Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2023.039451
Article

Lightweight Surface Litter Detection Algorithm Based on Improved YOLOv5s

Zunliang Chen[1,2], Chengxu Huang[1,2], Lucheng Duan[1,2] and Baohua Tan[1,2,*]

[1]College of Science (College of Chip Industry), Hubei University of Technology, Wuhan, 430068, China
[2]National "111 Research Center" Microelectronics and Integrated Circuits, Hubei University of Technology, Wuhan, 430068, China
*Corresponding Author: Baohua Tan. Email: tbh@hbut.edu.cn
Received: 30 January 2023; Accepted: 17 April 2023; Published: 09 June 2023

Abstract: In response to the problem of the high cost and low efficiency of traditional water surface litter cleanup through manpower, a lightweight water surface litter detection algorithm based on improved YOLOv5s is proposed to provide core technical support for real-time water surface litter detection by water surface litter cleanup vessels. The method reduces network parameters by introducing the deep separable convolution GhostConv in the lightweight network GhostNet to substitute the ordinary convolution in the original YOLOv5s feature extraction and fusion network; introducing the C3Ghost module to substitute the C3 module in the original backbone and neck networks to further reduce computational effort. Using a Convolutional Block Attention Mechanism (CBAM) module in the backbone network to strengthen the network's ability to extract significant target features from images. Finally, the loss function is optimized using the Focal-EIoU loss function to improve the convergence speed and model accuracy. The experimental results illustrate that the improved algorithm outperforms the original Yolov5s in all aspects of the homemade water surface litter dataset and has certain advantages over some current mainstream algorithms in terms of model size, detection accuracy, and speed, which can deal with the problems of real-time detection of water surface litter in real life.

Keywords: Surface litter detection; lightweight; YOLOv5s; GhostNet; deep separable convolution; convolutional block attention mechanism (CBAM)

1 Introduction

Along with the high-quality growth of China's economy and the rising living standards of its residents, the increasing richness of material life is accompanied by the corresponding phenomena of the massive output of garbage, random discarding, simple piling, and disposal [1]. Besides, the problems of water pollution and eutrophication caused by floating litter on the water surface have seriously affected the ecological civilization and human living environment in the watershed. At the present stage, the management of floating litter on water surfaces by domestic related departments and institutions is mainly based on interception and collection and manual salvage, but the manual

cleaning method is time-consuming, laborious, and long-period, and it cannot reach the demands of intelligent detection and automatic cleaning in real-time.

With modern information technology and the advancement of intelligent manufacturing technology, related researchers have proposed the combination of object detection technology and mechanical cleanup boats to detect and clean up floating litter on the water surface by using deep learning technology instead of manual labor [2]. Since these scenes use hardware that is mostly edge devices, under the terms of limited memory and arithmetic power, it is necessary to take into account not only its mechanical cleanup device drive system requirements, the required arithmetic power requirements of the model. Therefore, reducing the model size is more beneficial for porting to computing devices [3].

As artificial intelligence technology and computer vision develop, object detection, as one of the key branches, is extensively applied in Optical Character Recognition (OCR) analysis, contextual analysis, disaster management, and vehicle recognition. In the field of Unmanned Aerial Vehicles (UAV), Dilshad et al. [4] innovated the LocateUAV method for estimating UAV location using contextual analysis in an Internet of Things (IoT) environment. In the vision field, the traditional object detection technology based on machine learning mainly uses the size, shape, color, texture, and other information of the object, and then performs template matching and saliency detection after completing manual feature extraction by image segmentation technology [5]. However, this method suffers from insufficient model detection accuracy, speed, and poor robustness. Over the past few years, object detection techniques grounded on deep learning have obtained outstanding results in terms of detection effectiveness and model robustness under their convolutional neural networks' ability to automatically extract features. Yu et al. [6] proposed the High-resolution Network (HRNet) improves detection accuracy by fusing multi-scale feature maps using parallel structural networks to obtain different multi-scale feature information. Tang et al. [7] achieved good detection results in OCR by changing the detection scale of the network and incorporating a modified OCR branch in the network. Currently, there are two stages in the development of object recognition algorithms based on deep learning: one class is region-based two-stage object detection algorithms, with networks such as Regions Convolutional Neural Network (R-CNN) [8], Fast R-CNN [9], and Faster R-CNN [10] as typical representatives. This type of method performs feature and classification detection in two stages based on the divided regions, which have a higher detection accuracy but slow speed. Another class is the single-stage regression-based object detection algorithm, with a series of networks such as You Only Look Once (YOLO) [11,12] and Single Shot Multi-Box Detector (SSD) [13,14] as typical representatives. Mathias et al. [15] merged a Gaussian mixture model 2D empirical modal decomposition algorithm with a Yolov3 depth network and applied it to underwater object detection.

The YOLO series algorithm integrates features and classification into one network and identifies and locates them through regression calculation, significantly increasing the detection speed and meeting the demand for detection in real-time, but with a loss in detection accuracy. For that reason, this paper proposed a lightweight surface litter detection algorithm with improved YOLOv5s, which reduces the model size and makes the network more lightweight based on improving the detection performance of the original YOLOv5s methods. The main contributions of our work are as follows:

(1) Improving the ordinary convolution in the YOLOv5s network into a deeply separable convolution GhostConv, which effectively reduces the convolution layers and computational resources; introducing lightweight C3Ghost modules to substitute the C3 module, further minimizing the model parameters and computation, making it more lightweight.

(2) This paper fuses the CBAM module [16] in the backbone network of YOLOv5s, applying it to enrich the network's capacity to acquire target feature information and improve the model's prediction capability.
(3) A new loss function is adopted, and the Focal-EIoU (Focal and Efficient Intersection over Union) [17] loss function is invoked as the prediction frame regression loss function in the YOLOv5s network to optimize the deficiencies of the CIoU (Distance IoU) [18] loss function in the original method and accelerate the loss function convergence speed and model prediction accuracy.

In this paper, Section 2 reviews the work on water surface litter detection, YOLOv5s network, and lightweight network. Section 3 describes the water surface litter detection methods. Section 4 describes the experimental content, including the experimental datasets production, experimental environment, and evaluation metrics. Experiments and results are presented and reviewed in Section 5. Section 6 summarizes and gives an outlook.

2 Related Work

2.1 Water Surface Detection

Water surface litter detection is a branch of object detection, which has significant research implications for environmental management and water resource protection. Surface object detection has been gradually improved and enriched from traditional manually designed features and shallow classifier frameworks based on deep learning object detection frameworks.

As far as the traditional detection methods are concerned, they rely too much on key points, edges, and templates leading to their low detection accuracy. For example, Matsumoto's Histogram of Oriented Gradient (HOG)-Support Vector Machine (SVM) method [19], proposed in 2013, detects surface vessels by images taken by a shipboard camera. Kaido et al. [20] combined SVM methods and edge detection techniques for ship detection and ship number identification. However, a majority of these methods traditionally scan the entire image through a swiping window to detect objects, which greatly limits detection efficiency.

Deep learning techniques, with their powerful image processing power and expression performance of convolutional neural networks, are applied to obtain remarkable effects in surface object detection. Zhang et al. [21] optimized the Faster R-CNN by fusing high-level and low-level features of the network to enrich the real-time detection of surface objects. Panwar et al. [22] proposed the AquaTrash dataset using deep migration learning techniques to enrich the generalizability of the AquaTrash dataset. Li et al. [23] improved the YOLOv3 algorithm and embedded it into a water surface garbage cleaning robot only to achieve intelligent cleaning of water surface litter. The rich surface object dataset and constantly innovative and iterative object detection algorithms provide key technical support for surface object detection.

2.2 YOLOv5 Method

The YOLOv5 [24] network model is made up of four main parts: Input, Backbone, Neck, and Head network, as shown in Fig. 1. Input side uses Mosaic + Mixup data enhancement technology to randomly crop, scale, and stitch photos from the input into enroll the background of the dataset images, enhance the network's generalizability. The Backbone network mainly consists of Cross Stage Partial (CBS), CSP Bottleneck with 3 convolutions (C3), and Spatial Pyramid Pooling-Fast (SPPF) modules, where the CBS structure consists of the Conv + BN + SiLU activation function, which

convolves and normalizes the input images before passing them through the activation function to the next layer of convolution. The C3 structure is a reworked version of the CSP structure made in the correction unit, for dividing the feature mappings at the base layer before merging them via the inter-stage hierarchy to ensure the correct rate while reducing the computational bottleneck and strengthening the network's learning capability. The SPPF module uses three 5×5 maximum pooling to fuse feature map messages across different scales to enrich the network's ability to abstract features from images. The neck network uses Feature Pyramid Network (FPN) + Path Aggregation Network (PAN) [25,26] structure to further enhance the network feature fusion capability using a combination of top-down and bottom-up approaches. The head network screens the target candidate frames by non-maximal suppression and is used as the output of the prediction results.

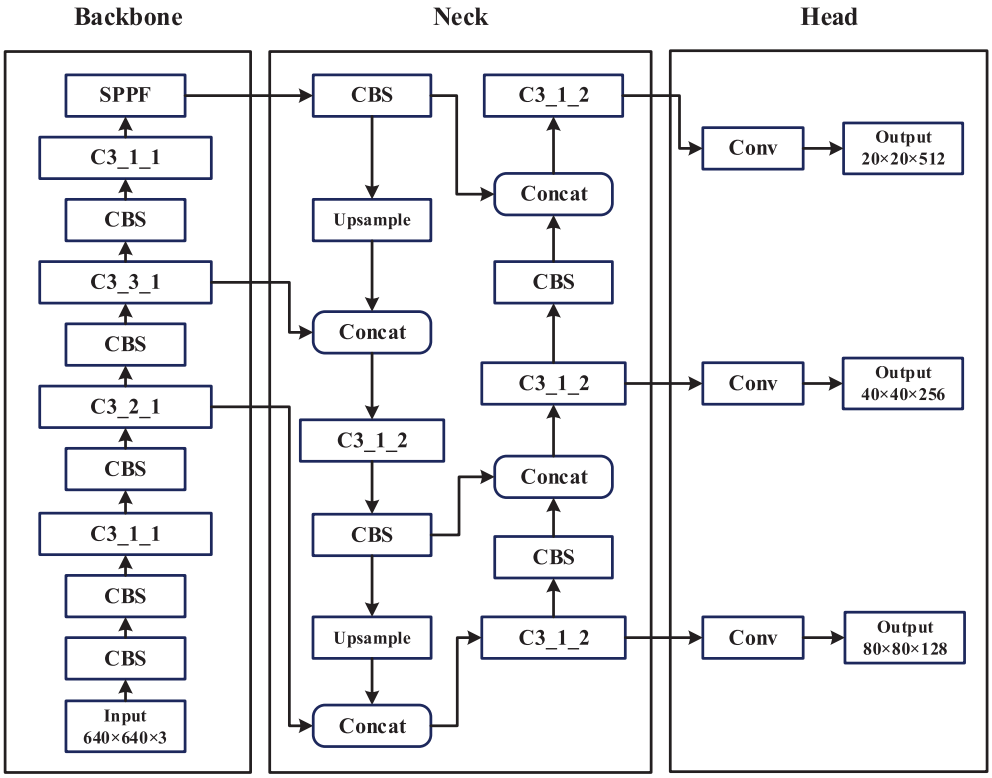


Figure 1: YOLOv5s overall network design chart

2.3 Lightweight Network

A lightweight network means that the size and parameters of the model are reduced as much as possible by optimizing the network structure to meet the requirements of low arithmetic power of edge devices while ensuring that the model is equally effective in detection. At present, most of the more popular lightweight solutions of the algorithm are considered from model compression and pattern structure design.

Model reduction aims to take an already trained network and reduce the number of parameters, usually with schemes such as model distillation, pruning, and quantization. With the improvement in the level of hardware conditions, the computational power of computers has grown by leaps and

bounds, and Neural Architecture Search (NAS) network is also a technique to find the best network using computational-level arithmetic power [27,28].

Among the model architecture design solutions, many excellent design solutions have been broadly applied in the past few years. The more classical lightweight network structures are listed from the time of computational introduction, such as SqueezeNet [29], which proposes Fire Module, ShuffleNet [30,31] series of ShuffleNet Unit feature fusion schemes, etc.

Google proposed MobileNetV1 [32] object detection algorithm using a depth-separable convolution structure and in the subsequent proposed MobileNetV2 [33] and MobileNetV3 [34] algorithms, the essence of depth-separable convolution involves splitting the normal convolution into deep and pointwise convolution [35]. In particular, in terms of network structure, MobileNetV3 derives the main network structure through a neural architecture search [36]. The network uses a 5×5 depth-separable convolutional structure by adding the Squeeze-and-Excitation (SE) attention mechanism [37], which assigns its weights on the feature maps through the process of network training, which is very easy to use, plug-and-play, and has shown good performance improvement in several mainstream networks.

3 Water Surface Litter Detection Method

For water surface litter detection, the main detection process can be divided into three steps: dataset production, model improvement, and model deployment. Since there is no publicly available water surface litter dataset, the first step requires a homemade water surface litter dataset. After filtering the collected images, the object garbage in the pictures is labeled with Labelimg labeling software, with the labeled garbage categories divided into 7 categories, namely: {lunch boxes, foam, bottles, plastic bags, leaves and branches, food packaging bags, paper drink boxes}, which are used to generate a standard VOC dataset format. Labelimg labeling software will automatically label the object boxes with the labeled information the generated XML file is stored in the Annotations folder, the TXT file of the dataset division is stored in the ImageSets folder, and the JPEGImages are used to store the original images. In the second step, the produced dataset is put into the enhanced YOLOv5s network for training, and the network will perform Mosaic + Mixup [38] data enhancement process on the dataset according to the pre-set ratio, and the CBAM module is inserted in the bottom position of the improved YOLOv5s backbone network to help extract features. It is then fed into the feature fusion network for training after the improved Focal-EIoU loss function balances the samples of positive and negative. Finally, to evaluate the trained model with test data, and after completing the evaluation, it is deployed into the embedded device on the water surface trash cleaning vessel to provide real-time detection and intelligent classification of water surface trash, as shown in Fig. 2.

3.1 Improving Convolution and C3 Module of YOLOv5 Network

YOLOv5 [24] network is an open-source object detection algorithm proposed by Ultralytics and is the fifth edition of the YOLO series developed to date. A typical single-stage object detection algorithm, YOLOv5 is divided into four versions in order of detection accuracy and model size, depending on network layers' depth and feature map width: YOLOv5s, YOLOv5m, YOLOv5l, and YOLOv5x, which they respond to the different demands of industrial applications concerning detection accuracy, detection speed, and the module size, respectively. In this article, YOLOv5s is chosen as the benchmark model from the perspective of edge devices oriented to low computing power.

The network structure diagram of YOLOv5s shows that the network contains many basic convolutional blocks of CBS are made up of Convolution (Conv), Batch Normalization (BN), Sigmoid Linear Unit (SiLU), and C3 structure consisting of CBS, Bottleneck, and Concat. More basic convolutional

blocks increase network computational parameters and affect network inference speed. Therefore, in this paper, from the perspective of lightweight, a more concise network structure is used to replace the more computationally intensive ordinary convolutional Conv and C3 structures in YOLOv5s based on the premise of equal detection effect. The ordinary convolutional Conv in the YOLOv5s network is substituted by the deep separable convolutional GhostConv, as displayed in Fig. 3.

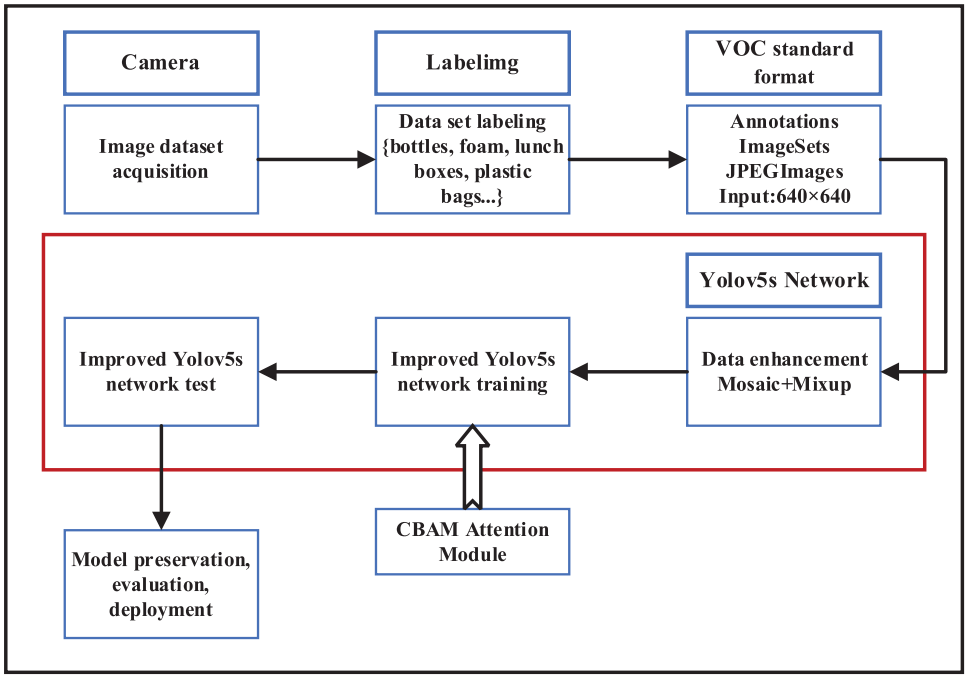


Figure 2: Detection flow chart

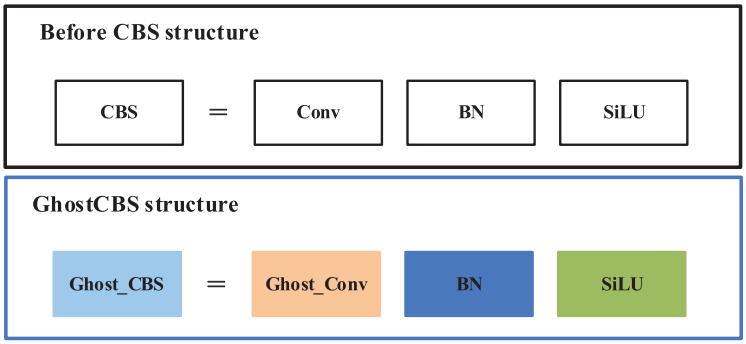


Figure 3: CBS comparison chart

GhostNet [39] network is an efficient network designed by Huawei Noah's Ark Lab, whose core idea is to use less computationally intensive Ghost modules to generate redundant feature maps that exist in neural networks. A feature of the Ghost module is to instead of the normal convolutional layer, it divides the normal convolution into two parts, which are divided into two parts. First, the necessary feature condensation map of the input features is obtained using the ordinary 1×1 convolution that acts similarly to feature integration; then the feature con map obtained in the previous stage to obtain

the similar character map of feature condensation using the depth-separable convolution. By this two-step operation, the number of model counts is reduced by obtaining redundant characteristic maps while minimizing the number of convolutional layers, as displayed in Fig. 4.

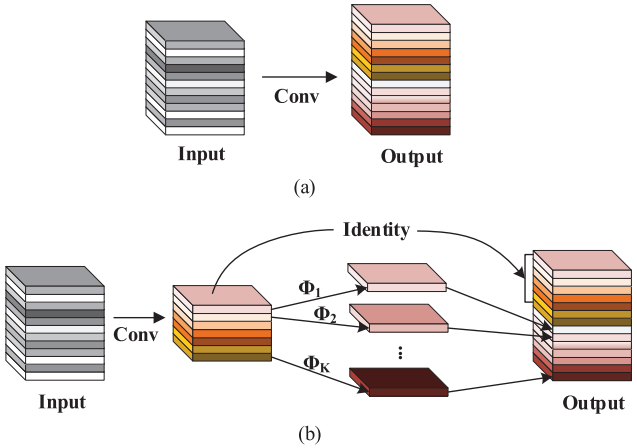


Figure 4: Schematic diagram of the two convolution processes, (a) normal convolution (b) deeply separable convolution GhostConv

The GhostNet network structure is composed of a GhostNet Bottlenecks section as the backbone and a residual edge section. Bottlenecks mainly consist of a bottleneck structure with two stacked Ghost modules. Add channels and set the expansion ratio of the number of output channels with the first ghost module; the second ghost module decreases the channel count to match the number of input channels. This structure can be categorized into two types, depending on the step size. A deep separable convolution with Stride = 2 is used for twice the down-sampling. In this paper, the C3 module in the YOLOv5s network is substituted with the C3Ghost module, which decreases and makes the overall YOLOv5s model more lightweight and upgrades the execution speed of the model. The new network is displayed in Fig. 5.

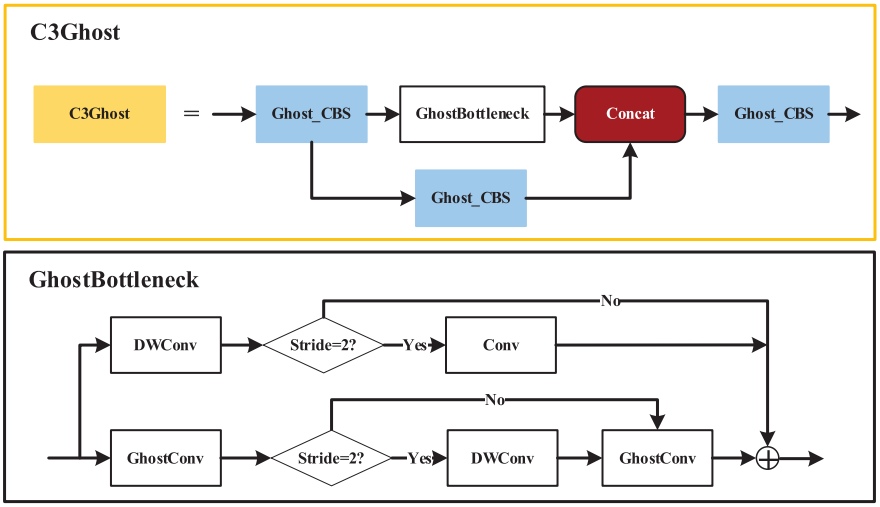


Figure 5: Structure of the improved C3Ghost network

3.2 Introduces CBAM Module

The CBAM module is a lightweight attention network presented by Woo et al. [16] that enhances the feature representation of the network with attention operations that can be performed in the channel and spatial dimension. In this article, the CBAM module is integrated before the SPPF module in the YOLOv5s backbone network to strengthen the extraction capability of small target feature information using the CBAM module. The modified YOLOv5s network structure is displayed in Fig. 6 below.

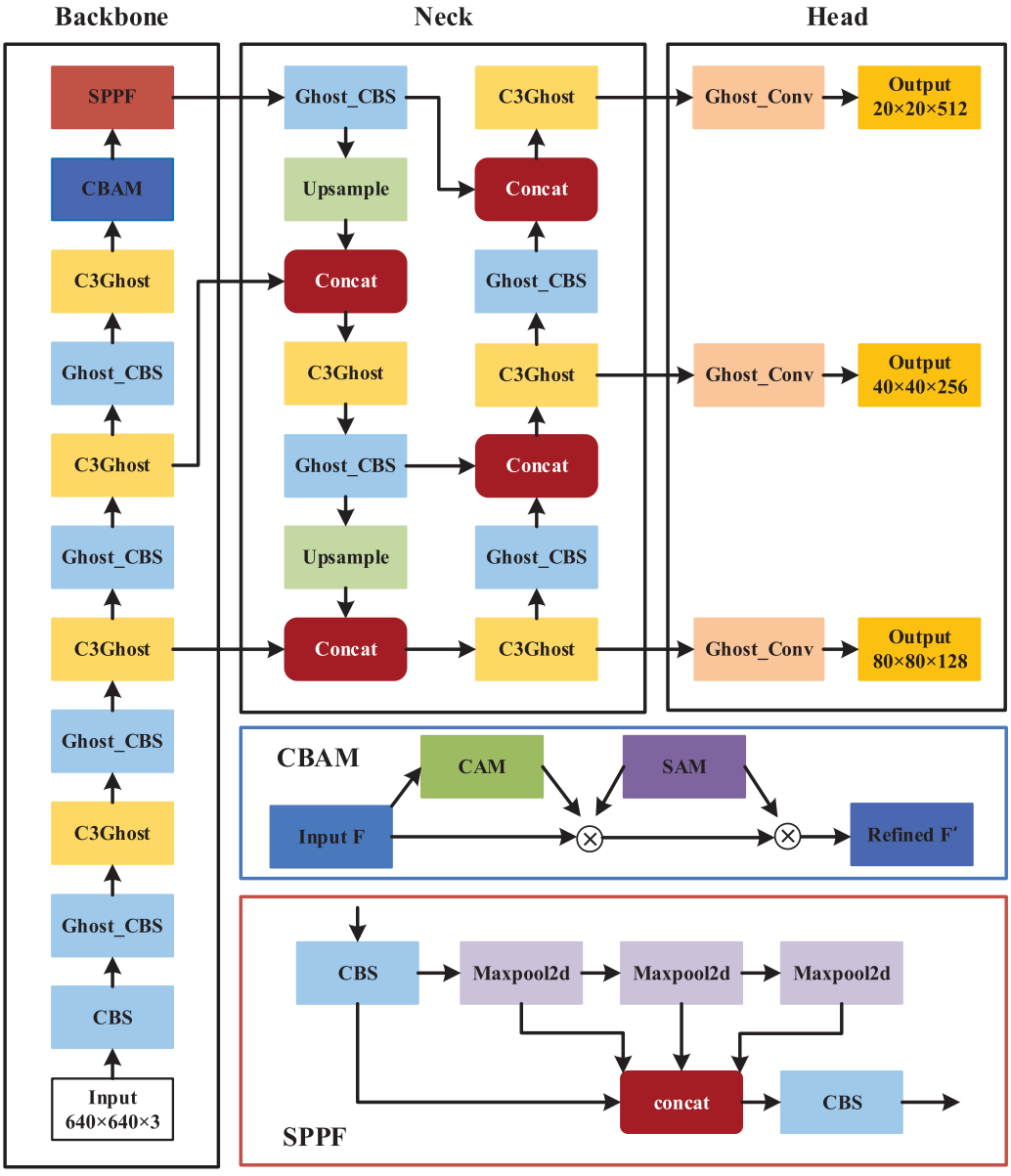


Figure 6: Improved YOLOv5s network structure

The CBAM module contains two independent sub-modules, Channel Attention Module (CAM) and Spatial Attention Module (SAM), and the overall structure is presented in Fig. 7. A feature map

$F \in R^{C\times H\times W}$ of an intermediate layer is given as input, a 1D channel attention feature map $M_c \in R^{C\times 1\times 1}$, and a 2D spatial attention feature map $M_s \in R^{1\times H\times W}$, with C denoting the channel of the feature map, and H and W denoting the height and width of the feature map. The input feature map F is first multiplied with the feature map M_C of F after channel attention module operation to get F_1, and then F_1 is multiplied with the feature map M_S of F_1 after spatial attention module operation to get F_2, and its whole calculation process is given in Eq. (1).

$$F_1 = M_C(F) \otimes F,\ F_2 = M_S(F_1) \otimes F_1 \tag{1}$$

where $\otimes$ denotes multiplication by elements.

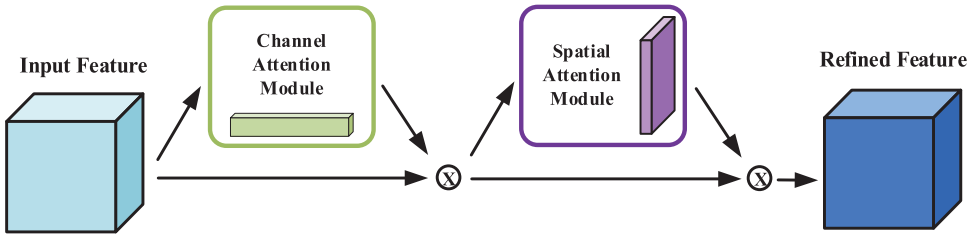


Figure 7: The CBAM module structure chart

The CAM is presented in Fig. 8. The input F is compressed in spatial dimension by two operations of maximum pooling and average pooling to get two $1 \times 1 \times C$ feature maps F_{max} and F_{avg}, which are fed into a shared network composed of Multilayer Perceptron (MLP) [40] for the calculation to obtain two different background description maps, then make it summed at the pixel level and then use a sigmoid function to activate it to finally get the channel attention map M_C, and the operation process is given in Eq. (2).

$$\begin{aligned} M_C(F) &= \sigma\left(MLP\left(AvgPool\left(F\right)\right) + MLP\left(MaxPool\left(F\right)\right)\right) \\ &= \sigma\left(W_1\left(W_0\left(F^c_{avg}\right)\right) + W_1\left(W_0\left(F^c_{max}\right)\right)\right) \in R^{C\times 1\times 1} \end{aligned} \tag{2}$$

where C is the output vector length; σ is the sigmoid activation function; MLP is the shared fully connected layer; $W_0 \in R^{\frac{C}{r}\times C}$ denotes the first layer of the shared full connectivity layer; $W_1 \in R^{C\times\frac{C}{r}}$ denotes the second layer of the shared full connectivity layer.

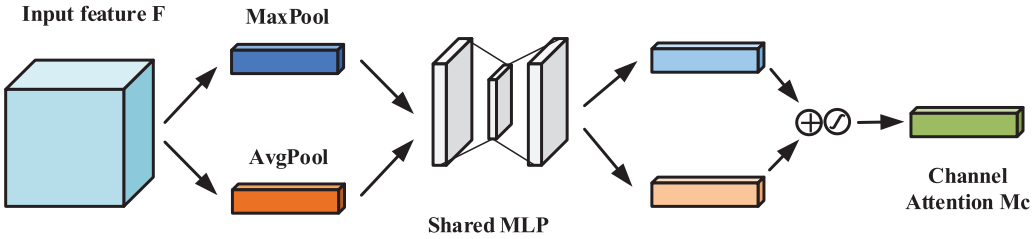


Figure 8: Channel attention module (CAM)

The SAM is given in Fig. 9. Firstly, maximum pooling and average pooling are applied in the channel dimension to acquire two $H \times W \times 1$ feature maps, then channel cascading is performed on the feature maps, followed by a convolutional layer to downscale to a single channel, and finally the spatial attention map M_S is obtained after activation by a sigmoid function, and the operation process

is illustrated in Eq. (3).

$$M_S(F) = \sigma\left(f^{7\times7}\left([AvgPool(F); MaxPol(F)]\right)\right) = \sigma\left(f^{7\times7}\left(\left[F^s_{avg}; F^s_{max}\right]\right)\right) \in R^{H\times W} \tag{3}$$

where $f^{7\times7}$ denotes a 7×7 convolutional layer.

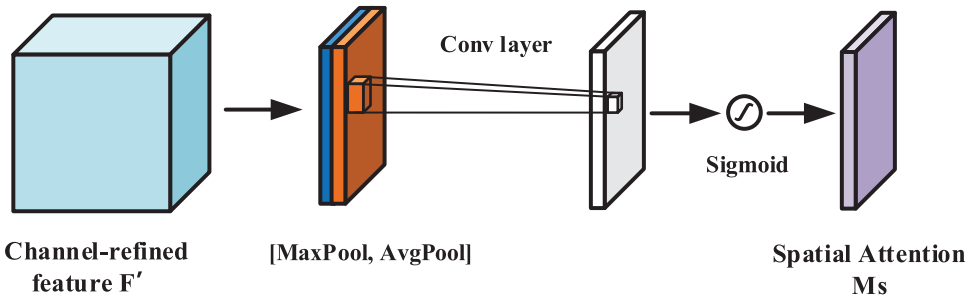


Figure 9: Spatial attention module (SAM)

3.3 Optimization of the Loss Function

The loss function of the YOLOv5s network is made up of three loss functions, which are localization loss ($loss_{box}$), confidence loss ($loss_{obj}$), and classification loss ($loss_{cls}$). The magnitude of the loss function value is the sum of the three loss functions, as given in Eq. (4). The CIoU [18] loss function is used as the bounding box regression loss function by default in the YOLOv5s network. The formula is shown in Eq. (5).

$$Loss = loss_{box} + loss_{obj} + loss_{cls} \tag{4}$$

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

$$\alpha = \frac{v}{(1 - IoU) + v} \tag{6}$$

$$v = \frac{4}{\pi^2}\left(\arctan\frac{w^{gt}}{h^{gt}} - \arctan\frac{w}{h}\right)^2 \tag{7}$$

In the formula, IoU indicates how the predicted frame intersects the true frame; $\rho^2(b,\ b^{gt})$ for the calculation of a Euclidean distance between two centroids; c shows the diagonal distance between the minimum external rectangle of the object box to be detected and the real object box; α indicates a weighting factor; v is a measure of aspect ratio consistency; $\frac{w^{gt}}{h^{gt}}$ indicates the aspect ratio of the real frame; $\frac{w}{h}$ indicates the length-to-width ratio of the prediction frame.

While CIoU considers accounting for the intersection area, centroid distance, and aspect ratio of the prediction frame regression, the difference in aspect ratio is reflected by v in the formula and is not the real difference between the width and height respectively, and its confidence level, which sometimes prevents effective optimization of the model. EIoU [17] splits the aspect ratio based on CIoU to explicitly measure the difference between the three geometric factors, accelerating convergence and improving regression accuracy. Simultaneously, Focal loss is added to tweak the

problem of unbalanced hard and easy samples. Therefore, in this paper, Focal-EIoU [17] is introduced as the prediction frame regression loss function in the YOLO v5s network, and a suppression factor γ is added, which is calculated as shown in Eqs. (8), (9).

$$L_{EIoU} = L_{IoU} + L_{dis} + L_{asp} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{(w^c)^2 + (h^c)^2} + \frac{\rho^2(w, w^{gt})}{(w^c)^2} + \frac{\rho^2(h, h^{gt})}{(h^c)^2} \tag{8}$$

$$L_{Focal-EIoU} = IoU^{\gamma} L_{EIoU} \tag{9}$$

4 Experimental Design

4.1 Dataset Production

Currently, there is no public open-source dataset for research on water surface litter classification. To evaluate the capabilities of the model, the types of surface litter were classified into seven categories: bottles, paper drink containers, lunch boxes, foam, plastic bags, food packaging bags, and leaves and branches, according to domestic litter classification standards and by combining the types, forms, and sizes of floating litter commonly found on the water surface of rivers and lakes. All data sets used in the experiments were acquired in the watershed of the XunSi River at the Hubei University of Technology. To ensure the comprehensiveness and complexity of the datasets and to enhance the generalization of the detector, the collection of photos needed to take into account different periods and weather. A total of 3000 photos in JPG format with 480×480 pixels were acquired. To overcome problems of overfitting caused by small data sets and the effects of water reflection and lighting on detection, image processing methods such as cropping, rotation, flipping, and stitching are used to improve the diversity of data sets and the richness of image backgrounds, which are helpful for feature extraction of small target objects. As shown in Fig. 10.

(a) (b)

Figure 10: Schematic diagram of different scenario datasets. (a) photos of water surface debris at different times (b) spliced photos

The images in the dataset are labeled according to the determined categories using the open-source software Labelimg, which generates standard VOC dataset format files, including three folders of Annotation, ImageSets, and JPEGImages, containing all the labeling information of the images. To

avoid the problem of model overfitting, the model should have the best generalization performance. This paper divides the produced dataset into training and test sets at the rate of 9:1, where 2700 images are used as the training set and 300 pictures are used as the test set, and using migration training method for 100 rounds of epoch training. As presented in Table 1.

Table 1: Table of the count of various types of water surface litter

Category	Lunch boxes	Foam	Bottles	Plastic bags
Category number	0	1	2	3
Training set	1520	1183	1768	709
Test set	166	125	218	76
Category	Leaves and branches	Food packaging bags	Paper drink boxes	
Category number	4	5	6	
Training set	1868	1060	1394	
Test set	212	86	163	

4.2 Experimental Configuration

In this paper, the experimental environment for the Windows 10 operating system, the choice of Pytorch 1.12 framework, the processor is intel i9-11900, the configuration of Nvidia GeForce RTX A4000 graphics card, the specific experimental environment hardware and software configuration as presented in Table 2.

Table 2: Experimental environment configuration

Configuration name	Configuration information
Operating system	Windows10
Memory	64G
CPU	Inter (R) Core (TM) i9-11900@2.50 GHz 2.50 GHz
GPU	Nvidia GeForce RTX A4000
Language	Python 3.7
GPU acceleration	Cuda11.6 Cudnn11.5
Software environment	Anaconda, PyCharm, Pytorch

4.3 Model Evaluation Indicators

To objectively and accurately judge the modified model performance, this paper adopts Precision (P), Recall (R), Average Precision (AP), and Mean Average Precision (mAP) as the hard metrics of the model; and adopts Frames Per Second (FPS), Parameters (Params) and model Size as an auxiliary indicator to measure the performance of the improved method. The hard indicators are calculated as follows.

$$P = \frac{TP}{FP + TP} \tag{10}$$

$$R = \frac{TP}{FN + TP} \tag{11}$$

$$AP = \begin{cases} \int_0^1 P(r)\, dr, \ \ PR \text{ continuous} \\ \sum_{n=1}^{N} P(n)\, \Delta r, \ \ PR \text{ discrete} \end{cases} \tag{12}$$

$$mAP = \frac{\sum_{i=1}^{K} AP(i)}{K} \tag{13}$$

In the formula, TP (True Positives) means the count of positive samples which are correctly identified; FP (False Positives) means the count of positive samples detected incorrectly; FN (False Negatives) means the count of negative samples identified incorrectly; K means the total count of identified categories (7 in this paper); AP(i) means the AP value of the ith category; mAP is the average value of the detected APs of various types of surface litter. The larger of mAP value, the higher the model detection accuracy.

5 Results and Analysis

5.1 Network Training

Before the network starts working, it needs to set some parameters of the network, so that the improved YOLOv5s network can reach the best training effect. This paper used the migration training idea to use the training weights of YOLOv5s over the COCO dataset as pretraining weights of the model backbone network and prevent random values of backbone weights to raise feature extraction. To further strengthen the generalization of the dataset, use the Mosaic + Mixup data boosting technique at the input side, and the data boosting technique is performed in the first 70 rounds according to the probability of 50% in each round. The input picture resolution is 480 × 480, and the model training batch size is 8. Meanwhile, the Adam optimizer with random gradient descent was selected to train the network optimally, setting the momentum to 0.937, the initial learning rate to 0.001, and the weight decay coefficient to 0 for 100 rounds of training.

To test the advantages of the modified method, the training process is visualized using the visualization tool Tensorboard, as shown in Fig. 11. The horizontal axis denotes the count of training rounds while the vertical axis denotes the corresponding parameters. From the figure, the improved model has a faster and more stable accuracy increase in the initial stage, and the overall improvement effect is slightly superior than before, and the advanced model has smaller loss values and faster convergence of the loss function.

5.2 Ablation Experiments

To verify the usefulness of the various methods proposed in this paper, we did ablation experiments based on YOLOv5s using the same hyperparameters and training strategies, as shown in Table 3. Experiment 1 is the original YOLOv5s, compared with Experiment 1, Experiment 2 replaces the Ghost module; Experiment 3 adds the CBAM module; Experiment 4 changes the CIoU loss function to the Focal-EIoU loss function; Experiment 5 combines the three improved methods. It can be seen from Experiment 2 that the model Size, GFLOPs, and Param are reduced by almost 50% after replacing the module with Ghost, but the reduction of model parameters leads to a decrease in accuracy. The results of Experiment 3 and Experiment 4 show that the addition of the CBAM module and the improvement of the Focal-EIoU loss function have improved the accuracy of the algorithm. The final results of Experiment 5 show that the model incorporating the three improved methods has a 3.1%

improvement in mAP, a 39% reduction in Size, a 44% reduction in GFLOPs, and a 40% reduction in Params compared to the original YOLOv5s model. Therefore, the comprehensive performance of the enhanced algorithm proposed in this paper is superior to the original YOLOv5s about surface litter detection, and it is also more appropriate for application in the computing devices of litter-cleaning vessels.

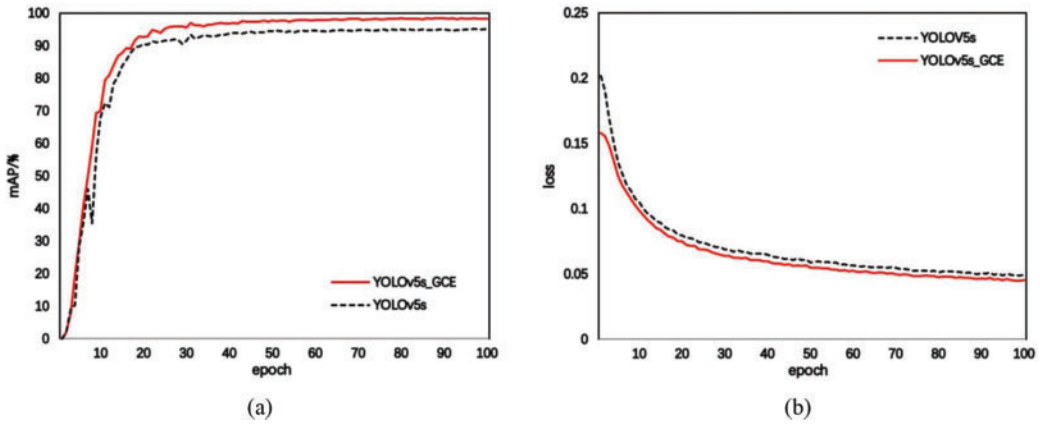


Figure 11: Comparison of model performance visualization before and after improvement. (a) average precision value (mAP) (b) loss function (Loss)

Table 3: Ablation experiment

Experiment	Models	G	C	E	mAP_0.5	Size/MB	GFLOPs	Params (M)
1	YOLOv5s				0.952	14.4	16.0	7.039
2	YOLOv5s-G	√			0.937	7.8	8.1	3.692
3	YOLOv5s-C		√		0.974	15.4	16.6	7.553
4	YOLOv5s-E			√	0.969	14.4	15.8	7.029
5	YOLOv5s-GCE	√	√	√	0.983	8.8	8.9	4.216

5.3 Comparison Experiments

In addition, based on the above parameter settings, the same datasets were used to train on the original YOLOv5s, YOLOv7-tiny, YOLOv3, SSD, and Faster R-CNN algorithm models, and the metric comparison of each model is listed in Table 4.

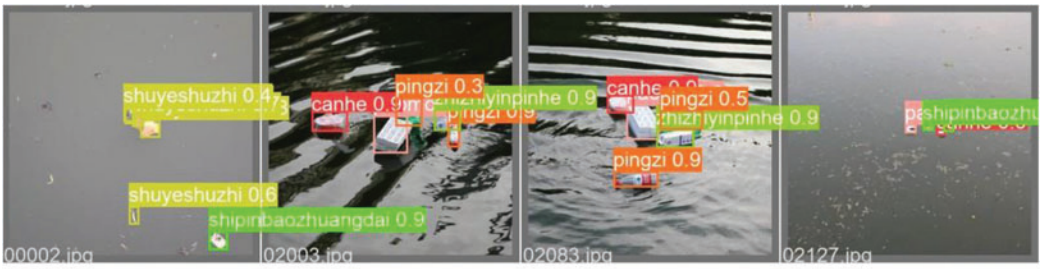
From the comparison of the performance of each model in Table 4, the improved algorithm in this paper performs better than other algorithms in aspects of mAP, precision, recall, and model size. mAP improved by 3.1% over the original YOLOv5s algorithm, 1.6% over YOLOv7-tiny, 6.7% over YOLOv3%, 9.4% over Faster R-CNN, and 7.6% over SSD; A 39% reduction in model size compared to the original YOLOv5s and a 25% reduction compared to the YOLOv7-tiny; FPS on the GPU is 13 higher than the original YOLOv5s and slightly lower than the YOLOv7-tiny. Params has much smaller than YOLOv3, Faster R-CNN, and SSD, and 30% less than YOLOv7-tiny. By comparing the performance of these models, the improved model has higher detection accuracy while ensuring real-time detection and lightweight volume of the algorithm.

Table 4: Comparison of metrics of mainstream algorithm

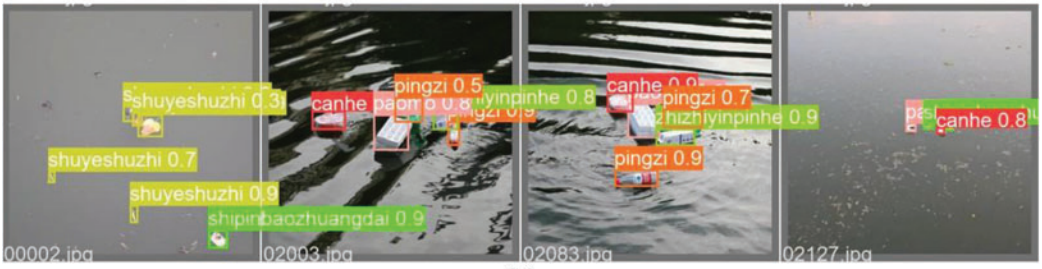
Models	mAP_0.5	P/%	R/%	Size/MB	FPS	GFLOPs	Params (M)
YOLOv3	0.916	91.25	82.99	235.2	71	87.4	61.556
SSD	0.907	92.04	67.56	93.7	50	155.1	24.414
Faster R-CNN	0.889	56.22	95.72	108.4	16	919.0	28.337
YOLOv5s	0.952	91.96	92.19	14.4	108	16.0	7.039
YOLOv7-tiny	0.967	92.62	95.61	11.7	128	13.1	6.024
YOLOv5s-GCE	0.983	97.70	97.17	8.8	121	8.9	4.216

5.4 Experiment Analysis

For verifying the actual detection effect of the revised algorithm, test validation is performed using test set images with complex backgrounds. The confidence threshold for the test set is set to 0.5, and the practical detection capacity of the algorithm before and after enhancement is presented in Fig. 12. From the detection comparisons in Figs. 12a and 12b, the detection accuracy of the improved method is greater than before. The original algorithm is prone to miss-detection when detecting images with more complex backgrounds, especially small target categories, and obscured target categories, while the improved algorithm in this paper adopts the CBAM module and optimized loss function, which can detect targets more accurately and effectively decline the miss-detection rate while significantly enhancing the feature acquisition capability of the network. Therefore, the algorithm proposed by this paper has better detection accuracy and detection speed while maintaining a small size, which is both feasible and superior and can fulfill the practical demands of real-time detection.



(a)



(b)

Figure 12: Comparison chart of model detection results. (a) original YOLOv5s test results (b) improved YOLOv5s detection results

6 Conclusion

For the traditional manual cleaning of water surface litter is time-consuming and inefficient, we proposed a lightweight water surface litter detection method by improved YOLOv5s to provide core technical support for water surface litter cleaning vessels. Based on the YOLOv5s network model, this model introduces a lightweight GhostNet network to replace the ordinary convolutional blocks and C3 modules in the original network, decreases the count of network parameters and computational overhead; embeds the CBAM module in the network to reinforce the network's capability to extract object feature information; optimize the loss function and use the Focal-EIoU loss function to enhance the regression accuracy and solve the positive and negative sample proportion imbalance problem. After practical testing, the mAP of the improved algorithm reaches 98.3% and the model size is 8.8 MB, which is 39% less, with 44% fewer GFLOPs, and 40% fewer parameters compared to the original model. The single image test speed is about 8 ms, which has high detection accuracy and detection speed while maintaining lightweight volume, it can fulfill the needs of real-time detection and processing of water surface litter.

After that, it will further enrich the count of categories in the water surface litter dataset, improve the generalization capability of the model, optimize the network structure, compress the model volume, facilitate the subsequent terminal deployment, and realize real-time detection in mobile.

Acknowledgement: The authors gratefully acknowledge the support of China University Industry-University Research Innovation Fund Project, Science and Technology Innovation R&D Project of the State General Administration of Sports of China, Major Project of Philosophy and Social Science Research in Higher Education Institutions in Hubei Province, Key Project of Hubei Provincial Key Laboratory of Intelligent Transportation Technology and Equipment Open Fund.

Funding Statement: Support for this work was in part from the China University Industry-University Research Innovation Fund Project (No. 2022BL052), author B.T, https://www.cutech.edu.cn; in part by the Science and Technology Innovation R&D Project of the State General Administration of Sports of China (No. 22KJCX024), author B.T, https://www.sport.gov.cn; in part by the Major Project of Philosophy and Social Science Research in Higher Education Institutions in Hubei Province (No. 21ZD054), author B.T, https://jyt.hubei.gov.cn; Key Project of Hubei Provincial Key Laboratory of Intelligent Transportation Technology and Equipment Open Fund (No. 2022XZ106), author B.T, https://hbpu.edu.cn.

Conflicts of Interest: The manuscript is submitted without conflict of interest and all authors have approved the manuscript for public release. An original study of the work described has not been submitted or published elsewhere, the revised manuscript has been approved by all listed authors.

References

[1] Y. Tong, J. Liu and S. Liu, "China is implementing "Garbage classification" action," *Environmental Pollution*, vol. 259, no. 11307, pp. 1–2, 2019.

[2] S. Kong, M. Tian, C. Qiu, Z. Wu and J. Yu, "IWSCR: An intelligent water surface cleaner robot for collecting floating garbage," in *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, no. 10, pp. 6358–6368, 2021.

[3] W. Tang, H. Gao and S. Liu, "Design and implementation of small waters intelligent garbage cleaning robot system based on raspberry pi," *Science Technology and Engineering*, vol. 19, no. 34, pp. 239–247, 2019.

[4] N. Dilshad, A. Ullah, J. Kim and J. Seo, "LocateUAV: Unmanned aerial vehicle location estimation via contextual analysis in an IoT environment," *Internet of Things Journal*, vol. 10, no. 5, pp. 4021–4033, 2022.
[5] P. Zhang, Z. Wang and F. Wang, "Research on image target detection algorithm based on depth learning," *Foreign Electronic Measurement Technology*, vol. 39, no. 8, pp. 34–39, 2020.
[6] C. Yu, B. Xiao, C. Gao, L. Yuan, L. Zhang *et al.,* "Lite-hrnet: A lightweight high-resolution network," in *2021 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Nashville, TN, USA, pp. 10440–10450, 2021.
[7] X. Tang, C. Wang, J. Su and C. Taylor, "An elevator button recognition method combining yolov5 and ocr," *Computers, Materials & Continua*, vol. 75, no. 1, pp. 117–131, 2023.
[8] R. Girshick, J. Donahue, T. Darrell and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *2014 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Columbus, OH, USA, pp. 580–587, 2014.
[9] R. Girshick, "Fast R-CNN," in *2015 IEEE Int. Conf. on Computer Vision (ICCV)*, Santiago, Chile, pp. 1440–1448, 2015.
[10] S. Ren, K. He, R. Girshick and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2017.
[11] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You only look once: Unified, real-time object detection," in *2016 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, pp. 779–788, 2016.
[12] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *2017 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, pp. 6517–6525, 2017.
[13] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed *et al.,* "SSD: Single shot multibox detector," in *2016 European Conf. on Computer Vision (ECCV)*, Amsterdam, Netherlands, pp. 21–37, 2016.
[14] Z. Shen, Z. Liu, J. Li, Y. -G. Jiang, Y. Chen *et al.,* "DSOD: Learning deeply supervised object detectors from scratch," in *2017 IEEE Int. Conf. on Computer Vision (ICCV)*, Venice, Italy, pp. 1937–1945, 2017.
[15] A. Mathias, S. Dhanalakshmi, R. Kumar and R. Narayanamoorthi, "Deep neural network driven automated underwater object detection," *Computers, Materials & Continua*, vol. 70, no. 3, pp. 5251–5267, 2022.
[16] S. Woo, J. Park, J. Y. Lee and I. S. Kweon, "CBAM: Convolutional block attention module," in *Proc. the European Conf. on Computer Vision (ECCV)*, Munich, Germany, pp. 3–19, 2018.
[17] Y. Zhang, W. Wen, Z. Zhang, Z. Jia, L. Wang *et al.,* "Focal and efficient IoU loss for accurate bounding box regression," arXiv preprint, pp. 146–157, 2021. https://arxiv.org/pdf/2101.08158
[18] Z. Zheng, P. Wang, W. Liu, J. Li, R. Ye *et al.,* "Distance-IoU loss: Faster and better learning for bounding box regression," arXiv preprint, pp. 12993–13000, 2019. https://arxiv.org/abs/1911.08287v1
[19] Y. Matsumoto, "Ship image recognition using HOG," *Journal of Japan Institute of Navigation*, vol. 129, pp. 105–112, 2013.
[20] N. Kaido, S. Yamamoto and T. Hashimoto, "Examination of automatic detection and tracking of ships on camera image in marine environment," in *2016 IEEE Techno-Ocean (Techno-Ocean)*, Kobe, Japan, pp. 58–63, 2016.
[21] L. Zhang, Y. Zhang, Z. Zhang, J. Shen and H. Wang, "Real-time water surface object detection based on improved faster R-CNN," *Sensors*, vol. 19, no. 16, pp. 3523–3539, 2019.
[22] H. Panwar, P. K. Gupta, M. K. Siddiqui, R. Morales-Menendez, P. Bhardwaj *et al.,* "AquaVision: Automating the detection of waste in water bodies using deep transfer learning," *Case Studies in Chemical and Environmental Engineering*, vol. 2, no. 100026, 2020.
[23] X. Li, M. Tian, S. Kong, L. Wu and J. Yu, "A modified YOLOv3 detection method for vision-based water surface garbage capture robot," *International Journal of Advanced Robotic Systems*, vol. 17, no. 3, pp. 1729–8806, 2020.
[24] G. Jocher, A. Stoken, J. Borovec, A. Chaurasia, L. Changyu *et al.,* "Ultralytics/yolov5: V5.0-YOLOv5-p6 models, AWS, supervisely and YouTube integrations," 2021. https://doi.org/10.5281/zenodo.4679653

[25] T. -Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan *et al.,* "Feature pyramid networks for object detection," in *2017 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, pp. 936–944, 2017.
[26] S. Liu, L. Qi, H. Qin, J. Shi and J. Jia, "Path aggregation network for instance segmentation," in *2018 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT, USA, pp. 8759–8768, 2018.
[27] H. Gao, Y. Tian, F. Xu and S. Zhong, "Survey of deep learning model compression and acceleration," *Journal of Software*, vol. 32, no. 1, pp. 68–92, 2021.
[28] D. Ge, H. Li, L. Zhang, R. Liu, P. Shen *et al.,* "Survey of lightweight netural network," *Journal of Software*, vol. 31, no. 9, pp. 2627–2653, 2020.
[29] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally *et al.,* "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5 MB model size," arXiv preprint, 2016. https://arvix.org/pdf/1602.07360
[30] X. Zhang, X. Zhou, M. Lin and J. Sun, "ShuffleNet: An extremely efficient convolutional neural network for mobile devices," in *2018 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT, USA, pp. 6848–6856, 2018.
[31] N. Ma, X. Zhang, H. T. Zheng, H. Zheng and J. Sun, "ShuffleNet v2: Practical guidelines for efficient cnn architecture design," in *2018 European Conf. on Computer Vision (ECCV)*, Munich, Germany, pp. 122–138, 2018.
[32] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang *et al.,* "MobileNets: Efficient convolutional neural networks for mobile vision applications," arXiv preprint, 2017. https://arvix.org/abs/1704.04861
[33] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov and L. -C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *2018 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT, USA, pp. 4510–4520, 2018.
[34] A. Howard, M. Sandler, B. Chen, W. Wang, L. Chen *et al.,* "Searching for mobilenetv3," in *2019 IEEE/CVF Int. Conf. on Computer Vision (ICCV)*, Seoul, Korea (South), pp. 1314–1324, 2019.
[35] X. Jiang, "Research on scene image classification algorithm based on deep learning," M. S. Dissertation, Beijing University of Posts and Telecommunications, China, 2019.
[36] S. Xie, H. Zheng, C. Liu and L. Lin, "SNAS: Stochastic neural architecture search," in *2019 European Conf. on Computer Vision (ECCV)*, New Orleans, LA, USA, 2019.
[37] J. Hu, L. Shen and G. Sun, "Squeeze-and-excitation networks," in *2018 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT, USA, pp. 7132–7141, 2018.
[38] H. Zhang, M. Cisse, Y. N. Dauphin and D. Lopez-Paz, "Mixup: Beyond empirical risk minimization," arXiv preprint, 2017. https://arxiv.org/abs/1710.09412
[39] K. Han, Y. Wang, Q. Tian, J. Guo, C. Xu *et al.,* "GhostNet: More features from cheap operations," in *2020 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA, 2020, pp. 1577–1586, 2020.
[40] Y. Ke, X. Lin, R. Liao and Z. Wei, "Research on hierarchical decomposition of convolution neural network," *Computer Engineering*, vol. 45, no. 11, pp. 191–197, 2019.