32G, the deep learning framework used is Pytorch 1.12.0, the programming language is Python 3.7, and the GPU acceleration libraries are CUDA 11.6 and CUDNN 8.3.2.

The operating system used during the running speed test was Windows 11. To simulate running the inspection program on a low computing power fishing boat hardware device, the CPU model was AMD_Ryzen_7_5800H, the memory size was 16G, the deep learning framework used was Pytorch 1.12.1 and the programming language was Python 3.7.3.

4.3. Model Training

In the model training session, the number of training rounds was set to 150, the Adam optimizer was used, the batch size was set to 16, no pre-training weights were used, the initial learning rate was set to 0.001, the network input image size was 640×640, using cosine annealing to reduce the learning rate. train_loss and val_losss changes during the training process are shown in Figure 11. The model was trained to 130 rounds when convergence was reached.

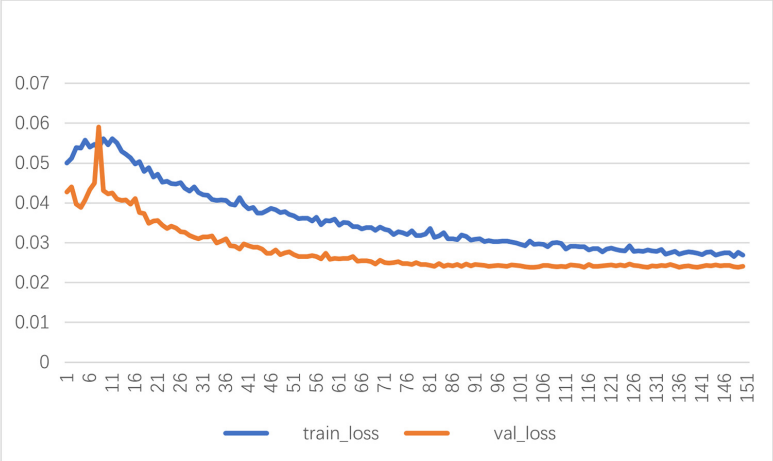


Figure 11. Change of train_loss and val_loss

4.4. Results

Table 4 presents a comparison of FLOPs, model parameters, and weight sizes between the YOLOv5s-SGC and YOLOv5 models. Table 5 displays the detection results of the YOLOv5s-SGC and YOLOv5 models on the fishing vessel operation image dataset.

Table 4. Comparison of Yolov5s and Yolov5s SGC model.

Model	Params	FLOPs	Weight
YOLOv5s	7.02×10^6	15.8G	13.7MB
YOLOv5s-SGC	1.44×10^6	3.2G	3.12MB

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 16 May 2023 doi:10.20944/preprints202305.1115.v1

Table 5. Test results of YOLOv5s and YOLOv5s-SGC model.

Model	Precision	Recall	AP	FPS
YOLOv5s	0.933	0.849	0.901	6.21
YOLOv5s-SGC	0.915	0.796	0.872	9.27

It is evident that YOLOv5s-SGC outperforms YOLOv5s in terms of the number of parameters and floating point operations based on the data in Table 3 and Table 4. The weight size of the method presented in this paper is only 3.12MB, which is 77.23% less than YOLOv5s' 13.7MB, and the FLOPs size is only 3.2GFLOPs. With 1.8% less accuracy, 5.3% less recall, and 3% less average precision than YOLOv5s, the algorithm presented in this paper obtains 9.27 FPS on CPU devices, a 49.28% improvement in detection speed. The comparative data demonstrates that the algorithm reduces the size and computation of the network model and increases the speed of computation while maintaining a certain level of accuracy, making it more suitable for deployment on hardware devices with limited computational capacity.

4.5. Performance Comparison of Different Models

This part aims to objectively and accurately evaluate the overall performance of the proposed model on the fishing vessel operation dataset, including detection accuracy, model size, FLOPs, and detection speed, and compare it to six other models.

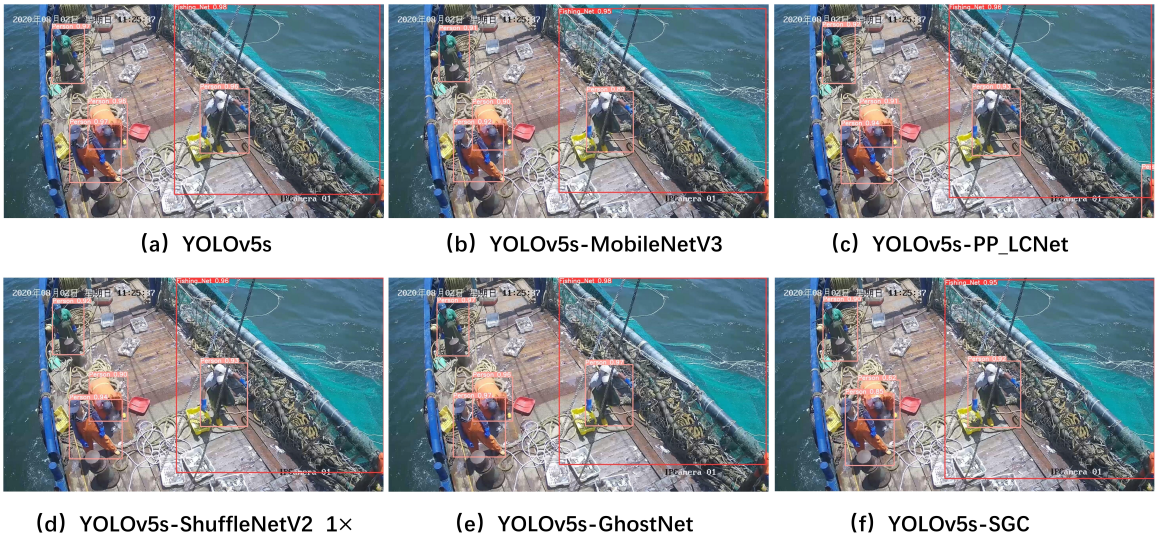
Table 6. Test results of YOLOv5s and YOLOv5s-SGC model.

Model	Params	FLPOS	Weight	FPS	mAP
YOLOv5s	7.02×10^6	15.8G	13.7MB	6.21	0.901
YOLOv5s-MobileNetV3	3.54×10^6	6.3G	7.08MB	6.54	0.875
YOLOv5s-PP_LCNet	3.71×10^6	8.1G	7.36MB	5.87	0.885
YOLOv5s-ShuffleNetV2 1×	3.61×10^6	7.5G	7.22MB	7.46	0.870
YOLOv5s-GhostNet	7.28×10^6	12.8G	14.3MB	3.61	0.897
YOLOv5s-SGC	1.44×10^6	3.2G	3.12MB	9.27	0.872

As seen in Table 6, the YOLOv5s-SGC model has slightly lower accuracy than YOLOv5s, YOLOv5s-MobileNetV3, YOLOv5s-PP_LCNet, and YOLOv5s-GhostNet, but the number of parameters is the lowest and FLOPs is substantially lower than YOLOv5s, YOLOv5s-ShuffleNetV3, YOLOv5s-PP_LCNet, YOLOv5s-GhostNet, MobileNetV3, YOLOv5s-PP_LCNet, YOLOv5s-ShuffleNetV2 1× and YOLOv5s-GhostNet model. This makes the YOLOv5s-SGC model the fastest detection model among the six models, and it also achieves an FPS of 9.27 on CPU devices, which can satisfy the requirement for real-time target detection on fishing vessels.

The detection results of each model on some of the test images are shown in Figure 12. The YOLOv5s, YOLOv5s-MobileNetV3, YOLOv5s-PP_LCNet, YOLOv5s-ShuffleNetV2 1×, YOLOv5s-GhostNet, and YOLOv5s-SGC models were all able to identify the crew in the images and fishing net locations, but YOLOv5s-PP_LCNet appeared to misidentify them. The minimal number of parameters and computational effort of YOLOv5s-SGC also caused the model to have a low confidence level for some of the detected targets, but the confidence level was still above 0.5.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 16 May 2023 doi:10.20944/preprints202305.1115.v1



(a) YOLOv5s (b) YOLOv5s-MobileNetV3 (c) YOLOv5s-PP_LCNet

(d) YOLOv5s-ShuffleNetV2 1× (e) YOLOv5s-GhostNet (f) YOLOv5s-SGC

Figure 12. Detection results of different models on test images.

4.6. Ablation Experiments

Ablation experiments were designed according to Table 7 to assess the efficacy of the various improvement strategies. Group 1 denotes the YOLOv5s model. Group 2 indicates the detection network with Ghost module introduced on the basis of Group 1. Group 3 represents the detection network with CBAM attention mechanism introduced on the basis of Group 2. Group 4 indicates the backbone network using ShuffleNetV2 lightweight network. Group 5 denotes the detection network with improved Ghost module introduced on the basis of Group 4. Group 6 represents the detection network with CBAM attention mechanism introduced on the basis of Group 5.

Table 7. Design of ablation experiments.

Model	YOLOv5s	ShuffleNetV2	Ghost Module	CBAM
1	✓			
2	✓		✓	
3	✓		✓	✓
4	✓	✓		
5	✓	✓	✓	
6	✓	✓	✓	✓

Table 8. Results of ablation experiments.

Model	Params	FLOPs	mAP	Weight	FPS
1	7.02×10^6	15.8G	0.901	13.7MB	6.21
2	5.65×10^6	13.9G	0.898	11.1MB	5.93
3	5.69×10^6	14.0G	0.911	11.2MB	5.77
4	2.86×10^6	5.6G	0.862	5.79MB	9.23
5	1.40×10^6	3.1G	0.868	3.03MB	9.67
6	1.45×10^6	3.2G	0.872	3.12MB	9.26

In comparison to Group 1, Group 2 decreased the number of parameters by 1.37×10^6 and the FLOPs by 1.9 GFLOPs through replacing the standard C3 module with GSConv

 doi:10.20944/preprints202305.1115.v1

and CSP_GSC, according to the results of the ablative analysis in Table 8. The average precision thus dropped by 0.3%. However, the FLOPs and memory access ratio of the Ghost module in the feature fusion network is rather low due to the huge amount of feature maps in the YOLOv5s backbone network, resulting in a drop in FPS compared to Group 1's. By incorporating the CBAM attention mechanism before the detection layer at the expense of a small number of parameters and computations, Group 3 increased the average precision in comparison to Group 2. Although Group 4's average precision decreased from Group 1's by 3.9%, it raised FPS to 9.26, decreased the number of parameters by 4.16×10^6, and increased FLOPs by 10.2 GFLOPs. Group 5 modified the feature fusion network to the Ghost module. Due to the significant reduction in the number of feature maps in the ShuffleNetV2 0.5× backbone network compared with the DarkNet53, there was no situation where the calculation amount and model weight size of yolov5s decreased but the FPS increased. The number of parameters was reduced by 1.46×10^6, FLOPs by 2.5 GFLOPs, weight size decreased by 2.76 MB, and FPS reached 9.67, and the average precision increased by 0.6%. Group 6's parameters, FLOPs, weight size, and FPS were somewhat higher than those of Group 5's, while the average precision increased by 0.4%.

The aforementioned ablation experiments demonstrate that replacing the backbone network with ShuffleNetV2 0.5× and adding the Ghost module to the feature fusion network can significantly reduce the number of parameter-level floating-point operations in the model with a small decrease in average accuracy, thereby decreasing the weight size and accelerating the network. The addition of a CBAM attention mechanism in front of the detection head can compensate for the loss of precision, ensuring a high level of precision while reducing the model's weight.

5. Conclusion

By installing two cameras on a trawler to record video data of fishing vessel operations at sea and creating an image dataset of fishing vessel operations from the video data, this paper proposes a lightweight real-time detection algorithm for deck crew and fishing nets based on improved YOLOv5s. This algorithm combines the features of YOLOv5s, ShuffleNetV2, GhostNet, and CBAM attention modules YOLOv5s-SGC, which decreases the number of parameters. The model has been proven to be deployable on low computing power devices and to provide real-time accurate target detection of fishing vessels in operational images. It has also been proven to lessen the workload of law enforcement officers by determining the current operational status of fishing vessels based on the model's detection results.

Author Contributions: Conceptualization, J.W. and G.L.; methodology, J.W. , X.Y. and G.L.;software, J.W.;validation, J.W.;formal analysis, J.W.;investigation, J.W.;resources, X.Y.;data curation, J.W. and X.Y.; writing - original draft, J.W. ;writing - review & editing, J.W. and G.L.; supervision, G.L.;Project administration, G.L.; funding acquisition, G.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the project entitled "Research on multi parameter fusion recognition and tracking method of single fish target for large aquaculture water body" funded by the National Natural Science Foundation of China (NSFC), China (32073026) , was supported by the project " Research and development of long-range stereo fish detection sonar equipment" financed from Sanya Yazhou Bay Science and Technology City Administration, China(SKJC-2020-01-013).

Institutional Review Board Statement: No institutional board review was necessary for this project.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data available on request from the authors.

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

References

1. Zou, L. Research on information technology for the regulation of fishing vessels. *The Farmers Consultant* **2017**, *No.568*, 347–349.

2. Yusuf, M.; Razak, A. Illegal Fishing Eradication: Comparative Study of Indonesia and Malaysia. *JL Pol'y Globalization* **2018**, *71*, 120.
3. Zou, L. Study On The Supervision Of Marine Fishing Vessel Of Cangnan County. *Northwest Agriculture and Forestry University* **2018**.
4. Feng, B.; Chen, X.; Zhu, G. Identification of complementary overfishing and its implications for fisheries management. *Resource Development Market* **2010**, *26*, 20–23.
5. Yan, W.; Jiang, Y.; Cai, L.; He, Y.; Wang, F. Analysis of Fishing Yield of Different Types of Fishing Boats on Operation by Grey Theory in Zhoushan. *Journal of Zhejiang Ocean University(Natural Science)* **2018**, *37*, 468–474.
6. Lin, Z. Study on the need for the application of the Fishing Vessel Dynamic Monitoring Information System. *China Agricultural Informatics* **2013**, p. 152.
7. Zhu, J. A Study on the Application of the Fishing Vessels Dynamic Monitoring Information System in Fishery Management. *South China University of Technology* **2010**.
8. Zhang, J.; Zhang, S.; Wang, S.; Yang, Y.; Dai, Y.; Xiong, Y. Recognition of Acetes chinensis fishing vessel based on 3-2D integration model behavior. *South China Fisheries Science* **2022**, *18*, 126–135.
9. Wang, S.; Zhang, S.; Tang, F.; Shi, Y.; Sui, Y.; Fan, X.; Chen, J. Developing machine learning methods for automatic recognition of fishing vessel behaviour in the Scomber japonicus fisheries. *FRONTIERS IN MARINE SCIENCE* **2023**, *10*.
10. Wang, S.; Zhang, S.; Zhu, W.; Sun, Y.; Yang, Y.; Sui, J.; Shen, L.; Shen, J. Application of an electronic monitoring system for video target detection in tuna longline fishing based on YOLOV5 deep learning model. *Journal of Dalian Ocean University* **2021**, *36*, 842–850.
11. Wang, S.; Sun, Y.; Zhang, S.; Sui, J.; Zhu, W.; Yang, S.; Fan, W. Automatic mapping of thermal diagram based on satellite AIS offshore ship position. *Fishery Information Strategy* **2021**, *36*, 45–53.
12. Pei, K.; Zhang, S.; Fan, W.; Hou, J.; Tang, X.; Zhu, W. Extraction method of stow net fishing intensity distribution in Zhejiang Province. *Journal of Fisheries of China* **2020**, *44*, 1913–1925.
13. Zhang, S.; Fan, W.; Zhang, H.; Yang, S.; Shen, J.; Zou, G. Text extraction from electronic monitoring video of ocean fishing vessels. *Fishery Information Strategy* **2020**, *35*, 141–146.
14. Tang, X.; Zhang, S.; Fan, W.; Pei, K. Fishing type identification of gill net and trawl net based on deep learning. *Marine Fisheries* **2020**, *42*, 233–244.
15. Wang, S.; Zhang, S.; Dai, Y.; Wang, Y.; Sui, J.; Zhu, W. Research on calculating fishing depth of krill by sonar data. *South China Fisheries Science* **2021**, *17*, 91–97.
16. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861* **2017**.
17. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.C. Mobilenetv2: Inverted residuals and linear bottlenecks. In Proceedings of the Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 4510–4520.
18. Howard, A.; Sandler, M.; Chu, G.; Chen, L.C.; Chen, B.; Tan, M.; Wang, W.; Zhu, Y.; Pang, R.; Vasudevan, V. Searching for mobilenetv3. In Proceedings of the Proceedings of the IEEE/CVF international conference on computer vision, pp. 1314–1324.
19. Tan, M.; Le, Q. Efficientnet: Rethinking model scaling for convolutional neural networks. In Proceedings of the International conference on machine learning. PMLR, pp. 6105–6114.
20. Cui, C.; Gao, T.; Wei, S.; Du, Y.; Guo, R.; Dong, S.; Lu, B.; Zhou, Y.; Lv, X.; Liu, Q. PP-LCNet: A lightweight CPU convolutional neural network. *arXiv preprint arXiv:2109.15099* **2021**.
21. Ma, N.; Zhang, X.; Zheng, H.T.; Sun, J. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In Proceedings of the Proceedings of the European conference on computer vision (ECCV), pp. 116–131.
22. Wang, C.Y.; Liao, H.Y.M.; Wu, Y.H.; Chen, P.Y.; Hsieh, J.W.; Yeh, I.H. CSPNet: A new backbone that can enhance learning capability of CNN. In Proceedings of the Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops, pp. 390–391.
23. Yun, S.; Han, D.; Oh, S.J.; Chun, S.; Choe, J.; Yoo, Y. Cutmix: Regularization strategy to train strong classifiers with localizable features. In Proceedings of the Proceedings of the IEEE/CVF international conference on computer vision, pp. 6023–6032.
24. Zhang, Q.; Lin, Q.; Xiao, L. Improved aerial image recognition algorithm of YOLOv5 [J]. *Changjiang Information & Communication* **2021**, *34*, 73–76.
25. He, K.; Zhang, X.; Ren, S.; Sun, J. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE transactions on pattern analysis and machine intelligence* **2015**, *37*, 1904–1916.
26. Rezatofighi, H.; Tsoi, N.; Gwak, J.; Sadeghian, A.; Reid, I.; Savarese, S. Generalized intersection over union: A metric and a loss for bounding box regression. In Proceedings of the Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 658–666.
27. Bodla, N.; Singh, B.; Chellappa, R.; Davis, L.S. Soft-NMS–improving object detection with one line of code. In Proceedings of the Proceedings of the IEEE international conference on computer vision, pp. 5561–5569.
28. Zhang, X.; Zhou, X.; Lin, M.; Sun, J. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In Proceedings of the Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 6848–6856.
29. Han, K.; Wang, Y.; Tian, Q.; Guo, J.; Xu, C.; Xu, C. Ghostnet: More features from cheap operations. In Proceedings of the Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 1580–1589.

 doi:10.20944/preprints202305.1115.v1

30. Du, K.L. Clustering: A neural network approach. *Neural networks* **2010**, *23*, 89–107.
31. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. Cbam: Convolutional block attention module. In Proceedings of the Proceedings of the European conference on computer vision (ECCV), pp. 3–19.